Reminiscences and Reflections of a Codebreaker

Peter Hilton*

Distinguished Professor of Mathematics Emeritus
Department of Mathematical Sciences
SUNY, Binghamton, New York 13902-6000

1 Introduction

Many books have now been published about the work of the Bletchley Park codebreakers during World War II. Outstanding among these are Alan Turing: The Enigma, by Andrew Hodges [Ho], a sensitive and enormously informative biography of a genius who made a unique contribution to winning the war while he was simultaneously inventing the computer; and Codebreakers, edited by F. H. Hinsley and Alan Stripp [Hin], a series of articles providing detailed information on the methods employed by the codebreakers of Bletchley Park. Particularly to be commended among the latter is the article by Professor I. J. (Jack) Good, entitled "Enigma and Fish", in which Jack, one of the key members of the teams working first on Naval Enigma and then on the even more sophisticated Geheimschreiber code (which we called Fish!), describes the machines employed by the Germans and the machines we developed to help to read messages encrypted by these machines. It is a great advantage, of course, for those able, like Jack Good, to provide precise descriptions of these machines and of our methods, that much of the necessary information has now, at long last, been declassified.

With so many good sources of information available, it would be pointless to write yet another technical article. On the other hand, there has not been the same wealth of information available about the more human side of our activities at Bletchley Park, so perhaps there is a gap to be filled. Of course, I will only speak for myself. I, too, like Jack Good, worked first on Naval Enigma (in 1942) and then on Fish until the end of the European War (May, 1945); but I had a period, at the end of 1942 and early in 1943, when I was withdrawn from the Enigma team and joined the research group actually trying to understand the modus operandi of the Geheimschreiber machine. I then was attached to the Testery, but liaised with the Newmanry. The Testery people largely used hand methods, that is, they did not themselves use the Colossus machine; but, of course, they routinely used the output of Colossus to complete the effective decryption of a message. The Newmanry ran the Colossi.

* The editor (wdj) would like to express his appreciation to Michael Ryan, editor of Global Intelligence Monthly, for permission to reproduce this article.

Even though this reminiscence is very informal and personal, it is relevant to point out that the teams to which I belonged were working on the highest grade Germany military and diplomatic ciphers. I do not believe that those working on lower grade (e.g., field) ciphers felt much of the excitement we felt; and I am sure that those who only came into the picture once the messages had been deciphered had an entirely different experience from our own.

What then are my most vivid recollections from those days? Let me start with the recruiting process.

2 The Road to Bletchley Park

It is now common knowledge (see e.g., Hinsley et al., Vol. II [Hin]) that in October, 1941, four top Bletchley Park cryptanalysts, including Alan Turing, wrote a letter to Churchill arguing that it was essential to give the highest priority to the recruitment of codebreakers and the provision of necessary equipment. Churchill might have reacted like a bureaucrat and said that the letter should have been properly routed through the corridors of Whitehall — but he didn't. He saw the good sense of what was proposed and its urgency; and he minuted his chief of staff "Action this day". Thus it came about — though I did not know this at the time — that an interviewing board came to Oxford in November, 1941, to look for "a mathematician with a knowledge of modern European languages". (Unfortunately, however, the dictates of security required that the candidates should not be told the nature of the work they would ge doing — it was my distinct impression that the members of the interviewing board did not know this themselves.)

Now the British educational system, at the time, being based on the principle of premature specialization, virtually guaranteed that there would be no such person, except by chance. [1]

My tutor recommended me to attend the interview although I was not a mathematician — merely an undergraduate student of mathematics — and my knowledge of German was rudimentary, since I had merely been teaching myself for a year. [2]

In the event, I believe I was the only candidate to present himself, and I was immediately offered a position — in the Foreign Office. However, the condition was imposed that I must start in January, 1942. This was a blow as my age group (I was born in 1923) was not due to be drafted till August, 1942. But my experience of training for the Royal Artillery as a student at Oxford — all university students had to undergo military training — had convinced me that, if I was conscripted into the Royal Artillery, I would

[1] There were, of course, many outstanding mathematicians among the Jewish refugees from Germany and Austria, but they could not be trusted as enemy aliens!

[2] It could not be doubted that German was the "modern European language" in question.

almost certainly die young — of sheer boredom! Thus it did not take me long to decide that, whatever the secret work I was to undertake at Bletchley Park, it was certain to be far more interesting than being an artilleryman, and, much as I regretted losing two terms at Oxford, the sacrifice was surely worthwhile. How right I was!

So it came about that, on 12 January 1942, I presented myself at the gates of Bletchley Park and was escorted to Hut 8. I met many people that day, but I didn't find out the nature of the work. For one person I met — none other than Alan Turing himself — asked me if I played chess and added, when I replied affirmatively, that he had a chess problem he had not been able to solve and invited me to help him to solve it. Fortunately, I was able to help him to solve it; and I like to think that the cordial relationship I enjoyed with Alan Turing for the remainder of his tragically short life (he committed suicide in 1954, just short of his 42^{nd} birthday) owed much to the fortunate circumstances of our first meeting. On my second day I discovered that I was to be involved in the decoding of Naval Enigma, especially of the highly secret Offizier messages, and I got my first instructions in the subtle methods developed by the Hut 8 team of cryptanalysts to achieve an amazingly high success rate and a remarkable speed of decryption. A uniquely exciting period of my life had begun!

3 A Tribute to My Colleagues

It goes without saying that my colleagues were all extraordinarily good at their wartime jobs at Bletchley Park — they were intelligent, quick, inventive, immensely hard-working and always encouraging each other. Almost all resumed or went on to academic jobs after the war, though some chose different careers. [3]

It is really invidious to pick out any for special praise or mention; yet I feel I should if only to point to the wide variety of attributes they displayed, either in common or individually, in addition to their mathematical flair. I will, rather arbitrarily, confine myself to seven names, which, to avoid gross favoritism, I will refer to in alphabetical order. Of course, it is understood that these people made a profound impression on me; most of them have continued to exert an influence on my life in the postwar years.

Hugh Alexander (C. H. O'D. Alexander, to give him full panoply of initials) was the British chess champion. He was a most colorful person, with an attractive personality and striking intelligence. He and Shaun Wylie taught me much of what I learned about Naval Enigma and the decoding problem in my early days in Hut 8 — he was at that time in charge of our Section. What struck me about him then, in addition to the qualities I have mentioned,

[3] One, Roy Jenkins, now Lord Jenkins of Hillhead, was Home Secretary in a Labour Government and is now Chancellor of Oxford University. Another, Peter Benenson, founded Amnesty International.

and his sense of humor, was his complete informality. This, combined with a total lack of self-regard, I was to come to recognize as the distinguishing mark of greatness in my colleagues. Unfortunately, I saw very little of Hugh after leaving Hut 8.

Jack Good (now I. J. Good, Distinguished Professor of Statistics Emeritus at Virginia Polytechnic Institute) was the nearest any of us came to being an applied mathematician — I will revert to this point later. He was, in fact, a probabilist, but he was — and is — a polymath. Both in Hut 8 and in the Newmanry he was enormously effective and productive, both of decrypts and ideas. He is possessed of a prodigious and totally accurate memory which makes him, today, the most reliable, and comprehensive, authority on the history of those times. His very individual sense of humor, together with the modesty which was characteristic of all those heroes of long ago, enrich our friendship, which persists to this day.

Donald Michie (now Professor of Artificial Intelligence at the University of Edinburgh) was an example of inspired recruitment. He came to the Testery (though he also liaised very effectively with the Newmanry) as a classical scholar, but showed remarkable adaptability to our work, acquiring an ability to think mathematically even though he knew very little mathematics. He was, and remains, truly brilliant. He became a very close friend of Alan Turing, Jack Good, myself and many others; and his sunny disposition and willingness to learn — together with a remarkable ability to do so very quickly — made him an invaluable colleague. It is perhaps not coincidental that, as he mutated from classical scholar to become a master of theoretical computer science, his politics moved simultaneously from right to left (though always reasonable!)

Max Newman (Professor M. H. A. Newman, F.R.S.) was already a distinguished topologist when he came to Bletchley Park to head the Section responsible for the machine aspects of the decryption of Fish, by 1943 certainly the most important high grade cipher being used by the Germany military. He was wonderfully effective in this role, and struck up a working relationship with Alan Turing which was resumed at Manchester University after the war when, in conjunction with the university electrical engineers and others at Ferranti, they designed (and built) a computer [4]. Both Alan Turing and I joined his department in 1948 — but at very different levels of seniority!

Max had excellent ideas, mathematical and administrative; but it is first and foremost as a facilitator that I remember him. Both in the Newmanry and in the Mathematics Department at Manchester University, he created conditions under which we, his colleagues, could work best. He never imposed on us a chore which could only be justified on bureaucratic grounds. From

[4] Max was appointed Fielden Professor of Pure Mathematics at Manchester University in 1945 on leaving Bletchley Park

his understanding and leadership I benefited enormously — at both places where he exercised them. See [H] for further remarks about Max Newman.

Alan Turing, it is generally agreed, was a genius. He had already shown this at Cambridge before the war, when he produced his strikingly original definition of a computable function in which he introduced the concept of a universal machine, now always referred to as a Turing machine. What very few knew then, and somewhat more know now, is that, even in those early days, his machine was not merely, in his mind, a metaphor but also a blueprint for a machine which could actually be built, that is, a computer. The history of the development of these ideas is very well treated in the book by Andrew Hodges, already referred to.

I will be saying more about Alan Turing later. Let me only add now that it was an extraordinary, and wonderful, experience to know him; and that he was the friendliest of men [5].

Henry Whitehead (Professor J. H. C. Whitehead, F.R.S.) has a special place in my affections, and not only because he was such a lovable man, so creative a mathematician, and so interesting and diverse a personality. Henry already had a reputation as a great — but difficult — mathematician when he came to Bletchley Park. He had done outstanding work in algebraic and combinatorial topology at Oxford, but his work was not well understood (he rewrote much of it after the war in the hope of achieving greater clarity). Nevertheless, he was recognized as an outstanding talent and, after the war, he was appointed Waynflete Professor at Oxford. He and I had become very friendly at Bletchley Park — we shared a common attitude to politics, cricket and beer, among other interests — and, after his return to Oxford, he invited me also to return to Oxford to become his doctoral student. I accepted his invitation entirely on the basis of my affection for him and my trust in his intellectual judgment. [6]

Thus Henry exerted a profound influence on my choice of career and hence on my life. I have never regretted that influence. Very unfortunately, Henry collapsed and died, suddenly and unexpectedly, on a street in Princeton in 1960 at the age of 55, at the height of his powers — a grievous loss to mathematics and all his many friends.

Shaun Wylie, like Hugh Alexander, inducted me into the work of Hut 8; but he and I remained close friends as he also moved to the Newmanry — and, subsequently, we became colleagues on the faculty of Cambridge University and wrote a book together, Homology Theory, which became a standard text among graduate students and algebraic topologists for many years. Shaun is a man of unmistakable brilliance, matched only by kindness. He is a great teacher and a very cultured scholar. I have benefited more than I can say from

[5] This needs to be said, as he has sometimes been presented as awkward and nervous and uncomfortable in the presence of others.

[6] I recall asking him "What is algebraic topology, Henry?" He replied, "Don't worry, Peter. You'll love it!" On the strength of that assurance, I decided to become his student.

his friendship — and that of his remarkable wife Odette, whom he married when she was a Wren [7] — a very senior Wren — working in the Newmanry. Long may they both flourish!

4 The Teaching of Mathematics

I learnt many lessons from my exciting three and a half years at Bletchley Park. I have already hinted at some; thus, for example, I learned of the friendliness and lack of conceit of good mathematicians, a fact I can now conclusively confirm after 50 years among academic mathematicians. However, there is one lesson I learnt, about the teaching of mathematics, which I regard as crucially important. It does, however, embody a very controversial principle.

We were, first in Hut 8 and then while working on Fish, a group of some 30 people (at our peak). We were, almost all, mathematicians or would-be mathematicians. But none of us — with the possible exception of Jack Good — could be described as applied mathematicians. We were pure mathematicians, in the sense that our main interest and love of research, actual or intended, lay inside mathematics itself. Yet we were all, at Bletchley Park, applying mathematics. True, we were not doing conventional applied mathematics — ordinary and partial differential equations, theoretical physics, and such. We were, of course, using (and developing) some statistical methods but their theoretical basis was neither new nor terribly profound. If there was one branch of mathematics which we could be said to be using systematically, it was mathematical logic. But a better description of our work would be to say that we were using a mathematical way of thinking in our approach to the problem at hand — the mathematics itself was not very sophisticated, but we would have been useless if we had not acquired this ability to think clearly in mathematical terms. It is also worth adding that we would have been useless if we had not been strongly motivated, that is, consumed by a fierce desire to solve the problems the enemy was confronting us with.

What has all this to do with the teaching of mathematics, let us say, at the university level? To me the obvious implication is that the essential features of a good mathematics education, designed to enable the student subsequently to use mathematics effectively in his or her chosen occupation are that it inculcate the ability to think mathematically, that is, that the student acquire, in Speiser's phrase, mathematische Denkweise; and that it build in the student a strong appetite for using mathematics to solve problems which originate outside mathematics. (Of course, this must then be supplemented by a real interest in the problem area with which the student is confronted in his or her chosen profession.) What do not seem to be essential components of a good mathematics education for the future user of mathematics are (i) any special attention to the areas of mathematics usually associated with

[7] Women's Royal Naval Service.

the occupation chosen by the student, or (ii) the acquisition of expertise in the area (of science, engineering, statistics, . . .) to which the mathematics is to be applied, or, indeed, in any other area. As to (i), it would seem to me that any part of mathematics could serve to prepare the student to apply mathematics, provided it is properly taught, that is, taught for genuine understanding and effective problem-solving and not merely for the acquisition of knowledge and mechanical skill. As to (ii), I remain convinced that the experience of applying mathematical reasoning to the study of some discipline would be very valuable to the student. But time is limited, and we must make choices; and there can be no case for impoverishing the student's mathematical education to provide time to acquire a working knowledge of some other discipline. As any enlightened employer will tell you, "We can teach you what we want you to know about our work. What we cannot teach you is the necessary mathematical know-how."

5 The Life and Death of Alan Turing

I have already, in this article, testified to my enormous respect for Alan Turing, whom I have described as an authentic genius; and to my incredible good fortune in being able to claim him as a friend, despite the vast difference in our intellectual capacities. His contribution to the work of the Bletchley Park codebreakers was unique and irreplaceable. This has been attested by many; and forms a theme of the excellent play "Breaking the Code" by Hugh Whitemore, and the remarkable novel Enigma by Robert Harris [8]. However, there is a particular feature of his life and his nature which must be set on record if one wishes to complete the picture of the man — Alan Turing was a homosexual. This fact is central to the drama of Hugh Whitemore's play, and is there treated very sympathetically; but the details of Turing's life given in the play are too far removed from reality for one to rely on this fine work of fictional drama to provide a basis for an assessment of the man.

In the first place, we, his colleagues at Bletchley Park had no idea that Alan was a homosexual, since he gave no evidence of this fact throughout his time at Bletchley Park; indeed, Jack Good has trenchantly and pertinently remarked "Fortunately, the authorities at Bletchley Park had no idea Turing was a homosexual; otherwise, we might have lost the war."

Unfortunately, in the early 1950's a vigorous campaign was mounted in Britain against male homosexuals — homosexual acts carried out in private by adult males were a criminal offense — and in 1952, in circumstances well described by Andrew Hodges in his biography, Alan Turing was arrested and brought before a magistrate on a charge of committing this "crime". The magistrate recognized that Alan was a very special person — a Fellow of the

[8] While Turing's contribution was unique, our work was no "one-man show" — contrary to the impression given in these two dramatic reconstructions of Bletchley Park days.

Royal Society, Reader in Mathematics at Manchester University, holder of the Order of the British Empire for (unspecified) services to his country during the war — and tried to be as lenient as possible. Alan was "bound over" — effectively, a verdict of guilty but with no penalty imposed, on condition that he underwent hormone treatment whose effect, he later bitterly remarked, was merely to enlarge his breasts. He lost his security clearance; and the U. S. authorities treated him as a felon and refused to grant him a visa (he had been engaged on joint work with Johnny von Neumann). Lonely and depressed, he committed suicide on June 7, 1954, during the Whitsuntide weekend, by eating some apple slices he had himself laced with cyanide. Clearly, he know that, sooner or later, he would find life intolerable and, in his typical way, he prepared himself and his circumstances for the arrival of that event.

It is shameful that civilized nations should enact vicious legislation capable of ruining the lives of some of its finest citizens, and then set the forces of "law and order" to hound those unfortunate people whom they might catch in their trap. It is alarming to find the same prejudices [9] which destroyed the life of a very great man, to whom all who love freedom and democracy owe so much, once again manifesting themselves today, doubtless strengthened by fears of the AIDS virus and its effects. (Even as I write, the radio is reporting a case in Wyoming where four young people tortured a student of the university, Matthew Shepard, till he was close to death, for no other apparent reason than that he was known to be gay [10]. Will we never learn?

References

[H] Peter Hilton, *Obituary*, M. H. A. Newman, Bulletin of the London Mathematical Society, 18 (1986), 67 - 72.

[Hin] F. H. Hinsley, et al., **British Intelligence in the Second World War**, 3 volumes, Her Majesty's Stationery Office.

[Ho] Andrew Hodges, **Alan Turing: The Enigma**, Simon and Schuster, N.Y, 1988

email: marge@math.binghamton.edu

[9] Let no one suppose these prejudices were then confined to Britain, and that official America was innocent of such barbarism. The Immigration and Nationality Act (1952) states "Aliens afflicted with psychopathic personality . . . shall be excludable from admission to the U. S." In 1967, the Supreme Court pronounced that "the legislative history of the Act indicates beyond a shadow of doubt that the Congress intended the phrase "psychopathic personality" to include homosexuals."

[10] The student died two days later.

FISH and I

W. T. Tutte*

Distinguished Professor Emeritus
C& O Department
University of Waterloo
Waterloo, Ontario, Canada N2L 3G1

1 Introduction

I have been asked to speak today about some cryptographic work I was engaged in at Bletchley Park, during the Forties. I was concerned mainly with a German machine-cipher known in Bletchley as "FISH". The network using this system grew to have many links and each link was given the name of a kind of fish. Thus the first link to be intercepted was called "Tunny" and I recall such names as "Bream", "Herring" and "Mackerel" for later links.

The text-book for this lecture is "Code Breakers", edited by F.H. Hinsley and Alan Stripp [HS]. It is subtitled "The Inside Story of Bletchley Park." Part 3 of this book tells the story of "FISH". It tells that the first FISH traffic to be intercepted was on a German Army radio link between Athens and Vienna from the middle of 1941. Much praise is due to the designers and operators of the intercepting equipment for producing accurate copies of the German messages, with few garbled letters and every letter, garbled or not, in its proper place.

The letters used were those of the International Teleprinter Code. There were two basic symbols, called at Bletchley "Dot" and "Cross". They would equally well have been called "Zero" and "One". Or, with electrical switches in mind, "On" and "Off". Each letter was a sequence of 5 basic symbols, so there were 32 letters in all. Table 1 sets out this International Code.

In Table 1 the five symbols of a letter are written in a row. It was more usual at Bletchley to write them in a column. Thus the beginning of a message in teleprinter code might appear as in Table 2. Here "9" stands for "Word Space". When a message or other sequence of letters was written like this we referred to the five rows as the five "impulses", five streams of dots and crosses.

* Professor Tutte, FRS, worked from 1941 to 1945 in the British cryptanalytic headquarters at Bletchley Park, the most successful intelligence agency in world history. His work, which combined elements of statistics and combinatorics, was instrumental in the breaking of FISH, a series of codes that were used by the German command for encrypting communications between the highest authorities. Subsequent cryptanalytic work on FISH included the development of Colossus, the world's first electronic computer.

•	•	•	•	•	All space (7)
•	•	•	•	X	T
•	•	•	X	•	Carriage Return (4)
•	•	•	X	X	O
•	•	X	•	•	Word Space (6)
•	•	X	•	X	H
•	•	X	X	•	N
•	•	X	X	X	M
•	X	•	•	•	Line feed (5)
•	X	•	•	X	L
•	X	•	X	•	R
•	X	•	X	X	G
•	X	X	•	•	I
•	X	X	•	X	P
•	X	X	X	•	C
•	X	X	X	X	V
X	•	•	•	•	E
X	•	•	•	X	Z
X	•	•	X	•	D
X	•	•	X	X	B
X	•	X	•	•	S
X	•	X	•	X	Y
X	•	X	X	•	F
X	•	X	X	X	X
X	X	•	•	•	A
X	X	•	•	X	W
X	X	•	X	•	J
X	X	•	X	X	Figures (3)
X	X	X	•	•	U
X	X	X	•	X	Q
X	X	X	X	•	K
X	X	X	X	X	Letters (2)

Table1. Teleprinter Code

9	S	P	R	U	C	H	N	U	M	M	E	R	9
•	X	•	•	X	•	•	•	X	•	•	X	•	•
•	•	X	X	X	X	•	•	X	•	•	•	X	•
X	X	X	•	X	X	X	X	X	X	X	•	•	X
•	•	•	X	•	X	•	X	•	X	X	•	X	•
•	•	X	•	•	•	X	•	•	X	X	•	•	•

Table2.

I started work at Bletchley Park in (I think) May 1941. It was several months later that I encountered Tunny.

2 On additive ciphers

In an additive cipher we convert the clear message ($\mathcal{C}$) into the cipher message ($\mathcal{Z}$) by adding to it, letter by letter, a sequence of letters called the key ($\mathcal{K}$). The addition has to be defined. One method is to number the letters of the alphabet, in order, from 1 to 26 and then add those numbers mod 26. Thus (see Table 3),

$$J + S = 10 + 19 = 29 \equiv 3 = C.$$

In the case of the teleprinter code an obvious method is to add the letters as 5-vectors mod 2. Thus

$$J + S = \begin{matrix}\mathrm{X}\\ \mathrm{X}\\ \bullet\\ \mathrm{X}\\ \bullet\end{matrix} + \begin{matrix}\mathrm{X}\\ \bullet\\ \mathrm{X}\\ \bullet\\ \bullet\end{matrix} = \begin{matrix}\bullet\\ \mathrm{X}\\ \mathrm{X}\\ \mathrm{X}\\ \bullet\end{matrix} = C$$

The key we may suppose is a string of letters produced by the cipher machine. We can write the process of encipherment as an algebraic equation

$$\mathcal{C} + \mathcal{K} = \mathcal{Z}.$$

A	1	N	14
B	2	O	15
C	3	P	16
D	4	Q	17
E	5	R	18
F	6	S	19
G	7	T	20
H	8	U	21
I	9	V	22
J	10	W	23
K	11	X	24
L	12	Y	25
M	13	Z	26

Table3.

Additive ciphers have a well-known weakness. Suppose two messages are carelessly sent on the same key. Then

$$\mathcal{C}_1 + \mathcal{K} = \mathcal{Z}_1$$
$$\mathcal{C}_2 + \mathcal{K} = \mathcal{Z}_2.$$

Therefore

$$\mathcal{C}_1 - \mathcal{C}_2 = \mathcal{Z}_1 - \mathcal{Z}_2.$$

We called such a pair a “depth of two”. If the enemy cryptanalyst has reason to suspect such a depth he subtracts $\mathcal{Z}_2$ from $\mathcal{Z}_1$ and knows that he probably has $\mathcal{C}_1 - \mathcal{C}_2$. This, with reasonable luck, he can separate into the two clear messages $\mathcal{C}_1$ and $\mathcal{C}_2$. He reads them both. Moreover by subtracting $\mathcal{C}_1$ from $\mathcal{Z}_1$ he can find $\mathcal{K}$. The procedure is to try a likely word, perhaps like LONDON, in successive positions in $\mathcal{C}_1$ calculating the corresponding six letters of $\mathcal{C}_2$ until one finds a position in which those six letters are plausible as plain text. Perhaps they are then IMPENE. Guessing that this continues as “IMPENETRABILITY” he writes as follows

$\mathcal{C}_1$ LONDONTHOUARTTH(E)
IMPENETRABILITYT

Soon he is announcing $\mathcal{C}_1$ as beginning “London thou art the flour[1] of cities all” and $\mathcal{C}_2$ “Impenetrability, thats what I say”.

3 HQIBPEXEZMUG

It was the German custom on the TUNNY link to give in the preamble of each message a sequence of 12 letters. At Bletchley people called this sequence the “indicator” and guessed that it gave the settings for 12 wheels in a cipher machine. Occasionally two cipher messages would come with the same indicator. The cryptanalyst would say “Same settings, therefore same key. Try it as a depth of two”. There was enough success to identify Tunny as an additive cipher using the mod 2 addition I have already mentioned.

One day there came two long cipher messages, each about 4000 letters long, with the same indicator HQIBPEXEZMUG. This depth of two was successfully read. It proved to be two attempts at the same message, one having more word-spacing and other punctuation than the other. This obviously was a great help in the depth-reading. Col. J. Tiltman read this depth and deduced some 4000 letters of key. Next problem: given that the machine produced this key, determine the structure of the machine. In the language of the time and place cryptanalysts sought to “break the Tunny key”.

All this was done before I had any dealings with “Tunny”.

Some three months later, the key still unbroken, Major G. W. Morgan, head of the Research Section, gave a copy of the key to me and said “See what you can do with this”.

Now at my pre-Bletchley cryptographic school in London I had learned that you can sometimes get results by writing out a cipher text on a period and looking for repeats. I resolved to do this with one or more impulses of the key. But on what period? I had been given some information about the

[1] old spelling

letters of the indicator. There seemed to be 25 possibilities for 11 of these but only 23 for the last. Perhaps I should try periods of 23 or 25. Or why not try both at once by writing the impulse on a period of $23 \times 25 = 575$? I can't say that I had much faith in this procedure but I thought it best to seem busy. So I wrote out the first impulse in 7 rows of length 575 and looked for repeats of short patterns of dots and crosses, vertical repeats from row to row.[2]

As expected there were not significantly many. But then I noticed a lot of repeats on a diagonal. It seemed that I would have got better results on a period of 574. So I wrote out the impulse again on that period and found pleasingly many repeats of dot-cross patterns of length 5 or 6.

Then I tried a period of 41, this being a prime factor of 574, with even better results. The upshot was that the first impulse of the key was a sum of two sequences that I named χ_1 and Ψ_1, of dots and crosses. χ_1 was periodic with period 41. Ψ_1 was basically periodic too, with period 43. But whereas χ_1 moved on one place for each letter, Ψ_1 sometimes moved on one place and sometimes stayed still.

At this stage the whole Research Section joined in to analyse each other impulse into a χ-wheel and a Ψ-wheel. In "Code Breakers" it is recorded that the χ-wheels, in order from the first impulse to the 5th, had periods 41, 31, 29, 26 and 23, while the Ψ-wheels had periods 43, 47, 51, 53 and 59. A major discovery was that the χ-wheels moved in step. Either all moved on one place or all stayed still. They moved whenever an 11th wheel showed a cross. (Period 37.) This 11th wheel moved on one place when a 12th wheel, of period 61, showed a cross, and the 12th wheel moved on one place for each letter. The 11th and 12th wheels were called the "motor wheels".

In "Code Breakers" I am said to have worked out the whole of this by myself, but that is an exaggeration. Note that the fifth χ-wheel had period 23 and the clue from the indicator letter of 23 possibilities was a valid one.

Presumably if I had not noticed the diagonal repeats I would have tried the method again on the 2nd, 3rd, 4th, and 5th impulses. And on the fifth it would have worked, the fifth χ-wheel having period 23. I suppose I would have been said to have broken the key by pure analytic reasoning. As it was I was thought to have a stroke of undeserved good luck. There must be a moral in this.

A cryptographer might criticize the German χ-patterns which contained too many sequences of 3 or more dots or 3 or more crosses. In the resulting χ-key most of these sequences were stretched out to greater lengths. Hence in a key impulse sequences of five basic symbols were significantly often the corresponding part of the χ-wheel, or the reverse thereof. The critic must point out two grave errors, first the poor Ψ-patterns and second the sending of a long depth of two. Either error without the other the Germans would I

[2] I hope no one is going to ask why I didn't use a computer.

think have got away with. But the two together gave away the structure of the machine.

With our knowledge of the machine we could work on some keys from shorter depths. We discovered that in the past there had been a change of wheel-patterns once a month.

It was early in 1942 that we got an opportunity to attack current traffic. Then a vulnerable depth came in. It was read and yielded about 1000 letters of key. I have a vague memory of a depth of 3 at this time, and this may have been it.

4 Attacks on current traffic

The 1000 letters of new key proved a disappointment. We discovered later that the Germans had corrected their Ψ-error. So the method that had been so successful with HQIBPEXEZMUG did not work.

Then we received a near-depth, two messages whose indicators agreed in all but one letter, that letter corresponding to a χ-wheel. I advocated an attempt to read this even though the reading would have to be from 4 impulses only. However the difference between the two keys would be periodic; after sufficient initial success the messages could be so corrected as to be read as a true depth. It was a very difficult task requiring skilled linguists. Such people existed at BP and some of them tackled the near depth. They read it and got the key, with extra information about the off-set χ-wheel. That was 30 or so possibilities for its pattern only a few being plausible. With that extra information it was possible to analyse the key.

It was Alan Turing who not long afterwards solved the problem of analysing a length of key obtained from an ordinary depth. He would assume the first two symbols in the χ_1 pattern to be •X, or perhaps ••. The possibilities X• and XX are not genuine other choices since the reversal of all χ and Ψ patterns leaves the key unaltered. Suppose he assumed •X. Then at each repetition of that part of the χ_1 pattern one can deduce Ψ_1 as either •X, X•, XX or ••. If one of the last two he provisionally assumed Ψ_1 had not moved. He then got the corresponding doublet (2 possibilities) in the other χ-wheels and repeated it through the key according to the χ-periods. And so on, making as few corrections as were necessary for consistency. It is a method requiring great artistry. I never used it successfully myself. But there were others with whom it worked well enough.

We were reading only those messages that the German operators were careless enough to send in depth or near-depth. That was too few to satisfy Bletchley's customers. We learned however to use known wheel-patterns to break messages not in depth. Basically a commonly occurring "crib" like SPRUCHNUMMER or OBERKOMMANDO9WEHRMACHT would be "dragged", that is tried in one position after another until a plausible stretch of key was obtained. It was plausible if some positions of the χ-wheels gave plausible Ψ patterns, i.e., not too many occurrences of •X• or X • X. In practice at least one χ-wheel had to have a known setting, that is have the same indicator letter as in some message already read. So the more messages that were read, the easier it became to read others.

It was even found possible to break the wheel-patterns for a month from indicators alone, exploiting stereotyped beginnings and information from indicators as to which wheels in which messages had the same setting. I remember trying this method myself, getting some initial success but soon losing control. Then Capt. J. M. Wyllie tried. In civil life he edited the Oxford Latin Dictionary. "This is just the job for a lexicographer" quoth he. And he broke the wheel-patterns for a past month, hitherto untouched.

The method was used. But since it required so many messages it was unlikely to succeed until late in the month. It might be cut short by the breaking of a depth and then its partial information would help in the reading of other messages.

In the second half of 1942, with all this progress we thought we were doing well. And so we went on through 1943.

It could not last. Eventually the Germans, noticing that the indicators were giving away information that need not be given away, abolished 12-letter indicators. Instead they gave a simple number. Presumably the operator looked up the number in a book and found his twelve letters against it.

We could still recognize depths, messages with the same number and, with luck, get wheel-patterns from them. But how to set those wheel-patterns on other messages?

5 The statistical method

The question now was as follows. Given a cipher message $\mathcal{Z}$, and given the corresponding wheel-patterns, how were we to set those wheels so as to decipher the message?

In the ith impulse we have

$$\mathcal{Z}_i = \chi_i + \Psi_i + \mathcal{C}_i. \tag{1}$$

For some purposes it is desirable to replace $\mathcal{Z}_i$ by $\Delta\mathcal{Z}_i$. The nth symbol of $\Delta\mathcal{Z}_i$ is the sum of the nth and $(n+1)$th symbols of $\mathcal{Z}_i$. We could call $\Delta\mathcal{Z}_i$ a difference. Addition and subtraction are the same in this arithmetic. Similarly for χ_i, Ψ_i and $\mathcal{C}_i$. We note that $\Delta\chi_i$ has the period of χ_i. Also $\Delta\Psi_i$ is zero whenever Ψ stays still.

Let us write (1) with $i = 1$ and then with $i = 2$ and let us add the two equations

$$(\Delta\mathcal{Z}_1 + \Delta\mathcal{Z}_2) = (\Delta\chi_1 + \Delta\chi_2) + (\Delta\Psi_1 + \Delta\Psi_2) + (\Delta\mathcal{C}_1 + \Delta\mathcal{C}_2). \tag{2}$$

I derived this equation because I suspected that $\Delta\Psi_1 + \Delta\Psi_2$ would be mostly dot. It is always so when Ψ stays still and sometimes so when χ does not. I calculated that it would be about 70% dot. Note that $(\Delta\chi_1+\Delta\chi_2)$ has period $41 \times 31 = 1271$. $\Delta\mathcal{C}_1 + \Delta\mathcal{C}_2$, constructed from a military German message, was expected to be 60% dot or a little more. I concluded that $\Delta\mathcal{Z}_1 + \Delta\mathcal{Z}_2$ agreed more often that not with $\Delta\chi_1 + \Delta\chi_2$. In favourable cases there might be as much as 55% agreement. It seemed that to set χ_1 and χ_2 we should try $\Delta\chi_1 + \Delta\chi_2$ against $\Delta\mathcal{Z}_1 + \Delta\mathcal{Z}_2$ in all the 1271 possible relative positions and pick the one with best agreement.

Extensions of the method would set the other χ-wheels.

I remember explaining the method to Gerry Morgan and Max Newman. There were rapid developments. Post Office engineers in consultation with applied mathematicians mechanized the process using first electrical relays and then vacuum tubes. This was the way to Bletchley's pioneering electronic computer "Colossus". In those days telephones and the associated engineering problems were the responsibility of the Post Office.

Soon χ-wheels were being set on current messages and Bletchleyites spoke of the process of "dechiing".

After dechiing there remained the sum $\mathcal{C} + \Psi$. Since $\mathcal{C}$ and Ψ each had its peculiarities this could be broken somewhat in the manner of a depth. Or you could say it was the old process of "dragging", simplified by all the χ wheels having been set. The process was called "depsiing". Or, when unsuccessful "deep sighing". This process was carried out by hand, mostly by members of the W.R.N.S.

It occurred to me that with a sufficiently long message this statistical method might be strengthened so as to find the unknown wheel-patterns. One day, having received a message 15,000 letters long I tried out the idea. It worked. I remember reporting to Major Tester, head of the appropriate Section (known as the Testery) with the news of "the first machine to be broken on a depth of one".

Statisticians at BP, notably Jack Good greatly strengthened the method and the famous Colossus computers were programmed to apply it. Now depsiing became a more tricky process being done with initially unknown psi patterns, which had to be determined in the process. For this work would normally be done in the absence of any helpful depth.

Meanwhile the Germans began to change the wheel-patterns every day and to make the Ψ movement depend partially on the χ-wheels, or even on the past plain text. But production at Bletchley continued up to the end of the European War.

It appears from the work at Bletchley that the main weakness of the Tunny machine was that the five Ψ-wheels kept in step. They either all moved on one place or they all stayed still. Turing's method and the statistical method all depended on this, and so did dragging in the days of indicators. The mere fact that the cipher was additive was also a weakness since depths could be read however subtle the machine. I suppose there was a switch allowing an effortless return to the initial position. With such as switch it would have been hard to avoid sending an occasional depth, especially in times of emergency, strain and overwork.

There was another teleprinter cipher machine that we called "Sturgeon", used by the Luftwaffe. It mixed the five impulses more thoroughly than did Tunny. There was a permutation of the five impulses in the course of key construction.

"Code Breakers" explains why the authorities decided to concentrate on Army Tunny rather than Air Force Sturgeon. One reason: resources were limited and it seemed better to make a full scale attack on one cipher system than to make half-hearted attacks on both. Another reason: Enigma was supplying much information about the German Navy and Air Force but little about the Army.

To which I might add that though we found out how Sturgeon worked we failed to think of a way to apply that knowledge to the reading of messages.

References

[HS] F.H. Hinsley and Alan Stripp (eds.) **Code Breakers: The Inside Story of Bletchley Park**, Oxford University Press, 1993.

Sturgeon, The FISH BP Never Really Caught

Frode Weierud*

CERN, Div. SL, CH-1211 Geneva 23, Switzerland

1 Introduction

The German armed forces employed three different types of teleprinter cipher machines during the Second World War, the Lorenz machines SZ40 and SZ42 also called Tunny by Bletchley Park (BP), the Siemens *Schlüsselfernschreibmaschine* (SFM) T52, and the one-time-tape machine T43, also manufactured by Siemens. The Lorenz machines, which existed in three different models, SZ40, SZ42a, and SZ42b, are well known as the machines that were broken at BP with the aid of Colossus. The Siemens T52 existed in four functionally distinct models, T52a/b, T52c and T52ca — which was a modified version of the T52c machine, T52d, and T52e, all going under the BP code name of Sturgeon, while the Siemens T43 probably was the unbreakable machine that BP called Thrasher. The T43 machine came into use relatively late in the war and appears to have been used only on a few selected circuits.

This paper will, for the first time in the open literature, explain in detail the events that led to BP breaking the Sturgeon machines. In 1964, the Swedish Under-Secretary of State Erik Boheman first revealed that Sweden had broken the German Geheimschreiber (T52) during the Second World War. [4] In 1967, David Kahn gave further details about this achievement. [16] However, it was only in 1984, when Hinsley et al. published part one of the third volume of "British Intelligence in the Second World War," that it was officially acknowledged that BP also had experienced some success against the Siemens T52. [13] Previously, many authors had confused the T52 with the Lorenz SZ40/42 machines and had erroneously linked the Siemens T52 to Colossus. Since 1982, Donald Davies has published detailed information about the electrical and mechanical construction of the machines. [6–8] And Wolfgang Mache has through his contacts and interviews with former Geheimschreiber operators and technicians presented the evolutionary history of the Siemens T52 machines. [17–19] Apart from Sir Harry Hinsley's and Professor Tutte's [22] references to BP's attack against the T52 there has so far not been any detailed account of this part of BP's history. It is hoped the present paper will fill this void.

The first section of this paper gives a short overview of the German teleprinter cipher machines and their use, followed by a short section explaining how and when BP first encountered the Sturgeon traffic. The third

* This article represents the views of the author but not necessarily those of his employer or any other third party.

section explains the cryptographic principle used by the Siemens T52 machines. Here, for the first time, the "Pentagon" is introduced and an explanation is given of how important this device was for BP's attack against the first T52 model it encountered. The following two sections continues the historical presentation of BP's attack on the T52 and its struggle to keep abreast with the German cryptographers continuous changes to the machines. For the first time, it is revealed that BP broke the T52d, a machine with irregular code wheel movement. This was indeed a major achievement. Sections seven and eight explain what knowledge BP gained from the captured machines and the information they acquired through both FISH and Enigma decodes. The section entitled The Cryptanalytical Problem gives new and detailed cryptanalytical information about the structure of the T52 key generators and how this information was used to attack the machines. A constructed example of how to perform an attack on T52 messages in depth[1] concludes this section.

2 The Machines and Their Use

All the German teleprinter cipher machines were on-line machines. This means that when an operator types his plain text message on the transmitting machine, A, the same plain text appears immediately on the receiving machine, B. Neither of the operators ever sees the cipher text. The Lorenz machines were from their inception designed to be suitable for use on high frequency radio circuits operating in the 3 to 30 MHz bands. Radio signals in this frequency range are affected by both slow and fast fading, Doppler shift and multipath propagation which can easily play havoc with the digital teleprinter signals. All these machines used the standard teleprinter speed of that time, 50 Baud, which results in an element time of 20 ms. They were asynchronous machines using a start and stop pulse for each transmitted character. The SZ40/42 machines had a better receiver design than the T52 and were therefore more successful in reconstituting severely distorted teleprinter pulses. Towards the end of the war Lorenz worked on the development of an improved machine, the SZ42c, which applied the cryptographic process directly to the radio signal itself.[2] It was used in conjunction with a continuously operating, synchronous teleprinter which maintained its speed with the help of a crystal controlled oscillator. The SZ42c was an advanced design and the German engineers were clearly leading in this area.

It may therefore seem that technical reasons led to the Lorenz machines being used on radio teleprinter circuits. However, the author believes that lo-

[1] Two or more cipher texts or messages are said to be in depth when the texts have been aligned such that the entire texts or parts thereof have been enciphered by the same key. This process, that messages are enciphered by the same key, can occur when a cipher machine or system is used incorrectly or from the use of keys that have been constructed wrongly.

[2] "European Axis Signal Intelligence in World War II – Vol.2", 1 May 1946, A TICOM Publication released under the US Freedom of Information Act (FOIA).

gistics are more likely to have been the reason. The Lorenz SZ40/42 machines were a German Army development, while from an early stage the T52 machines were adopted by the Air Force and the Navy. The T52 machines were only allowed to remain on board naval ships while they were in harbour. It is evident that they would mainly be connected to the well developed telegraph line network which covered the most of German occupied territory. This was also the situation for the machines used by the German Air Force. On the other hand, a large part of the German Army tended to be continually on the move and it was relatively seldom that they could connect their machines to the fixed telegraph network. With time the T52 machines also appeared on radio circuits. Initially they were used on radio relay connections using frequencies in the VHF and UHF range, while later they would also appear on circuits in the HF (3–30 MHz) area.

3 The First Encounter

BP first observed Siemens T52 traffic in the summer and autumn of 1942. Most of the traffic passed on a radio link between Sicily and Libya, which BP called the "Sturgeon" link. [1] In the same period there was also another link from the Aegean to Sicily that BP called "Mackerel". The operators on these links were in the habit of sending a large number of cipher text messages using the same machine settings. When using the machine, they sent a short cipher text, followed by some operator chat in clear text. They then transmitted in clear the signal "UM UM" (*Umschalten* — switch over) and the cipher text continued but with the machine set to its initial setting. These interruptions and operator exchanges were frequent and the cipher texts in depth would continue to accumulate. The depths allowed the BP cryptanalysts to analyse the machine in detail and they soon discovered that the machine had 10 code wheels whose patterns appeared to be fixed. At least that was their assumption based on the intercepts during September and October and the first two days of November. After that, the Sturgeon link and its traffic came to an end. In the period before September the interception was too bad to allow any of the traffic to be read.

4 The Cryptographic Principle

The analysis of the intercepts showed that the Sturgeon machine was using two operations, a modulo two addition (XOR) and a permutation of the resulting five teleprinter code elements. The modulo two key was called the *subtractor* and represented by the symbol $\varSigma$, while the permutation key was called the *permutor* and represented by $\varPi$. The cryptographic algorithm, transforming a plain text character P into its cipher text character C, is given by

$$C = \varPi(P \oplus \varSigma) \tag{1}$$

where ⊕ signifies modulo two addition. The plain text character is first added to the subtractor modulo two and the permutor then permutes the result. On reception the inverse permutation took place before the addition of the subtractor, which gives

$$P = C\Pi^{-1} \oplus \Sigma \tag{2}$$

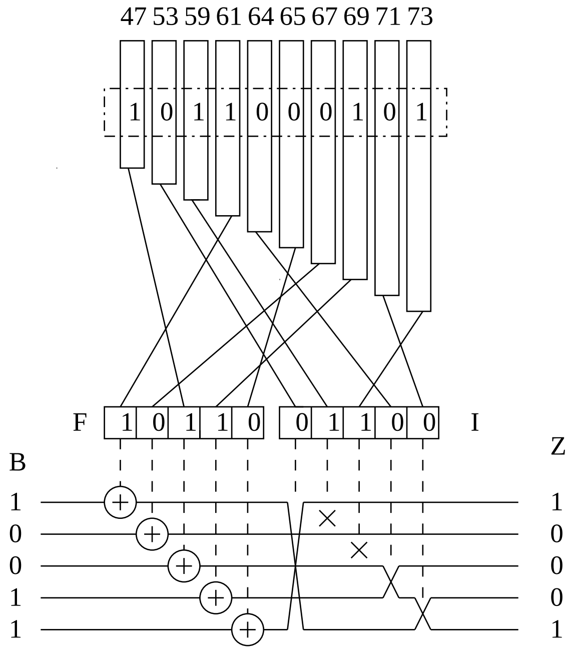


Figure1. SFM T52's functional diagram.

A schematic diagram of the basic operations of all the T52 machines is given in Fig. 1. The ten rectangles of varying heights symbolize the ten code wheels whose circumference carried bit patterns of different lengths. The wheels were bakelite disks with protrusions which were sensed by one or more electrical contacts. A more modern analogy for the code wheels is shift register sequences of different lengths. In Fig. 1 the length of the code wheel sequences is written above each of the ten wheels. The code wheels were labelled A to K from right to left, omitting I. These wheel identities are used later in Fig. 4 which gives a description of the wheel stopping logic for the T52d machine.

Below the wheels, the plug connections that make up the main inner key are shown connecting each of the ten wheels to the various elements of the XOR and transposition circuits. The figure is an accurate representation of the functioning of the T52a/b and T52d machines. In these two models each code wheel consisted of four identical cams, each fitted with a changeover contact which was used in either the XOR circuits or the transposition circuits of the transmitter and receiver part of the machine. The plugs connected to the code wheel contacts were labelled with the corresponding wheel identities A to K, each wheel being equipped with two plugs, one red and the other black. The corresponding sockets in the transposition circuit, ten in total, were labelled from 1 to 10. Sockets 1 and 2, 3 and 4, etc. were paired together but we will see later that any of the two plugs of a given wheel can be plugged into any of the transposition sockets. Further, the red/black order had no electrical significance and the two plugs could be swapped. The ten sockets in the XOR circuit were labelled with Roman numerals from I to V in pairs, where each socket in a pair carried an additional a or b label, e.g. sockets Ia and Ib. For the XOR circuit the plug order had to be strictly adhered to and the two plugs of a given wheel had to be plugged to the sockets with the same Roman numeral pair, e.g. red K would plug to IIa and black K to IIb. If the plugs of a given wheel were connected to two different Roman socket pairs a short circuit of the ± 60 volt signalling supply would be the result.

The T52c/ca and T52e machines modified this relatively complex circuit by using relays with multiple contact sets for the functions in the XOR and transposition circuits. These so-called SR relays were controlled via a logic circuit driven by the cam contacts on four different code wheels. On these machines the code wheels had one single cam on each wheel; the other three cams became superfluous and were therefore removed. The relays SR1–SR5 were used in the permutation circuit, while SR6–SR10 made up the substitution circuit. The machines also did away with the flexible transposition circuit of the T52a/b and d models which allowed full freedom in the configuration of the circuit as will be explained later. The T52c/ca and T52e machines used a standard configuration of the transposition units which were wired permanently in place.

Instead of changing wheel order by plugging, these machines used ten switches, one for each wheel, which could be set to one of ten positions labelled $1, 3, 5, 7, 9, \mathrm{I}, \mathrm{II}, \mathrm{II}, \mathrm{IV}$, and V. There were no longer any pairs of plugs and sockets such that the previous paired designations, e.g. 1–2 and IIa–IIb would be represented simply by respectively 1 and II. The ten outputs from the wheel order selection circuit carried the same labels as the switch positions; here the outputs are called the output channels. Any of the ten wheels could be connected to any of the ten output channels via the ten switches with the restriction that a given output channel could only be selected once. If this rule was not obeyed a short circuit of the ± 60V supply would occur. Furthermore, the labels had lost their previous meaning of Arabic numerals belonging to the transposition circuit and the Roman numerals belonging to the XOR circuit. Instead the three machines, T52c, T52ca and T52e, controlled each of the SR relays via a wheel combination logic which consisted of the modulo two sum of four different output channels. The wheel combination logic for the T52c has previously been published by Donald Davies in his paper on the T52 machines [7] and is reproduced here in Fig. 2. The wheel combination logic was different on each of the three machines. The logic for the T52e machine has also been published by Donald Davies in his paper on the T52e machine, [6] while the logic for the T52ca machine will be presented later. The information in Fig. 2 has also been compared with information from the archives of the Swedish signal intelligence organization, FRA,[3] and found to be correct.

Relays		Code Wheel Outputs									
		1	3	5	7	9	I	II	III	IV	V
Permutor	SR1	X	X				X		X		
	SR2		X	X				X		X	
	SR3			X	X				X		X
	SR4				X	X	X			X	
	SR5	X				X		X			X
Subtractor	SR6			X	X			X		X	
	SR7		X	X					X		X
	SR8	X	X					X			X
	SR9	X				X	X		X		
	SR10				X	X	X			X	

Figure2. Wheel combination logic for T52c.

The T52c and T52ca machines introduced yet another complexity, the message key unit. This unit, which consisted of 15 transposition units and which will be introduced later, was connected between the code wheel cam contacts and the wheel order selection circuit. Its function was to further permute the order of the wheels before their contacts were selected in the wheel order selection circuit which was the main inner key. As explained later, a new setting of the message key unit would be selected for each new message. This meant that even if the main inner key would remain the same the wheels would still have a different function for each new message.

The T52d and e models also had irregular movement of the code wheels, a so-called stop-and-go movement. The movement of each wheel was controlled by contacts on two of the other wheels. These two machines also had a switchable autokey[4] element where the third bit of each plain text character would control the movement of the wheels in addition to the control given by the wheels themselves.

[3] FRA, Försvarets Radioanstalt. See [3, 23].

[4] Autokey or autoclave is where a part of the key is generated from the plain text or the cipher text.

Here is how the machine works in an example as shown in Fig. 1. First, a plain text character, B say, will be represented by its Baudot code equivalent 10011 or ×••××[5] as given in the Baudot alphabet in Fig. 5. The plain text character B is then added bitwise modulo two to the subtractor character, F say, and the result routed through the transposition circuit, which is controlled by the permutor character, I say. The resulting cipher text character is Z. The two key characters, F and I, are determined from the code wheel setting and the inner key configuration once the plain text character B enters the machine. In addition, the figure shows that an element of the transposition circuit, the transposition unit, is active when the controlling bit is 0 or, as BP said, a dot.

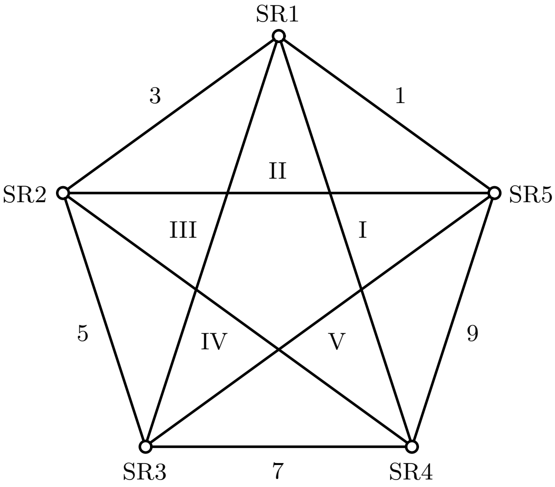


Figure3. The Pentagon

The analysis of the T52 key generator showed that the 10 code wheels were combined in fours. They named this circuit the "Pentagon". The author has not been able to find any documentary information about the Pentagon, however, largely inspired by Professor William Tutte's beautiful little book on graph theory, [21] he thinks he has found the answer.

The graph in Fig. 3 is constructed from the wheel combining logic in Fig. 2. The code wheel output channels are labelled $1, 3, 5, 7, 9$ and $\mathrm{I}, \mathrm{II}, \mathrm{III}, \mathrm{IV}, \mathrm{V}$. A cross in the row for one of the SR relays means that the control of the relay depends on the marked output channels, e.g. the function for the SR4 relay is given by

$$\mathrm{SR4} = 7 \oplus 9 \oplus \mathrm{I} \oplus \mathrm{IV} \tag{3}$$

[5] BP used the terms cross and dot to describe the Baudot code elements mark and space, logical 1 and 0.

In the graph in Fig. 3 the SR relays are represented by the vertices and the controlling wheel output channels by the edges which join in a given vertex. The advantage of the graph is that it quickly shows the relationship between the different SR relays; it clearly shows the topology of the circuit. The symmetry of the graph is such that it is highly likely that it corresponds to what BP called the Pentagon.

The Pentagon was cryptographically a weak device. Only four different subtractors could be associated with a given permutation. Furthermore, the subtractor character was always even, i.e. the 5 code impulses always summed to zero. [1] Therefore the plain text character was even whenever the cipher text character was even, and odd whenever the cipher text character was odd. For the cryptanalyst this was similar to the Enigma's peculiarity that no letter can encipher to itself, and it was of great help in reading depths and placing cribs.

The first Sturgeon message to be read was at a depth of 40, an almost incredible depth, which clearly shows that the German operators had no idea of the detailed functioning of the machine and that they must have disobeyed orders or been wrongly instructed. Eventually, with the detailed knowledge of the limitations imposed by the Pentagon device, depths of four or five could be read fairly easy. The 10 code wheels were set once a day and this initial setting remained in force during the whole day. However, the machine was equipped with a small crank which allowed the operator to easily bring the machine back to its initial code wheel settings. This was the main reason for the large number of messages in depth. With this knowledge, it was possible to read messages at depths of two or three as soon as the daily wheel settings had been recovered. When they could make a guess at a crib of about six letters even single messages could be broken with the help of the Pentagon limitations.

The different messages were sent using different wheel orders. There was some form of message key device that changed the connections between the code wheels and the Pentagon. However, as the machine was brought back to its initial position, the binary streams from each of the wheels were always the same. Five letters were given as a message key, and these always came from the reduced alphabet: P S T U W X Y Z. A letter could appear more than once in the group of five — once the indicator WWWWW was even observed. BP noticed that when two indicators agreed in n positions, then usually but not always, $2n$ of the wheels had the same function in the Pentagon. However, this rule did not apply to indicators sent on different days. The indicator system of this machine was never broken cryptanalytically.

Comparing the above description with what is known about the different Siemens T52 models it is evident that BP was confronted with the T52c machine. [6–8, 17–19] This machine had a code wheel combination logic like the one described for the Pentagon. It also had a message key unit with five levers that could be set in eight different positions indicated by the letters P S T U W X Y Z. Like the T52a/b, the c model also had the small crank that allowed the code wheels to be brought back to an index position. This was a conceptual error in this model as the main reason for this wheel resetting mechanism was to allow the operator to set the message key easily on the wheels. The T52a/b machines were not equipped with a message key unit like the T52c and therefore the message key was set directly on five of the code wheels. The 10 wheels were therefore brought back to the initial position and the five wheels selected as message key wheels would be set to their new position. It is debatable whether even this limited wheel resetting on the T52a/b was a good idea. However, it is evident that the complete wheel resetting used on the T52c machines was a blunder of some magnitude.

The Sturgeon and Mackerel links came to an end with the second battle of El Alamein which started at the end of October 1942. One other signal transmitted on a T52c machine was intercepted later in November. It was believed to have come from the Caucasus. It consisted of the usual messages in depth and was successfully attacked. The messages dealt with the situation on the Russian front. That was the last appearance of traffic from a T52c machine.

5 The Reappearance

In the first six months of 1943 other teleprinter links appeared which also used "UM UM". Some of the links were known to use the Tunny machine and from this moment it was often difficult to distinguish between links using the two machines. Both types of link gave only a QEP number for the indicator. The only exception to this rule was a link named Salmon where some groups of letters were sent, apparently as indicators. They were quite different from the normal Sturgeon indicator groups. Messages on Salmon, which linked Königsberg and Mariupol, were intercepted from 11 January to 6 February 1943. The machine was of a much simpler construction than the Pentagon machine and there was no combination of the wheels. Five of the wheels made up the subtractor key while the other five wheels constituted the permutor key. The messages consisted mainly of operator chat.

Even though the new machine was simpler than the Pentagon machine (T52c), it was more difficult to break. The absence of the Pentagon meant that the parity of the cipher letter was no longer the same as the parity of the corresponding plain text letter. And instead of having only 60 different alphabets this new machine had 960. From this description it is evident that the machine must have been the T52a/b.

In May 1943, a new link, codenamed Sardine by BP, started to operate between Sicily and Sardinia. This link was never broken. Later in the year, two operator log books were captured which contained references to the intercepted traffic on the Sardine link. Time, numbers and priority codes corresponded to those of the intercepted traffic. Also the same type *Luftwaffe* addresses that had earlier been used on Sturgeon appeared on this link.

A new link codenamed Halibut by BP appeared in July 1943. The link, which operated between Königsberg and Munich,[6] ceased to operate in August but reappeared in a changed form in 1944. In its first period, from July to August, a few depths of four and one of five were found. One depth of four from August was read and was found to have been enciphered in the same way as the depths that had earlier appeared on Salmon (T52a/b). Like the Salmon messages it consisted of operator chat. However, the July depth of five resisted all attempts to break it. It only succumbed a year later, in June 1944, to a sustained attack. It then turned out that it was enciphered on a new machine, the T52d.

6 A Historic Achievement

This break constituted the first break of the T52d machine, a machine similar in construction to the T52a/b but with irregular, stop-and-go, code wheel movements. The Halibut message did not use the autokey element, *Klartextfunktion*, of this machine but in June 1944 other Sturgeon links were suspected of using this machine with the autokey function. The break was nevertheless an outstanding achievement. The T52d was completely broken from reading a depth of five for a part of the message, while for the remainder it was only a depth of four. [12] From BP's subsequent analysis of the machine a depth of four appeared to be the absolute minimum. How was it possible to break such a complicated machine from only one message in depth of four and five? One answer is that BP was not confronted with a completely new machine. It was mainly the stop-and-go code wheel movements which differentiated this machine from the T52a/b. The code wheels themselves had the same patterns as on the T52a/b and T52c machines. It would turn out later that all the machines in the T52 series used the same code wheel patterns. The patterns were fixed and no changes were ever made to them. This constituted a very serious weakness of these machines.

The break itself was a manual operation, but assisted by a large number of catalogues which showed the possible alphabets that resulted from an assumption of a plain-cipher text pair of characters. BP did not develop a machine to assist in deciphering. All the operations were done by hand so that even developing the subtractor and permutor keys from a given wheel order and setting was a very slow and tedious process. BP also tried to use masks and inverse probability calculations, but it is not known if this was successful. As will be shown later, the permutation circuit only produced 30 out of the 120 possible permutations. Thirty-two permutations should have been possible with the five double changeover contacts used for the

[6] A list of FISH links in one of the Fried reports gives the link as operating between Memel and Königsberg. [10] However, as the distance between Memel (Klaipeda) and Königsberg is only 120 km, mainly over water, the use of an HF link does not sound right.

permutation function, but / and Z produced identical permutations, as did T and E.[7]

The break was a success, but it also showed the difficulty this machine presented cryptanalytically. BP launched a substantial research effort to understand the T52d machine fully and to explore possible cryptanalytical attacks against it. BP realised that solutions through depths could not be relied upon in the future because of the increasing use of the autokey function. Another problem that presented itself was how to differentiate between this traffic and ordinary Fish traffic generated by the Lorenz SZ40/42 machines. BP hoped to find statistical techniques that would allow it to identify the traffic.

Wheel		Controlled by
ID	Length	
K	47	E crosses, D dots[a]
J	53	K crosses, A dots
H	59	K dots, J crosses
G	61	J dots, H dots
F	64	H crosses, G crosses
E	65	G dots, F crosses
D	67	F dots, E dots
C	69	F dots, E dots
B	71	F dots, E dots
A	73	F dots, E dots

[a] Dot and cross are BP parlance for 0 and 1, space and mark.

Figure4. Wheel stopping logic for T52d.

It is not known how long the July 1943 message was but it is nevertheless an extraordinary feat to have fully deduced the “motor wheel” logic of the T52d. In contrast with the Lorenz SZ40/42, the T52d did not have separate “motor wheels.” Instead, each “motor” was formed by the modulo two addition of two other wheels, sometimes with inverted logic for one or both of the wheels. The “motor” or wheel stopping patterns were read from a different part of the code wheels than those used for the subtractor and permutor keys. And of course the movement of these wheels was again controlled by others. Four of the wheels, with the lengths 73, 71, 69 and 67, were controlled in parallel by two of the other wheels. This was presumably done to ensure a periodicity of at least $73 \cdot 71 \cdot 69 \cdot 67 = 23\,961\,009$. The wheel stopping logic as derived cryptanalytically by BP is given in Fig. 4. [11] The figure shows

[7] BP replaced the six teleprinter control characters *carriage return*, *line feed*, *letter* and *figure shift*, *space*, and *null* with the special characters *3,4,8,+,9,* and */*. See the teleprinter alphabet in Fig. 5 and Appendix A of [23].

how the movement of a given wheel depends on two other wheels, e.g. the K wheel, which is the leftmost wheel in the machine and with a sequence length of 47, will not move if there is a cross (1) on the E wheel and a dot (0) on the D wheel. The other wheels have similar relationships to two other wheels.

The deciphered messages referred to experiments with a machine the operators called T52d, which gave BP the final proof that it had broken a new Sturgeon model. Later two captured T52d machines were found to contain the same logic as had been derived cryptanalytically from the Halibut message.

In September 1943, the link named Conger appeared between Athens and Berlin. Hundreds of messages were sent and all were in depth so there was no great difficulty in reading them. However, their intelligence value was nil. The messages contained only operator chat.

Conger contained references to the T52b, a machine that had previously been captured in Tunisia. By correlating the recovered code wheel sequences with those of the actual machine it was found that the initial position corresponded to that of all wheels set to one. The wheels were used in the order of their periods, while the operation of the machine corresponded to what had earlier been observed on Salmon, and in the August Halibut messages. In November, similar Conger messages in depth were sent; this time the wheels were all set to two.

The description of the Conger usage is frankly amazing and shows a complete disregard for applying secure keying instructions for the machines. It would seem that the machines were used by operators who had never read the instructions and who had not been issued with operational keys for these machines. One also gets the very strong impression that the majority of these links were not operational links, but reserve channels kept open mainly with operator chat and test messages. However, their usage was cryptographically damaging to the machines.

Both Conger and Halibut reappeared early in 1944 in a slightly changed form. The new Halibut messages were all short, while earlier they had often been very long. Conger, on the other hand, often contained long messages. Depths, in this case messages with the same QEP number, of up to four occurred. However, the messages had no repeats, which strongly indicated that the autokey function was being used. This hypothesis was further supported by the intercept logs which contained phrases like "*Mit KTF* " and "*Ohne KTF* " where KTF was the abbreviation for "*Klar Text Funktion*". BP did find one depth of two without the autokey function, but a depth of two was considered to be unbreakable.

Shortly afterwards it was decided to cease the interception of links using the Sturgeon machines as it was considered to be unprofitable. In the autumn of 1944 many Tunny links, which also used an autokey element, ceased to use this function and Enigma messages were found ordering the Sturgeon operators to stop using autokey on the T52d and T52e machines. During the same period, one day's traffic on Conger was intercepted. It was found to be in depth of two and without the autokey function. However, there are no further indications that a lot of effort was invested in the Sturgeon machines and their traffic.

7 The Captures

The first Sturgeon machine to be captured was a T52b which was found in Tunisia. It was discovered that the code wheels on this machine moved regularly and that they did not combine. It was therefore evident to BP that it was not the Pentagon machine (the first Sturgeon type of machine to be intercepted and broken).

Later a full technical description of a machine which combined the functions of the T52a/b and T52c was captured on Elba. It appeared from this description that the T52c machine was related to the Pentagon machine as it combined the code wheels in fours. However, the number of alphabets was found to be 256 instead of 60 as for the Pentagon machine. It will be shown later that this T52c machine was the modified version, T52ca. The T52a/b mode showed that the machine could have been used for the Salmon, August Halibut messages and the early Conger traffic.

The Elba description also showed that the T52c machine was equipped with a wheel permuting mechanism corresponding to the message key unit described earlier. It was found that the unit consisted of five levers each of which controlled three switches out of a set of 15. Each switch interchanged two wheels in its active position and left their order unaffected in the inactive position. A switch was active or inactive depending on the position of the controlling lever, but the correlation of active switch position and lever position was different for the three switches controlled by a given lever. This circuit has been described in Donald Davies' paper on the T52 machines. [7]

In addition, it was found that all the machines were equipped with a set of switches or plugs which constituted the main inner key setting. The switches or plugs selected which of the ten code wheels controlled a given functionality in the cryptographic process. After the capture of the Elba description, an actual machine of this type was captured at Naples. This was clearly a T52c machine, but the message key unit with the five levers had been removed. It was noted that the machine was very similar to the first captured T52b machine; the T52b also had room for a message key unit although none was actually fitted. Yet another machine was captured at Naples. On this machine the original type number, T52b, had been altered to T52d. This machine was equipped with the wheel stopping logic and had a switch to enable or disable the autokey function KTF. Without KTF the code wheels had the same movement as the one derived cryptanalytically from the July Halibut message. When the KTF was active, the wheel movement logic became more symmetrical and the third impulse of the clear text governed part of the logic. Two of the wheels were controlled by a plain text cross (1), while two others were controlled by a dot (0). This logic has also been described in detail by Donald Davies. [6–8]

Later yet another T52d machine was captured, which had been altered from a T52a. Comparing this machine with the T52b, it became obvious that the two models must have been very similar. It is known from German sources that the only real difference between the two machines was that the T52b was fitted with extra filters to reduce interference to radio installations. [18, 20]

Together with the T52c machine description captured at Elba, allied forces also captured two key book pages, one for the T52d, and one for the T52a/b and T52c machine. One side of each page gave the table for 3 June 1944, while on the other side was the table for 4 June. Each table consisted of 25 rows labelled with the letters from A to Z, omitting J. A similar table for the T52d/e machine is reproduced in Appendix A. The message key QEP FF OO PP AA ZZ VV CC MM HH UU corresponded to setting the leftmost code wheel to 19, as can be found in column 1, row F. The wheel to its right is set to 11 as given in column 2, row O etc. The complete code wheel setting for this message key was: 19 11 56 31 59 33 13 46 02 25.

The corresponding table for the T52c machine is reproduced in Appendix B. The same method of indicating the code wheel setting applies to this table, but in addition the lever settings for the message key unit are in the first five columns. The same QEP message as above would give the code wheel settings: 47 23 09 27 34 45 26 09 02 48 here, with the message key levers at: p t p s x. The use of these tables and the method of disguising the code wheel settings that were transmitted as QEP numbers or letters changed several times throughout the war, but the tables themselves largely retained their original structure and layout. The main instructions for the use of teleprinter cipher machines, *Wehrmacht Schlüsselfernschreibvorschrift (SFV)* [9], indicate there were three basic key tables in use, *Fernschreibgrundschlüssel* (main inner key), *Fernschreibwalzenschlüssel* (code wheel key), and *Fernschreibspruchschlüssel* (message key). An example of the *Fernschreibgrundschlüssel* for the T52d is reproduced in Appendix C.

8 Intelligence From Decodes

References to the Sturgeon machines were frequent in both Tunny and Enigma traffic. In 1942 the decodes referred only to the T52a/b and T52c machines. The *Wehrmacht SFV* as referred to above was issued on 1 December 1942 and also refers only to the T52a/b, T52c and SZ40 type of cipher machines. It is therefore very likely that these were the only machines available in 1942. BP also appears to have captured a copy of the *Wehrmacht* instructions some time before November 1944.

On 17 October 1942 a message[8] from C.S.O.[9] *Luftflotte 2* to *Fliegerführer Afrika* mentioned that T52c had inadequate security. It gave orders that

[8] Message on the *Luftwaffe's* Red (the main Air Force) key, 121-2-3, 17/10, 6610. The author has so far not been able to trace any of these messages.

[9] C.S.O. = Chief Signal Officer.

"Secret" and "Secret Commands Only" (probably translation of *Geheime Kommandosache* — Top Secret) are to be enciphered on Enigma before being sent over *Sägefisch* (Sawfish) links.

This message passed between stations served by the Sturgeon link using the T52c machine. Nevertheless, seemingly important messages continued to pass over this link without being previously enciphered on the Enigma. However, many Enigma messages also passed over this link before it ceased operation on 2 November 1942.

This message doubting the security of the T52c stands in contrast with the *Wehrmacht SFV* which contains a clear instruction not to use the T52a/b over radio and radio relay connections (*Richtstrahlverbindungen*). The T52c was the only machine authorized for use over radio and radio relay links. However, we have seen that the *Luftwaffe* for some reason did not obey these instructions and that they used the T52b machine for practice messages on the Salmon, Halibut and Conger links. This shows that *Luftwaffe* cipher officers must have been unaware of the close links and similarities between the different T52 models and that they did not see the danger these practice transmissions were to the other machines.

In February 1943, decodes show that the Germans suddenly had discovered that something was seriously wrong with their *Sägefisch* machines. A message from Madrid to Paris[10] said that the T52 was very badly compromised and that enemy decipherment was possible. "Secret" and "Top Secret" messages were no longer to be sent over the T52.

On 18 February 1943, a new set of instructions for using the T52 machines were issued:[11]

1. The indicator systems in use with the T52a/b and c are cancelled.
2. Henceforth the ten wheel settings are to be given instead and sent on a specific emergency key.
3. A new method of indicating the settings of the five message key levers is to be used.
4. The device for setting back all the wheels to the so-called zero position is to be removed.

Point four of these new instructions shows that the Germans had finally discovered the faulty operator practice of sending many messages on the same key due to the facility for doing so offered by the T52 wheel resetting mechanism. Apparently they also suspected some weakness in the use of the message key procedure and therefore introduced new, temporary measures. They would later abandon the use of QEP numbers and use the QEP structure with ten bigrams that has already been presented in the previous section. It is not clear why this was considered a better procedure but it is possible it

[10] Message on the *Abwehr* link Madrid–Paris, RSS 6713/2/43.

[11] Message on the Army's Bullfinch II (Italy) key, 1735/18/2/43.

offered more flexibility in choosing messages keys than the previous method using QEP numbers.

On 19 February, yet another message[12] gave further instructions:

1. T52a/b is not to be used for "Secret" and "Top Secret" messages, except when other means are not available.
2. If teleprinter links are used there must be previous encipherment on Enigma.
3. After the changes to the T52c, and after a change in the indicator system, "Secret" and "Top Secret" messages may again be forwarded without previous encipherment on Enigma.

In March two messages[13] said that traffic on the *Aptierte* (adapted) T52c no longer needed to be enciphered on the Enigma. From then on there were references to the T52ca, which probably stands for T52c *Aptierte*. Then finally on 14 June 1943 there was a message[14] to the Naval Communications Officer in Sulina and other addressees that said: "On the completion of the adaptation to SFM T52c, the designation T52ca will no longer be used. The designation T52c only is to be used from now on." The changes made to the T52c concerned the wheel combining logic which BP had found to be of such great help when breaking the Pentagon machine. This indicates that the Germans must have made a detailed analysis of the machine and found this part of the logic to be particularly weak.

The knowledge of German security evaluations and analysis of their own cipher machines has not yet been fully declassified and released. It is therefore not yet possible to give a detailed picture of what the Germans knew and suspected with respect to the security of their crypto systems. However, it is known that Dr. Eric Hüttenhain, the chief of the cryptanalytic research section of OKW/Chi (*Oberkommando der Wehrmacht/Chiffrierabteilung*), examined the T52a/b machine in 1939.[15] He found that this machine had an extraordinarily low degree of security and could be broken with about 100 letters of cipher text without a crib. This study could have resulted in the *Wehrmacht SFV* instruction prohibiting the use of the T52a/b on any form of radio channel. However, it is perhaps more likely the discovery by the Germans on 17 June 1942 of the Swedish success in breaking this machine led to the restriction. [23] OKW/Chi suggested changes in the machine, including ways of producing non-uniform code wheel stepping but for engineering reasons Siemens refused to accept these changes. Instead a new machine, the T52c, was produced which overcame some of the more obvious weaknesses of the earlier model. The T52c was studied by the Army cryptanalyst, Doering,

[12] Message on the Army's Merlin (Southern Europe) key, 19/2/43.

[13] Message on the *Luftwaffe's* Red key, Nos. 322/4 and 387/7 of 6 March 1943.

[14] Naval message 14/6/43, 77, Mediterranean.

[15] "European Axis Signal Intelligence in World War II – Vol.3", 1 May 1946, A TICOM Publication released under the FOIA.

from OKH/Gen d Na (*Oberkommando des Heeres/General der Nachrichten Aufklärung*) in 1942. He showed that it could be broken on a text of 1 000 letters. This study was apparently assisted by cryptanalytical machinery in use by OKW/Chi, but it is not known how involved Dr. Hüttenhain and his people were in the actual study and its recommendations. The investigations resulted in the design and production of the T52d. The security analysis of the T52d was continued, mainly by Doering, and early in 1943 he showed that this machine was also insecure. This resulted in the production of the T52e. However, it was known that both the T52d and T52e machines were open to attacks through messages in depth and that at a depth of ten messages could be read without a crib.

However, the cries of alarm from the German cryptographers were not heard, or at least not acted on, by the German Army and Air Force. In the summer of 1942 the totally insecure model T52a/b was still in use and the equally insecure T52c was being distributed. The Army's position was that the teleprinter traffic went over land lines and could not be intercepted, hence there were no need to worry about inadequate security. Evidence of tapping of the teleprinter lines that appeared in Paris in 1942 and 1943 gave the Army a serious jolt and the Army's signal authorities were forced to reconsider their views on teleprinter cipher security. However, it was too late and the newly developed T52e was only slowly being introduced at the end of 1944.

The first reference to the T52d machine appeared in the decodes in October 1943.[16] Subsequently, there were frequent references to all three models, T52 a/b, c, and d. From September 1944 onwards, there were also references to the newly developed machine T52e. Traffic from this machine was never observed or at least identified as such by any of the allied cryptanalytical services and the machine remained unknown to them until the end of the war.

9 The Cryptanalytical Problems

On 29 July 1944 Captain Walter J. Fried, the US Army Signal Security Agency's (SSA)[17] liaison at BP, sent his report No. 68, [12] which he devoted entirely to the Sturgeon problem, to the SSA headquarters at Arlington Hall. He started the report with the following assessment: "The problem of solving current traffic seems completely hopeless. The only feasible method of solving messages enciphered on the T52d machine seems to be through depths. Sometimes the "motor" action is switched off and this gives rise to several

[16] Message on the *Luftwaffe's* Red key, 279/0, 4/10/43.

[17] The agency went through a number of changes in both name and organization during the period 1939–1945. It was named Signal Intelligence Service, Signal Security Division, Signal Security Service, Signal Security Branch, etc. before it was redesignated Signal Security Agency on 1 July 1943, later to be changed to Army Security Agency on 15 September 1945.

possible techniques of solution.[18] For the most part, however, the problems which seem capable of solution are comparatively trivial. The fundamental difficulty of the general problem arises from the fact that that a crib does not yield key."

To give a better feeling for the fundamental cryptanalytical problems I will attempt to give an overview of what is involved in breaking the T52 machines, and how certain features of the machine hampered this task, while other features made it easier for the cryptanalyst. The basic algorithm of the machine has already been explained. To recapitulate, a five element teleprinter plain text character will first be added modulo two to a five element *subtractor* character and then permuted under the control of another five element *permutor* character as given by the encipherment formula (1).

	0	1					2										3										4					5
	/	E	4	9	3	T	A	S	D	Z	I	R	L	N	H	O	U	J	W	F	Y	B	C	P	G	M	K	Q	+	X	V	8
1	●		●	●	●	●					●	●	●	●	●	●							●	●	●	●					●	
2	●	●		●	●	●		●	●	●				●	●	●				●	●	●				●				●		
3	●	●	●		●	●	●		●	●		●	●			●		●	●			●			●				●			
4	●	●	●	●		●	●	●		●	●		●		●		●		●		●			●				●				
5	●	●	●	●	●		●	●	●		●	●		●			●	●		●			●				●					
[a]	#	3	#	#	#	5	-	’	#	+	8	4	)	,	*	9	7	#	2	*	6	?	:	0	*	.	(	1	#	/	=	#

[a] In the figure shift row control characters and other special functions are marked with #, while the national special characters are marked with *.

Figure5. International Telegraph Alphabet No. 2 in class order

A simple way of representing the relationship between the four elements P, C, Σ and Π is through a $32 \times 32 \times 32$ cube. One of the elements P, C or Σ can be placed in the cube and the other three elements along the three axes. Π cannot be placed inside the cube as it is not uniquely defined by P, C and Σ. The cube can then be cut by planes along any of the axes and it will then be represented by 32 squares slices each of the size $32 \times 32 \times 1$. The choice of the representation will entirely depend on the problem to be solved. It is now easily seen that a plain text character from the 32 element teleprinter alphabet will be transformed into a cipher text character through $32 \cdot 32 = 1024$ cipher alphabets. However, this theoretical limit was seldom achieved in practice. If we analyse the basic permutation circuit used in the T52c and T52e machines we will find that / and Z produce identical permutations, as do T and E. This means that, instead of producing 32 permutations, the

[18] The author's studies of the T52d and e models have not revealed any possibility of switching off the "motor" or wheel stopping function on these machines. It is more likely the observed absence of wheel stopping was due to the use of the T52a/b machine.

circuit only generate 30 unique permutations. Therefore these machines only have $32 \cdot 30 = 960$ cipher alphabets. However, this was only achieved in the T52e. In the T52c and T52ca machines the wheel combination logic reduced the number of cipher alphabets even further.

	Subtractor																															
Permutor	/	E	4	9	3	T	A	S	D	Z	I	R	L	N	H	O	U	J	W	F	Y	B	C	P	G	M	K	Q	+	X	V	8
/	★						★							★													★					
E																																
4																																
9																																
3																																
T																																
A								★	★		★	★																				
S								★	★		★	★																				
D															★	★												★	★			
Z	★						★							★													★					
I	★						★							★													★					
R										★			★																	★	★	
L								★	★		★	★																				
N										★			★																	★	★	
H								★	★		★	★																				
O															★	★												★	★			
U																																
J																																
W																																
F																																
Y																																
B																																
C																																
P																																
G																																
M																																
K															★	★												★	★			
Q	★						★							★													★					
+										★			★																	★	★	
X										★			★																	★	★	
V															★	★												★	★			
8																																

Figure6. Alphabet distribution for T52c.

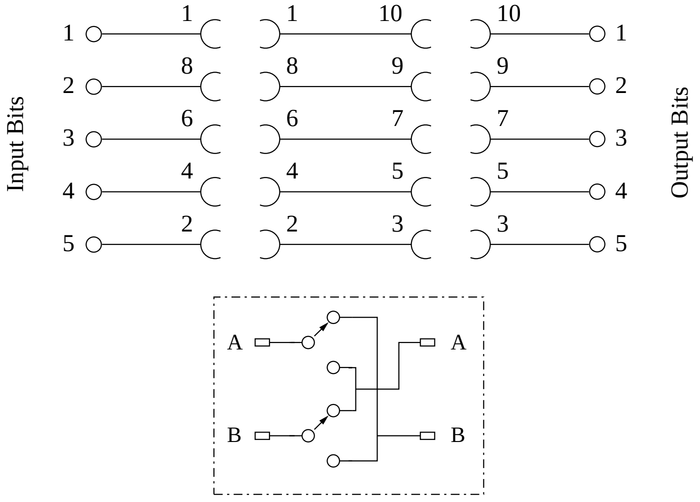
Before we use the cipher squares in our analysis it is useful to introduce the concept of Baudot classes. The class of a Baudot character is defined as the number of crosses (or 1's) that it contains. It is clear that we have six classes labelled from 0 to 5 inclusive. There are various ways of arranging these classes but the method used here is the one used at BP, and is shown in Fig. 5. The Baudot classes are indicated in the top row with the letter shift alphabet used by BP in the row below. The Baudot control characters have been given the special BP values as previously indicated in footnote 7 on page 11. Below the alphabet are the five bits of each character's Baudot code value indicated by dots and crosses. The bottom row shows the corresponding figure shift characters.

Using computer simulations, the T52c's wheel combination logic has been analysed: a plot of the 32 × 32 permutor/subtractor square is given in Fig. 6. The alphabets along the permutor and subtractor axes are in the Baudot class order: an asterisk indicates the existence of an alphabet. We see that there are no alphabets in the odd classes 1, 3 and 5. All the alphabets are clustered in the even classes 0, 2 and 4. This is a confirmation of BP's finding that the parity of the subtractor character was always even. We further see that there are $16 \cdot 4 = 64$ alphabets which, with our knowledge of the reduced permutor alphabet, gives a total number of 60 cipher alphabets. As the parity of the characters T and E is odd, the doublet T–E is not possible. Only the doublet /–Z exists, hence we get $15 \cdot 4 = 60$ cipher alphabets. We also see that for each permutor character there are only four possible subtractor characters as mentioned by BP. The plot clearly shows that this machine was extremely insecure.

Relays		Code Wheel Outputs									
		1	3	5	7	9	I	II	III	IV	V
Permutor	SR1				X	X	X			X	
	SR2	X	X	X	X						
	SR3				X		X	X			X
	SR4	X				X		X		X	
	SR5	X				X			X		X
Subtractor	SR6	X	X					X			X
	SR7						X	X	X	X	
	SR8		X	X		X				X	
	SR9			X	X		X		X		
	SR10		X	X					X		X

Figure7. Wheel combination logic for T52ca.

The wheel combining logic of the modified T52ca machine has been reconstructed using data from the FRA archives. The truth table is given in Fig. 7 while the corresponding permutor/subtractor plot is in Fig. 9. In the plot in Fig. 9 the alphabets are in the binary order, not the Baudot class order, since such a representation shows more clearly the inherent structure of the wheel combining logic. As we can see, the alphabets are well spread out and are no longer exclusively of even parity. However, the linear structure is there and changing one single entry in the truth table will drastically change both the structure and number of possible alphabets. Each permutor character is associated with eight subtractor characters, which is twice as many as for the T52c logic. However, if we plot the permutor/subtractor square in Baudot class order, we find that when a permutor character is even, the alphabets have an even subtractor character, and when the permutor character is odd, so is the subtractor. This information can still be exploited by the cryptanalyst. The possible number of alphabets is $32 \cdot 8 = 256$ but, due to the reduced permutor alphabet, there are only 240 unique cipher alphabets.



Permutation Unit

Figure8. SFM T52's transposition circuit.

The T52a/b and T52d machines use the same layout of the transposition[19] circuit as the T52c and T52e, but instead of using relays for the transposition units, these machines directly use the cam contacts on each coding wheel. What distinguishes the a/b and d models from the others is that the transposition units, which consisted of double changeover contacts, were not wired permanently into the transposition circuit. Each of the five contact sets was equipped with two plug connections which were then plugged into the transposition circuit. Figure 8 shows the layout of the transposition circuit together with the circuit of a single transposition unit. The figure shows that there are two possible contact points in each Baudot bit or element branch.

The connection 1–3 means that either the A or B plug of a transposition unit will connect to the socket marked with 1's, while the other plug will go to the socket marked with 3's. If A goes to socket one, the left part of the A plug will plug into the left-hand side of socket one, while the right part of the A plug goes to the right-hand side of the socket. In this particular case, bit one will end up in position five when the transposition unit is inactive, while

[19] The terms transposition circuit and transposition unit reflect the cryptographic usage; mathematically speaking the circuit performs a permutation.

		Subtractor																															
		/	E	4	A	9	S	I	U	3	D	R	J	N	F	C	K	T	Z	L	W	H	Y	P	Q	O	B	G	+	M	X	V	8
Permutor	/	★						★			★						★		★						★	★						★	
	E		★						★	★						★		★						★			★						★
	4			★		★							★		★						★		★					★		★			
	A				★		★					★		★						★		★							★		★		
	9			★		★							★		★						★		★					★		★			
	S				★		★					★		★						★		★							★		★		
	I	★						★			★						★		★						★	★						★	
	U		★						★	★						★		★						★			★						★
	3		★						★	★						★		★						★			★						★
	D	★						★			★						★		★						★	★						★	
	R				★		★					★		★						★		★							★		★		
	J			★		★							★		★						★		★					★		★			
	N				★		★					★		★						★		★							★		★		
	F			★		★							★		★						★		★					★		★			
	C		★						★	★						★		★						★			★						★
	K	★						★			★						★		★						★	★						★	
	T			★		★							★		★						★		★					★		★			
	Z				★		★					★		★						★		★							★		★		
	L	★						★			★						★		★						★	★						★	
	W		★						★	★						★		★						★			★						★
	H	★						★			★						★		★						★	★						★	
	Y		★						★	★						★		★						★			★						★
	P			★		★							★		★						★		★					★		★			
	Q				★		★					★		★						★		★							★		★		
	O				★		★					★		★						★		★							★		★		
	B			★		★							★		★						★		★					★		★			
	G		★						★	★						★		★						★			★						★
	+	★						★			★						★		★						★	★						★	
	M		★						★	★						★		★						★			★						★
	X	★						★			★						★		★						★	★						★	
	V				★		★					★		★						★		★							★		★		
	8			★		★							★		★						★		★					★		★			

Figure9. Alphabet distribution for T52ca.

in the active position bit one will leave on the branch connected to socket ten. Its final position will depend on the connection that is made from socket ten.

There are $9 \cdot 7 \cdot 5 \cdot 3 \cdot 1 = 945$ different ways that the five contact sets can be inserted into the transposition circuit. Computer simulations show that each of these 945 connection variants results in unique permutation sets. However, the majority of the permutation sets, a total of 561, are degenerate in the sense that each set contains only from 1 to 16 unique permutations.

The set with only one single permutation is a special case — it contains the identity permutation, hence no transposition takes place. There are further variants on this where one, two or three of the Baudot character pulses will not be permuted. There are in total 300 cases where one pulse remains in place, 80 cases where two pulses are fixed and 20 instances where three pulses are unaffected. All of these cases belong to the set of the degenerate permutations. Figure 10 gives an overview of the distribution of the different permutation sets. The figure shows that among the remaining 384 permutation sets, 24 sets have 27 unique permutations, 240 sets have 30 permutations and 120 sets contain all the 32 permutations. Figure 10 shows that of the degenerate sets only the sets with 10 and 12 unique permutations also have normal permutations, in the sense that none of the bits remain in place. All the other degenerate sets have one or more bits that are not affected by the permutations.

	Number of Unique Permutations in a Set												
Bits Stuck	1	2	4	5	6	10	12	14	16	27	30	32	Total
1 bit			60				30	180	30				300
2 bits				20	60								80
3 bits		20											20
5 bits	1												1
None						40	120			24	240	120	544
Total	1	20	60	20	60	40	150	180	30	24	240	120	945

Figure10. Permutation distribution for T52d.

Looking at the *Wehrmacht* SFM T52d Key table reproduced in Appendix C, it can be shown that all the connections in this table belong to the two groups with 30 and 32 unique permutations. This means that in reality only 360 permutation sets were used by the German cryptographers during the period this key list was in use. It also means that there are not always only 960 cipher alphabets — there can be as many as 1024. This might be an indication that the Germans were aware of the fact that not all of the permutations could be used for cryptographic purposes. This knowledge may have been of a relatively recent nature. The T52a/b machine may have been used earlier with connections which resulted in degenerate permutation sets. When the Swedish cryptanalyst Lars Carlbom analysed the transposition circuit, he found four main permutation families, of which two could be divided further into three sub-groups. One of these families, he said, consisted of connections where one of the transposition units was inactive or disconnected. It is not possible to disconnect a transposition unit and still expect the machine to function, but Lars Carlbom did not know this as he had never seen a T52 machine. He based his analysis entirely on cryptanalytical evidence. In practice, what happened was that an input impulse exited the transposition circuit at the same level as it entered; hence no Baudot element permutation was taking place. The identity permutation referred to earlier is caused by such a set of connections: 1–10, 2–3, 4–5, 6–7 and 8–9, which leave all the bits in their original positions. If one or more of these special connections are combined with other more random connections, the other cases of one or more bits stuck will occur.

			C															
			/	E	4	9	3	T	A	S	D	Z	I	R	L	N	H	O
			8	V	X	+	Q	K	M	G	P	C	B	Y	F	W	J	U
Φ	/	8	32															
	E	V		9	9	2	4	8										
	4	X		8	8	16	0	0										
	9	+		4	4	8	16	0										
	3	Q		2	2	4	8	16										
	T	K		9	9	2	4	8										
	A	M							8	6	1	2	6	1	2	2	4	0
	S	G							4	2	6	1	2	6	1	4	2	4
	D	P							2	1	2	6	1	2	6	0	4	8
	Z	C							4	2	4	8	2	4	8	0	0	0
	I	B							0	8	4	0	8	4	0	8	0	0
	R	Y							0	4	2	4	4	2	4	4	8	0
	L	F							8	6	1	2	6	1	2	2	4	0
	N	W							0	0	4	2	0	4	2	8	4	8
	H	J							4	2	6	1	2	6	1	4	2	4
	O	U							2	1	2	6	1	2	6	0	4	8

Figure11. SFM T52's dibit distribution.

During the year when BP struggled with the July Halibut message enciphered on the T52d it developed and tried out various methods of attack. Several of them were of a statistical nature and were based on knowledge gained through the use of statistical techniques on the Lorenz SZ40/42 machines. The statistical methods BP developed only applied to the "motorless" machines and would not work on machines with wheel stopping. The T52 code wheels had an almost even distribution of dots and crosses with a slight preponderance of crosses. This meant that the modulo two addition was nearly random. However, this was not the case for the permutations, since certain impulses were more likely to go to some positions than others. Therefore the statistical techniques were based on developing statistics for certain impulse combinations of the "pseudo plain text" character, Φ, and their probability of ending up in certain positions in the cipher text character. Here the "pseudo plain text" is the real plain text transformed by the subtractor key.

$$\Phi = P \oplus \Sigma \tag{4}$$

The method applies to both single impulses or to pairs, dibits, but plain text characteristics are more pronounced when using a pair of impulses. For a given permutation it was possible to enumerate how often dibits of a given "pseudo plain text" character, Φ, and its inverse would be associated with dibits in different cipher text characters, C. This is shown in Fig. 11 where the permutation is generated by the transposition circuit used on the T52c and e models, and which used the connections: 1–2, 3–4, 5–6, 7–8 and 9–10.

0	1					2										3										4					5
/	E	4	9	3	T	A	S	D	Z	I	R	L	N	H	O	U	J	W	F	Y	B	C	P	G	M	K	Q	+	X	V	8
A	4	E	U	J	W	/	I	R	L	S	D	Z	K	Q	+	9	3	T	C	P	G	F	Y	B	8	N	H	O	V	X	M
B	O	+	X	Z	D	G	M	T	3	8	W	J	Y	F	E	V	L	R	H	N	/	Q	K	A	S	P	C	4	9	U	I
C	K	N	R	I	V	F	J	U	8	3	9	M	4	G	P	D	S	X	A	+	Q	/	O	H	L	E	B	Y	W	T	Z
D	3	J	F	E	B	R	N	/	O	K	A	+	S	X	Z	C	4	G	9	M	T	U	8	W	Y	I	V	L	H	Q	P
E	/	A	S	D	Z	4	9	3	T	U	J	W	F	Y	B	I	R	L	N	H	O	K	Q	+	X	C	P	G	M	8	V
F	N	K	D	S	X	C	3	9	M	J	U	8	E	B	Y	R	I	V	/	O	H	A	+	Q	Z	4	G	P	T	W	L
G	+	O	V	L	R	B	8	W	J	M	T	3	P	C	4	X	Z	D	Q	K	A	H	N	/	I	Y	F	E	U	9	S
H	Y	P	T	M	9	Q	Z	X	S	L	V	I	O	/	N	W	8	U	B	E	F	G	4	C	3	+	A	K	D	R	J
I	U	9	4	C	P	S	A	K	Q	/	N	H	R	L	V	E	F	Y	J	W	8	3	T	M	G	D	Z	X	+	O	B
J	R	D	K	A	+	3	C	4	G	F	E	B	U	8	W	N	/	O	I	V	L	S	X	Z	Q	9	M	T	P	Y	H
K	C	F	J	U	8	N	R	I	V	D	S	X	A	+	Q	3	9	M	4	G	P	E	B	Y	W	/	O	H	L	Z	T
L	W	T	P	G	4	Z	Q	+	A	H	O	/	V	I	R	Y	B	E	8	U	J	M	9	3	C	X	S	D	K	N	F
M	X	V	O	H	N	8	B	Y	F	G	P	C	T	3	9	+	Q	K	Z	D	S	L	R	I	/	W	J	U	E	4	A
N	F	C	3	9	M	K	D	S	X	R	I	V	/	O	H	J	U	8	E	B	Y	4	G	P	T	A	+	Q	Z	L	W
O	B	G	M	T	3	+	X	Z	D	V	L	R	H	N	/	8	W	J	Y	F	E	P	C	4	9	Q	K	A	S	I	U
P	Q	H	L	V	I	Y	W	8	U	T	M	9	G	4	C	Z	X	S	+	A	K	O	/	N	R	B	E	F	J	3	D
Q	P	Y	W	8	U	H	L	V	I	Z	X	S	+	A	K	T	M	9	G	4	C	B	E	F	J	O	/	N	R	D	3
R	J	3	C	4	G	D	K	A	+	N	/	O	I	V	L	F	E	B	U	8	W	9	M	T	P	S	X	Z	Q	H	Y
S	9	U	E	F	Y	I	/	N	H	A	K	Q	D	Z	X	4	C	P	3	T	M	J	W	8	B	R	L	V	O	+	G
T	Z	L	H	O	/	W	Y	B	E	P	G	4	M	9	3	Q	+	A	X	S	D	V	I	R	N	8	U	J	F	C	K
U	I	S	A	K	Q	9	4	C	P	E	F	Y	J	W	8	/	N	H	R	L	V	D	Z	X	+	3	T	M	G	B	O
V	8	M	G	P	C	X	+	Q	K	O	H	N	L	R	I	B	Y	F	W	J	U	T	3	9	4	Z	D	S	A	/	E
W	L	Z	Q	+	A	T	P	G	4	Y	B	E	8	U	J	H	O	/	V	I	R	X	S	D	K	M	9	3	C	F	N
X	M	8	B	Y	F	V	O	H	N	+	Q	K	Z	D	S	G	P	C	T	3	9	W	J	U	E	L	R	I	/	A	4
Y	H	Q	Z	X	S	P	T	M	9	W	8	U	B	E	F	L	V	I	O	/	N	+	A	K	D	G	4	C	3	J	R
Z	T	W	Y	B	E	L	H	O	/	Q	+	A	X	S	D	P	G	4	M	9	3	8	U	J	F	V	I	R	N	K	C
3	D	R	N	/	O	J	F	E	B	C	4	G	9	M	T	K	A	+	S	X	Z	I	V	L	H	U	8	W	Y	P	Q
4	A	/	I	R	L	E	U	J	W	9	3	T	C	P	G	S	D	Z	K	Q	+	N	H	O	V	F	Y	B	8	M	X
8	V	X	+	Q	K	M	G	P	C	B	Y	F	W	J	U	O	H	N	L	R	I	Z	D	S	A	T	3	9	4	E	/
9	S	I	/	N	H	U	E	F	Y	4	C	P	3	T	M	A	K	Q	D	Z	X	R	L	V	O	J	W	8	B	G	+
+	G	B	8	W	J	O	V	L	R	X	Z	D	Q	K	A	M	T	3	P	C	4	Y	F	E	U	H	N	/	I	S	9
/	E	4	9	3	T	A	S	D	Z	I	R	L	N	H	O	U	J	W	F	Y	B	C	P	G	M	K	Q	+	X	V	8

Figure12. Baudot XOR square in class order

The alphabets in the figure have only a length of 16, as the normal 32 element Baudot alphabet has been folded in half, with each position in the alphabet occupying a given Baudot character and its inverse, e.g. E and V, which have the Baudot vectors ×•••• and •××××. The characters /–8 (which are all dots and all crosses) can only go to one place under all the 32 different permutation, while in all the other cases there are varying distributions. The characters belonging to classes 1 and 4 have single cross/dot distributions, while the characters in classes 2 and 3 have double cross/dot distributions. This is the reason for the clustering of the distributions in the two squares of size 5 and 10.

But on the T52a/b and d models the permutations were not fixed but variable depending on the connections of the transposition units. Therefore, the permutation probabilities, and hence the statistics, depended on the given permutation set which, of course, was unknown until the machine was broken. So the statistical techniques available in 1944 were nothing more than tools for getting a better knowledge about the cryptanalytical problem. They were not of much use in attacking the machines.

It appears that messages in depth were the only viable attack on these machines in 1944. It is far too involved to illustrate a full blown attack on a real example, but looking at a very small constructed example with a depth of two and a known crib will give a feeling for the problem. As mentioned earlier, the attacks in depths were helped by the use of tables and catalogues. One such table is the Baudot XOR or modulo two square. However, the table becomes a lot more useful when one of the alphabets is arranged in class order. This is illustrated in Fig. 12, where the plain text alphabet is in its normal order along the left hand column and the key alphabet is arranged in class order along the top row. The intersection of a plain text character and a key character will give the resulting cipher text character. However, due to the properties of modulo two addition any of the two alphabets, the one in normal order or the one in class order, can be used for any of the three elements plain, cipher or key characters.

To see what is actually taking place and how one might attack two messages in depth it is of interest to return to the principal encipherment equation (1)

$$\Pi(P \oplus \Sigma) = C$$

It is easily shown that permutation is distributive under modulo two addition

$$\Pi(X \oplus Y) = \Pi X \oplus \Pi Y \tag{5}$$

If we apply (5) to (1) we get

$$\Pi P \oplus \Pi\Sigma = C \tag{6}$$

In other words, the cipher character can also be obtained by first applying the permutation on the plain text character and the subtractor before combining these two transposed elements by modulo two addition. In the case of two messages P and Q enciphered in depth by the subtractor key Σ and the permutor key Π we can write the following

$$\Pi P \oplus \Pi\Sigma = C \tag{7}$$

$$\Pi Q \oplus \Pi\Sigma = D \tag{8}$$

Combining (7) and (8) by modulo two addition eliminates the $\Pi\Sigma$ term and gives at the basic equation for messages in depth

$$\Pi(P \oplus Q) = C \oplus D \tag{9}$$

Equation 9 shows that if either P or Q is known the value of the other cannot be automatically determined, as with pure Vernam [24] encipherment where there is only a subtractor function and no permutor function. In reality there might be as many as ten possible solution for P or Q depending on the Baudot class in which the operation took place. If the operation takes place in class 0 or 5, P and Q are uniquely determined, while in class 1 and 4 there are five possibilities and in class 2 and 3 there are ten possible solutions.

One opening for attack is the fact that the permutation only reorders the Baudot code elements: it does not change the elements themselves. Therefore if $C \oplus D$ contains m crosses and n dots, it must also be the same for $P \oplus Q$. So if we know or can make a guess at P we will have a limited number, from 1 to 10, of choices for Q. This is the basis for an attack on messages in depths enciphered on the Siemens T52.

The two messages in Fig. 13 have been enciphered with the same key on a computer simulation of the T52d machine.[20] They are enciphered without the KTF and the main inner key, *Fernschreibgrundschlüssel*, is 6–8, 1–2, 5–7, III, 4–10, IV, II, I, 3–9, V, which is the key for day one in the *Norwegen Nr. 7* key table in Appendix C. The message key, *Fernschreibspruchschlüssel*, is the same as given on page 14, QEP FF OO PP AA ZZ VV CC MM HH UU. The second message is suspected to start with "three" or "four", since message numbers in the region of three to four hundred are expected.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	P	G	I	+	L	X	G	L	T	N	E	3	Y	O	X	P	J	3	B	V
2	Y	N	+	8	4	I	P	P	9	E	B	D	K	W	E	8	E	I	4	H
n	2	3	4	1	1	4	2	1	2	3	2	1	3	3	3	2	2	3	4	2

Figure13. Two messages in depth.

The class numbers appearing in the last row are found by forming the modulo two sum of the two cipher text characters and looking up in which class the resulting character belongs. Taking the first two cipher text characters T and E and combining them modulo two results in Z. This result is found by using a simple Baudot XOR square or using the class XOR square in Fig. 12. Looking up T in the vertical left hand alphabet and E in the horizontal class alphabet, we find Z at their intersection. Looking in the top row class alphabet, we find that Z belongs to class 2. The class information can also be found from the Baudot class alphabet in Fig. 5. Another, perhaps even faster, method is to look up one of the cipher text characters, say T, in the left hand vertical alphabet and then searching down the row to find the other cipher text character, E. Doing so we find E situated in one of the columns for class 2.

[20] The T52 computer simulation will be made available on the Cipher Simulation Group's (CSG) Web servers which are accessible through the author's Cryptology Web page at URL: http://home.cern.ch/~frode/crypto/

Trying the word "three" with a space, here represented by 9, as a crib for the beginning of the second message gives the following possible solutions for the characters of the beginning of the first message, as shown in Fig. 14. The possible solutions for each character is given in the generatrices[21] which have been obtained from the class XOR table. Looking up the first clear text letter of the crib, T, in the left hand vertical alphabet the corresponding generatrix is found further along the row in the columns for the Baudot class 2. The generatrix characters are: WYBEPG4M93 which have been entered in alphabetical order to ease the search for a possible plain text word.

	1	2	3	4	5	6
n	2	3	4	1	1	4
2	T	H	R	E	E	9
	B	B	H	A	**A**	A
	E	C	Q	D	D	**G**
	G	**E**	**S**	**S**	S	J
	M	F	X	Z	Z	W
1	P	G	Z	/	/	8
	W	U				
	Y	W				
	3	3				
	4	4				
	9	8				

Figure14. Trying the crib "three" in message no. 2.

The most prominent plain text word is the beginning of the word "MESSAGe". We can now try to extend the plain text in the second message by using the expected "E9" (E and a space) as a further crib in the first message. This is shown in Fig. 15a.

Since the beginning of the first message is suspected to contain a message number the continuation is expected to be another number. Of the numbers from one to ten the only possible solutions are "THree" or "FIve". "ThREE" and 9 do not give any promising plain text in message number one but "fiVE" and 9 give "ONE" as shown in Fig. 15b. This is even a unique solution as none of the other characters needed for the other numbers are present in the first generatrix. The rest of the solution is left as an exercise for the reader. However, solutions are not always as straightforward as here: often it will not be possible to carry on with only two messages in depth. Very often the messages contained numbers or abbreviations which made it extremely

[21] Generatrix, plural generatrices, is a decipherment or encipherment out of a set of decipherments or encipherments of the same text under a given hypothesis or cryptographic principle.

	7	8
n	2	1
1	E	9
2	B	H
	F	I
	J	N
	T	S
	U	/
	W	
	Y	
	3	
	4	
	9	

(a)

	9	10	11
n	2	3	2
2	V	E	9
1	H	H	C
	I	I	**E**
	K	K	F
	L	L	M
	N	**N**	P
	O	O	T
	Q	Q	U
	R	R	Y
	X	X	3
	+	+	4

(b)

Figure15. Continuing the cribs in messages no. 1 and no. 2.

difficult, if not impossible, to extend the messages with only a depth of two or three.

It is one thing to break a number of messages in depth. However, the aim is to break the machine, so as to be able to recover the key streams and hence to break all other messages for the rest of the key period. For this purpose it is necessary to be able to uniquely determine the permutation Π for each encryption step. It can be shown that at least a depth of four is necessary, but that it is generally not sufficient. With a depth of four one has only a 20 % probability of finding a unique permutation. With a depth of seven or eight the probabilities are such that a workable key extraction can take place. As the code wheel patterns are fixed, it is possible to determine from the extracted key streams which code wheel is used where and for what purpose. From this information it is then possible to recover the plug connections and starting positions of the machine.

10 Conclusion

Not only did Bletchley Park intercept traffic enciphered on the Siemens SFM T52, but it also broke all the different models that it discovered. However, it was clear from the very beginning that the T52 was a very difficult machine to break. It probably would have remained unbroken had it not been for the German security blunders in using the machines. The blame should not be put entirely on the German teleprinter operators. The Siemens designers of the machine are equally responsible for not listening to the advice of the German cryptographic experts. The Siemens engineers appear to have been more focused on the engineering problems than on the cryptographic security of the machine. The T52a/b and the original T52c machines were basically machines with very limited security. The T52c is an extraordinary example of how not to go about designing cryptographic algorithms. The wheel combining logic, which clearly was meant to strengthen the machine, had exactly the opposite effect — it eased the task of breaking the machine.

On the other hand, the T52d was a relatively well-designed machine. If this machine been the first to see service and the teleprinter operators had been properly instructed in using the machine, it is highly unlikely that it would have been broken. Another weakness of all of these machines is the fixed code wheel patterns. It is understandable that the designers thought that with the complexity of the machine it would not be necessary to vary the code wheel patterns. However, with variable code wheel patterns the machines would have been strengthened considerably. Due to the transposition circuit, cribs would not have led to the recovery of the key stream and even complete plain text of thousands of characters would not have resulted in recovered code wheel patterns.

Sir Harry Hinsley's statement, [13–15] that BP decided to concentrate its non-Morse interception, cryptanalytical, and decryption resources on the Army's Tunny traffic because of a need to husband resources and the need for good intelligence on the German Army, is undoubtedly correct. However, these were probably not the only reasons why BP abandoned its efforts against the Sturgeon machines. The cryptanalytical difficulties BP faced in attacking these machines, the small number of Sturgeon links, and the very limited intelligence that could be derived from the traffic must have played important roles in the outcome of BP's decision to concentrate on the Tunny traffic.

11 Acknowledgements

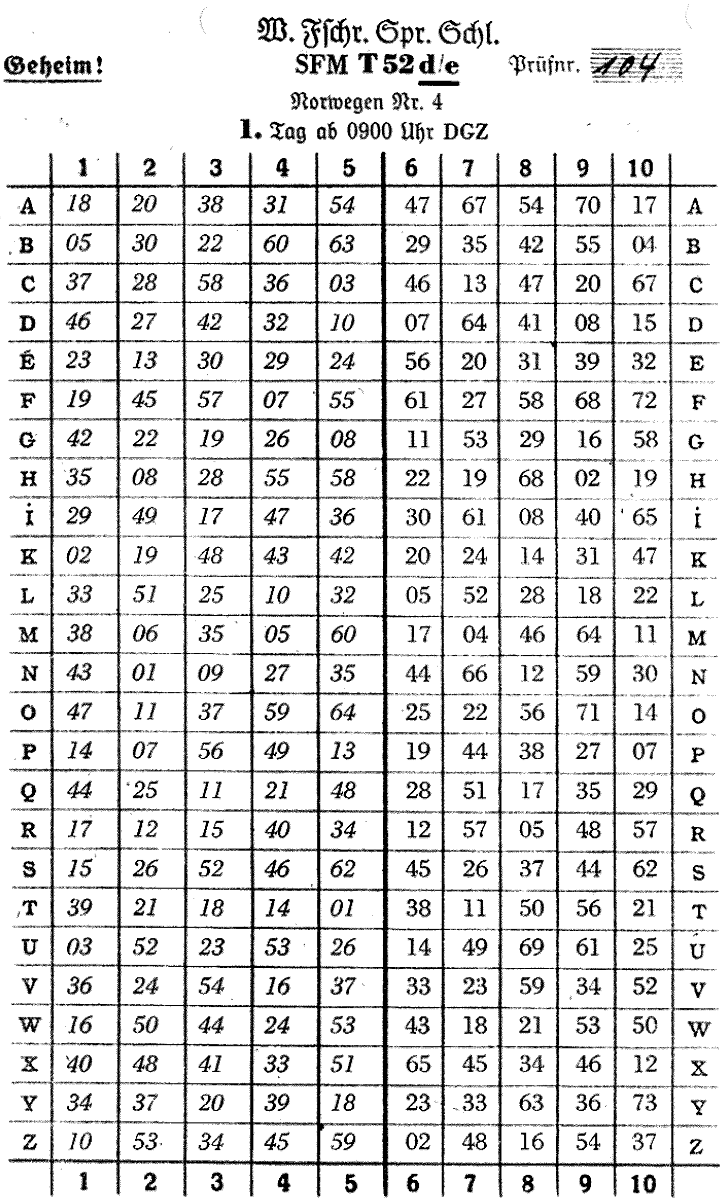
The author should like to thank Bengt Beckman who, through his friendship over the last five years, has been a constant inspiration for my research into the history of the Siemens SFM T52 machines. His help with obtaining material about the Swedish cryptanalysts and their success against these machines has been crucial to this work. As usual, Ralph Erskine has been very helpful with suggestions and improvements, not to forget his help with proof reading and archive material. David Alvarez has given generous support and supplied several documents. Special thanks go to Captain Jon Ulvensøen and The Armed Forces Museum (Forvarsmuseet) in Oslo for supplying many German documents and for giving me access to their collection of cipher machines. I should also like to thank Donald Davies for answering my questions about the T52c wheel combining logic and generally for his help over a great many years. Furthermore, I am very grateful to Geoff Sullivan who has helped me with the simulations of the permutation circuit, and whose computer simulation of the complete cipher machine in all its versions and models has been of the utmost importance to this research.

References

1. Unknown Author: Sturgeon Type Ciphers (Research Section, November 1944). Addendum to Captain Walter J. Fried's report No. 116 of 17 Nov. 1944. Henceforth called Fried reports. National Archives and Records Administration (NARA) RG 457 NSA Historical Collection Box 880 Nr. 2612
2. Unknown Author: Band-Transposition Systems. Technical Paper, Signal Security Agency, Cryptanalytic Branch, Washington, June 1944 NARA RG 457 NSA Hist. Col. Box 1029 Nr. 3304
3. Beckman, Bengt: **Svenska kryptobedrifter (Swedish Crypto Achievements)**. In Swedish. Stockholm: Albert Bonniers Förlag (1996)
4. Boheman, Erik: **På Vakt. Kabinettssekreterare under andra världskriget (On Duty. Under-Secretary of State During the Second World War)**. In Swedish. Stockholm (1964)
5. Campaigne, Howard: Report on British Attack on "FISH". National Archives and Records Administration RG 457 NSA Hist. Col. Box 579 Nr. 1407 (1945)
6. Davies, Donald W.: The Siemens and Halske T52e Cipher Machine. Cryptologia **6(4)** October (1982) 289–308
7. Davies, Donald W.: The Early Models of the Siemens and Halske T52 Cipher Machine. Cryptologia **7(3)** July (1983) 235–253
8. Davies, Donald W.: New Information on the History of the Siemens and Halske T52 Cipher Machine. Cryptologia **18(2)** April (1994) 141–146
9. Deutsche Wehrmacht: Schlüsselfernschreibvorschrift (SFV). H.Dv. g 422, L.Dv. g 704/3b, M.Dv. Nr. 924a Geheim, 1 Dezember 1942
10. Fried, Walter J.: Fish Notes. Fried Report No. 43 of 27 May 1944. NARA RG 457 NSA Hist. Col. Box 880 Nr. 2612
11. Fried, Walter J.: Fish Notes. Fried Report No. 46 of 12 June 1944. NARA RG 457 NSA Hist. Col. Box 880 Nr. 2612
12. Fried, Walter J.: Fish Notes (Sturgeon). Fried Report No. 68 of 29 July 1944. NARA RG 457 NSA Hist. Col. Box 880 Nr. 2612
13. Hinsley, F.H.: Geheimschreiber (Fish). In F.H. Hinsley et al. **British Intelligence in the Second World War**. London: HMSO Vol. **3** Part 1 Appendix 2 (1984) 477–482
14. Hinsley, F.H.: Cracking the Ciphers. Electronics & Power IEE July (1987) 453–455
15. Hinsley, F.H.: An Introduction to Fish. In ed. F.H. Hinsley and Alan Stripp. **Codebreakers, The Inside Story of Bletchley Park**. Oxford: Oxford University Press (1993) 141–148
16. Kahn, David: **The Codebreakers**. New York: Macmillan (1967)
17. Mache, Wolfgang: Geheimschreiber. Cryptologia **10(4)** October (1986) 230–242.
18. Mache, Wolfgang: The Siemens Cipher Teletype in the History of Telecommunications. Cryptologia **13(2)** April (1989) 97–117
19. Mache, Wolfgang: Der Siemens-Geheimschreiber — ein Beitrag zur Geschichte der Telekommunikation 1992: 60 Jahre Schlüsselfernschreibmaschine. In German. Archiv für deutsche Postgeschichte Heft **2** (1992) 85–94
20. Oberkommando der Kriegsmarine: Die Siemens-Schlüsselfernschreibmaschine SFM T52d (T typ 52 d). M.Dv. Nr. 35IV, D.(Luft) T.g.Kdos. 9105d. Geheime Kommandosache, Berlin März 1944

21. Tutte, William T.: **Graph Theory As I Have Known It**. Oxford Lecture Series in Mathematics and Its Applications Vol. **11** Oxford: Oxford University Press (1998)
22. Tutte, William T.: FISH and I. In these proceedings. **Coding Theory and Cryptology: From Enigma and Geheimschreiber to Quantum Theory**. New York: Springer Verlag, Lecture Notes in Computational Science and Engineering (1999)
23. Ulfving, Lars and Weierud, Frode: The Geheimschreiber Secret: Arne Beurling and the Success of Swedish Signals Intelligence. In these proceedings. **Coding Theory and Cryptology: From Enigma and Geheimschreiber to Quantum Theory**. New York: Springer Verlag, Lecture Notes in Computational Science and Engineering (1999)
24. Vernam, Gilbert S.: Cipher Printing Telegraph Systems For Secret Wire and Radio Telegraphic Communications. Transactions A. I. E. E. Vol. **XLV** Feb. (1926) 295–301

12 Appendix A



W. Fſchr. Spr. Schl.

Geheim! SFM T52d/e Prüfnr. 104

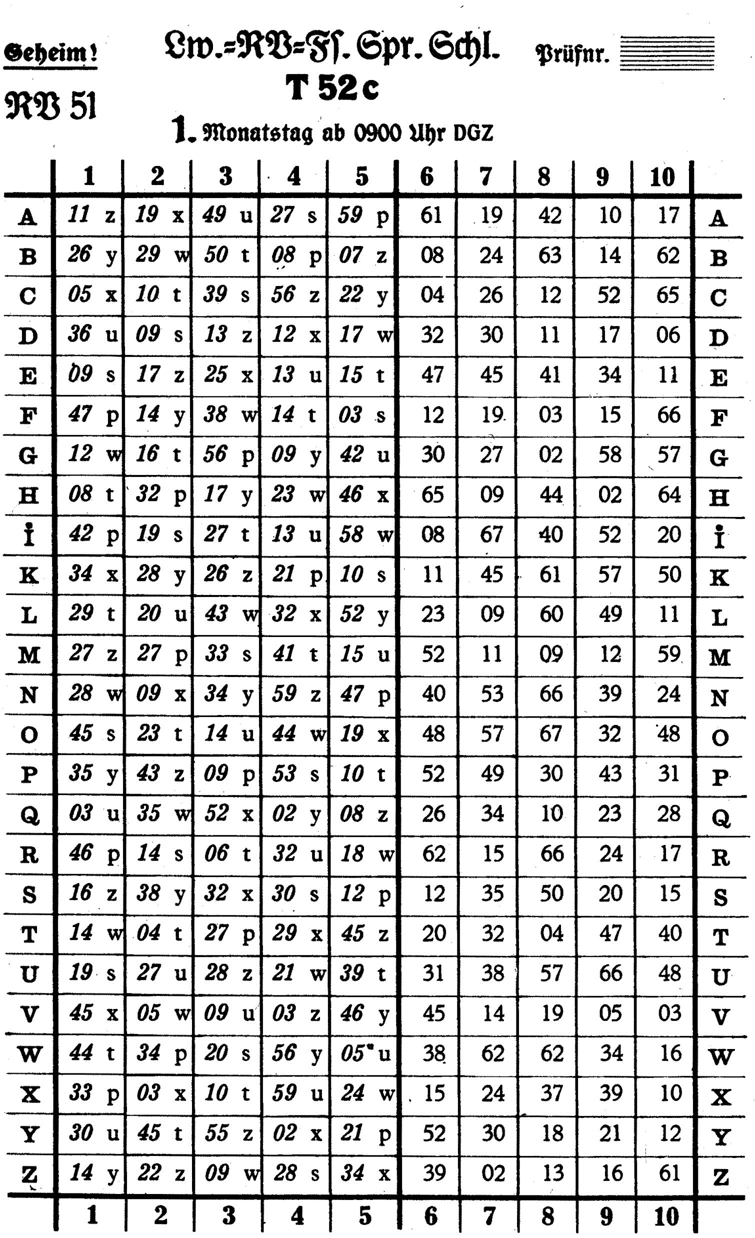
Norwegen Nr. 4

1. Tag ab 0900 Uhr DGZ

	1	2	3	4	5	6	7	8	9	10	
A	18	20	38	31	54	47	67	54	70	17	A
B	05	30	22	60	63	29	35	42	55	04	B
C	37	28	58	36	03	46	13	47	20	67	C
D	46	27	42	32	10	07	64	41	08	15	D
É	23	13	30	29	24	56	20	31	39	32	E
F	19	45	57	07	55	61	27	58	68	72	F
G	42	22	19	26	08	11	53	29	16	58	G
H	35	08	28	55	58	22	19	68	02	19	H
İ	29	49	17	47	36	30	61	08	40	65	İ
K	02	19	48	43	42	20	24	14	31	47	K
L	33	51	25	10	32	05	52	28	18	22	L
M	38	06	35	05	60	17	04	46	64	11	M
N	43	01	09	27	35	44	66	12	59	30	N
O	47	11	37	59	64	25	22	56	71	14	O
P	14	07	56	49	13	19	44	38	27	07	P
Q	44	25	11	21	48	28	51	17	35	29	Q
R	17	12	15	40	34	12	57	05	48	57	R
S	15	26	52	46	62	45	26	37	44	62	S
T	39	21	18	14	01	38	11	50	56	21	T
U	03	52	23	53	26	14	49	69	61	25	U
V	36	24	54	16	37	33	23	59	34	52	V
W	16	50	44	24	53	43	18	21	53	50	W
X	40	48	41	33	51	65	45	34	46	12	X
Y	34	37	20	39	18	23	33	63	36	73	Y
Z	10	53	34	45	59	02	48	16	54	37	Z
	1	2	3	4	5	6	7	8	9	10	

Figure16. T52d *Spruchschlüssel* — message key.

13 Appendix B



Geheim! Lw.=RV=Fſ. Spr. Schl. Prüfnr.

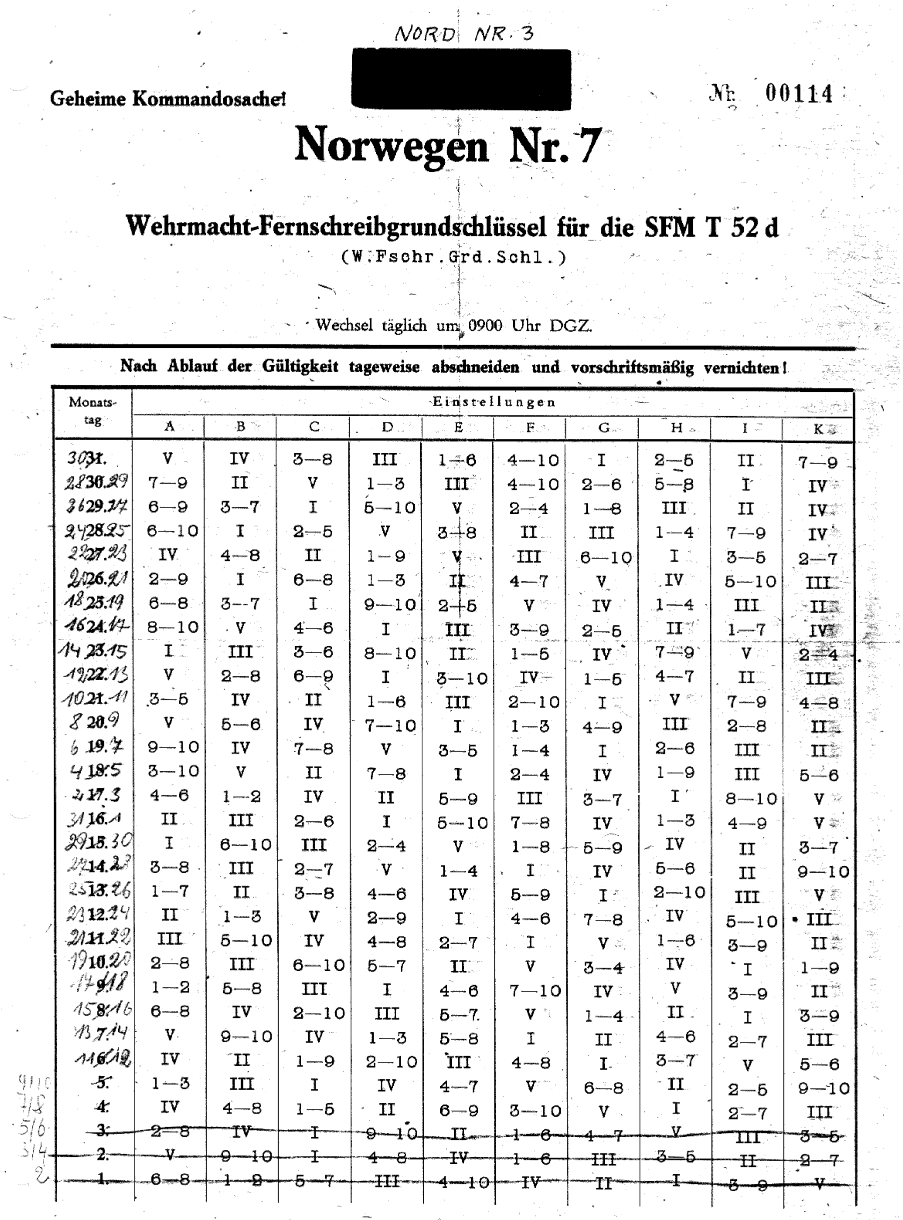
RV 51 T 52c

1. Monatstag ab 0900 Uhr DGZ

	1	2	3	4	5	6	7	8	9	10	
A	11 z	19 x	49 u	27 s	59 p	61	19	42	10	17	A
B	26 y	29 w	50 t	08 p	07 z	08	24	63	14	62	B
C	05 x	10 t	39 s	56 z	22 y	04	26	12	52	65	C
D	36 u	09 s	13 z	12 x	17 w	32	30	11	17	06	D
E	09 s	17 z	25 x	13 u	15 t	47	45	41	34	11	E
F	47 p	14 y	38 w	14 t	03 s	12	19	03	15	66	F
G	12 w	16 t	56 p	09 y	42 u	30	27	02	58	57	G
H	08 t	32 p	17 y	23 w	46 x	65	09	44	02	64	H
I	42 p	19 s	27 t	13 u	58 w	08	67	40	52	20	I
K	34 x	28 y	26 z	21 p	10 s	11	45	61	57	50	K
L	29 t	20 u	43 w	32 x	52 y	23	09	60	49	11	L
M	27 z	27 p	33 s	41 t	15 u	52	11	09	12	59	M
N	28 w	09 x	34 y	59 z	47 p	40	53	66	39	24	N
O	45 s	23 t	14 u	44 w	19 x	48	57	67	32	48	O
P	35 y	43 z	09 p	53 s	10 t	52	49	30	43	31	P
Q	03 u	35 w	52 x	02 y	08 z	26	34	10	23	28	Q
R	46 p	14 s	06 t	32 u	18 w	62	15	66	24	17	R
S	16 z	38 y	32 x	30 s	12 p	12	35	50	20	15	S
T	14 w	04 t	27 p	29 x	45 z	20	32	04	47	40	T
U	19 s	27 u	28 z	21 w	39 t	31	38	57	66	48	U
V	45 x	05 w	09 u	03 z	46 y	45	14	19	05	03	V
W	44 t	34 p	20 s	56 y	05 u	38	62	62	34	16	W
X	33 p	03 x	10 t	59 u	24 w	15	24	37	39	10	X
Y	30 u	45 t	55 z	02 x	21 p	52	30	18	21	12	Y
Z	14 y	22 z	09 w	28 s	34 x	39	02	13	16	61	Z
	1	2	3	4	5	6	7	8	9	10	

Figure17. T52c *Spruchschlüssel* — message key.

14 Appendix C



NORD NR. 3

Geheime Kommandosache! [redacted] Nr. 00114

Norwegen Nr. 7

Wehrmacht-Fernschreibgrundschlüssel für die SFM T 52 d

(W.Fsohr.Grd.Schl.)

Wechsel täglich um 0900 Uhr DGZ.

Nach Ablauf der Gültigkeit tageweise abschneiden und vorschriftsmäßig vernichten!

Monatstag	Einstellungen									
	A	B	C	D	E	F	G	H	I	K
30 31.	V	IV	3—8	III	1—6	4—10	I	2—5	II	7—9
28 30. 29	7—9	II	V	1—3	III	4—10	2—6	5—8	I	IV
26 29. 27	6—9	3—7	I	5—10	V	2—4	1—8	III	II	IV
24 28. 25	6—10	I	2—5	V	3—8	II	III	1—4	7—9	IV
22 27. 23	IV	4—8	II	1—9	V	III	6—10	I	3—5	2—7
20 26. 21	2—9	I	6—8	1—3	II	4—7	V	IV	5—10	III
18 25. 19	6—8	3—7	I	9—10	2—5	V	IV	1—4	III	II
16 24. 17	8—10	V	4—6	I	III	3—9	2—5	II	1—7	IV
14 23. 15	I	III	3—6	8—10	II	1—5	IV	7—9	V	2—4
12 22. 13	V	2—8	6—9	I	3—10	IV	1—5	4—7	II	III
10 21. 11	3—5	IV	II	1—6	III	2—10	I	V	7—9	4—8
8 20. 9	V	5—6	IV	7—10	I	1—3	4—9	III	2—8	II
6 19. 7	9—10	IV	7—8	V	3—5	1—4	I	2—6	III	II
4 18. 5	3—10	V	II	7—8	I	2—4	IV	1—9	III	5—6
2 17. 3	4—6	1—2	IV	II	5—9	III	3—7	I	8—10	V
31 16. 1	II	III	2—6	I	5—10	7—8	IV	1—3	4—9	V
29 15. 30	I	6—10	III	2—4	V	1—8	5—9	IV	II	3—7
27 14. 28	3—8	III	2—7	V	1—4	I	IV	5—6	II	9—10
25 13. 26	1—7	II	3—8	4—6	IV	5—9	I	2—10	III	V
23 12. 24	II	1—3	V	2—9	I	4—6	7—8	IV	5—10	III
21 11. 22	III	5—10	IV	4—8	2—7	I	V	1—6	3—9	II
19 10. 20	2—8	III	6—10	5—7	II	V	3—4	IV	I	1—9
17 9. 18	1—2	5—8	III	I	4—6	7—10	IV	V	3—9	II
15 8. 16	6—8	IV	2—10	III	5—7	V	1—4	II	I	3—9
13 7. 14	V	9—10	IV	1—3	5—8	I	II	4—6	2—7	III
11 6. 12	IV	II	1—9	2—10	III	4—8	I	3—7	V	5—6
9/10 ~~5.~~	1—3	III	I	IV	4—7	V	6—8	II	2—5	9—10
7/8 ~~4.~~	IV	4—8	1—5	II	6—9	3—10	V	I	2—7	III
5/6 ~~3.~~	~~2—8~~	~~IV~~	~~I~~	~~9—10~~	~~II~~	~~1—6~~	~~4—7~~	~~V~~	~~III~~	~~3—5~~
3/4 ~~2.~~	~~V~~	~~9—10~~	~~I~~	~~4—8~~	~~IV~~	~~1—6~~	~~III~~	~~3—5~~	~~II~~	~~2—7~~
2 ~~1.~~	~~6—8~~	~~1—2~~	~~5—7~~	~~III~~	~~4—10~~	~~IV~~	~~II~~	~~I~~	~~3—9~~	~~V~~

Figure18. T52d *Grundschlüssel* — main inner key.

Subject Index

ENIGMA and PURPLE: How the Allies Broke German and Japanese Codes During the War

David A. Hatch

Director of the Center for Cryptologic History, National Security Agency, Fort George Meade, MD

1 Introduction

Cryptology consists of two aspects: Signals Intelligence (SIGINT), which seeks to exploit the encrypted communications of enemies or potential enemies, and Information Systems Security, which seeks to protect American communications from those who might wish to exploit them. Americans utilized cryptology even before the foundation of the United States, particularly in the American Revolution and the Civil War. However, it was not until the Twentieth Century that the United States began sustained Communications Intelligence (COMINT) activities.

One probable reason for this was the feeling of security two oceans gave Americans; without a sense of immediate external menace, there was no stimulus for an American government to obtain regular and timely information about potential overseas enemies. The United States was late in forming organizations for military and naval intelligence, and even once in existence, these organizations remained rudimentary until two world wars brought American leaders to the realization that American territory was now vulnerable.

The "golden age" for American cryptology was the Second World War. Both the U.S. Navy and Army solved and exploited a wide variety of enemy codes and ciphers at all levels. These achievements in reading enemy systems enabled U.S. commanders to make wiser decisions that saved thousands of American lives and shortened the war. Out of the many achievements of wartime cryptanalysis, this article will discuss PURPLE, the high-level Japanese diplomatic system, and ENIGMA, the high-level German military cryptographic machine.

2 Origins of radio intelligence

The invention and widespread use of wireless radio made collection of foreign communications less of a technical challenge. With this Twentieth Century invention, it became possible to acquire a potential enemy's messages without knocking over couriers or tampering with telegraph wires. Elementary efforts at radio intercept began in the first decade of the Twentieth Century, and most industrialized countries engaged in Signals Intelligence to some extent in the years immediately prior to World War I. However, few countries created organizations to manage the activity until war made it necessary for them to do so.

The War to Make the World Safe for Democracy found American radiomen intercepting German communications for both strategic and tactical purposes. The revitalization of Military Intelligence on the eve of the war provided a suitable organizational structure for the exploitation and dissemination of Communications Intelligence. After the war, however, the radio intelligence organization was eliminated in the general demobilization.

With a nascent sense of international menace and perhaps based on the favorable wartime experience, at the conclusion of the Great War, the United States for the first time created a national-level intelligence organization, a cryptologic organization. This was MI-8, jointly funded by the U.S. Army and the State Department, with some Navy participation, and headed by Herbert O. Yardley. The story of MI-8, usually known by its colorful nickname, the "Black Chamber," is well-known, so it is not necessary to recount its exploits here; suffice it to say that this organization was successful against a number of foreign diplomatic cryptosystems during the 1920s.

Equally well-known is the demise of the Black Chamber, marked by the mythical moral utterances of Secretary of State Henry Stimson, who is alleged to have said "Gentlemen don't read other gentlemen's mail!" Stimson withdrew the State Department's funding from MI-8 largely for budgetary reasons in the Great Depression, effectively closing it down. Subsequent to the closure of MI-8 in 1931, Yardley published his expose of American codebreaking, The American Black Chamber. This book caused Japan, among other countries, to change its communications security practices, with increasing dependence on machine systems in the 1930s.

Whatever the truth of Stimson's statement, the U.S. Army and Navy, for their part, wanted to go on reading other gentlemen's mail. The Navy from the 1920s began building an organization for cryptologic activities, and expanded this service in the 1930s; it was primarily concerned with training intercept operators and cryptanalysts who would be ready for operations in case of war. The Army, for its part, in 1929 hired one of its consultants, William F. Friedman, to form a modern Signal Intelligence Service (SIS).

3 PURPLE

On April Fool's Day in 1930, William Friedman welcomed the first of three cryptologists he had hired for the SIS. His first employee was Frank B. Rowlett, a young mathematician who was brought to Washington as a junior cryptanalyst at $ 2,000 per year. Shortly after, Friedman brought in Abraham Sinkov and Solomon Kullback – two graduates of the City College of New York – into the service. John Hurt, a talented Japanese linguist, who had the added advantage of a Congressman for an uncle, was hired later. For much of the decade they served as the nucleus of the Army's cryptologic service.

The organizational placement of SIS, it should be noted, was not in Military Intelligence, but the Army Signal Corps. Its primary mission was devising cryptosystems for U.S. use, with the solution of foreign systems as a secondary activity. The Army at this time evinced minimal interest in peacetime SIGINT, except to keep skills current in case of war.

William Friedman proved to be a genius at training and put his staff through rigorous exercises, including codes used in the Great War, files of the Black Chamber, and rudimentary machine systems. This regimen was validated by the eventual achievements of this group: in the period from 1933 to 1941, SIS cryptologists studied eleven Japanese diplomatic and military attache systems – and solved them.

The Navy's cryptologic organization developed considerable skill in exploiting Japanese communications during fleet maneuvers. Their discoveries of Japanese naval preparedness and likely Japanese strategies in case of war between the two countries helped the U.S. naval senior commanders to re-evaluate their own policies and practices.

Initial progress was made against some Japanese diplomatic messages done in traditional systems, when, in 1935, working as a team, Frank Rowlett and Solomon Kullback detected exploitable weaknesses in a Japanese machine cipher "Angooki Type A" used by the Foreign Ministry. As they began to read messages enciphered in this system, the Americans gave it the codename RED – the first color of the spectrum for the first cipher machine actually solved.

Both Army and Navy cryptanalysts solved the Japanese RED system, independently of each other and in the same general time frame. The production of real intelligence from the cipher machines of a potential foreign enemy quickly came to the attention of the Army authorities. Rowlett recalled that for the first time their military superiors began according SIS personnel real respect. More importantly, the Army increased SIS resources, and had Friedman draw up regulations on distribution of the decrypted material, closely restricting those who would be given access to it. This may have been the first instance of a compartmentation program in modern American intelligence activities.

Cryptologic operations never remain static, however, and the Americans all too soon lost their access to this inside information. In February 1939, Japan's Foreign Ministry introduced the TYPE-B Cipher Machine, nicknamed PURPLE by American cryptanalyts. The Japanese distributed this machine to their most important embassies – Washington, Berlin, Rome, and London – precisely those locations from which U.S. policymakers most wanted information. The Americans were able to exploit PURPLE partially until May 1939, when the Japanese introduced significant security improvements to the system.

It took U.S. Army cryptanalysts until late November 1940 to recover and produce usable decrypts from this improved PURPLE system. (If this seem excessive, it should be noted that after the war, American cryptologists discovered that their German counterparts had also attempted the PURPLE system – and failed!) While the cryptosystem itself was named PURPLE, the product of SIS efforts, that is, decrypts prepared for circulation, were stamped with the restrictive codeword MAGIC – a possible reference to William Friedman's fond joke that his cryptanalysts were "magicians."

In addition to solving the system, SIS personnel designed items of equipment to speed the processing of PURPLE. Drawing on the talents of Leo Rosen, a reserve officer with a background in electrical engineering, SIS created a range of equipment, including an "analog" which would enable automatic decryption of PURPLE-based messages.

The ability to read RED and PURPLE led SIS into cooperation with the U.S. Navy. The amount of material involved, compounded by lengthy processing time, had brought the Army to share the secret of RED with its sister service. The ability to exploit PURPLE greatly increased the need for inter-service cooperation. The Navy's organization, OP-20-G, had assets the Army did not: a better position to collect Japanese communications and a larger pool of Japanese-trained language officers, to begin with.

The already-complicated relationship between the Army and Navy became considerably more complex in the struggle to exploit PURPLE. When Army analysts designed an analog machine to speed PURPLE recoveries, it was constructed by the Navy at the Washington Navy Yard, which had more experience in making cryptographic devices. Subsequently, the Navy cooperated with the Army in collecting message traffic, and making code recoveries. However, the two services found themselves competing in exploiting and distributing the product, particularly vying for the privilege of disseminating decrypts to senior civilian officials.

The two services tried various schemes to avoid duplication of effort in processing PURPLE until finally they agreed that the Navy would process PURPLE messages on odd-numbered days, the Army on even-numbered ones. This was a cumbersome system with frequent points of conflict but it was an important step toward interservice cooperation in the whole process of cryptology. During the war, another step was taken as the two services exchanged liaison officers to each others' cryptologic organization.

As the military itself expanded in the late 1930s, SIS and OP-20-G also expanded, and when war came, the expansion became more rapid. With the advent of war, SIS moved from its tiny quarters in a vaulted area in the Munitions Building in downtown Washington, D.C., to Arlington Hall Station, a former girls' school across the Potomac in Virginia. The Navy also moved its cryptologic headquarters to Mt. Vernon Academy, a former girls' school in northwest Washington.

Just prior to the U.S. entry into the Second World War, the United States and Great Britain began cautious exchanges of technical information, including some sharing of data on cryptanalysis. Each was delighted to learn that the other had made major solutions to the cryptosystems used by their common enemies, Germany and Japan – the U.S. against PURPLE, Great Britain against ENIGMA, the German high-level system. (Both the U.S. Army and Navy had separate agreements with the British organization). This sharing was just one factor among many aspects of wartime cooperation between the two countries, but it certainly helped build a solid bilateral working relationship.

4 ENIGMA

The ENIGMA began as a commercial machine, around the time of World War I, but was not an initial success – it was too far ahead of its time and too expensive for businesses. However, in the 1920s, the German Navy learned that the British Navy had broken traditional German codes during World War I, and sought a new method of keeping its communications secure. The Navy adopted a modified version of the commercial ENIGMA, and, eventually, the German Army and Air Force also adopted it, making it the workhorse of German military communications security prior to and during World War II. (Note that ENIGMA – a Greek word meaning a puzzle or mystery – was the actual trademark name for the German machine, not a codename bestowed on it by the Allies).

Each German service used ENIGMA machines with slight variants from the others to suit its own special military needs. For example, the German Navy, most security conscious of the services, added mechanical features and changed procedures from time to time. In general, however, the ENIGMA appears to be a simple device: it is an electromechanical device which uses a combination of rotors (rotating disks) and plugs to encipher messages letter by letter. Its mechanical and electrical operations were state-of-the-art in the 1930s, and the Germans had good reason to believe that the ENIGMA would keep their communications absolutely secure.

Despite its obvious strengths, the ENIGMA device was solved by mathematical analysis in the 1930s. The Polish Cipher Bureau, assisted by a little inside information about the machine, came to an understanding of its operational methods and devised methods of solving ENIGMA-based messages. Most of the Polish methods were too slow to be used in combat situations, but the Cipher Bureau's mathematicians invented a machine they called a bombe, which would apply their decryption principles faster than the human hand could work.

Despite good intelligence on the German military, Poland fell when the Nazis attacked – the German military was the best in the world. However, before this happened, the Polish Cipher Bureau shared the secret of the ENIGMA machine, including how to construct bombes with the French and the British.

Great Britain's Government Code and Cipher School (GC&CS), its cryptologic organization, put great efforts into improving the speed and efficiency of the bombe. Among those who worked on it were Alan Turing and Gordon

Welchman. GC& CS also created an efficient organization for intercepting German military communications, processing them and passing the resultant intelligence to commanders quickly. The secret intelligence derived from ENIGMA decrypts was marked with the codeword ULTRA.

When the United States and Great Britain began cooperating in Signals Intelligence during the War, U.S. cryptologic organizations learned much from the British, and, in fact, SIS and OP-20-G modeled their distribution systems on that of their ally. Eventually, to avoid confusion, all high-level decrypts, whether from German or Japanese communications, were labeled "ULTRA."

Signals Intelligence proved a labor-intensive business, even with advanced machines such as the bombe, but, together, the U.S. and British organizations developed a working system that gave their senior leaders and commanders unprecedented types and amounts of secret information.

5 Cryptology during the war

It might be argued that World War II was the first true "information war." It is also true that the United States in many ways found itself ill-prepared for global war at the beginning, intelligence information being one important instance: America's military was forced to fight on unfamiliar fronts where the state of intelligence ranged from barely adequate to non-existent. It should not be surprising then that all PURPLE/MAGIC and ENIGMA/ULTRA material was eagerly ingested by consumers anxious for reliable information. PURPLE, though used by the Japanese for diplomatic communications, became a major ingredient in Allied military decision-making.

The German and Japanese military used a variety of cryptographic systems to protect their communications prior to and during the war. Allied cryptanalysts solved many of these systems over this period, but for the purposes of this article, we will discuss PURPLE and ENIGMA only.

Prior to December 1941, exploitation of PURPLE gave American policymakers access to the instructions sent by Tokyo to their negotiators in Washington and the negotiators' reports of the meetings with State Department officials. This was very useful for U.S. diplomats in day-to-day interaction with the Japanese, and somewhat helpful to the U.S. military, but did not reveal to the American eavesdroppers the most valuable secret of all – the question of war and peace.

The Japanese military did not trust the Japanese Foreign Ministry with details of its operational planning, least of all that a large strike force was moving through the north Pacific toward Hawaii. Thus, although Japanese diplomatic messages in early December 1941 implied that war was a matter of days, perhaps hours, away, the MAGIC material contained no specific clues about the impending attack at Pearl Harbor.

American Army and Naval cryptanalysts exploited several Japanese military systems after Pearl Harbor, enabling U.S. commanders to get an unprecedented window on enemy military operations in the Pacific and South Asian theaters. With the coming of war, the Navy discontinued its work against non-Naval system, thus the continued exploitation of the PURPLE diplomatic system became strictly an Army effort, although, of course, Navy consumers still received distribution of the decrypt intelligence.

By an interesting turn of fate, PURPLE, even though a Japanese system and intended for diplomatic communications, became an essential ingredient in Allied military operations in Europe.

With their fate linked inextricably to Germany's, Japanese leaders demanded accurate and detailed data about all aspects of the war in Europe, and this was forthcoming from their diplomatic stations there – much of it transmitted to Tokyo in the PURPLE system. Thus, the source known as MAGIC provided detailed insight into the thinking and activities of Nazi leaders, Mussolini, and the leaders of Vichy France, among others. MAGIC provided anecdotal evidence of the coordination – or lack of it – among the Axis nations. MAGIC also provided the most reliable source of hard data on the Russo-German front.

The Japanese Ambassador to Germany was Baron Oshima, concurrently a lieutenant general on the active list, who was well-treated by his hosts. His tours sponsored by Albert Speer, for example, led him to write several detailed reports about Germany's new weapons and weapons production.

In late 1943, Ambassador Oshima and some of his subordinates were given an inspection tour of German defenses erected along the coast of France. Their detailed reports not only listed fortifications and shore emplacements, but unit identities, locations and areas of responsibility, training, and locations of reserves. This proved rather useful when Eisenhower was planning Operation OVERLORD.

As the war turned against Germany, U.S. policymakers read in MAGIC the scary prospect of a Russo-German peace settlement, which the Japanese were promoting. MAGIC reports from Japanese diplomats in Berlin and Moscow, however, reassured American readers that Germany declined to pursue this settlement, and that the Soviet Union in any case was standing firm against it.

MAGIC enabled American policymakers to follow both the beginning and the end of the Russo-Japanese Neutrality Pact. MAGIC reports showed the progress of its negotiation and signing in 1941, the vain attempts of Japan to find out whether the Soviets would extend it in 1946, and, finally, the shock to Japan when the USSR announced its intention to abrogate the pact in 1945.

With the defeat of Germany, Japanese diplomats in neutral nations, such as Sweden and Switzerland, began proposing feeble schemes by which Japan could seek a brokered end to the war without unconditional surrender. MAGIC reported these, and also showed that the idea of unconditional surrender was anathema to leaders in Tokyo.

In the European Theater, the ability to decrypt ENIGMA-based military messages quickly became an essential ingredient in combat planning for the British, and the Americans, after their entry into the war, also learned to use its essential insights.

Early in the war, ULTRA enabled the outnumbered Royal Air Force to outfox the German Air Force in its bombing campaigns against Britain. ULTRA intelligence was important to General Montgomery, and later to General Eisenhower, in fighting Rommel's Afrika Corps.

ULTRA was an essential ingredient in winning the Battle of the Atlantic. German U-Boats were a serious threat to Allied shipping, and could possibly have prevented American intervention in Europe. However, decryption of submarine messages enabled the Allied navies to hunt and destroy a large number of subs as well as their replenishment ships. Here, ULTRA had to be augmented carefully with other sources of data, including sightings and direction finding, but was the sine qua non of the U-Boat battles.

Decrypts from ENIGMA-based messages became the staple of Allied secret intelligence in the Italian campaigns and in the drive across France after D-Day. Messages provided copious amounts of data on enemy order of battle and often gave Allied commanders advance warning of impending German attacks.

6 Conclusions

I would offer the following conclusions about the American and Allied experience with PURPLE/MAGIC and ENIGMA/ULTRA:

The U.S. solution of the PURPLE system and British exploitation of ENIGMA gave Allied decision makers continuous insight into what the enemy was saying to itself. This unprecedented – and sustained – insight into the activities, thinking, and intentions of our enemies worldwide and at the highest levels gave the Allies an incalculable advantage.

The ability to read the PURPLE system provided much wider access than might have been expected. It not only produced information on the immediate users, that is, the Japanese diplomatic service, it gave the Allies a superior source of information useful in both theaters of war and for military purposes as well. It might be argued that PURPLE at some stages was more important in the struggle against Germany than it was in the fight against Japan itself.

The ability to exploit the RED system was the catalyst which spurred the growth of SIS from a tiny cadre into a large and active organization and prepared the organization to take advantage of the later solution of PURPLE. It also necessitated the development of mechanisms for widespread collection and dissemination of SIGINT materials and the doctrine under which these activities would be conducted.

The facts that exploitation of the PURPLE system was manpower-intensive and speed of processing was imperative, forced the two branches of the military services, even while they were domestic rivals for resources, to cooperate to produce this source in a timely way. This helped prepare the ground for increased interservice cooperation in the post-war era.

Most senior and many mid-level military and civilian leaders in the war had some level of access to MAGIC reports. They retained their memories of World War cryptologic successes, and supported the continuation of cryptologic capabilities in the brave new post-war world.

The solutions of PURPLE and ENIGMA were intellectual accomplishments of the first brilliance. Both were team efforts and the members of these teams ought to be listed among the important contributors to victory in World War II. Unfortunately, however, for most of the contributors, their names and deeds are generally known only to the few who are interested in intelligence activities.

Finally, SIGINT did not win World War II. The war was won by those sailors, soldiers, airmen, and marines who took the fight to the enemy at the risk and, sometimes, cost of their own lives. However, Signals Intelligence gave commanders inside information about the enemy in such detail that it allowed them to make wise decisions that saved uncounted thousands of American and British lives and shortened the war by many months.

That is what is expected of an intelligence organization in wartime, and, in World War II, the American and British SIGINT personnel delivered their product with war-changing and lifesaving effects.

The Geheimschreiber Secret

Arne Beurling and the Success of Swedish Signals Intelligence [1)]

Lars Ulfving[1] and Frode Weierud[2⋆]

[1] HKV/MUST, S-10786, Stockholm, Sweden
[2] CERN, Div. SL, CH-1211 Geneva 23, Switzerland

Preface

The present paper appears under joint authorship, however, the responsibilities of the two authors are divided and well defined. Lars Ulfving is the sole author of the original Swedish version of *The Geheimschreiber Secret*[1], while Frode Weierud alone is responsible for the translation into English, the translator's notes, and postscript, as well as the bibliography and the appendixes.

The translation has been kept as close to the original language as possible. Specialist terms and expressions have been retained where possible. Otherwise, a substitute term with the closest possible meaning has been chosen and an explanation given in the text.

The original notes appear as in the original, as footnotes at the bottom of the page. Other short notes in brackets appear as in the original. Original references are marked with superscript numbers, while the references themselves are placed at the end of the text.

The translator's notes are of two types: short notes included in the text in brackets and in italic (*translator's note*), and normal notes which are marked by bold, italic numbers in square brackets, e.g. [***1***].

The postscript, based mainly on information from Bengt Beckman's book, fills in some of the missing personal histories and brings the account up-to-date with present historical knowledge. Two appendixes and a bibliography have been added to place this account in its cryptological context and as an incentive to further study.

1 A short historical résumé until spring 1941 [2)]

At the beginning of 1941 Sweden was in a situation that had drastically deteriorated during the previous year. The end of the Winter War (*Russo–*

[⋆] This article represents the views of the author but not necessarily those of his employer or any other third party.

[1] Original Title: "Geheimschreiberns hemlighet, Arne Beurling och den svenska signalspaningens framgångar" In "I Orkanens Öga, 1941 – Osäker neutralitet" edited by Bo Hugemark, Probus Förlag, Stockholm 1992

Finnish war) and the signing of peace in Moscow on 13 March 1940 were traumatic events in Sweden as well as in Finland. However, the conditions were not so devastating as they could have been had the Soviet war aims been achieved. Finland survived as an independent state, but within tighter borders.

The surprising and successful German attack on Denmark and Norway on 9 April 1940 brought serious consequences for Sweden. Sweden's political freedom became seriously limited. Threats from new directions had quickly to be taken into consideration. In one area, however, the German demands on Sweden created unexpected possibilities that were exploited well. Their request to hire telegraph lines going through Sweden made great successes for Swedish signals intelligence possible.

In mid-summer 1940, after the French capitulation and the Soviet occupation of the Baltic states, the Swedish intelligence service was faced with two important questions. Would Germany carry out "Operation Seelöwe", a naval invasion of Great Britain, during the autumn? Would the Soviet Union again attack Finland to re-establish Russian borders on the northern shore of the Gulf of Finland while Germany was engaged on the western front? There existed no guarantees that Stalin would wait until the German air war might lead to air supremacy over the British Isles or that he would wait for the German invasion. After developments in the Baltic, Stockholm considered it likely that the Soviet Union would attack Finland at the turn of the month July-August even before, or perhaps even without, a German invasion attempt on Great Britain. Germany would then probably take the same attitude as during the Winter War, i.e. benevolent neutrality. Indications, chiefly reported by attachés from Riga, Berlin and Moscow, showed such a development. The Finnish government was under enormous Soviet pressure, while Moscow-led communists tried to create internal trouble in the country. The pattern was the same as that before the Baltic states were occupied by the Soviet Union in June.

When, on 13 August 1940, Foreign Minister Günther explained the current situation to the Foreign Affairs Committee, his opinion was that there existed a real danger of such an attack. The German military attaché in Riga was given as the source. It is remarkable that no Swedish sources or Finnish military authorities' views were used or referred to. The Defence Staff's intelligence department was therefore also obviously restrained in its written reports to the military command and the government.

German fear of a Russian attack, however, caused Hitler to free the arms embargo against Finland shortly before 10 August. The arms' deliveries that had stopped under the Winter War were now let through. This political change of course indicated a renewed German interest for Finland's continued existence as an independent nation. The tension in Helsingfors was relieved. The immediate danger of a Russian attack was estimated to be over for the time being. As a result it was considered necessary that the Defence Staff's intelligence department concerned with Finland should be strengthened. Now, it was not only Finnish–Soviet relations but also German–Finnish relations that had to be watched. Since neither Finnish nor German military contacts were very communicative any longer, the task was not particularly easy.

The extraordinary Swedish military contacts in Finland were evidently about to be reduced because of Finland's German connections. The attaché reports from Berlin also showed a clear German interest in Finland. Speculations about a German attack on the Soviet Union started to appear in the reports from Berlin. Neither the Swedish attachés nor others could obtain any further information about Hitler's intentions. They were reduced to making assumptions about probabilities and possibilities. A German two-front war, however, was not considered probable by the Defence Staff's intelligence department. In autumn 1940 the situation did not seem particularly alarming to the Swedes. The concessions made, in response to German demands for troop transit and continued iron ore exports, appeared to be sufficient.

The reports from the military attaché in Moscow during the winter of 1940 no longer indicated a Russian build-up and deployment of forces against Finland. The Russian build-up of forces was instead concentrated in the border areas in the south and south-west, against Bessarabia and Bukovina. In Finland, however, they still felt that a new Russian attack was a real possibility.

As a consequence of the commonly perceived threat, the military intelligence exchange between Sweden and Finland which, as already explained, had diminished after the end of the Winter War, was again improved. However, once more the Finnish interest for co-operation diminished considerably during spring 1941, principally during and after April. Attaché von Stedingk in Helsingfors reported simultaneously that German–Finnish relations had dramatically improved. He also reported that Finnish contacts appeared to be prepared to take part in a German war against the Soviet Union, which German contacts said, with a surprising frankness, would start in early or mid-June. Colonel Carlos Adlercreutz, the chief of the Defence Staff's intelligence department, had during a conversation with the chief of the Finnish General Staff, General Heinrichs, clearly seen the possibility of Finland joining in a German attack on the Soviet Union, or being forced to participate in such an attack.

German probes about the transit of German troops from Norway to Finland through Sweden had already taken place in February. Sweden feared that these explorations would change into direct demands, but wondered if Hitler would be content with demands for transit if he intended to start a two-front war against the Soviet Union before Great Britain had been finally defeated.

The uncertainty about Hitler's intentions was brought to a head during the so-called "March crisis" that culminated in a large preparedness alert on 15 March. The alert was actually caused by a German communications error. In fact, there was no other intelligence about a German attack. However, the information obtained through signals intelligence made Swedish preparatory measures possible against regions where German units were positioned in Norway.

The deployment and build-up at the German eastern border now started to be obvious and was difficult to conceal. However, an uncertainty still prevailed as to whether Hitler really intended to attack the Soviet Union before Great Britain was conquered. Perhaps the build-up aimed only at applying pressure — not a two-front war. How — or if — the received information was analysed at the Defence Staff and how — or if — the information in the military reports was weighed together with the rather uncertain and speculative diplomatic information, can no longer be clarified. At a presentation for the government on 21 April General Olof Thörnell, the chief of the Defence Staff, judged that a German–Soviet war was probable, and that in such a conflict Finland would participate on Germany's side. From the presentation it became clear what vague information was the basis for this judgement. The deception measures taken to protect the planning were hard for other intelligence services, including the Swedish, to penetrate.

The Soviet military intelligence service was informed about the coming attack. Nevertheless, Stalin refused to believe that the attack on the Soviet Union would start before Great Britain was conquered.

In one area the Swedish intelligence service was considerably in the lead. In spring 1941 the Swedish signals intelligence service furnished very interesting, extensive, accurate and unique information about German military dispositions in the vicinity of Sweden.

2 Swedish signals intelligence and intelligence service before and during the Second World War [3)]

In 1936 a resolution was passed about a new defence order which came into force on 1 July 1937. The resolution included provision for the establishment of an intelligence department, a signals intelligence department and a cryptology department.

However, the prerequisites for an effective intelligence service were not so good. Swedish intelligence services in the modern sense of the word had indeed been already established in the beginning of this century. The armed forces intelligence service had increased in 1905, during the Union crises, and in the First World War. The General Staff and Naval Staff of that time both had their own signals intelligence and cryptographic units. However, in the inter-war period less and less was done. The knowledge acquired in signals intelligence and cryptanalysis was lost. Politicians of that time did not understand the importance of a well-functioning intelligence service and consequently they did not grant any appropriations for this purpose. Nor did the Defence Commission, which was appointed in 1930 and on whose report the 1936 Defence Resolution was based, take any appreciable interest in the intelligence service. The General Staff's foreign department did not constitute a solid enough foundation for such a service. No special agency for cryptanalysis existed before the Defence Staff was established, although the cryptographic departments at the Naval and General Staffs had some success during the First World War. The encrypted radio traffic of the Russian Baltic Fleet could be broken to some extent. The General Staff probably broke German diplomatic traffic periodically.

The breaking may have taken place even earlier, but solid information about this is missing. These breaks, however, were probably quite sporadic and had a "chamber character", i.e. rather amateurish.

From the summer of 1928 signals intelligence operations were carried out fairly regularly under naval direction. In the beginning it took place from the warship *Sweden* (*Sverige*) and from the summer of 1929 from several ships in the coastal fleet. From October 1929, signals intelligence activities were also carried out from naval coastal radio stations.

The first attempts to develop this branch of the intelligence service were made by the Navy. During the years 1930–31, the Naval Staff had already organized a course in cryptology and cryptanalysis. Ships in the coastal fleet started the systematic interception of foreign radio traffic in spring 1931. Later professional intercept operators were trained on the warship *Queen Victoria* (*Drottning Victoria*). The first successful attempts to break foreign cipher traffic were made in spring 1933, when they succeeded in breaking the cipher then used by the OGPU (later the KGB). These breaks into foreign military ciphers were probably the first to be made in Sweden after the First World War. The naval cryptography courses of 1930–31 were repeated in 1932–33 and 1934–35. An agreement was reached between the Naval and General Staffs to run these courses alternatively every second year.

The instruction was based on theory with special exercises. The real material available was too complicated to be used in the teaching. Even if these cryptanalysis courses did not result in real breaks, they were nevertheless of great importance as they created a small cadre of trained theoretical cryptanalysts, consisting of both active and reserve officers together with conscripted students. Later on civilians from the University of Uppsala, among others, were also trained as cryptanalysts. One of these students was the mathematics professor Arne Beurling.

When the future Defence Staff organization was analysed in 1935–36, cryptology-committed interest groups succeeded in pushing through the establishment of a department for cryptography and cryptanalysis — the crypto department. In some quarters a crypto department was considered unnecessary but, in spite of opposition, one was set up during the final stages of establishing the Defence Staff. Sections I to III were intended to deal with the cryptographic security of the Army, Navy and Air Force. The fourth section, crypto section IV, was intended to be a cryptanalytic section. Thus, the foundation was created for a central cryptanalytic organization. It was in crypto section IV that the Geheimschreiber traffic would later be broken.

Radio interception was taken care of by the Defence Staff's signals department. However, the actual signal-intercept work was completely carried out by the Navy, which was the only force with access to qualified intercept personnel.

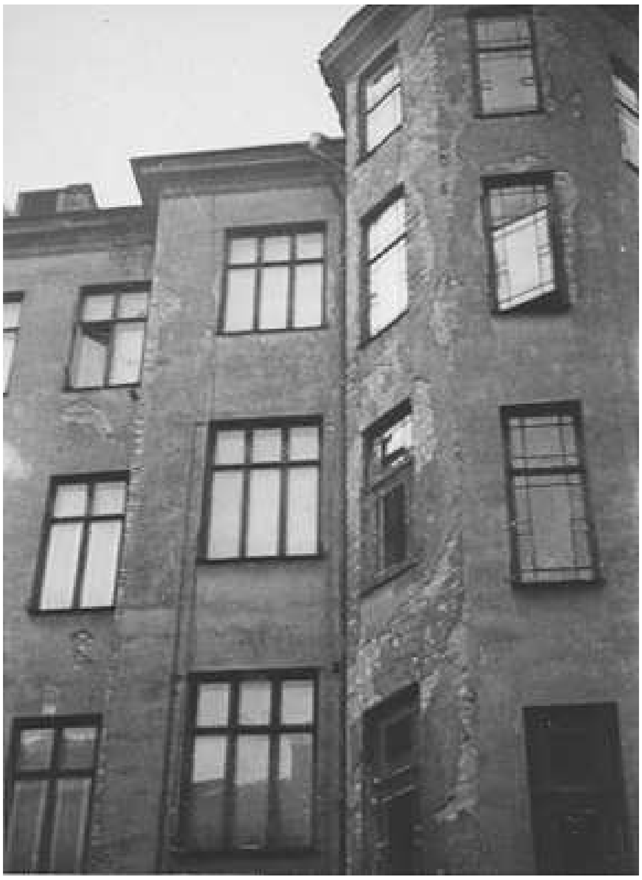


Figure1. Dilapidated house in the back garden on Karlaplan 4.

The crypto department led an ambulatory existence during 1936–40. In the beginning, it was housed in the staff building "Grå Huset" (*The Grey House*) in Östermalmsgatan 87; later in a house on Lützengatan. [*1*] During the summer of 1939 the approaching war became more evident and the department intensified its mobilization preparations. At the outbreak of war it transferred to the premises of the Military Academy, where it had the top floor at its disposal. Soon it became overcrowded, which is why crypto section IV moved to a property on Karlaplan 4, consisting of a building facing the street and a dilapidated house in the back garden. The conditions were rather primitive. The furnishing was of the utmost simplicity, consisting of folding tables and simple wooden chairs. However, it was here that the crypto department was going to perform great achievements.

3 Breaking of the German encrypted telex traffic — the breaking of the Geheimschreiber [4)]

At the time of the German attack on Denmark and Norway 9 April 1940, Germany demanded that there should be no interruption of their telecommunications transmitted over Swedish lines. After a positive answer, the Germans gradually hired lines in Sweden for the connections Oslo – Copenhagen – Berlin, Oslo – Trondheim, Oslo – Narvik and Oslo – Stockholm. The German embassy in Stockholm already possessed a line for their traffic to Berlin. Later on they hired a line for the connection Stockholm – Helsingfors. The interception of the German telegraph lines was the fundamental condition for the future successful breaking of the German encrypted messages. The Swedes had, by these means, access to large quantities of genuine information. Wartime arrangements allowed foreign rented telecommunication lines passing through Swedish territory to be tapped, without breaking Swedish law.

We shall here for the first time in the open literature show the principles involved in the breaking of the Geheimschreiber. The Germans considered this cipher machine to be extremely secure, but their confidence resulted in an imaginary security. German carelessness and Professor Arne Beurling's genius exposed the secret of the Geheimschreiber.

3.1 A note about teleprinters

To facilitate an understanding of the subsequent explanations we will give a short technical review of teleprinters.

The first teleprinters were constructed at the end of the 19th century. The principles have remained largely unchanged since then. Every character transmitted consists of a combination of pulses of two types. The number of pulses in a character is always five, contrary to the varying number of pulses in the Morse code alphabet. All pulses are of the same length, and are indicated by positive or negative polarity or alternatively, current or no current. Five pulses taking on two different states give 32 different combinations, but that is not sufficient for all the characters that have to be transmitted. There therefore exists an arrangement with the same function as the letter/figure shift key on a typewriter. A combination of pulses causes all subsequent characters to be received as number or punctuation characters, and another combination signals in a similar way that all the following characters will be received as letters. Several teleprinter alphabets have existed. The one mostly used is called the Murray code after its inventor, or the International Teleprinter Code (*International Telegraph Alphabet No. 2*). The text is punched on paper tape which is fed into the transmitter. 'Hole' or 'no hole' in the tape corresponds to 'current' or 'no current', or to 'positive polarity' or 'negative polarity'.

3.2 Teleprinter encryption

Soon after the first teleprinters were put into operation, equipment for the encryption of teleprinter signals was constructed. The method first used was simple. Two identically-punched key tapes, one for the sender, the other for the receiver, are produced. They are then glued together in a loop of manageable length, about 1000 characters. The sender punches his plain text tape and places it in a tape-reader. He then comes to an agreement with the receiver about how the key tapes will be placed in their respective tape-readers, and the transmission starts. In a simple relay circuit a modulo-two addition is performed on the characters from the sender's two tape-readers. A character on a tape can be regarded as a binary number: a combination of 'holes' representing ones, and 'no holes' representing zeros. A modulo-two addition of two such numbers signifies an addition, without carry, of each bit in corresponding positions. The result of this addition forms the cipher character that is transmitted. The same procedure is used in the receiver. After each transmitted character all tape readers are stepped one position forward and the whole process repeats. The cipher methods gradually evolved. Instead of key tapes a number of code wheels with pins were introduced, e.g. five wheels , one for each channel in the key tape. An active pin had the same function as a hole in the corresponding channel on the tape. Thus it was no longer necessary to punch the plain text on tape and then transmit it later. The teleprinter could be directly connected to the cipher equipment, hence it was possible to transmit and receive in "real time", so saving a lot of time. In the crypto department this encryption method, consisting of adding a key character to a plain text character, irrespective of how the key character was generated, was called "overlaying". This term is used in the following text.

3.3 The Geheimschreiber

The German company Siemens developed a mechanical teleprinter cipher machine in the 1930s that was the first in a series of such machines. The generic name was "Der Geheimschreiber", which the crypto department called "G-skrivare" (*G-writer*). In addition to the previously mentioned classical type "overlaying", it also made a permutation[2] of the pulse order as another encryption function.

[2] Permutation: A permutation of a sequence of n numbers corresponds to a reordering of the sequence. Two permutations are equal when the numbers are placed in the same order. For n = 2 there are two permutations (1,2) and (2,1), for n = 3 there are six (1,2,3), (1,3,2), (2,1,3), (2,3,1), (3,1,2) and (3,2,1). Generally it can

The crypto department later used the expression “transposition” for this permutation. A polarity inversion was made with five relays, and five others took care of the transposition. [2] The relays were controlled by ten coding wheels which, through a set of plugs and jacks, could be connected to relays in an arbitrary way. The principle of transposition of the teleprinter pulses is not suitable for all types of teleprinters. The five pulses must be available simultaneously in the transmitter and the receiver for a permutation to take place. In the transmitter this is not so difficult to arrange, but it is much more difficult in the receiver. However, Siemens had solved the problem for the receiver even without access to modern digital technology. All functions were mechanical in the cipher machines of those days. An incoming character’s five pulses, positive or negative, charged five capacitors in sequence. When the fifth pulse was received the information stored in the capacitors was simultaneously transferred to five polarized relays. These relays were part of a circuit that selected the character to be printed. During this transfer it was possible to produce a transposition by changing the connections between the capacitors and the relays.

The Geheimschreiber’s ten code wheels had the periods 47, 53, 59, 61, 64, 65, 67, 69, 71 and 73. In the first models all the wheels moved one step for each enciphered character. Since the wheel periods were relatively prime, that is they had no common factor, the total period of the machine — the number of steps the machine must make to return to its starting position — was equal to the product of all the individual wheel periods, that is 893 622 318 929 520 960 steps. This number also indicates the number of possible wheel starting positions.

The “transposition circuit”, that is the insertion of the “transposition relays” between the rows, could be varied. Eight basic patterns were possible, each with 2 612 736 000 variations. [3] The combinations of connections and wheel adjustments were, before the creation of the computer, considered to be extremely large numbers. In addition there were the number of ways of connecting the code wheels to the relays. This may have given the Germans the impression that the Geheimschreiber was a very secure cipher machine. It was probably considered to be more secure than the Enigma machine which was intended for tactical use. The Enigma had, for example, a period of 17576. [4]

The Geheimschreiber was gradually developed and several models were brought into service. The first machine the crypto department came in contact with was called T52a/b. Later T52c, d and e came into service. There were also variants of the different models. However, they were all based on the same basic principle.

At the end of 1941, a new machine designated Z appeared in the traffic. It did not belong to the A/B-series and it was called Geheimzusatz 40 [5] and

be shown that the number of permutations of n numbers is $1 \cdot 2 \cdot 3 \cdots (n-1) \cdot n = n!$ (n factorial).

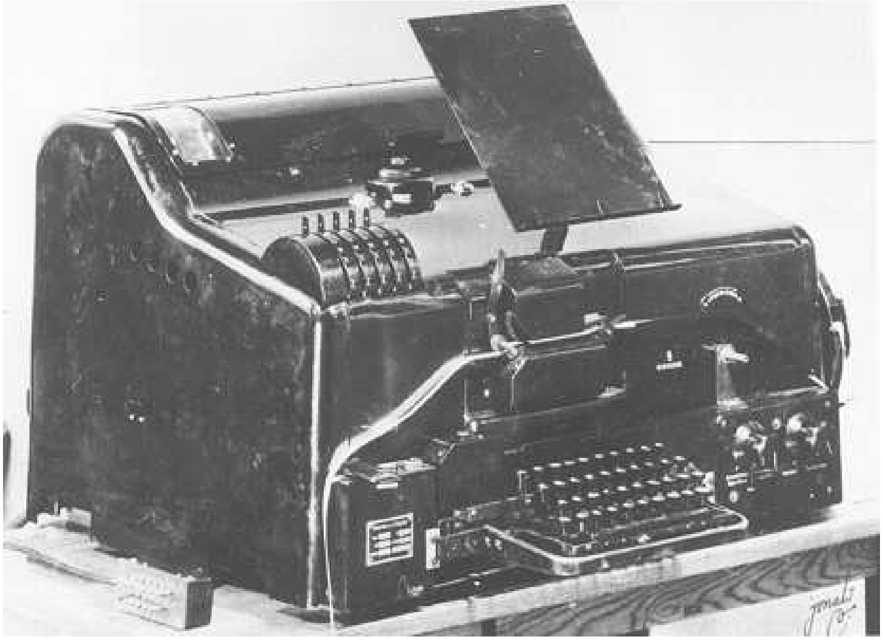


Figure2. “Geheimschreiber” or “G-skrivaren” (*Schlüsselfernschreibmaschine T52c*).

not Geheimschreiber. It was a stand-alone attachment that was connected between the teleprinter and the transmission line. It could therefore have been used together with teleprinters other than the Siemens machine that the Geheimschreiber was based on.

3.4 Arne Beurling

Who was Arne Beurling? According to *Svensk Uppslagsbok* (*Swedish Encyclopaedia*) he was “Arne Karl August Beurling, born 5 February 1905, mathematician. Beurling defended his thesis in Uppsala in 1933 (Etudes sur un problème de majoration), senior lecturer same year, Ph.D in 1934, professor in 1937. Beurling is an ingenious and all-round scientist who has attained beautiful results in function theory, prime number theory, modern integration theory and in several other areas.” After the war Arne Beurling was offered an excellent position at Princeton University in USA in 1954. In 1965 he was given Albert Einstein’s office, No. 115, at The Institute for Advanced Study, a distinction granted to very few people. Arne Beurling died in 1986.

It is now known that Professor Arne Beurling was the man behind the breaking of the German Geheimschreiber. David Kahn writes in “The Codebreakers”, page 482: “Quite possibly the finest feat of cryptanalysis performed during the Second World war was Arne Beurling's solution of the secret of the Geheimschreiber." Arne Beurling's greatness is given by the fact he had at his disposal only the teleprinter tapes with the cipher text. He had no access to any machine, no plain text and no knowledge about the logical construction of the cipher machine. Everything had to be reconstructed, something which was done in a remarkably short time.

It is known that he based his analysis on only 24 hours of traffic intercepted on 25 May 1940. A quick analysis showed that the first assumptions probably were correct. A check was made with the traffic intercepted for 27 May. Two weeks later the construction principles for the cipher machine were solved.

On the other hand it is not known how he set about it. That secret Arne Beurling took with him in the grave. However, a reconstruction has been made by FRA (*Försvarets Radioanstalt*). The credit for this reconstruction goes to Carl-Gösta Borelius who served at the Defence Staff's crypto section, later on FRA, from 1941 to 1985. Borelius' description of the reconstruction work is the basis for what is shown here.

3.5 The reconstruction

There was, of course, a procedure for indicating the Geheimschreiber key settings. One setting was the inner one, that is the selective connection [***6***] used for the connection between the ten code wheels and the previously mentioned inversion and transposition relays. The inner setting was in force for three to nine days, starting at 9 o'clock on the first day.

When a message was to be transmitted, the code wheels had first to be set to a given position. These settings had to be the same for both sender and receiver. The transmitting station would select a setting for five consecutive wheels. This setting was transmitted to the receiving station with a three-character so-called "QEP indicator". The five remaining wheels were set to a predetermined key value that was valid for all messages during a 24-hour period. This setting was called "QEK". The daily key list indicated which wheel would be "QEP-wheel" and which setting the "QEK-wheels" should have.

It should be pointed out that the number 3 = letter shift, 4 = figure shift and 5 = space in this teleprinter alphabet. This has great importance in the following explanation. [***7***]

When the transmission of a cipher message was about to start, the transmitting station would present itself with "Hier MBZ" (*MBZ here*) and would then ask if the message could proceed, "QRV". If this was the case, the receiving station answered with "KK". The transmitting station then sent "QEP" succeeded by five two-digit numbers (e.g. 12 25 18 47 52). Both operators then adjusted the wheels in their respective machines, partly the "QEK-numbers" after the key list for the current day, and partly the "QEP-numbers". When the transmitting station was ready it would transmit "UMUM" (umschalten

— *switch over*) and when the receiving station was ready it would answer "VEVE" (verstanden — *understood*). Then they switched over to cipher mode and the transmission of the text itself started. The cipher texts were consequently always preceded by "UMUM" and were therefore easy to retrieve in the large number of signals.

It is possible that Beurling had knowledge of the Siemens & Halske patent, but this is not certain. Borelius recounts that when Beurling visited FRA on 15 November 1976, he reacted strangely to questions about the first break. He evidently did not like the questions to be put. He nevertheless said that he made use of "threes" and "fives" in the texts.

Telecommunication technical problems were a great help during the breaking. The telegraph lines were long, sometimes bad, and therefore often exposed to interference, which could distort a transmitted character. The readability was nevertheless not disturbed except when the character changed to a "4" (= figure shift), because then all succeeding text became an unintelligible sequence of numbers and punctuation characters. If the distortion affected only the receiving station, the transmitting station did not notice anything, and continued the transmission. To reduce the problem, the operators normally used to write "35" (= letter shift, space) instead of only "5" (= space) between the words. Thus the consequence of a false "4" would be restored at the next space between the words.

Beurling discovered that when the plain text of "3" and "5" had one pulse the same and four different, this had also to be the case in the enciphered state. For a guessed "3" there consequently existed only five possible "5" or vice versa. It was therefore relatively easy to establish spaces between words, which would have facilitated further work. It was probably this which Beurling talked about when he alluded to "threes" and "fives". Hence a guessed "3" gave only five possibilities for Q and V in "QRV", which asked if the message could proceed. In this way further work was greatly eased.

It also seemed natural to suppose that a part of the encryption process consisted of a transposition of the five-pulse-characters pulse positions. A number of comparisons could give the transposition arrangement.

Beurling tried in this case to trace back the cipher character to its appearance before the transposition. Then he made his next observation. The change from one character in the plain text to a cipher character after the "overlaying" consisted of a change in polarity for some of the pulses of the plain text character, and for all characters in a column it is always the same character that changes.

We can assume that Beurling now introduced his observations on his "work sheets" (*"avvecklings-papper"*).[3] In five rows under the examined text he placed in every column a dot for pulses with inverted polarity and a

[3] Avveckling: A technical term for the elements of work that took place between the interception of a message and until the plain text could be extracted. That is, the work consisted of extracting the current cipher key.

circle for those that did not change. When the emerging five-dot combination resembled a teleprinter character, it was called an "overlaying" character. Under this character was written the permutation order, that is the so-called transposition. Gradually it was detected that the pattern of circles and dots in the five rows of the "overlaying" repeated after a number of characters. Beurling then supposed that the pattern was produced by pin-wheels like those in the Swedish cipher machine, invented by Boris Hagelin.

Subsequently he continued the work with the transposition. It turned out that if for example "pulse 2" ended up on "place 3" then the fourth circuit connection had to be open, otherwise it was not possible. If this circuit connection was controlled by a wheel with even distribution of active and inactive pins, the pulse would end up on "place 3" in half of the events, in the rest it would fall on some other place. A converse conclusion should have been possible, using an inverted argument. The complete circuit and hence the details of the remaining wheels were obtained through hypothesis tests. By these means Beurling would have got the break that revealed the Geheimschreiber secret, as Borelius' reconstruction shows.

3.6 Imperfections, errors and laziness

It has been mentioned earlier that the telecommunication connections were bad. But it was not only single characters that were distorted and therefore gave an opening for breaking.

The abundant number of "parallel texts", that is an enciphered text sent several times with the same key setting on the cipher machine, were a big help in breaking the Geheimschreiber. In extreme cases the same message was sent 20 to 40 times with the same setting. How could this happen? One of the basic rules of cryptological security is never to send the same or a different message with the same key setting. Bad connections and distortions together with laziness gave many openings of this type.

The keying procedures have already been described. It takes a certain time to adjust the wheels by hand. To facilitate the adjustment of the "QEK-wheels", that is the current key for the day, there existed a cursor that easily could be moved around on the wheel and positioned on an arbitrary key number.

The wheels could be freed with a locking arm. All the wheels could be turned backwards with a handle until their cursors came to a home position when the wheel stopped. This handle sat on the right on the front of the machine under a plate with the inscription "LANGSAM DREHEN" (*turn slowly*). The idea was that every morning the cursors would be set on the five wheels that were intended for the daily key (the QEK-wheels). Later these wheels could quickly be returned to their agreed positions before every new transmission. Unfortunately it became a habit that even when the "QEP-numbers" were set, the cursors were set to the key values and then the wheels were cranked back. This was against the existing rules, as new "QEP-numbers" should be selected for every new message.

Since the lines were long and sometimes poor there were often distortions. These sometimes caused one of the machines to interpret the distortion as a character. The operator of the other machine would not notice anything. One of the machines would then be one step ahead of the other. The cipher machines were no longer in phase and the message became unintelligible. When this happened, it was necessary to break the transmission, switch over to plain text, choose a new message key and continue the transmission.

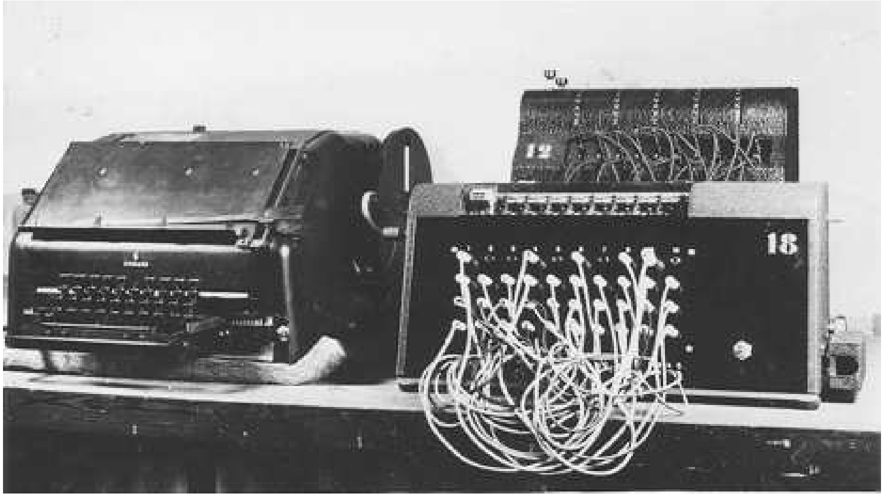


Figure3. Decryption unit for German line traffic, ex. "APP".

Now the big mistake was made. For each break in the transmission of a message due to continuous distortions on the line, the operators chose the easy solution of simply cranking back the wheels to the previous setting. In this way the cryptanalysts got their parallel texts with all their errors and flaws which gave a large number of opportunities for breaking it. The high security of the Geheimschreiber became therefore in many ways simply illusory.

3.7 Interception and preparation

As mentioned earlier, the traffic on the German hired telecommunication lines through Sweden was intercepted. It was quickly discovered that apart from plain text, encrypted text was also transmitted. When Beurling found how the Geheimschreiber functioned, Swedish technicians under the leadership of

Viggo Bergström started to construct special machines for decryption after directions from Beurling. [*8*] These machines very soon made it possible to follow the frequent changes in cipher keys and subsequently quickly extract the plain text. The machines were later built in quite large numbers in L.M. Ericson's workshop for precision mechanics.

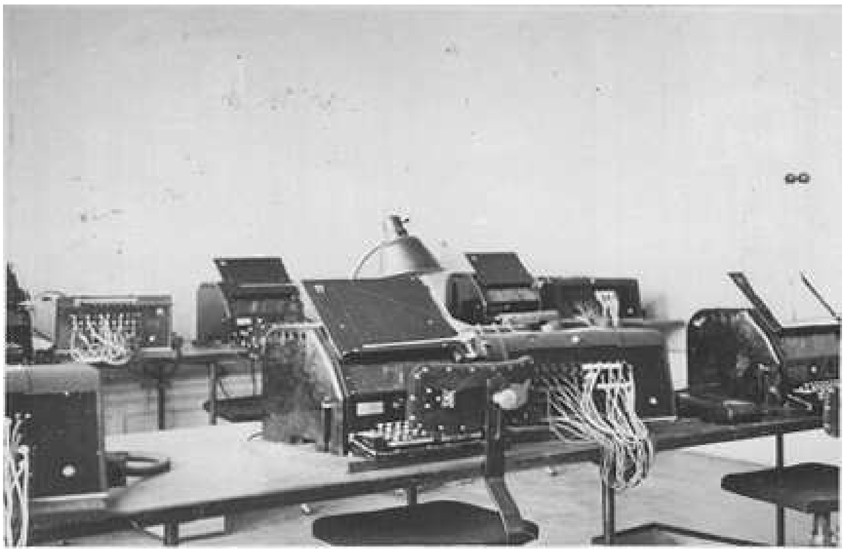


Figure4. The decryption units, "APPARNA", at their working places.

The German Geheimschreiber traffic was broken and deciphered from June 1940 until May 1943. This went on even when new models were introduced and the key procedures were gradually changed.

The intercepts came from a number of teleprinters in a room in Karlaplan 4 which were connected to the different lines between Germany – Norway, Sweden and Finland. The machines, which were very noisy, were supervised 24 hours a day. The texts came out in a never-ending stream of paper tapes, which were then glued on to big sheets of paper.

The daily routine was the following: In the morning (after 9 o'clock, when the daily key was changed) the cryptanalysts examined the incoming traffic, waiting for a case with sufficiently many "parallel texts" to occur. As soon as possible, the cipher key extraction ("avvecklingen") took place. When the new key settings were produced they were given to the staff who worked with the deciphering machines, and the deciphering of the day's harvest could start. Subsequently the plain texts were cleaned up and typed. They were later given to the various consumers of intelligence.

As mentioned above, two types of traffic were observed and studied. The most extensive were the military traffic and the traffic between Berlin and

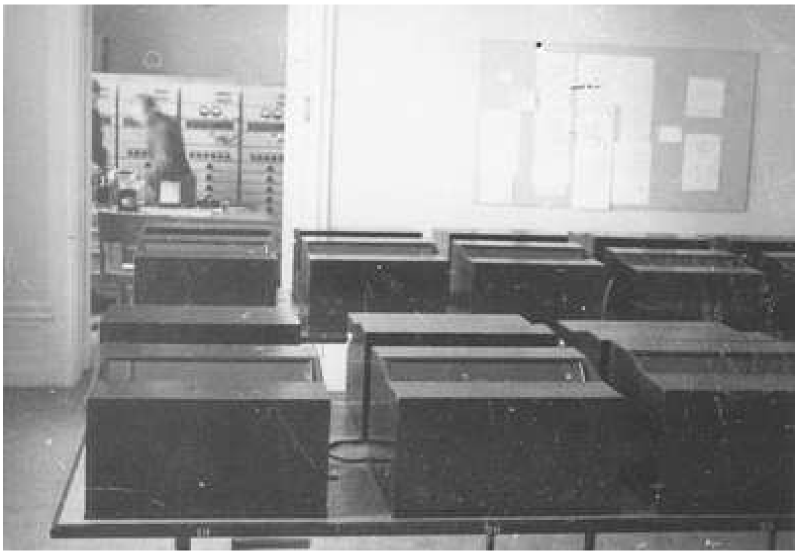


Figure5. In the machine room. (*Room with teleprinter-receiving machines connected to the intercepted lines.*)

the embassy in Stockholm. The diplomatic traffic had the highest priority, as this concerned Swedish–German relations. At least two key settings therefore had to be determined every day.

At first, the teleprinters used in Karlaplan 4 were machines from the American firm Teletype. Teleprinters were, however, in short supply as importing them was difficult during the war. However, the Royal Telecommunication Administration (*Kgl. Telegrafverket*) were persuaded to surrender a number of their teleprinters, which resulted in a return to Morse telegraphy on some lines. When the crypto department later succeeded in obtaining a batch of Siemens teleprinters, these replaced the Teletype machines, which was certainly reasonable considering their use.

As mentioned earlier, when newer versions of the Geheimschreiber were later introduced, attachment units were constructed and connected to some of the deciphering machines that were used for the C model traffic. For the Z-traffic only one deciphering machine was built. [***9***]

Large quantities of messages were decrypted and distributed. This extensive traffic resulted in an increase in the number of staff. The number of teleprinters and deciphering machines also increased until they reached a total of 32.

The decryption of the Geheimschreiber traffic developed gradually into a real industry needing a lot of people. In 1941 the staff increased to 500 people and later on it became even bigger from time to time.

Table1. Number of distributed messages.

Year	Encrypted	Unspecified	Unencrypted	Total
1940		7100		7100
1941		41400		41400
1942	101000		19800	120800
1943	86600		13000	99600
1944		29000		29000
Total	187600	77500	32800	297900

The highest number of messages distributed in one day (October 1943): 678.

The breaking of the Geheimschreiber was a large contributory factor to the establishment of FRA (formed from the Defence Staff's signals and crypto departments) as an independent authority on 1 July 1942.

In May 1943 the keying principle was changed with the result that further deciphering became impossible. A small group stayed on to handle those messages that for different reasons had not been deciphered earlier.

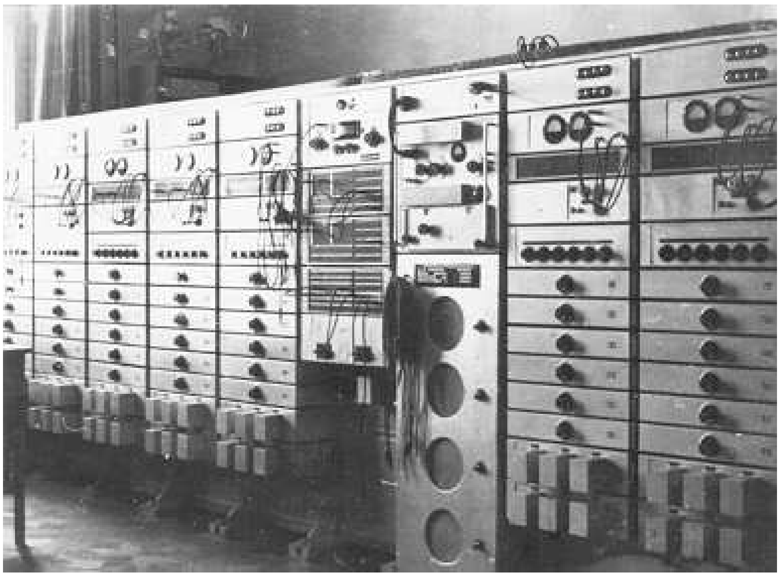


Figure6. View from the rack room. (*Room with equipment racks where the intercepted telecommunication lines entered.*)

4 On the eve of Operation Barbarossa: How was the intelligence used? [5)]

What has just been described, it must be emphasised, was a spectacular performance even on an international scale. As far as known the Siemens Geheimschreiber was not broken in any other country. [***10***] It was also an extraordinary yield based on a relatively limited investment. Beurling's reluctance to explain how he did it could be due to the fact that he solved the problem so easily and quickly that he found it too easy to arrive at the solution. But in reality genius shows itself in simplicity and its accompanying excellence.

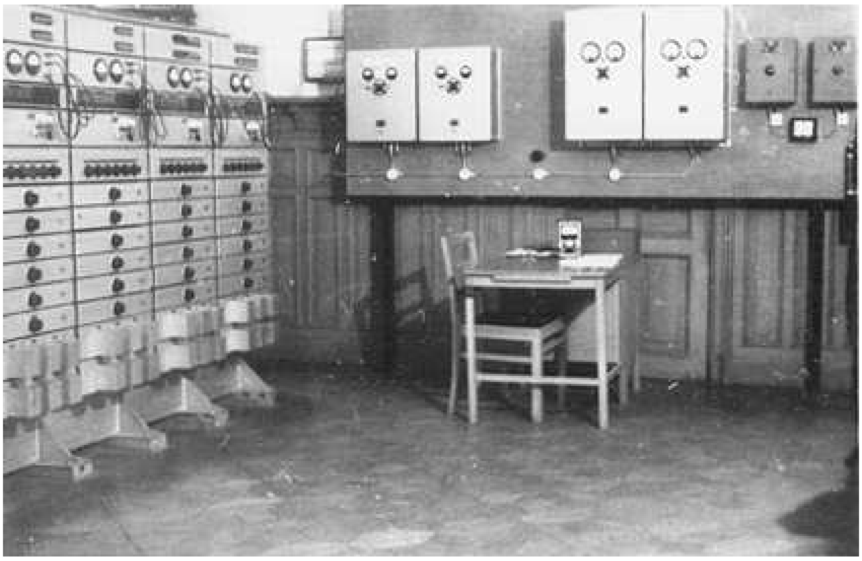


Figure7. Another view from the rack room.

In 1940 the crypto department had been developed with mostly rather new staff. Its technical and organizational achievements were therefore considerable, since during autumn that year it began to distribute German messages in ever-increasing numbers. German unit compositions and their position, together with military and political deliberations and directives were in this way known to the Swedish authorities, sometimes almost at the same time as the real addressee.

The German traffic was not particularly alarming during the autumn of 1940. Hitler was hardly interested in Sweden politically. The planning for the attack on the Soviet Union was still in its infancy. The directives for Operation Barbarossa were first drawn up in December 1940. This gave the crypto department a respite that was used to build up and render more effective the breaking, analysis and delivery routines. The Defence Staff could therefore prepare methods for handling the decrypted material at a time when it was not under any particular pressure.

Texts "of strategic importance or of an obviously secret nature" were delivered directly to Adlercreutz, as chief of the intelligence department, who normally submitted them to his superiors Commander-in-Chief General Olof Thörnell and the chief of the Defence Staff Major-General Axel Rappe. Then the texts went to the intelligence department's own sections. Routine messages went there directly. Afterwards the messages were burned, apart from those judged to have long-term value or that were of great strategic importance. Otherwise, distribution within the Defence Staff was very restricted. A distribution list was never established. The material was delivered to the recipient after an assessment in each particular case. (Comment: Destruction of the decrypted original texts has probably created problems for modern historical research.)

Distribution outside the Defence Staff, to the Foreign Office and the Security Service, took place in the beginning through Adlercreutz's personal service. The distribution to UD (*Utrikesdepartementet = Foreign Office*) was very restricted as Adlercreutz doubted the Foreign Office's security consciousness. However, that the co-operation between the Foreign Office and the crypto department was as good as it became, was due to the fact that foreign minister Richard Sandler (1932–1939, later governor of Gävle) was a very keen amateur cryptologist although, as a cryptanalyst he had no real success. His great services consisted of arranging for UD to inform the crypto department when important events were under way and when encrypted messages might be sent to Germany. This could assist the cryptanalysts, by allowing them to check that the decrypts were correct. [***11***]

Different considerations contrasted with each other here. The necessity to deliver information to suitable Swedish authorities conflicted with the requirements for secrecy in order to minimize the risk of betrayal of this unique intelligence source.

As the decryption work was done by a department of the Defence Staff, Adlercreutz wanted the intelligence department to be the first to receive all information and even to have a right to direct the work in the crypto department. However the crypto department successfully resisted these attempts to boost the intelligence department's power.

In the beginning, few objections were raised against Adlercreutz's control over distribution. But when the German preparations for Operation Barbarossa eventually came to their final stage, the Foreign Office started to feel that they did not get all the intelligence that they needed. The chief of the crypto department then decided to change the distribution routines after consultation with the Foreign Office, who also suspected that they did not get all the information in time.

During the winter of 1941 the incoming information became more and more alarming for Sweden, much more so than during the autumn and winter of 1940. However, it was now possible to get a continuously clear view from the decrypts of the groups, composition and combat readiness of the German forces in Norway and therefore also of changes in the situation. A lengthy force enumeration, intercepted on 21 April 1941, indicated a general movement of troops towards the north. Nevertheless, as on previous occasions no concentrations or deployment could be shown to be directed against Sweden. Nor was there any indication in the numerous reports from military border patrols, customs officers, police commissioners and officials interrogating Norwegian refugees that a German offensive against Sweden was imminent. The decrypted diplomatic traffic gave no special reason for alarm. The repeated German threatening warnings to Swedish contacts were reflected neither in the incoming intelligence nor in their own signals. During a period when German–Soviet tensions increased rapidly, it was nevertheless impossible to ignore the warnings. The government, principally the prime minister and the foreign minister, as well as the military command, were not prepared to cause trouble and accordingly create German irritation. It was not therefore apparent that they reckoned with a German–Soviet war.

Many analysts consider that war preparations serve only as instruments of pressure during negotiations. However, that ignore the dynamics of future military developments which are created by a deployment as large as that which occurred here. Economic factors and military logistics make it almost impossible to keep large, inactive troop concentrations in place as a trump card during long negotiations, just as it is damaging for the units' fighting spirit. It is too expensive not to use the troops, therefore they must either be used in combat or be demobilized and returned to civilian life. Only victory justifies the price — even if it is high. For example, consider the collapse of the economic, political and ecological systems now affecting the states of the former Soviet Union as a consequence, during a long period, of a highly forced "war economy" that did not result in any gains.

On 4 June a message was received indicating that strong German forces would in the near future be transferred east of Rovaniemi in northern Finland. Units would come from Germany as well as from Norway. The deployment was to be ready for 15 June. However, no demands on Sweden for the transit of troops could be gathered from the messages. Two divisions would be transferred by sea to Stettin, then in turn to Oslo and on to Rovaniemi.

The information in the decrypted messages clearly indicated a German attack on the Soviet Union.

On 11 June three further messages came which showed that this assumption was probably correct, as well as other intelligence revealing that Finland could not avoid becoming involved in the war. On 16 June came a teleprinter message that AOK (*Armeeoberkommando*) Norwegen had taken military command of Finnish Lappland and that the troop transports were going as planned. The same day, 16 June, the teleprinter connection Berlin

– Helsingfors via Stockholm was established following an earlier request. In spite of different speculations about a negotiated agreement, the incoming messages in the week before 27 June increasingly pointed towards an imminent outbreak of war.

Despite the intelligence, the Defence Staff did not cancel leave for the mid-summer weekend. An assessment that there would be a negotiated settlement, which would not require a high military preparedness, clearly had some validity. However, it is also possible that when the assessment was made that the war would not affect Sweden, it was more important to conceal the possession of this extraordinary source, which consisted of access to the Geheimschreiber traffic. Perhaps, therefore, they took it easy and allowed themselves to be "taken by surprise".

5 What happened later? When and how was the unique source exposed? [6)]

The decrypted German messages had, as mentioned earlier, been the most valuable sources during the weeks before the German attack on the Soviet Union on 22 June 1941. This would remain the position for a few years. With the attack, the teleprinter traffic to the German commands in Oslo and Rovaniemi, as well as the diplomatic traffic Stockholm – Berlin increased. The information received by the Swedish authorities became more detailed than before.

During the first year, from summer 1941 to summer 1942, when the German campaign against the Soviet Union took place, the decryption of the German teleprinter traffic provided extraordinary intelligence. German military plans and German politics towards Sweden could be clarified with the utmost certainty. However, no reliable knowledge was obtained about Adolf Hitler's political and strategic intentions. Intelligence throwing any light on the innermost reasoning of the people close to Hitler rarely or never existed.

The supreme army commands in Oslo and Rovaniemi did not command any of the decisive operations of German warfare. Hitler therefore seldom interfered in what went on in these theatres of operations. Nevertheless, even if the embassy in Stockholm and the commands in Norway and northern Finland were on the periphery of German interests, the intercepted internal German briefings and compilations had a great intelligence value for Sweden.

Adlercreutz's restrictions on distributing the decrypts to external recipients, except for the senior officers of the Defence Staff and the Intelligence Department, were mainly aimed at not exposing this exclusive source. Special instructions about other aspects of handling the material were issued in September 1941 by Samuel Åkerhielm in his capacity as deputy chief of the Defence Staff. The purpose was, of course, not to reveal the source. The decrypted messages had to be communicated and handled in secure ways. It was not permitted to refer to this material in conversations, and even less so on the telephone or in writing. When the messages were no longer needed, they had to be burned in controlled conditions.

The German confidence in the Geheimschreiber's security was, as explained earlier, an illusion. However, even the Swedish belief that the Geheimschreiber's secret still was a secret, except in Sweden, soon also became an illusion.

Some time in August 1941 the Soviet Union obtained access to the decrypted material. The courier Allan Emanuel Nyblad had the task of transporting the decrypted messages from Karlaplan 4 to the Staff building "Grå Huset" on Östermalmsgatan 87. He was a rather quiet man. He had been recruited as an agent, on ideological grounds, by the Soviet Union. The espionage was carried out in the following manner. On his way to the Grå Huset, Nyblad went to a rented flat situated along his usual route and photographed the messages he carried. The photos were given to the Soviet representatives, who had promised Nyblad a prominent position in a future communistic Sweden. He is not likely either to have received or demanded any money worth mentioning.

It is a reasonable assumption that the Soviet intelligence services (the NKVD and GRU) followed the Swedish success with interest. Moscow must have welcomed the likelihood that Sweden, with the aid of the intelligence, would take as strong as possible a position against Germany. But at the same time, the decrypted texts could also create doubts about Swedish power and will to withstand the German pressure.

Here we may reflect that the Soviet Union was taken by total surprise in 1941. During spring 1941 information about the coming German attack was leaked from Sweden to Great Britain (via the naval attaché) who passed on the information to the Soviet Union. However, General Golikov, then the chief of the GRU (the military intelligence service), actively contributed to the surprise by playing down the warnings from western sources, especially British, for reasons of political expediency. (The British intelligence service, SIS or MI 6, had until the Second World War primarily been working against the Soviet Union. Distrust can therefore be considered to be the explanation.)

Nyblad's spying was exposed in January 1942. It could not be exactly established which messages came into Russian hands through Nyblad, as he could not remember clearly on which days he had copied the material (T. Thorén) [***12***]. It is not known whether the Soviet Union derived any benefit from the information.

Less than six months after Nyblad's spying stopped, the Defence Staff discovered a new leak about the successful Swedish codebreaking activities. This time the leak soon had devastating consequences.

On 22 June 1942, Colonel Carl Björnstjerna, chief of the newly created foreign affairs section, which was directly subordinated to the chief of the Defence Staff, wrote to Major-General von Stedingk, the military attaché in Helsingfors, "A serious mishap has taken place. The Germans have been warned by the Finns that we have succeeded in breaking their G-schreiber. For this reason they are changing keys, message channels and everything . . . ”.

No contemporary information exists about how “the serious mishap” happened. However, one need not be surprised. Swedish service personnel had been so open towards their Finnish colleagues that a leak was made possible. The openness shown during the winter of 1941, when it was conceivable that both Sweden and Finland could have common defence interests, continued after the summer of 1941. The Finnish military attaché, Colonel Stewen, was even treated like an insider in the higher Swedish staffs, and the Germans suspected him of communicating sensitive information to Sweden, who they presumed then passed it on to the USA and Great Britain. However, it was equally possible that both of these great powers were capable of acquiring the alleged “leaked information” themselves. When the Germans criticized their Finnish colleagues for gossiping, the latter defended themselves by revealing that the Swedes had intercepted the teleprinter connections and were breaking the messages. It is even possible that Colonel Stewen had seen decrypted messages; at least he knew about them. The Finnish intelligence about the Swedish codebreaking activity probably reached the Germans some days before 17 June 1942, when the big alert hit the German communications. In the beginning, the counter-measures were quite incoherent, but they were soon concentrated in two directions. One consisted of introducing new cipher machines or attachments to them.

On 21 July 1942 a new machine appeared in the traffic, T52c, “Cäsar”. In the beginning it appeared on only a few lines, while on the others the old machine remained in use. As time went, more and more C-machines were put into operation.

At first inspection the C-machine appeared to be completely normal. When the crypto department received parallel texts it could attack these as before, but the texts no longer fitted the previously known patterns. The sequences were no longer periodic or they had at least no short periods. Was this a new encryption method?

At last the crypto department hit upon the solution. Two texts had been solved and it appeared that two “pin series” were identical in the long sequences. They had already earlier worked on the hypothesis that the seemingly infinite sequences were generated by addition, modulo-two, of the results of two wheels and hence obtained very long periods. The identical sequences allowed this hypothesis to be tested. This had failed earlier. Was it true that the old A/B-machines’ QEK-settings also were used in the C-machine? It was known that the C-machine could be used like the A/B-machine. For the A/B-machine they knew which five code-wheels were QEK-wheels, and also the settings. They then combined the wheels two by two in the ten combinations, but that did not work. However, when this procedure was repeated by leaving out four of the wheels it turned out to be correct. In this way it was possible to reveal the functioning of the C-machine.

The Germans had in a hurry made the mistake of keeping an element from an older, simpler system in a new cipher system. In the C-machine they used the same wheel-lengths and pin-patterns as in the A/B-machine and the keying principle was the same. They should have made the new machine completely independent of the old one, but it is likely that production difficulties created obstacles. The A/B-machine could be connected to the C-machine. Therefore the quick change of machines did not have any appreciable effect.

The other counter-measure taken by the Germans consisted in avoiding transmitting particularly important messages over Swedish telecommunication lines. It was therefore no longer possible to maintain the excellent intelligence about the German armed forces in Norway and Finland. A further deterioration occurred in October 1942 when all teleprinter traffic to and from Oslo and Rovaniemi was sent over Danish–Norwegian, or Finnish–Baltic cables. In certain cases cables were laid specially for this purpose. However, the teleprinter traffic to and from the German Embassy in Stockholm could still be intercepted and broken.

In October 1942 the Germans ordered the introduction of so-called "Wahlwörter" (*randomly chosen words*). The idea was actually sound. It is a good cryptological practice to avoid stereotype beginnings, which are usually where the codebreaker starts to look for an entry. The "QET-texts" were of course necessarily monotonous. Now the text would begin with a "Wahlwort" and in this way move the stereotype, fixed text further on to an undefined place in the message. However, the good intention failed. Many people follow instructions to the letter. Most people used the word given as an example in the instruction, which probably was SONNENSCHEIN (*sunshine*), as it occurred very often at first. Some managed to produce the word MONDSCHEIN (*moonlight*). The record was the word DONAUDAMPFSCHIFFSFARTSGESELLSCHAFTSKAPITÄN (*Danubesteamshipcompanycaptain*). When the procedure with Wahlwörter was used correctly it became more difficult, but not impossible, to decrypt the incoming messages.

In May 1943 such radical changes in the keying procedures were introduced that codebreaking became almost impossible. The breaking of the Geheimschreiber's teleprinter messages therefore diminished considerably. A smaller group was left on the task to try, if possible, to do something with the current material. Otherwise they were engaged in decrypting older material that had not been dealt with earlier, and in following the traffic.

During 1944 the Geheimschreiber appeared in versions D and E and a mysterious machine called Y. The crypto department never succeeded in breaking these machines. Models D and E were further developments of the machines that carried the designations A, B and C. The Y-machine [***13***] could perhaps have been a further development of the Z-(Zusatz)gerät (*Z-(additional)device = Z-attachment*). But, as just mentioned, the traffic transmitted with these machines could never be decrypted, only followed.

Other leaks also occurred, but were never of any significance. It was the Finnish military attaché in Stockholm, Colonel Stewen, who exposed the secret.

6 Experiences and lessons [7)]

When the new Defence Staff started functioning on 1 July 1937 the conditions were not the best for the intelligence and crypto departments. They did not have a solid foundation to build on. The procedures applied in the intelligence department were rather simple: namely, they were suited to producing compilations of open material and to making glossaries. Incoming messages from signals intelligence and e.g. attachés were passed on "in extenso" to the concerned parties without making any overall assessments and conclusions. This was left to the individual reader. The information was therefore not turned into intelligence.

The crypto department worked under very frugal conditions during its first development phase. However, through considerable efforts, where limited means were used in the best possible way, impressive results were achieved.

Swedish military intelligence collection deserves very good marks for the period just before Operation Barbarossa and during its opening phase. The Swedish military attaché in Helsingfors had given a warning in good time about a German attack on the Soviet Union in which Finnish participation was highly probable. The continuous decryption of the Geheimschreiber's messages also indicated clearly and unambiguously an imminent outbreak of war.

No inquiry will probably ever be conducted to see if the organization was a pure military endeavour, or a common one for the foreign department and the military commands to study and analyse the incoming information jointly. At least no documents exist showing this to be the position. Evidently it was considered adequate to circulate attaché reports and decrypted messages without any attached intelligence evaluation. However, the most essential messages were probably discussed by representatives from the Foreign Office and the military commands at their weekly meetings which they started in the beginning of April 1941.

Yet the incoming information gave the impression that Sweden would not be pulled into the war on the German side. The Swedish government and the commander-in-chief were therefore not surprised on 22 June 1941 as they had been on 9 April 1940. Nor was it necessary for Sweden to take any precipitate measures. They could afford to lie low and show surprise in order not to expose the exclusive source of the decrypted German teleprinter messages. No information exists about this, but the view is not unreasonable.

As mentioned, no organization existed for compilation, study, analysis and synthesis in preparing intelligence evaluations. Nor was the establishment of such an organization considered. The assessment of the circulating messages' intelligence value and the conclusions were left to the individual readers, whether with good or bad results.

Any long-term analysis and assessment of the war's course and end was never carried out, but considering the turbulent developments ahead that was perhaps best.

When one looks back to see whether it is possible to learn from that time's events one must also take into consideration the classical dilemma of a professional intelligence service. Incoming intelligence rarely or never gives information about planning prerequisites, considerations and objectives of the supreme command's inner circles, and even less about chosen alternatives. For this to be possible one must have access to a traitor or a planted spy in the enemy's supreme command (e.g. the CIA's Oleg Penkovski in the Soviet Union or Mossad's Eli Cohen in Syria). Normally one is simply reduced to using information which can give intelligence about possible actions and when they are likely to be realized. When different readers, each studying from varying positions, preconceived opinions and needs to assert his preserve, draws intelligent conclusions from raw information which will be the basis for decisions, the result can be disastrous. The lonely decision-maker may very well take non-optimal, irrational decisions. However, if the decision is made in full session, the delay caused by the decision-making process can produce catastrophic consequences before everybody agrees after a long discussion. These are two of the conditions for a strategic attack to succeed, like the German attack on Denmark and Norway. A third risk also exists. A joint, balanced intelligence estimate, which is carried out by a whole organization, can in the end be so diluted that it has no value for the decision-maker. The balance between the different extremes demands a lot from those carrying out intelligence work, irrespective of grade or service rank. It should also be pointed out that it is nearly impossible to assess all undercurrents influencing a historic event so as to make a forecast. Unknown as well as known events, which an intelligence service will not be able to interpret and describe, can create unknown forms of interference in a chain of events so that they can hardly be predicted.

7 Conclusion

On New Year's Eve 1941, Sweden was seemingly in the same geostrategic situation as the year before. However, a change in the wind was under way. In front of Moscow's gates, the Germans' storm wave had collapsed when Marshal Zhukov's Siberian troops launched a violent counter-attack on 6 December 1941. The Red Army had good help from "General Winter", who would assist in several winters to come. But in reality the strategic initiative was on its way to slipping out of German hands. However, it would take a whole year before this became evident for the world, when a complete German army was annihilated near Volga.

On 7 December 1941, the day after the counter-attack outside Moscow, Japan attacked the American fleet in Pearl Harbor. With that an industrial giant arose in terrible anger — an anger that in the end would turn the fortunes of war.

In the Second World War, the breaking of the cipher from the German Enigma machines and the Japanese Purple machines [*14*] was of crucial importance. The considerably greater intellectual effort needed to break the Geheimschreiber messages, which was accomplished in Sweden, did not in any way have the same decisive significance for the war. Therefore this accomplishment has not been so well known. However, from a Swedish perspective, it was of considerable importance as it actively contributed to keep Sweden out of the war.

8 Bibliography

Unpublished documents

Documents of various types concerning the breaking of the Geheimschreiber in FRA's secret archives.

Published works

Carlgren, Wilhelm M, *Svensk underrättelsetjänst 1939–1945*, Stockholm 1985.
Kahn, David, *The Codebreakers*, New York 1967.

Annotation

The publication of information from FRA's archives has taken place with due permission.

9 Notes

1. Where no other source is given, the account is based on information from Försvarets Radioanstalt's (FRA) documents.
2. Carlgren, p. 32.
3. Resumé of experts in FRA's documents, compare with Kahn, p. 478.
4. After Carl Gösta Borelius' unpublished documents in FRA's archive.
5. Carlgren, p. 68.
6. Carlgren, p. 80, compare with Borelius' documents.
7. Carlgren, p. 179.

10 Translator's Notes

1. Both of these addresses are in Stockholm. The majority of the FRA installations were in or around Stockholm, many of them adopting names with the ending "bo" which means room or house. On Lidingö, a small island on the eastern side of Stockholm, there were altogether five different FRA installations. Krybo and Rabo where FRA intercepted radio traffic and did cryptanalytical work, Petsamo which intercepted radio teleprinter traffic, Utbo for training intercept operators, and Matbo ("the food house") where there was a restaurant and housing quarters. In Stockholm city were Karlbo, Karlaplan 4, and Lebo on Strandvägen 57 which housed the FRA administration. Elsewhere in the country there were a few intercept stations and other installations. Sydbo intercepted Baltic radio traffic, while Norbo covered the Arctic radio traffic and traffic on the Finnish–German–Russian fronts. Ostbo covered the eastern parts of the Baltic Sea, the Baltic States and Poland.
2. The author refers to the use of relays for the inversion and the transposition circuits, which is how Carl-Gösta Borelius describes the circuits in his manuscript. However, only the T52c and T52e machines used relays for these circuits. The other machines, T52a/b and T52d, used the cam contacts on each coding wheel. The term "transposition circuit" reflects the cryptographic usage; mathematically speaking the circuit performs a permutation.
3. The number of combinations given by the author, $2\,612\,736\,000 = 10!\cdot 720$, is presumably the number of ways the 10 coding wheels can be selected and the number of permutation sets that can be obtained from the transposition circuit. This circuit has five double changeover contacts or transposition units which will give a total number of 2^5 permutations, which we call a permutation set, for a given set of connections. Furthermore, there are $9\cdot 7\cdot 5\cdot 3\cdot 1 = 945$ ways that the five contact sets, each equipped with two plug connections, can be inserted into the transposition circuit. Computer simulations show that each of these 945 connection variants results in unique permutation sets. However, the majority of the permutation sets, a total of 561, are degenerate in the sense that each set contains only from 1 to 16 unique permutations. The case of the set with only one single permutation is special because it is the identity permutation. Of the remaining 384 sets, 24 sets have 27 unique permutations, 240 sets 30 permutations and 120 sets contain all the 32 permutations.

   Inspection of a *Wehrmacht* SFM T52d Key table from May 1945 shows that all of the permutation connections belong to the two groups with 30 and 32 unique permutations, which means that in reality only 360 permutation sets were used by the German cryptographers during this period. The given number of 720 permutation sets probably is the result of a too superficial analysis of the T52 transposition circuit.

   The pluggable permutation units are only available on the T52a/b and T52d machines. On the T52c and T52e machines the transposition circuit uses relays instead of directly using the code wheel contacts. The five relays are permanently wired in one of the most basic permutation configurations. Therefore the only selection available on these machines is which code wheel controls the different transposition relays.

4. Due to an anomaly of the rotor movement in the Enigma machine, the middle rotor will step twice every time the left rotor advances. Therefore the period is $26 \times 25 \times 26 = 16900$ instead of $26 \times 26 \times 26 = 17576$, but the machine has a total of 17576 starting positions.
5. The translator previously believed that FRA had confused the terminology. The early Siemens T52a/b machine (1937) was called Geheimzusatz while the Lorenz SZ40 and SZ42 machines were called Schlüsselzusatz (cipher attachment). However, recent archive research has shown that the German teleprinter operators used the term G-Zusatz for the SZ 40/42 machines.[4] The following communication was decoded by Bletchley Park on the link they named Stickleback (Berlin – H.Gr. Südukraine) on 13 September 1944:
   ”Do you have a *G-Zusatz* 40 available? *Fundament* 40? . . . So you have no 40 *G-Zusatz* any longer . . . good . . . many thanks . . . a *Fundament* . . . So you have a forty'er after all . . . good . . . good.”
   Fundament 40 and 42 were used by the operators to refer to the SZ40 and SZ42 machines. Later this usage would cease and instead would appear Fundament A and Fundament B. These two terms would refer to the SZ42A and SZ42B variants. The dialogue above shows nevertheless that all these cryptic terms for the different machines were not understood by all. The use of the Z designation for the SZ40 machine probably comes from the German use of the letter Z to describe this machine while setting up an encrypted communication circuit. To set up encrypted communication with the T52a/b they would transmit in clear the Q-code “QEK”, while for communication with the T52c and the SZ40 machines they would transmit “QEK C” and “QEK Z” respectively.
6. The author uses the term “transposition connection”, which is correct in the sense that it transposes the function of a given code wheel, but it is a term likely to be confused with the machine's permutation circuit or the transposition unit. The translator has therefore decided to call this part the selective or programmable connection.
7. Because the enciphered text can contain any of the 32 possible teleprinter combinations, including the six control signals, the teleprinters used for interception had to be modified. In these specially modified teleprinters the signals *carriage return*, *line feed*, *letters*, *figures*, *space* and *null* were replaced with the numbers 1,2,3,4,5 and 6. All the other combinations were represented by their corresponding letter as normal. The British codebreakers at Bletchley Park (BP) used a similar arrangement where

[4] “Log Procedure Relating to The Use of “Limitation” on Non-Morse Army Links”, addendum to Captain Walter J. Fried's report No. 118 of 21 Nov. 1944. NARA RG 457, NSA Historical Collection, Box 880, Nr. 2612.

they replaced the six control signals by the characters *4*, *3*, *8 or -*, *5 or +*, *9 or .* and */*. See Appendix A.

8. The construction of the decryption units, Apparna, was led by engineer Vigo Lindstein of L.M. Ericson's cash register department. He would later join Hagelin's Cryptoteknik as technical chief and eventually end up as deputy director of AB Transvertex, another Swedish cipher manufacturer.
9. The breaking of the SZ40 machine was based on intercepted cable traffic, while intercepted radio transmissions later allowed the Swedish cryptanalysts to break the more modern SZ42. The cable traffic covered the period from 26 November 1941 until March 1943 and the first Swedish break into the SZ40 took place on 9 April 1943.
10. The codebreakers at Bletchley Park made a break into the Siemens T52 machine during the summer of 1942. They had followed the use of this machine that BP called Sturgeon for some time. The T52 machine was mainly used on radio teleprinter links belonging to the German Air Force and the German Navy. Due to a question of priorities BP allocated their resources on the German Army links that used the Lorenz SZ40 and SZ42 machines. However, the Swedish codebreakers were the first to break the Siemens T52 machines.
11. It is more likely that the crypto department was looking for probable words or "cribs" than testing for correct decryption.
12. The author refers here to Commodore Torgil Thorén, the chief of the Defence Staff's crypto department. The review of the Nyblad case is part of Thorén's analysis, which is included as Appendix 6 in the 1946 report "Investigation into the Defence Staff's handling of decrypted messages from FRA".
13. In his book, Bengt Beckman explains that the Y or QEKY machine was the Siemens one-time-tape machine T-43 which towards the end of the war was used on radio communication circuits that also used the SZ40/42.
14. The Japanese machines, Purple, Coral and Jade, were used for high-level diplomatic communications and therefore never carried the same kind of tactically important traffic that was the case for the Enigma. Nevertheless, intelligence from these machines was important for the conduct of the war, and the reports from Japan's ambassador in Germany, Hiroshi Ōshima, contained much information of great strategic value.

11 Translator's Postscript

Lars Ulfving's account of Swedish codebreaking during the Second World War has largely been superseded by Bengt Beckman's book *Svenska kryptobedrifter (Swedish Crypto Achievements)*, [5,41] as Lars Ulfving himself acknowledges. Ulfving's account, which is largely based on Carl-Gösta Borelius' internal FRA history, cannot be compared with Beckman's complete historical treatment of Swedish cryptography. Bengt Beckman, who is the former chief of FRA's cryptanalytical department, interviewed many people who were directly or indirectly involved in FRA's work during the war and had full access to the archives. He did not participate in FRA's wartime work, since he joined the organisation in 1946. However, he personally knows most, if not all, of those who took part.

Although, Lars Ulfving had limited access to the FRA archives and sources he did a good job with the material at his disposal. A strong point in his presentation is the setting of the cryptological exploits in their true historical context, which shows their importance for Swedish defence and foreign policy. However, it lacks a more profound explanation of the cryptanalytical problems and the personal histories of those who were involved.

Arne Beurling has a central place in both Lars Ulfving's presentation and Bengt Beckman's book, which he clearly merits. Arne Beurling was in many ways Sweden's and FRA's Alan Turing. Like Turing he was a genius who always worked from first principles and received great pleasure in seeking simple solutions to problems. However, unlike Turing, he was not socially awkward. He liked an enjoyable evening in town, while his good looks and great personal charm made him very attractive to women. He was also a typical outdoors man who liked trekking, sailing and hunting. However, he had a darker side. He could be stubborn and difficult. Throughout his life he had many conflicts with other people and could then be physically violent. He is known to have settled one argument with the famous Swedish cryptographer Yves Gyldén with his fists. Bengt Beckman dedicates three full chapters to Arne Beurling. The picture that emerges is of a person with a complex character, but who is full of life and nevertheless inspires both trust and friendliness.

Like Turing, Arne Beurling would make the initial breaks into a problem and lead the way, but afterwards he would take on other tasks, allowing others to continue the work. Before Beurling decided to attack the Geheimschreiber problem, he had worked together with Åke Lundqvist, botanist and chess Grandmaster,[5] on superenciphered Russian codes. The Swedish cryptanalysts made great inroads into the Russian code and cipher systems. Olle Sydow and Gösta Wollbeck were two of the major cryptanalysts working on the Russian problems, but there were many others. They made up a variegated group of professors in Slavic languages and literature, mathematics and astronomy including a few art historians. Not to forget all the young women of "good" families who, as at Bletchley Park, attended to the more humdrum tasks.

Another of Beurling's great achievements at FRA was his solution together with Robert Themptander, an actuarial mathematician, of a very difficult double transposition problem. These ciphers had made their first appearance in the autumn of 1940. They were sent by two spy transmitters, with the callsigns CDU and MCI, which were located on the continent and communicated with a station in England. In June 1941 the same traffic, which

[5] Åke Lundqvist received the title of Grandmaster in correspondence chess in 1962.

always started with the indicator CXG, appeared in the transmissions of the British embassy in Stockholm.

Double transposition can be a difficult cipher to break. In this case it was made even more difficult by applying a monoalphabetic substitution before the double transposition. By analysing the cipher texts they discovered that the digits 0,1,2,3 and 4 had a higher frequency than expected, something which indicated a substitution alphabet in the range 01–45. In October 1941, when Arne Beurling struggled with this problem, he finally succeeded in breaking six messages enciphered with the same key. He discovered that double transposition was indeed used with keys of different length for each transposition. However, his greatest difficulty was not the transpositions but rather to reconstruct the substitution alphabet. He apparently guessed that the alphabet would be in its ordered sequence, but the plain text that emerged did not fit the English language as expected.

After many trials Beurling finally succeeded in extracting one word that made sense: "Baltik". However, the rest was mainly incomprehensible. Arne Beurling then brought the text to his good friend Richard Ekblom, professor in Russian at the University of Uppsala. After slightly rearranging the text, Ekblom said: "This looks like Czech". And it turned out to be telegrams from Vladimir Vanek who was the Czech Exile Government's representative in Stockholm. As the telegrams showed that Vanek was involved in espionage against both Germany and Sweden, he was arrested in his home on 27 March 1942. A search resulted in the identification of the book used as the base for the transposition keys. It was Jan Masaryk's *Světová Revoluce (World Revolution)*. The meaning of the indicator CXG escaped the FRA cryptanalysts during the war but now, more than 50 years later, Robert Themptander says it must have stood for Czech Exile Government. It is said that Arne Beurling himself considered this solution of a double transposition cipher with a monoalphabetic substitution and in an unknown language to be a greater feat than his solution of the Geheimschreiber cipher.

Arne Beurling was not the only master cryptanalyst at FRA during the war. He was perhaps the only genius, but there were also other excellent cryptanalysts, who performed great achievements. A group of three people, Carl-Gösta Borelius, Tufve Ljunggren and Bo Kjellberg, under the leadership of Lars Carlbom succeeded in breaking the Lorenz SZ40 and SZ42 machines. The SZ40 traffic had been observed on the German cable connections in November 1941 and in January 1943 FRA also observed the same traffic on radio circuits. Their first break occurred on 9 April 1943, based on the cable traffic, while they also succeeded in breaking the SZ42, which was then used on radio, in September 1943.

The solution of the SZ40 came later than the first British solution of the same machine in January 1942. However, Arne Beurling's solution of the T52a/b machine in June 1940, based on a set of messages in depth intercepted on 25 and 27 May, constituted the first break of a modern, on-line teleprinter cipher. As with the Siemens T52 solutions the Lorenz machines were also broken by hand methods. The only tool was a SZ40 “replica” which was constructed using bicycle chains of different length to simulate the action of the twelve coding wheels.

FRA’s resources were very limited. With a peak staff of 384 people in 1942 it was a tiny organization compared with Bletchley Park (BP) and Arlington Hall. It is therefore of interest to compare FRA’s Geheimschreiber group with the corresponding Fish[6] sections at Bletchley Park. FRA’s Geheimschreiber group, which consisted of the sections 31f, 31g, 31m, and 31n, saw its peak performance in November 1942 while the Fish sections at BP had their peak in the autumn of 1944. In November 1942, 31f, which was responsible for tidying up the texts and typing the final results, provided 10638 messages. It was staffed by 56 people handling the tidying up process and 18 typists. At the same time, the line intercept section, 31n, had nine technician and eight young women who glued the printed teleprinter tapes on sheets of paper. The Post Office supplied up to three maintenance people on demand who would tend to the 72 line receivers and the 36 teleprinters. The cryptanalytic section, 31g, had 14 cryptanalysts and 60 women who manned the decoding machines, the “Apps”. There were a total of 32 “Apps” of which 22 had the special T52c adapter and 26 specially connected teleprinters. Finally, in the translation and compilation section, 31m, there were seven compilers and 13 translators including a few persons taking care of the odd jobs. In total the Geheimschreiber group had about 185 people.

In September 1944,[7] at the inauguration of BP’s new Block H which was built to house Max Newman’s section and his machines, there were a total of 345 people working on the Tunny problem in the two sections, the Newmanry and the Testery. The Newmanry comprised 20 male civilians, at least 10 of them with honours degrees in mathematics, one US Navy officer, 2 US Army officers, and 186 Wrens from the Women’s Royal Naval Service, a total of 209 people. In Major Tester’s section, the Testery, there were at this time 21 officers, including two US Army officers, 77 other ranks, 25 ATS (Auxiliary Territorial Service) women, and 26 male civilians, a total of 136 people. They included six mechanics and 37 machine operators, while 30 people were working on registration of traffic and 20 others were engaged on breaking “dechis”[8] and anagramming depths. In addition there were 34 “setters” whose job it was to carry a break back to the beginning of a message and compute the settings for the machine operators. The remaining nine people were taking care of a variety of jobs.

[6] Fish was the BP codename for the non-morse, teleprinter, transmissions and the ciphers they employed. The Lorenz SZ40/42 was labelled Tunny while the Siemens T52 machines were known as Sturgeon.

[7] Captain Walter J. Fried’s report No. 96 of 29 Sep. 1944. National Archives and Records Administration (NARA), RG 457, NSA Hist. Col. Box 880, Nr. 2612.

[8] Dechi is a kind of “pseudo plain text” as given by the expression $D = Z + \mathcal{X} = P + \Psi$. The dechi is part of the method used to strip off the influence of the Chi wheels of the Lorenz SZ40/42 machine.

At the end of September 1944, Colossus 5 had been installed in the new Block H and was fully operational, while Colossus 6 had also been installed but was not yet connected. A total of 12 Colossi were planned and ten machines were in operation at the end of the war. The sections also had up to 16 Tunny machines to decode messages on already broken keys. Tunny was also used for the "dechi" process and the Newmanry was equipped with three of these machines. During September over 2.5 million plain text characters were produced by the two Fish sections from a total of 40 million intercepted characters.[9] This is only 6.25 %, however, the majority of the intercepted signals, 82 %, consisted of transmissions of fewer than 2000 characters, which were too short to be broken.

What is immediately apparent in this comparison is the difference in approach. The FRA group appears more like a production unit where the daily cryptanalytical problem was well in hand and was solved with a minimum of specialist staff and without any machines. In BP's Tunny sections the cryptanalytical problem clearly demanded the biggest resources both in cryptanalysts and machine operators who attended to the Colossi and other specialised machines. One reason for this is due to the differences in the two cipher machines. In the Siemens T52 the code wheel patterns remained fixed while for the Lorenz SZ40/42 they had to be broken every day for some of the links. This required a major effort from the cryptanalysts. FRA also had the advantage of working with intercepted transmission that were as good as the intended German recipient. This was never the case for BP who very often had to work with marginal material due to the difficulty of receiving and transcribing the very faint Fish signals. However, the T52's cryptographic algorithm, which used permutation together with modulo two addition, was more difficult than the principle used for the SZ40/42 machines.

Seen in this light, FRA's results are astounding, which can only partly be explained by the dedication of the people and their leadership. One important factor must have been the quality, professionalism and experience of those involved. The majority were highly educated and those who lacked formal education were extremely talented. One person who comes to mind is Gösta Wollbeck, then Gösta Eriksson, who, when he joined FRA, lacked formal academic education but had a good knowledge of Russian acquired through self-study. He would, over the years, assimilate one language after the other and undertake whatever translations were needed.

However, there is yet another factor. Even though there appear to have been tensions within the organisation, something that probably can be explained by the close proximity of so many strong and somewhat eccentric personalities, most of them clearly loved the work they were doing. As Arne Beurling's student and very good friend over many years, Bo Kjellberg, said

[9] Captain Walter J. Fried's report No. 101 of 14 Oct. 1944. NARA, RG 457, NSA Hist. Col. Box 880, Nr. 2612.

recently at a meeting of FRA veterans: "That was the most wonderful time of my life. To have all those unsolved problems in front of me."

12 Acknowledgements

The translator should like thank Dipl. Ing. Wolfgang Mache for his request for help in translating parts of Lars Ulfving's paper into German or English. This request, from an old friend, started the process which has resulted in the paper finally being published here. I should also like to thank Alan Stripp and Ralph Erskine for proof reading earlier versions of the translation and their many suggestions. I am further indebted to the author, Lars Ulfving, and Lieutenant-Colonel Bo Kjellander of Probus Förlag, Stockholm, for their friendliness and courtesy in granting permission to publish the translation together with the original illustrations. Geoff Sullivan has helped me with the T52 simulations and I am very grateful for his assistance. Finally I wish to thank Bengt Beckman for permission to quote from his book and FRA monograph and for helping me to obtain photos from the FRA archives.

13 Translator's Bibliography

References

1. Ahlfors, Lars: The Story of a Friendship: Recollections of Arne Beurling. Mathematical Intelligencer Vol. **15** No. 3 (1993) 25–27
2. Banks, David L.: A Conversation with I. J. Good. Statistical Science Vol. **11** No. 1 (1996) 1–19
3. Bauer, Friedrich L.: **Entzifferte Geheimnisse, Methoden und Maximen der Kryptologie**. In German. Berlin: Springer Verlag (1995)
4. Bauer, Friedrich L.: **Decrypted Secrets, Methods und Maxims of Cryptology**. Berlin: Springer Verlag (1996)
5. Beckman, Bengt: **Svenska kryptobedrifter (Swedish Crypto Achievements)**. In Swedish. Stockholm: Albert Bonniers Förlag (1996)
6. Beckman, Bengt: **A Swedish Success: Breaking the German Geheimschreiber during WW2**. Monograph produced by Försvarets Radioanstalt, Stockholm, Sweden (1997)
7. Beckman, Bengt: **Svenska Kryptotriumfer Under Andra Världskriget (Swedish Crypto Triumphs During the Second World War)**. In Swedish. Monograph produced by Försvarets Radioanstalt, Stockholm, Sweden (1999)
8. Beckman, Bengt: **Så Knäcktes Z-Maskinen (This is how the Z-Machine was broken)**. In Swedish. Monograph produced by Försvarets Radioanstalt, Stockholm, Sweden (1999)
9. Carter, Frank: Codebreaking with the Colossus Computer. The Bletchley Park Trust Reports No. **1** November (1996)
10. Carter, Frank: Codebreaking with the Colossus Computer: An Account of the Methods Used for Finding the K-wheel Settings. The Bletchley Park Trust Reports No. **3** May (1997)

11. Carter, Frank: Codebreaking with the Colossus Computer: Finding the K-wheel Patterns. An Account of Some of the Techniques Used. The Bletchley Park Trust Reports No. **4** June (1997)
12. Chandler, W.W.: The Installation and Maintenance of Colossus. Annals of the History of Computing **5(3)** July (1983) 260–262
13. Coombs, Allen W.M.: The Making of Colossus. Annals of the History of Computing **5(3)** July (1983) 253–259
14. Davies, Donald W.: The Siemens and Halske T52e Cipher Machine. Cryptologia **6(4)** October (1982) 289–308
15. Davies, Donald W.: The Early Models of the Siemens and Halske T52 Cipher Machine. Cryptologia **7(3)** July (1983) 235–253
16. Davies, Donald W.: New Information on the History of the Siemens and Halske T52 Cipher Machine. Cryptologia **18(2)** April (1994) 141–146
17. Davies, Donald W.: The Lorenz Cipher Machine SZ42. Cryptologia **19(1)** January (1995) 39–61
18. Deavours, Cipher A. and Kruh, Louis: Mechanics of the German Telecipher Machine. Cryptologia **10(4)** October (1986) 243–247
19. Flowers, Thomas H.: The Design of Colossus. Annals of the History of Computing **5(3)** July (1983) 239–252
20. Glünder, Georg: Als Funker und "Geheimschreiber" im Krieg 1941–1945. In German. Pionier Nr. **11/12** (1989) and Nr. **2** (1990) Translated into English by Paul K. Whitaker. Unpublished.
21. Gollman, Dieter and Chambers, W.G.: Clock-Controlled Shift Registers: A Review. IEEE Journal on Selected Areas in Communications **7(4)** May (1989) 525–533
22. Good, I.J.: Enigma and Fish. In ed. F.H. Hinsley and Alan Stripp. **Codebreakers, The Inside Story of Bletchley Park**. Oxford: Oxford University Press (1993) 149–166
23. Halton, Ken: The Tunny Machine. In ed. Hinsley and Stripp. **Codebreakers**. (1993) 167–174
24. Hayward, Gil: Operation Tunny. In ed. Hinsley and Stripp. **Codebreakers**. (1993) 175–192
25. Heider, Franz-Peter: A Colossal Fish. Cryptologia **22(1)** January (1988) 69–95
26. Hilton, Peter: Reminiscences and Reflections of a Codebreaker. In these proceedings. **Coding Theory and Cryptology: From Enigma and Geheimschreiber to Quantum Theory**. New York: Springer Verlag, Lecture Notes in Computational Science and Engineering (1999)
27. Hinsley, F.H.: Geheimschreiber (Fish). In F.H. Hinsley et al. **British Intelligence in the Second World War**. London: HMSO Vol. **3** Part 1 Appendix 2 (1984) 477–482
28. Hinsley, F.H.: Cracking the Ciphers. Electronics & Power IEE July (1987) 453–455
29. Hinsley, F.H.: An Introduction to Fish. In ed. Hinsley and Stripp. 1993. **Codebreakers**. (1993) 141–148
30. Kahn, David: The Geheimschreiber. Cryptologia **3(4)** October (1979) 210–214
31. Kjellberg, Bo: Memories of Arne Beurling. Mathematical Intelligencer Vol. **15** No. 3 (1993) 28–31
32. Mache, Wolfgang: Geheimschreiber. Cryptologia **10(4)** October (1986) 230–242
33. Mache, Wolfgang: The Siemens Cipher Teletype in the History of Telecommunications. Cryptologia **13(2)** April (1989) 97–117

34. Mache, Wolfgang: Der Siemens-Geheimschreiber — ein Beitrag zur Geschichte der Telekommunikation 1992: 60 Jahre Schlüsselfernschreibmaschine. In German. Archiv für deutsche Postgeschichte Heft **2** (1992) 85–94
35. Mache, Wolfgang: **Lexikon der Text- und Daten-Kommunikation**. In German. München: R. Oldenbourg Verlag 3rd. revised edition (1993)
36. Randell, Brian: The Colossus. In ed. N. Metropolis et al. **A History of Computing in the Twentieth Century**. New York: Academic Press (1980) 47–92
37. Sale, Tony: **The Colossus Computer 1943–1996**. Kidderminster UK: M & M Baldwin (1998)
38. Selmer, Ernst S.: From the Memoirs of a Norwegian Cryptologist. In ed. Tor Helleseth. Advances in Cryptology — EUROCRYPT '93. New York: Springer Verlag, Lecture Notes in Computer Science **765** (1993) 142–150
39. Selmer, Ernst S.: The Norwegian Modification of the Siemens and Halske T52e Cipher Machines. Cryptologia **18(2)** April (1994) 147–149
40. Tutte, William T.: FISH and I. In these proceedings. **Coding Theory and Cryptology: From Enigma and Geheimschreiber to Quantum Theory**. New York: Springer Verlag, Lecture Notes in Computational Science and Engineering (1999)
41. Weierud, Frode: Sweden, Cryptographic Superpower. A Book Review. Cryptologia **22(1)** January (1998) 25–28
42. Weierud, Frode: Sturgeon, The FISH BP Never Really Caught. In these proceedings. **Coding Theory and Cryptology: From Enigma and Geheimschreiber to Quantum Theory**. New York: Springer Verlag, Lecture Notes in Computational Science and Engineering (1999)
43. Wermer, John: Recollections of Arne Beurling. Mathematical Intelligencer Vol. **15** No. 3 (1993)
44. Zorpette, Glenn: Breaking the enemy's code. IEEE Spectrum **24(9)** September (1987) 47–51

Documents

45. Oberkommando der Kriegsmarine: Der Geheimzusatz der Siemens-Fernschreibmaschine T.typ.52. M.Dv. Nr. 35 Geheim, Berlin Oktober 1937
46. Deutsche Wehrmacht: Schlüsselfernschreibvorschrift (SFV). H.Dv. g 422, L.Dv. g 704/3b, M.Dv. Nr. 924a Geheim, 1 Dezember 1942
47. Oberkommando der Kriegsmarine: Die Siemens-Schlüsselfernschreibmaschine SFM T52d (T typ 52 d). M.Dv. Nr. 35IV, D.(Luft) T.g.Kdos. 9105d Geheime Kommandosache, Berlin März 1944
48. Small, Albert W.: Special Fish Report. National Archives and Records Administration RG 457 NSA Historical Collection Box 1417 Nr. 4628 (1944)
49. Campaigne, Howard: Report on British Attack on "FISH". National Archives and Records Administration RG 457 NSA Historical Collection Box 579 Nr. 1407 (1945)

14 Appendix A

Modified teleprinters with International Telegraph Alphabet No.2

Lower Case	Code elements	Upper Case	
(letters)	1 2 3 4 5	(figures)	
A	1 1 0 0 0	–	
B	1 0 0 1 1	?	
C	0 1 1 1 0	:	
D	1 0 0 1 0	”Who are you”	
E	1 0 0 0 0	3	
F	1 0 1 1 0	*\)	
G	0 1 0 1 1	*\)	
H	0 0 1 0 1	*\)	
I	0 1 1 0 0	8	
J	1 1 0 1 0	Bell	
K	1 1 1 1 0	(	
L	0 1 0 0 1	)	
M	0 0 1 1 1	.	
N	0 0 1 1 0	,	
O	0 0 0 1 1	9	
P	0 1 1 0 1	0	
Q	1 1 1 0 1	1	
R	0 1 0 1 0	4	
S	1 0 1 0 0	’	
T	0 0 0 0 1	5	
U	1 1 1 0 0	7	
V	0 1 1 1 1	=	
W	1 1 0 0 1	2	
X	1 0 1 1 1	/	
Y	1 0 1 0 1	6	
Z	1 0 0 0 1	+	
1	0 0 0 1 0	Carriage Return	(BP code: 3)
2	0 1 0 0 0	Line Feed	(BP code: 4)
3	1 1 1 1 1	Letters	(BP code: 8 or –)
4	1 1 0 1 1	Figures	(BP code: 5 or +)
5	0 0 1 0 0	Space	(BP code: 9 or .)
6	0 0 0 0 0	No Action	(BP code: /)

Note: **)* Unassigned, reserved for domestic use.

15 Appendix B

Chronological List of Swedish Geheimschreiber Events[††]

1940			
	Apr	T52a/b, first observations, to/from Norway	
	Jun	T52a/b, first solutions	Dec 40: 7100 msgs
	Sep	T52a/b, first routine production	20–30 staff 1–2 “apps”
1941			
	May	T52a/b, to/from German Legation in Stockholm	
	Jun	T52a/b, to/from Finland	Dec 41: 41400 msgs
	Nov	SZ40 observed on cables	94 staff 10 “apps”
1942			
	Jun	Rumours in Berlin about Swedish breaks	
	Jul	T52c, first observations	Dec 42: 120000
	Sep	T52c, first solutions	msgs 185 staff
	Dec	New key system	32 “apps”
1943	Jan	SZ40 on radio, first observations	
	Feb	T52ca, first observations	
	Mar	T52ca, first solutions	
	Apr	SZ40 on cables, first solutions	
	May	New key system	
	Jun	SZ40 on radio, first solutions	
	Sep	SZ42 on radio, first solutions	Dec 43: 71000 msgs
	Dec	T52d, first observations	51 staff
1944			
	Feb	No more solutions of cable traffic	
	Sep	T52e, first observations	

[††] Reproduced with permission from FRA’s Monograph *A Swedish Success* [6]

Subject Index

The RSA Public Key Cryptosystem

William P. Wardlaw

Mathematics Department, U. S. Naval Academy, Annapolis, MD, 21146

1 Introduction

The RSA (Rivest, Shamir, Adleman) cipher algorithm has captured the imagination of many mathematicians by its elegance and basic simplicity ever since it was introduced in 1978. Numerous descriptions of the algorithm have been published. Readers with a knowledge of a little basic number theory will find the original paper [RSA] by the inventors of the algorithm, Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman, quite readable. Perhaps the most famous description is Martin Gardner's expository article [G], which is written for readers of *Scientific American.* Martin E. Hellman [H] wrote another good *Scientific American* article describing the RSA algorithm and the knapsack cipher algorithm. The goal of this paper is to lead the reader who has some mathematical maturity but no knowledge of number theory, say a first year calculus student, a clever high school student, or an interested engineer, through the basic results needed to understand the RSA algorithm. The prerequisites are only a knowledge of the elementary school arithmetic of the integers, high school algebra, some familiarity with the notions of sets and of functions, and, most importantly, a real desire to understand how the RSA algorithm works. We begin with a discussion of general crypto systems and the differences between classical systems and public key systems. Then the treatment will give an informal but fairly rigorous introduction to the division algorithm, divisibility properties, greatest common divisors, the Euclidean algorithm, modular arithmetic, repeated squaring algorithm for $b^a \pmod m$, time estimates for these algorithms, Euler's totient function, Euler's Theorem, and, as a corollary, Fermat's Little Theorem. Don't worry if you are unfamiliar with some of these terms now - they will all be explained when they arise. These ideas will then be used to explain the RSA algorithm in detail. We will mention but not go into detail on the notions of primality testing and methods of factoring. The student who wishes a deeper understanding of these things is strongly recommended to read the pertinent sections of Neal Koblitz' excellent book [Ko], *A Course in Number Theory and Cryptography.*

2 General Cryptosystems

A **cryptosystem** is a method of secret communication over public channels between members of some group of people, which we call the **crypto group**.

The term **public channels** refers to the possibility that people outside of the crypto group can intercept messages sent between members of the group. Broadcast radio, telephone lines, ordinary mail, and e-mail are all examples of public channels. A cryptosystem will be made up of one or more (usually many) units, called **crypto cells**, each of which provides for communication from one member of the group to another.

Suppose that Bob wants to send a message x to Alice. Bob uses an **encryptor** E to act on x and transform it to the encrypted message $y = xE$. Then he sends the encrypted message y to Alice. When Alice receives the message, she uses a **decryptor** D to convert y back to the plaintext message x: $yD = (xE)D = x$. Thus, the decryptor D undoes or **inverts** the action of the encryptor E. In practice, E (as well as D) might be a mechanical device, a computer program, or an algorithm) which converts x into y (or y into x, in the case of D). This encryptor-decryptor pair (E, D) is the simple cryptosystem, or crypto cell, that Bob uses to contact Alice.

We model this situation mathematically by taking E to be a one to one function with domain $X = dom(E)$ and range $Y = ran(E)$, and taking $D = E^{-1}$ with domain $Y = dom(D)$ and range $X = ran(D)$ to be the inverse function of E. Thus, X is the set of all plaintext messages that can be encrypted by E, and Y is the set of all encrypted messages. The functions E and D satisfy the relationship

$$xE = y\text{if and only if } yD = x. \tag{1}$$

A simple example is given by letting $X = Y = \{a, b, c, d, ..., x, y, z\}$ be the alphabet and taking E to be the permutation

$$E = \begin{pmatrix} \text{a b c d e f g h i j k l m n o p q r s t u v w x y z} \\ \text{k r y f m t a h o v c j q x e l s z g n u b i p w d} \end{pmatrix} \tag{2}$$

Now E acts on a letter in the top row by transforming it to the letter below it in the second row. Thus, aE = k, bE = r, ... , and zE = d. It is not difficult to construct $D = E^{-1}$ from E to obtain

$$D = \begin{pmatrix} \text{a b c d e f g h i j k l m n o p q r s t u v w x y z} \\ \text{g v k z o d s h w l a p e t i x m b q f u j y n c r} \end{pmatrix} \tag{3}$$

It may seem silly to let every message consist of a single letter, but we do not need to stretch our imagination far to see how E encrypts “to err is human” as “ne mzz og huqkx” and D decrypts “ne tezaobm foboxm” as “to forgive divine”. The encryptor E is an example of an important class of $26! = 1 \times 2 \times 3 \times ... \times 25 \times 26$ encryptors called **monoalphabetic ciphers** that are really just permutations of the alphabet. It is a simplifying convenience to think of the set X of all plaintext messages as consisting of 26 letters, rather than the infinite collection of words and phrases that can be written with these letters!

As we mentioned above, a cryptosystem consists of crypto cells (E, D) which allow one member of the group to send a private message to another. In a *classical* cryptosystem, the situation is as described in our example. If Bob knows an encryptor E to send a message to Alice, he can easily figure out the corresponding decryptor D, and so he can use D to understand messages that Alice sends to him using E. The situation is *symmetrical*; Alice and Bob share E and D, and use them to communicate. In order to set up private communication over a public channel, Bob and Alice would first have to get together privately and share the information (E, D), usually in the form of a "key". Unfortunately, the problem of key exchange is made worse by the fact that if Bob and Alice use the same key for too long, an easedropper may be able to break their key and decrypt their communications, so Bob and Alice must get together frequently to exchange keys. David Kahn describes the following consequences of not changing keys in Chapter 17 of [K] , especially page 567. After their Pearl Harbor attack, the tremendous success of the Japanese in spreading their forces throughout the Pacific delayed their intended change of codebooks from 1 April to 1 June 1942. This enabled the cryptanalysts in Hawaii to glean enough information from Japanese coded messages to predict the Japanese attack on Midway and to get U. S. carriers in the right place to surprise the Japanese and win a decisive victory.

Because of this need to avoid sending too many messages using the same crypto cell (E, D), a cryptosystem, even one involving only two people, will often use a large number of different crypto cells. Each crypto cell will be given an identifying key. Different messages will then be encrypted using different encryptors, with keys determined by the date, the time of day, or somehow hidden in the message itself.

Another reason that a crypto system may require a number of different crypto cells is to establish different *cliques* in the crypto group. A **clique** within a crypto group is a set of people within the group who exclusively share a given set of crypto cells. Any pair of a clique can communicate with each other without members of other cliques being able to decrypt their messages. In a military crypto group, a clique of all officers might have access to those crypto cells used to send "confidential" messages, but only commissioned officers would be allowed to read "secret" messages, and a still smaller clique of officers with a special clearance would have access to the crypto cells used for "top secret" traffic.

One useful organization of a crypto group is to make each pair a clique, so that any two members of the group can communicate secretly with each other. In general, a crypto group of N people would have $N(N-1)/2$ pairs, and it would require that many crypto cells (E, D) in order to allow any two people in the group to enjoy private communication. For example, in a group of 10 people, we have 10 ways to pick the first person x in an ordered pair (x, y), and for each such x there would be 9 choices for y, for a total of

10×9 *ordered* pairs of people. Since this method counts each *unordered pair* {x, y} twice, there are $10 \times 9/2 = 45$ unordered pairs.

For several thousand years, only classical (also called symmetrical) crypto systems were available. But in 1976, Diffie and Hellman [DH] introduced the idea of a **public key cryptosystem**. In such a system, each user secretly obtains a crypto cell (E, D) and then *publishes* the encryptor E. Clearly, the central requirement of such a system is that it be *prohibitively difficult* to figure out the decryptor $D = E^{-1}$ from a knowledge of E. For example, if a given computer takes a millisecond to encrypt a given message using the encryptor E, that same computer might take several thousand years to compute the decryptor D using a knowledge of E. (This is called **breaking** the encryptor.) Of course, several thousand computers working together might break the encryptor in only one year, and a single computer which computes several million times as fast as the original computer might break the encryptor in nine or so hours. It is clear that the security of the system is dependent on the present state of technology.

Diffie and Hellman suggested the use of a **trapdoor function** E for which the possession of certain secret information would make it easy to calculate its inverse D, but for which D would be very difficult to discover without this information. Then each member M of a crypto group would calculate and publish an encryptor E_M and privately calculate its inverse, the decryptor D_M, which he would keep secret. If Bob wants to send a message x to Alice, he needs no private meeting to exchange keys. He simply looks up Alice's published encryptor E_A, uses it to encrypt the message, and sends the encrypted message $y = xE_A$ to Alice. Since only Alice knows her decryptor D_A, only Alice can decrypt y to obtain $x = yD_A = xE_AD_A$. If she wishes to reply, Alice can use Bob's published encryptor E_B.

But how can Alice be sure that it was Bob who sent her the message? After all, everyone has access to her encryptor E_A and could use it to send her a message masquerading as Bob. But authentification is possible at the cost of two extra messages. Bob could append a ten digit random number b as part of his first message. Alice could generate a ten digit random number a and send Bob the number $a + b$, using the encryptor E_B that only Bob can decrypt. Bob then authenticates his original message as well as future messages by appending a, which he obtains by subtracting b from Alice's appended $a + b$.

For certain public key crypto systems, Bob and Alice can even sign their messages in a way that can be verified by an arbiter later. Let $X_M = dom(E_M) = ran(D_M)$ be the set of all plaintext messages and $Y_M = ran(E_M) = dom(D_M)$ be the set of all encrypted messages for the crypto cell (E_M, D_M) belonging to crypto group member M. Suppose that every plaintext message can be considered to be an encrypted message, and vice versa. That is, $X_M = Y_M$ for every member M. If $X_A = Y_A \subseteq X_B = Y_B$, then Bob can send a signed message x in X_A to Alice by first encrypting it using Alice's encryptor E_A to obtain $y = xE_A$ in $X_A = Y_A$. Since $Y_A \subseteq Y_B$, y is also in Y_B, and so Bob's decryptor can be applied to y to obtain $z = yD_B = xE_AD_B$ in X_B. Then Bob sends the signed encrypted message z to Alice. When Alice receives the message z, she knows it is supposed to have come from Bob and that $X_A \subseteq X_B$, so she first applies Bob's publicly known encryptor E_B and then her secret decryptor D_A to read the message $zE_BD_A = (xE_AD_B)E_BD_A = xE_AD_A = x$: E_B "undoes" D_B and then D_A "undoes" E_A. Alice knows the message was originated by Bob, because only Bob is able to use his secret decryptor D_B to construct his encrypted message z. If another decryptor had been inserted instead, then Alice's use of Bob's encryptor E_B would have produced gibberish instead of the intelligible plaintext message x. Only Bob could have sent the message! Moreover, if Bob later denies sending the message (which could be a contract of some sort), Alice can show the messages x, y, and z to the judge. The judge can then observe that $xE_A = y = zE_B$ to verify Alice's claim that Bob sent the message: Although Alice could make up x and construct $y = xE_A$, she could not construct $z = yD_B$ without knowing Bob's decryptor D_B. Bob must have sent the message!

Similarly, Alice can send a signed message u to Bob by first applying her decryptor D_A to produce $v = uD_A$ in $Y_A \subseteq X_B$, and then applying Bob's encryptor E_B to produce $w = vE_B = uD_AE_B$ in Y_B, which she sends to Bob. Bob then is able to decrypt and read $u = wD_BE_A$.

Note that the order in which the encryptors and decryptors are applied is important if the containment $X_A \subset X_B$ is strict and $X_A \neq X_B$. If instead of $z = xE_AD_B$ Bob tried to form $z' = xD_BE_A$, he would start with x in $X_A \subset X_B$ and apply D_B to obtain $y' = xD_B$ in X_B, but maybe *not in* X_A ! If y' is not in X_A, Bob cannot apply E_A to y' to get $z' = xD_BE_A$ because E_A only works on elements of its domain $dom(E_A) = X_A$. A similar problem could occur if Alice changed the order in which the encryptors and decryptors are applied in sending a signed message to Bob.

Here are some advantages of public key crypto systems over classical crypto systems:

(1) There is no need for private meetings to exchange keys.

(2) Only N crypto cells are needed for private communication between each pair of a crypto group of N people using a public key system, but $N(N-1)/2$ are needed for a classical system, an increase by a factor of $(N-1)/2$.

(3) Some public key systems allow signatures on messages.

The major disadvantage of public key cryptosystems is that those that have been invented so far are up to a thousand times slower in encrypting and decrypting messages. For this reason, a major use of public key systems may be to exchange keys for a faster classical system.

The first and still most popular public key cryptosystem is the RSA algorithm, which was introduced by its inventors, Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman in their 1978 paper [RSA].

3 Arithmetic

Now we look into what the RSA is and why it works, beginning with a close look into the arithmetic of integers. We will let $\mathbf{N} = \{1, 2, 3, ...\}$ be the set of positive integers or natural numbers and $\mathbb{Z} = \{0, -1, 1, -2, 2, ...\}$ be the set of all integers. (The $\mathbb{Z}$ comes from the German word "Zahl" for "number".) All of the numbers we discuss in this section will be integers; that is, elements of $\mathbb{Z}$. We assume the truth of the following axiom.

Theorem 1. *(***Well Ordering Axiom***) Every nonempty subset of nonnegative integers has a smallest element.*

In grade school you learned a method of long division of a positive integer a (the dividend) by another positive integer d (the divisor) which produced successive remainders and continued until the final remainder r was smaller than the divisor. This process actually proves the next theorem for positive integers a and d, and you can figure out from the positive case why it is true regardless of the signs of a and d. We give another demonstration using (1).

Theorem 2. *(***Division Algorithm***) If a and d are integers and $b \neq 0$, then there are unique integers q and r which satisfy*
(a) $a = dq + r$ *and*
(b) $0 \leq r < |d|$.

Proof. Consider the set S of all integers of the form $a - dx$, where x can be any integer. Since x can be large and positive or large and negative and $d \neq 0$, it is clear that S contains a nonnegative integer. Let $r = a - dq$ be the smallest nonnegative integer in S, obtained when $x = q$. Then $a = dq + r$, so (2a) holds. If $r \geq |d|$ we let $s = sgn(d) = \pm 1$ so that $ds = |d|$; then $x = q + s$ gives $0 \leq a - dx = a - d(q + s) = a - dq - ds = r - |d| < r$, contradicting the choice of r as the smallest nonnegative element of S. Thus, $0 \leq r < |d|$ and r satisfies (2b). To show uniqueness, we assume that $a = dq + r = dq' + r'$ and $0 \leq r' \leq r < |d|$. Then $0 \leq r - r' = d(q' - q) = |d||q' - q|$. It is clear that $r - r' \leq r < |d|$, so we must have $|q' - q| < 1$. Hence we must have $q' - q = r - r' = 0$, so $r = r'$ is unique and $q = q'$ is unique. □

When the remainder $r = 0$ and $a = dq$, we say that d **divides** a, or equivalently, a is a **multiple** of d, and we write $d \mid a$. Otherwise, d does not divide a and we write $d \nmid a$. To reiterate, $d \mid a$ if and only if there is an integer m such that $a = dm$. For example, $3 \mid 12$ and $6 \mid 18$, but $5 \nmid 12$ and $15 \nmid 18$. The following divisibility properties are easily shown to be true.

Lemma 3. *(***Divisibility Properties***) If a, b, c, d, x, and y are integers, then*

(a) $a \mid b$ *and* $b \mid c$ *implies that* $a \mid c$. *(Divisibility is* **transitive***.)*
(b) $a \mid b$ *and* $b \mid a$ *implies that* $a = \pm b$. *(Divisibility is* **antisymmetric***.)*

(c) $d \mid a$ *and* $d \mid b$ *implies that* $d \mid xa + yb$.

An integer d which divides both of the integers a and b is called a **common divisor** of a and b, and the largest of these (when a and b are not both 0) is called the **greatest common divisor (gcd)** of a and b. We write $d = \gcd(a, b)$ for this necessarily positive integer, and define $\gcd(0, 0) = 0$.

Theorem 4. *(***GCD Theorem***) If a and b are any integers, then there are integers x and y such that* $\gcd(a, b) = xa + yb$.

Proof. This is clear for $a = b = 0$. If either a or b is nonzero, let S be the set of all positive integers of the form $sa + tb$, where s and t are integers. Since S is not empty, it has a smallest element by (1). Let $d = xa + yb$ be the smallest element of S. The division algorithm (2) tells us that there are integers q and r satisfying $a = dq + r$ and $0 \leq r < d$. Now $r = a - dq = (1 - x)a + (-y)b$ would be in S if it were positive, which would contradict our choice of d as the smallest element of S. Therefore, $r = 0$ and $a = dq$ is divisible by d. Similarly, $d \mid b$, and so it is a common divisor of a and b. If c is another common divisor of a and b, then $c \mid d = xa + yb$ by (3c) and $d > 0$ implies $c < d$ and $d = \gcd(a, b)$. □

Now it is easy to see that $\gcd(a, 0) = |a|$ and $\gcd(a, b) = \gcd(|a|, |b|)$, so we only need to find gcds of positive integers. Note that if $0 < b < a$, we can use the division algorithm (2) to obtain $a = bq + r$ and $0 \leq r < b$. It is clear from (3c) that the common divisors of a and b are exactly the same as the common divisors of b and r, and so $\gcd(a, b) = \gcd(b, r)$. This suggests that a sequence of divisions can determine the gcd of two positive integers:

$$\begin{array}{ll}
(0) & a = bq_1 + r_1 \\
(1) & b = r_1 q_2 + r_2 \\
(2) & r_1 = r_2 q_3 + r_3 \\
 & \vdots \\
(k) & r_{k-1} = r_k q_{k+1} + r_{k+1} \\
 & \vdots \\
(n) & r_{n-1} = r_n q_{n+1} + r_{n+1}
\end{array} \tag{4}$$

If we continue until the remainder $r_{n+1} = 0$ we will have $\gcd(a, b) = \gcd(b, r_1) = \gcd(r_1, r_2) = \gcd(r_k, r_{k+1}) = \gcd(r_n, 0) = r_n$. This method of finding the gcd was published by Euclid in his *Elements* more than 2000 years ago. It is called **the Euclidean algorithm**. Next we give an example with $a = 54321$ and $b = 12345$.

$$\begin{array}{ll}(1) & 54321 = 12345 \times 4 + 4941 \\ (2) & 12345 = 4941 \times 2 + 2463 \\ (3) & 4941 = 2463 \times 2 + 15 \\ (4) & 2463 = 15 \times 164 + 3 \\ (5) & 15 = 3 \times 5 + 0 \end{array} \tag{5}$$

Thus $\gcd(54321, 12345) = 3$ is the remainder in row (4), the last remainder before the remainder becomes 0. In a later section we will see an extension of this algorithm which finds x and y such that $\gcd(a, b) = xa + yb$.

The divisors of 1 are called **units**. Actually, 1 and -1 are the only units among the integers. An integer a is a **composite** if there are integers b and c such that $a = bc$ and $1 < |b| \leq |c|$, that is, neither b nor c are units. The first ten positive composites are 4, 6, 8, 9, 10, 12, 14, 15, 16, and 18. An integer p is a **prime** if $p > 1$ and the only divisors of p are ± 1 and $\pm p$. The first ten primes are 2, 3, 5, 7, 11, 13, 17, 19, 23, 29. Euclid presented a proof that every positive integer was a product of primes which was unique except for the order of the factors, and he showed that there were infinitely many primes. Two integers are said to be **relatively prime** if their only common factors are ± 1; that is, a and b are relatively prime if and only if $\gcd(a, b) = 1$.

4 Modular Arithmetic

Let m be an integer greater than 1. We say that integers a and b are **congruent modulo** m if and only if $m \mid a - b$, and we denote this by $a \equiv b \pmod{m}$. For example, $75 \equiv 9 \pmod{33}$ because $75 - 9 = 66 = 2 \times 33$ is divisible by 33. The relation of congruence mod m behaves pretty much like equality. The relation is an equivalence relation on the set $\mathbb{Z}$ of all integers which preserves addition and multiplication of integers.

(4.1) **Properties of congruence**: Let m be an integer greater than 1. Then, for any integers a, b, c, a', and b', the following properties hold:

(a) Reflexive: $a \equiv a \pmod{m}$.
(b) Symmetric: $a \equiv b \pmod{m}$ implies $b \equiv a \pmod{m}$.
(c) Transitive: $a \equiv b \pmod{m}$ and $b \equiv c \pmod{m}$ implies $a \equiv c \pmod{m}$.
(d) Addition: $a \equiv a' \pmod{m}$ and $b \equiv b' \pmod{m}$ implies $a + b \equiv a' + b' \pmod{m}$.
(e) Multiplication: $a \equiv a' \pmod{m}$ and $b \equiv b' \pmod{m}$ implies $ab \equiv a'b' \pmod{m}$.

Proof. (a)-(d) are left to the reader. For (e), $a \equiv a' \pmod{m}$ and $b \equiv b' \pmod{m}$ implies $m \mid a - a'$ and $m \mid b - b'$ implies $m \mid (a - a')b + a'(b - b') = aa' - bb'$ implies $ab \equiv a'b' \pmod{m}$. □

For example, $37 \equiv 4 \pmod{33}$ and $45 \equiv 12 \pmod{33}$, so $37 + 45 = 82 \equiv 4 + 12 = 16 \pmod{33}$ and $37 \times 45 = 1665 \equiv 4 \times 12 = 48 \equiv 15 \pmod{33}$, which can be seen directly by the facts that $33 \mid 82 - 16 = 66 = 2 \times 33$ and $33 \mid 1665 - 15 = 1650 = 50 \times 33$.

The Division Algorithm (2) shows that every integer a can be written in the form $a = mq + \overline{a}$ with $0 \leq \overline{a} < m$. Thus, every a can be reduced $(\operatorname{mod} m)$ to an integer $\overline{a}$ in the set $\mathbb{Z}_m = \{0, 1, 2, ..., m-1\}$ of **residues** modulo m. The example above illustrates that it is usually easier to reduce first and then perform arithmetic operations rather than the other way around!

Now we can consider only the m elements in $\mathbb{Z}_m$, perform addition and multiplication on these elements, and reduce them to again get elements of $\mathbb{Z}_m$. The result is an arithmetic modulo m on the set $\mathbb{Z}_m$ that is much the same as the arithmetic on $\mathbb{Z}$. But there are differences! For example, in $\mathbb{Z}_{33}$, $6 \times 9 \equiv 21 (\operatorname{mod} 33)$ and $6 \times 31 \equiv 21 (\operatorname{mod} 33)$, so the law of cancellation does not hold for multiplication by 6 modulo 33. Why not? We get a clue when we subtract the first of these congruences from the second: $6 \times (31 - 9) = 6 \times 22 = 132 \equiv 0 (\operatorname{mod} 33)$. Although $6 \not\equiv 0 (\operatorname{mod} 33)$ and $22 \not\equiv 0 (\operatorname{mod} 33)$, we still have $6 \times 22 \equiv 0 (\operatorname{mod} 33)$. But now we see why. $33 = 3 \times 11$; 6 has a factor of 3 and 22 has a factor of 11, so when multiplied together, the product has a factor of 33, and $33 \equiv 0 (\operatorname{mod} 33)$. Suppose we multiply by a number with no factor in common with 33, for example, 10. Then $10z \equiv 0 (\operatorname{mod} 33)$ means that $10z$ is divisible by 33. But then z must have both a factor of 3 and a factor of 11, since 10 has neither. That means that z is divisible by 33, and so $z \equiv 0 (\operatorname{mod} 33)$. Moreover, it follows that $10x \equiv 10y (\operatorname{mod} 33)$ implies $10(x - y) \equiv 0 (\operatorname{mod} 33)$ implies $x - y \equiv 0 (\operatorname{mod} 33)$ implies $x \equiv y (\operatorname{mod} 33)$. Multiplication by 10 is cancellative modulo 33. These facts are illustrated in the table below.

x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
$6x$	0	6	12	18	24	30	3	9	15	21	27	0	6	12	18	24	30	3	9	15	21	27	0	6	12	18	24	30	3	9	15	21	27
$10x$	0	10	20	30	7	17	27	4	14	24	1	11	21	31	8	18	28	5	15	25	2	12	22	32	9	19	29	6	16	26	3	13	23

Multiplication of elements of $\mathbb{Z}_{33}$ by 6 gives all multiples of 3, each repeated 3 times. But multiplication by 10 gives every element of $\mathbb{Z}_{33}$ exactly once in another order. Note that 10 is a unit in $\mathbb{Z}_{33}$ because $10 \times 10 \equiv 1 (\operatorname{mod} 33)$.

Now let m be a fixed integer that is greater than 1, and let a be any integer. If $d = \gcd(m, a) > 1$, then $m' = m/d$ and $a' = a/d$ are both integers and $m' \not\equiv 0 (\operatorname{mod} m)$ since $0 < m' < m$ but $am' = a'm \equiv 0 (\operatorname{mod} m)$. If there is an $x \not\equiv 0 (\operatorname{mod} m)$ such that $ax \equiv 0 (\operatorname{mod} m)$ we call a a **zero divisor** modulo m.

On the other hand, if $\gcd(m, a) = xm + ya = 1$ by (3.4) then $ya \equiv 1 (\operatorname{mod} m)$ shows that a is a unit modulo m and a has an **inverse** $a^{-1} = y$. (This is why a unit is called **invertible**.) In this case, a is **cancellable** modulo

m, because the congruence $ax \equiv ay (\bmod m)$ need only be multiplied on both sides by a^{-1} to show that $x \equiv y (\bmod m)$. In particular, $az \equiv 0 (\bmod m)$ implies $az \equiv a \cdot 0 (\bmod m)$ implies $z \equiv 0 (\bmod m)$, and so a is not a zero divisor. Thus we have the following result.

Theorem 5. *Let m be an integer that is greater than 1 and let a be any integer. Then the following are equivalent:*

(a) $\gcd(m, a) = 1$.

(b) $az \equiv 0 (\bmod m)$ implies that $z \equiv 0 (\bmod m)$; that is, a is not a zero divisor modulo m.

(c) $ax \equiv ay (\bmod m)$ implies $x \equiv y (\bmod m)$; that is, a is cancellable modulo m.

(d) a has an inverse modulo m; that is, there is an element b in $\mathbb{Z}_m$ such that $ab \equiv 1 (\bmod m)$.

We define $\mathbb{Z}_m^* = \{a \in \mathbb{Z}_m : \gcd(a, m) = 1\}$ to be the set of units in $\mathbb{Z}_m$. If a is in $\mathbb{Z}_m^*$, then multiplication by a is cancellable modulo m, and so multiplying all of the elements in $\mathbb{Z}_m$ by a simply moves the elements around. Moreover, if a and b are both in $\mathbb{Z}_m^*$, then $abb^{-1}a^{-1} = a1a^{-1} = 1$, so ab is invertible and is also in $\mathbb{Z}_m^*$. That is, $\mathbb{Z}_m^*$ is closed under multiplication, and if a is any unit, multiplying the set $\mathbb{Z}_m^*$ of all units by a simply moves the elements of $\mathbb{Z}_m^*$ around, or permutes them. The following tables of multiples of $\mathbb{Z}_{33}^*$ illustrate this.

x	1	2	4	5	7	8	10	13	14	16	17	19	20	23	25	26	28	29	31	32
$7x$	7	14	28	2	16	23	4	25	32	13	20	1	8	29	10	17	31	5	19	26
$10x$	10	20	7	17	4	14	1	31	8	28	5	25	2	32	19	29	16	26	13	23

The **Euler totient function** φ is defined by $\varphi(m) = |\mathbb{Z}_m^*|$, that is, $\varphi(m)$ is the number of units modulo m. For example,

$$\varphi(33) = |\{1, 2, 4, 5, 7, 8, 10.13, 14, 16, 17, 19, 20, 23, 25, 26, 28, 29, 31, 32\}| = 20.$$

For general $m > 1$ and any a with $\gcd(m, a) = 1$, we can write $\mathbb{Z}_m^* = \{a_1, a_2, \ldots, a_{\varphi(m)}\} = a\mathbb{Z}_m^* = a\{a_1, a_2, \ldots, a_{\varphi(m)}\} = \{aa_1, aa_2, \ldots, aa_{\varphi(m)}\}$, since multiplication by a simply permutes the elements of $\mathbb{Z}_m^*$. Now let A be the product of every element in $\mathbb{Z}_m^*$. Then $A \equiv a_1 a_2 \cdots a_{\varphi(m)} \equiv aa_1 aa_2 \cdots aa_{\varphi(m)} \equiv a^{\varphi(m)} a_1 a_2 \cdots a_{\varphi(m)} \equiv a^{\varphi(m)} A \ (\bmod m)$. Cancellation of A modulo m, which is valid since $A \in \mathbb{Z}_m^*$, gives $1 \equiv a^{\varphi(m)} (\bmod m)$. This proves Euler's Theorem, which is the mathematical basis of the RSA algorithm!

Theorem 6. *(**Euler's Theorem**) If* $\gcd(a, m) = 1$, *then* $a^{\varphi(m)} \equiv 1 \ (\bmod m)$.

Here are a couple of examples modulo 33: $\gcd(33, 10) = 1$ and $10^{20} = 100000000000000000000 = 99999999999999999999 + 1 \equiv 1 (\bmod 33)$;

$\gcd(33,7) = 1$ and $7^{20} = (49)^{10} \equiv (16)^{10} = (2^4)^{10} = 2^{40} = (2^5)^8 = (32)^8 \equiv (-1)^8 = 1(\mathrm{mod}\, 33)$.

Now if p is a prime it is easy to see that $\varphi(p) = p - 1$; moreover, for any integer a, it is easy to see that $\gcd(a, p) = 1$ if $p \nmid a$ and $\gcd(a, p) = p$ if $p \mid a$. In the first case Euler's Theorem (4.3) shows that $a^{p-1} \equiv 1(\mathrm{mod}\, p)$, and multiplication on both sides by a gives $a^p \equiv a(\mathrm{mod}\, p)$. In the second case, $a \equiv 0 \equiv a^p(\mathrm{mod}\, p)$. Thus, in any case, we have the following corollary to Euler's Theorem.

Theorem 7. *(***Fermat's Little Theorem***) Let p be a prime. If a is any integer, then $a^p \equiv a(\mathrm{mod}\, p)$ and $a^{p-1} \equiv 1(\mathrm{mod}\, p)$ if and only if $p \nmid a$.*

Theorem 7 can be used to show that a number is *not* prime. For example, $2^{32} = 2^2(2^5)^6 = 4(32)^6 \equiv 4(-1)^6 = 4(\mathrm{mod}\, 33)$ shows that 33 is not prime. However, $2^{10} = (2^5)^2 = (32)^2 \equiv (-1)^2 = 1$ $(\mathrm{mod}\, 11)$ shows that 11 *may* be prime. More evidence is given by the fact that $3^{10} = (3^5)^2 = (243)^2 \equiv (1)^2 = 1(\mathrm{mod}\, 11)$, but it still doesn't *prove* that 11 is prime.

Corollary 8. *Let p and q be different odd primes, let $m = pq$, and suppose that $r \equiv 1(\mathrm{mod}\,(p-1))$ and $r \equiv 1(\mathrm{mod}\,(q-1))$. If a is any integer, then $a^r \equiv a(\mathrm{mod}\, m)$.*

Proof. If $p \nmid a$, then $a^r = a^{k(p-1)+1} = (a^{p-1})^k(a) \equiv (1)^k(a) = a(\mathrm{mod} p)$. If $p \mid a$, then $a \equiv 0 \equiv a^r(\mathrm{mod} p)$. In either case, $a^r \equiv a(\mathrm{mod}\, p)$ and $p \mid a^r - a$. Similarly, $q \mid a^r - a$. Since both p and q divide $a^r - a$, it follows that $m = pq$ divides $a^r - a$ and hence that $a^r \equiv a(\mathrm{mod}\, m)$. □

5 The RSA Algorithm

Now we are finally able to describe the RSA public key cryptosystem! The RSA algorithm is actually a **cipher**, which means that it works on letters of the alphabet or on the symbols used to write a language rather than on words or meaningful phrases of the language. It really acts on a collection of numbers, so the first job is to get a uniform method of converting the symbols we want to transmit into numbers. One method would be to replace A with 01, B with 02, ..., Z with 26, and communicate only with these 26 letters. Another method would employ the ASCII code. We assume that some such uniformly understood technique has been established throughout the cryptosystem, and will not concern ourselves with it any more. For us, a message will be a number!

Since the RSA algorithm is a cipher, we will use the terms "encipher" and "decipher" that apply to ciphers ("encode" and "decode" are the corresponding terms for codes) rather than the more general terms "encrypt" and "decrypt" that apply to both ciphers and codes, and will make other changes in terminology as appropriate.

Here is what to do in order to set up an RSA cipher. We will discuss how to do it in §6 and §7.

(1) **Secretly** choose two large primes p and q, say each of about 100 digits, with $100 < q/p < 10000$, so that q has 2 to 4 more digits than p. For a small example, let $p = 3$ and $q = 11$.

(2) Let $m = pq$. Note that $\mathbb{Z}_m = \{0, 1, 2, \dots, m-1\}$ contains q multiples of p, namely $0 \cdot p$, $1 \cdot p$, $2 \cdot p$, ... , $(q-1) \cdot p$, and p multiples of q. These are the only elements of $\mathbb{Z}_m$ which have factors in common with m, and so $\mathbb{Z}_m^*$ contains the remaining $m - p - q + 1 = (p-1)(q-1)$ elements of $\mathbb{Z}_m$ that are relatively prime to m. (The "+1" compensates for the fact that 0 was counted twice, once among the q multiples of p and again among the p multiples of q.) Thus, $\varphi(m) = |\mathbb{Z}_m^*| = m - p - q + 1 = (p-1)(q-1)$. Abbreviate $\varphi(m)$ by φ. In our example, $m = 33$ and $\varphi = 20$.

(3) Choose $e > 10^5$ such that $\gcd(\varphi, e) = 1$ and **secretly** find d such that $ed \equiv 1 (\operatorname{mod} \varphi)$. That is, $d \equiv e^{-1} (\operatorname{mod} m)$, and $ed = k\varphi + 1$ for some integer k. For our example,

$$\varphi = 20, e = 7, \varphi - 2e = 6, -\varphi + 3e = 1, d = 3.$$

(4) Publish the enciphering key (m, e). Keep the deciphering key (m, d) *secret.*

The security of the RSA cipher is based on the ease of finding the deciphering number d when the factorization of $m = pq$ is known and the difficulty of finding d from m and e when the factorization is not known. This will be discussed at greater length in the next two sections.

How is the RSA cipher used to send a message? Anyone can use the public key (m, e) to encipher the message x in $X = \mathbb{Z}_m$ by raising x to the power e and reducing modulo m to obtain $y = xE \equiv x^e (\operatorname{mod} m)$. To decipher the message y, the holder of the secret deciphering key (m, d) raises y to the power d and reduces modulo m to obtain $x = yD = y^d (\operatorname{mod} m)$. For our example with $(m, e) = (33, 7)$, the message $x = 17$ is enciphered as $y = 17E = 17^7 \equiv (17)(-16)^6 \equiv (17)(16^2)^3 \equiv (17)((32)(8))^3 \equiv (17)(-8)^3 = (17)(64)(-8) \equiv (17)(-2)(-8) \equiv (-34)(-8) \equiv (-1)(-8) = 8 (\operatorname{mod} 33)$. Then $y = 8$ is deciphered with the secret key $(m, d) = (33, 3)$ to obtain $x = y^3 = 8^3 = 8(8^2) = 8(64) \equiv 8(-2) = -16 \equiv 17 (\operatorname{mod} 33)$.

Why does this work? It is a result of Corollary (4.5): $yD = (xE)D \equiv (x^e)^d = x^{ed} = x^{k\varphi+1} \equiv x (\operatorname{mod} m)$ because $k\varphi + 1 = k(p-1)(q-1) + 1 \equiv 1$ modulo $p - 1$ and modulo $q - 1$. Similarly, $x = yD \equiv y^d (\operatorname{mod} m)$ implies $xE = (yD)E \equiv (y^d)^e = y^{de} \equiv y (\operatorname{mod} m)$. Thus, E and D are one to one functions from the set $X = Y = \mathbb{Z}_m$ onto itself; that is, they are *permutations* of $\mathbb{Z}_m$. They have the property that for any x and y in $\mathbb{Z}_m$, $xE = y$ if and only if $yD = x$. The **decipherer** (decryptor for a cipher) D is the **inverse function** $D = E^{-1}$ of the **encipherer** (encryptor for a cipher) E, and vice versa, $E = D^{-1}$. Each undoes what the other does!

Let us suppose that Alice and Bob have independently taken the above four steps to set up their RSA ciphers. Alice has published her enciphering key (the encrypting key used with a cipher) (m_A, e_A) and has kept her deciphering key (m_A, d_A) secret. Alice's set of plaintext and enciphered "messages" are both the same set $X_A = Y_A = \mathbb{Z}_{m_A} = \{0, 1, 2, \dots, m_A - 1\}$ of all nonnegative integers less than m_A. Alice's encipherer E_A transforms any x in $X_A = \mathbb{Z}_{m_A}$ to $y = xE_A = x^{e_A}(\operatorname{mod} m)_A$, which is also in $\mathbb{Z}_{m_A}$. So Bob sends $y = xE_A = x^{e_A}(\operatorname{mod} m)_A$ to Alice. When Alice receives y, she applies her decipherer D_A to y to obtain $yD_A = y^{d_A}(\operatorname{mod} m)_A = x$.

Now Bob has public enciphering key (m_B, e_B), private deciphering key (m_B, d_B), and message set $X_B = Y_B = \mathbb{Z}_{m_B} = \{0, 1, 2, \dots, m_B - 1\}$. Assume that $m_A < m_B$. Then $X_A \subsetneqq X_B$. If Bob wants to send a signed message x to Alice, he first applies her public encipherer E_A to obtain $y = xE_A$ in $X_A \subsetneqq X_B$. Since y is also in X_B, he can apply his decipherer D_B to obtain $z = yD_B = xE_AD_B$ in X_B, which he sends to Alice. When Alice receives the message z, she knows it is supposed to have come from Bob and that $X_A \subsetneqq X_B$, so she first applies Bob's publicly known encipherer E_B to obtain $zE_B = y$ in X_B. But y *is also in* X_A, so Alice can then apply her secret decipherer D_A to read the message $zE_BD_A = yD_A = x$. As explained in §**2**, Alice knows the message was originated by Bob, and she can prove that Bob sent the message to an impartial arbiter if Bob later denies it. Bob must be careful to apply E_A first and D_B second, because $y' = xD_B$ might not be in the domain $X_A = \mathbb{Z}_{m_A}$ of E_A and so $y'E_A$ may be undefined. Worse than this, if Bob carelessly calculates $z' = (y')^{e_A}(\operatorname{mod} m_A) = ((x^{d_B})(\operatorname{mod} m)_B)^{e_B}(\operatorname{mod} m)_A$, then Alice may not be able to recover x from z'. We will see what can happen in the examples below.

In like manner, Alice can send a signed message u to Bob by forming $v = uD_A$ and then $w = vE_B = uD_AE_B$ that Bob can read as $u = wD_BE_A$. Alice also must exercise care in applying D_A and E_B in the correct order.

For example, suppose Alice has enciphering key $(m_A, e_A) = (33, 7)$ and deciphering key $(m_A, d_A) = (33, 3)$, and Bob has keys $(m_B, e_B) = (65, 11)$ and $(m_B, d_B) = (65, 35)$, and Bob wants to send the signed message $x = 18$ to Alice. (The reader can follow the calculations by using the repeated squaring algorithm described in the next section or a computer program like MAPLE.) Bob then calculates $y = 18E_A = (18^7)(\operatorname{mod} 33) = 6$ and then $z = 6D_B = (6^{35})(\operatorname{mod} 65) = 11$, which he sends to Alice. Alice then applies Bob's encipherer E_B to $z = 11$ to obtain $y \equiv 11^{11}(\operatorname{mod} 65) = 6$, and then applies her decipherer D_A to $y = 6$ to obtain $x = 6D_A \equiv 6^3(\operatorname{mod} 33) = 18$ in order to read the original message $x = 18$.

But what happens if Bob applies his deciphering key first and then his enciphering key? Bob transforms $x = 18$ to $y' = 18D_B \equiv 18^{35}(\operatorname{mod} 65) = 47$, and then calculates $z' = 47E_A = 47^7(\operatorname{mod} 33) = 20$, not realizing that $y' = 47$ is not in the domain of E_A. If Alice calculates either $y_1 = z'D_A = 20^3(\operatorname{mod} 33) = 14$ and then $x_1 = y_1E_B = 14^{11}(\operatorname{mod} 65) = 14$ or $y_2 = z'E_B = 20^{11}(\operatorname{mod} 65) = 15$ and then $x_2 = y_2D_A = 15^3(\operatorname{mod} 33) = 9$, neither $x_1 = z'D_AE_B = 14$ nor $x_2 = z'E_BD_A = 9$ is the message $x = 18$ that Bob intended. *The order of application of operators is important!* When *sending* a signed message the operator (encipherer or decipherer) associated with the smaller modulus m must be applied first, and when *receiving* a signed message the operator associated with the larger modulus must be applied first.

The reader is invited to construct more examples of signed messages between Alice and Bob. The author was amused to find that Bob could have successfully sent the sample message $x = 17$ to Alice as a signed message with operators reversed because $17D_B = 17^{35} (\bmod 65) = 23$ turned out to be in X_A.

6 Algorithms and Time Estimates

In order to *set up* an RSA cipher algorithm, one must first find two large primes p and q, find a number e relatively prime to $\varphi = (p-1)(q-1)$, and then find the inverse d of e modulo φ. Then, in order to *use* the algorithm, one must calculate the residues of x^e and y^d modulo $m = pq$, and all of this has to be "easy" for numbers d, e, φ, m, x, y with up to 200 digits each! But it has to be "hard" to factor m without a prior knowledge of p, q, φ, or d. In this section and in the next we explain why these tasks are "easy" or "hard". Much of this material in this section is covered in greater detail in sections 1 and 3 of Chapter I of [Ko].

The ease or difficulty of performing an arithmetic procedure can be measured by the number of **bit operations** needed to carry it out. A bit operation is the basic computer step used to calculate a single bit in the addition or subtraction of two numbers in binary notation. Thus the addition or subtraction of two k-bit integers requires k bit operations. Now a positive integer $a = a_0 + a_1 b + \cdots + a_{k-1} b^{k-1}$ with $0 \leq a_j < b$ and $a_{k-1} \neq 0$ has $k - 1 \leq \log_b a < k$ and hence has $k = 1 + \lfloor \log_b a \rfloor$ base b "digits". (Here $\lfloor t \rfloor$ denotes the greatest integer less than or equal to t. $\lfloor t \rfloor$ is called the **floor** of the real number t, and it satisfies $\lfloor t \rfloor \leq t < \lfloor t \rfloor + 1$. For example, $\lfloor 3.14 \rfloor = 3 = \lfloor 3 \rfloor$ and $\lfloor -3.2 \rfloor = -4 = \lfloor -4 \rfloor$. Most computer languages and calculators use $INT(t)$ for $\lfloor t \rfloor$.) Thus the positive integer a has $1 + \lfloor \log_2 a \rfloor$, or about $\log_2 a$, bits. For integers a and b we write

$$T(a \pm b) = O(\log_2 a) \text{ when } 0 < b \leq a \tag{6}$$

to indicate that addition (or subtraction) takes time proportional to the number of bits in the largest addend.

The "big O-notation" is defined as follows. If f and g are functions of n variables $x_1, x_2, \cdots, x_n$, we say that f is bounded by g, and write $f(x_1, x_2, \cdots, x_n) = O(g(x_1, x_2, \cdots, x_n))$ or $f = O(g)$, if there are constants B and $C > 0$ such that $0 < f(x_1, x_2, \cdots, x_n) < Cg(x_1, x_2, \cdots, x_n)$ whenever all of the $x_j > B$. The reader may find it helpful to simply interpret the big "O" as a fixed but unknown positive constant.

Consider the product $27 \times 11 = 297$ in binary (base 2) notation.

$$\begin{array}{r} 1\,1\,0\,1\,1\\ 1\,0\,1\,1\\ \hline 1\,1\,0\,1\,1\\ 1\,1\,0\,1\,1\\ 1\,1\,0\,1\,1\\ \hline 1\,0\,0\,1\,0\,1\,0\,0\,1\end{array}$$

From this example we see that if a has k bits and b has j bits with $j \leq k$, we can find the product by writing one copy of the multiplicand a with its unit bit aligned beneath each 1 in the multiplier b and then adding these staggered copies of a pairwise to form the product. There are no more than j additions, each taking no more than $j + k \leq 2k$ bit operations, and so the whole process requires no more than $j(j + k) \leq 2jk = O(\log_2 b \log_2 a)$ bit operations. This yields

$$T(a \times b) = O((\log_2 a)(\log_2 b)) = O((\log_2 a)^2) \text{ when } 0 < b \leq a. \tag{7}$$

A similar analysis of the grade school long division algorithm leads to the same result for division of the positive integer a by the positive integer d to obtain the quotient q and the remainder r satisfying the division algorithm (2). We indicate the time required by

$$T(q, r : a = dq + r) = O((\log_2 a)(\log_2 d)) = O((\log_2 a)^2) \text{ when } 0 < d \leq a. \tag{8}$$

Note that if we want to find the residue $\overline{a}$ modulo m of an integer a we need only use the division algorithm to find q and $\overline{a}$ such that $a = mq + \overline{a}$ with $0 \leq \overline{a} < m$, so it follows that the time needed for the procedure is

$$T(\overline{a} : a \equiv \overline{a} (\operatorname{mod} m) \text{ and } 0 \leq \overline{a} < m) = O((\log_2 a)(\log_2 m)). \tag{9}$$

We should remark that we are obtaining crude upper bounds for the times required to perform various calculations which can be lowered considerably by carefully examining more sophisticated algorithms presently in use. Our goal is just to get a rough idea of the difficulty involved in the RSA calculations.

We want to extend the Euclidean algorithm (4) to one that will find not only $d = \gcd(a, b)$, but also find x and y so that $d = \gcd(a, b) = xa + yb$. We use vector notation $\overline{v} = \langle v_1, v_2, v_3 \rangle$ and write the algorithm in pseudo computer code, using the notation "$\overline{u} \leftarrow \overline{v}$" to mean "replace $\overline{u}$ by the value of $\overline{v}$".

Theorem 9. *(***Extended Euclidean Algorithm***) Algorithm for* $d = \gcd(a, b)$ $= xa + yb$ *for* $0 \leq b < a$.

(a) $\overline{u} \leftarrow \langle a, 1, 0 \rangle$

```
(b) $\overline{v} \leftarrow \langle b, 0, 1\rangle$
(c) do $\overline{w} \leftarrow \overline{u} - \lfloor u_1/v_1 \rfloor \overline{v}$, $\overline{u} \leftarrow \overline{v}$, $\overline{v} \leftarrow \overline{w}$ while $v_1 \neq 0$
(e) $d \leftarrow u_1$,  $x \leftarrow u_2$, $y \leftarrow u_3$
(f) return $d, x, y$.
```

We repeat example (3.6) in matrix form, with the sequence of vectors $\overline{u}$ and $\overline{v}$ written as rows of the matrix. On the right we write the example as we might do it "by hand". Both forms illustrate that $\gcd(54321, 12345) = 3 = (-822)(54321) + (3617)(12345)$.

$$(6.6)\quad \begin{bmatrix} 54321 & 1 & 0 \\ 12345 & 0 & 1 \\ 4941 & 1 & -4 \\ 2463 & -2 & 9 \\ 15 & 5 & -22 \\ 3 & -822 & 3617 \\ 0 & 4115 & -18107 \end{bmatrix} \qquad \begin{array}{rrr} a & & = 54321 \\ & b & = 12345 \\ a\ - & 4b & = \ 4941 \\ -2a\ + & 9b & = \ 2463 \\ 5a\ - & 22b & = \ 15 \\ -822a\ + & 3617b & = \ 3 \\ 4115a\ - & 18107b & = \ 0 \end{array}$$

This process really amounts to the row reduction of the matrix $\begin{bmatrix} a & 1 & 0 \\ b & 0 & 1 \end{bmatrix}$ to $\begin{bmatrix} d & x & y \\ 0 & u & v \end{bmatrix}$ with $d = \gcd(a, b) = xa + yb$ and $0 = ua + vb$. At the k^{th} step, starting at $k = 0$ with $r_{-1} = a$, $r_0 = b$, $x_{-1} = 1$, $x_0 = 0$, $y_{-1} = 0$, and $y_0 = 1$, we have

$$\begin{bmatrix} r_k & x_k & y_k \\ r_{k+1} & x_{k+1} & y_{k+1} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & -q_{k+1} \end{bmatrix} \begin{bmatrix} r_{k-1} & x_{k-1} & y_{k-1} \\ r_k & x_k & y_k \end{bmatrix},$$

where the r_k and $q_{k+1} = \lfloor r_{k-1}/r_k \rfloor$ are the remainders and quotients displayed in (4). Note that at every stage we have $r_k = x_k a + y_k b$. This technique has been discussed several times (See [Bl] and [MW].) in the literature.

Consider rows (k) and (k+1) of (4):

$$\begin{array}{rl} \text{(k)} & r_{k-1} = r_k q_{k+1} + r_{k+1}, \\ \text{(k+1)} & r_k = r_{k+1} q_{k+2} + r_{k+2}. \end{array}$$

If $r_{k+1} \leq \frac{1}{2} r_k$ it follows from the division algorithm that $0 \leq r_{k+2} < r_{k+1} < \frac{1}{2} r_k$, and if $r_{k+1} > \frac{1}{2} r_k$, then $q_{k+2} = 1$ and $r_{k+2} = r_k - r_{k+1} < \frac{1}{2} r_k$. Eventually $r_{2k} < \frac{1}{2^k} a < 1$, so the algorithm must terminate in about $n = 2 \log_2 a$ steps. The calculation involved for each step occurs in line (c) of (9) and involves one division and three multiplications of numbers no larger than a, and so requires no more than $4(\log_2 a)^2$ bit operations, by (8) and (7). The whole algorithm can be accomplished with no more than $8(\log_2 a)^3$ bit operations. This proves

$$T(d, x, y : d = \gcd(a, b) = xa + yb) = O((\log_2 a)^3) \quad \text{when} \quad 0 < b < a. \tag{10}$$

Now we turn our attention to the problem of evaluating $x^e (\mathrm{mod}\, m)$. For this purpose we consider the following construction.

Theorem 10. *(***Mod Power algorithm***) To evaluate* $x^e (\mathrm{mod}\, m)$*, perform the following algorithm:*

```
(a) $E \leftarrow e,\ B \leftarrow x,\ P \leftarrow 1,$
(b) do until $E = 0$
(c)     if $E$ even
(d)         $E \leftarrow E/2,\ B \leftarrow B \cdot B (\mathrm{mod}\, m)$
(e)     else
(f)         $E \leftarrow E - 1,\ P \leftarrow P \cdot B (\mathrm{mod}\, m)$
(g)     end if
(h) return $P$.
```

Proof. It suffices to show that $P \cdot B^E \equiv x^e (\mathrm{mod}\, m)$ at every step. This is clear in line (a). If true before line (d) is executed, it holds afterward, because $P \cdot B^E = P \cdot (B \cdot B)^{E/2}$. And if true before line (f) is executed, it holds afterward, because $P \cdot B^E = (P \cdot B) \cdot B^{E-1}$. Therefore $P \equiv x^e (\mathrm{mod}\, m)$ when $E = 0$. □

When E is in binary form, line (c) is performed by checking whether the final bit is 0 or not. If so, $E \leftarrow E/2$ is performed by dropping final 0, and if not, $E \leftarrow E - 1$ is performed by changing the final 1 to a 0, so these operations take negligible time. Since B and P are both less than m, we see that each of the products $B \cdot B$ and $P \cdot B$ takes no more than $O((\log_2 m)^2)$ bit operations, by (6.2). Then replacing a by m^2 in (9) shows that each of the operations $B \leftarrow B \cdot B (\mathrm{mod}\, m)$ and $P \leftarrow P \cdot B (\mathrm{mod}\, m)$ takes $O((\log_2 m)^2)$ bit operations. Adding these two $O((\log_2 m)^2)$ gives $O(2(\log_2 m)^2) = O((\log_2 m)^2)$ for the execution of either line (d) or (f). After each execution of line (f), E becomes even, so each execution of (f) must be followed by an execution of (d), thereby halving E. Therefore, there can be at most $2 \log_2 e$ passes through the loop (b). Hence the algorithm requires no more than $O((\log_2 e)(\log_2 m)^2)$ bit operations.

$$T(x^e (\mathrm{mod}\, m)) = O((\log_2 e)(\log_2 m)^2). \tag{11}$$

We end this discussion of $x^e (\mathrm{mod}\, m)$ with a numerical example in which we use the algorithm (10) to evaluate $18^7 (\mathrm{mod}\, 33)$, $47^7 (\mathrm{mod}\, 33)$, $6^{35} (\mathrm{mod}\, 65)$, and $18^{35} (\mathrm{mod}\, 65)$. The reader should refer back to (10) to be sure he understands the changes in B, E, and P.

$(\bmod 33)$		
B	E	P
18	7	1
18	6	18
27	3	18
27	2	24
3	1	24
3	0	6
$18^7 (\bmod 33) = 6$		

$(\bmod 33)$		
B	E	P
47	7	1
47	6	47
31	3	47
31	2	5
4	1	5
4	0	20
$47^7 (\bmod 33) = 20$		

$(\bmod 65)$		
B	E	P
6	35	1
6	34	6
36	17	6
36	16	21
61	8	21
16	4	21
61	2	21
16	1	21
16	0	11
$6^{35} (\bmod 65) = 11$		

$(\bmod 65)$		
B	E	P
18	35	1
18	34	18
64	17	18
64	16	47
1	8	47
1	4	47
1	2	47
1	1	47
1	0	47
$18^{35} (\bmod 65) = 47$		

We conclude this section with the following definition which is crucial to the study of time estimates for algorithms.

Definition 11. An algorithm to perform a computation involving n integers $x_1, x_2, \cdots, x_n$ of $k_1, k_2, \cdots, k_n$ bits, respectively, is a **polynomial time algorithm** if there are positive integers $d_1, d_2, \cdots, d_n$ such that the number of bit operations required to perform the algorithm is $O(k_1^{d_1} k_2^{d_2} \cdots k_n^{d_n})$. In this case, we say it is of **degree** d_j **in** k_j (or $\log_2 x_j$) and that it is of **total degree** $d_1 + d_2 + \cdots + d_n$.

All of the algorithms presented in this section are of polynomial time. Basically, polynomial time algorithms are considered to be easy, and non-polynomial time algorithms are considered to be hard!

7 Implementation

The first job in implementing an RSA cipher is to secretly find two primes p and q of about $98-99$ and $101-102$ digits, respectively, so that $m = pq$ has about 200 digits. For greater security we can choose larger primes. In order to make them hard for our antagonists to discover, we should *randomly* select the primes. One way of doing this is to use a random number generator to pick an integer with the appropriate number of digits, add one if it is even, and test the resulting integer n for primality.

But how can we tell that an odd integer n is prime? We could try the "dumb test": Try to divide n by every positive integer $d \leq \sqrt{n}$. This could take about 10^{50} trial divisions, each taking an average of $(\log_2 10^{50})^2 \approx 27,000$ bit operations, for a total of about 3×10^{54} bit operations. If our computer can do 10^9 bit operations a second, this could take 3×10^{45} seconds, or about 8×10^{38} years.

Fortunately, there are better ways! We know by Fermat's Little Theorem 7 that if n is prime and $1 < b < n$, then

$$b^{n-1} \equiv 1(\bmod n). \tag{12}$$

Hence, if (12) does not hold for some b satisfying $1 < b < n$, then we *know* that n is composite, and we call b a **witness** to the fact that n is not prime. On the other hand, n could be an odd composite number and (12) could still hold for some b satisfying $1 < b < n$, but this is not likely.

Definition 12. If n is an odd composite number and (12) holds for some b satisfying $1 < b < n$, then we say that n is a **pseudoprime to the base** b.

For example, 91 is a pseudoprime to the base 3 and to the base 10, since $3^{90} \equiv 10^{90} \equiv 1(\bmod 91)$. However, both 2 and 5 are witnesses to the compositeness of 91, since $2^{90} \equiv 5^{90} \equiv 64(\bmod 91)$. These congruences are fairly easy to calculate once you realize that $3^6 \equiv 1(\bmod 91)$ because $3^6 \equiv 1(\bmod 91)$ implies that $10^6 = (10^2)^3 \equiv 9^3 = 3^6 \equiv 1(\bmod 91)$ and $64 \equiv -27(\bmod 91)$ implies that $64^2 \equiv 27^2 = 3^6 \equiv 1(\bmod 91)$, so $2^{90} = (2^6)^{15} = 64^{15} = 64(64^2)^7 \equiv 64(1)^7 = 64(\bmod 91)$ and $5^6 \equiv (2^6)^{-1} = 64^{-1} \equiv 64(\bmod 91)$ because $(2^6)(5^6) = 10^6 \equiv 1(\bmod 91)$.

Unfortunately, there are odd positive composites n, called **Carmichael numbers** , such that (12) holds whenever $\gcd(n, b) = 1$. As recently as 1994, Alford, Granville, and Pomerance showed in [AGP] that there are infinitely many Carmichael numbers. The smallest Carmichael number is $561 = 3 \times 11 \times 17$. Note that (12) holds for $b = 2, 4, 5, 7, 8, 10$, and 13, but of course it can't hold for $b = 3, 6, 9, 11, 12, 15$, or 17, or any other b with $\gcd(561, b) > 1$, since $b^k \equiv 1(\bmod n)$ implies $gcd(n, b) = 1$.

We remarked that $2^{560} \equiv 1(\bmod 561)$. Note that $560 = 35 \times 2^4$ and $2^{35} \equiv 263(\bmod 561)$, $2^{35\times 2} \equiv 263^2 \equiv 166(\bmod 561)$, $2^{35\times 2^2} \equiv 166^2 \equiv 67(\bmod 561)$, $2^{35\times 2^3} \equiv 67^2 \equiv 1(\bmod 561)$. This last congruence implies that $561 \mid 67^2 - 1 = (67-1)(67+1) = 66 \times 68$. But clearly $561 \nmid 66$ and $561 \nmid 68$, and so both $\gcd(561, 66) = 33$ and $\gcd(561, 68) = 17$ are nonunit factors of $561 = 33 \times 17$. Several useful lessons can be learned from this example.

Lemma 13. *If a and n are positive integers with n odd and $1 < a < n-1$ such that $a^2 \equiv 1(\bmod n)$, then $\gcd(n, a-1) > 1$, $\gcd(n, a+1) > 1$, and $n = \gcd(n, a-1)\gcd(n, a+1)$ is composite.*

Proof. By hypothesis, $n \mid a^2 - 1 = (a-1)(a+1)$. Note that we have $3 < a < n-1$, since $a = 2$ and $a = 3$ yield contradictions. But $a \not\equiv \pm 1(\bmod n)$ implies $n \nmid a \mp 1$ implies $\gcd(n, a-1) > 1$ and $\gcd(n, a+1) > 1$. The fact that $n \mid (a-1)(a+1)$ implies that $n \mid \gcd(n, a-1)\gcd(n, a+1)$. Since $\gcd(a-1, a+1) = \gcd(a-1, 2) = 1$ or 2 and n is odd, $\gcd(n, a-1)$ and $\gcd(n, a+1)$ are relatively prime, and they both divide n, and so their product divides n. That is, $n \mid \gcd(n, a-1)\gcd(n, a+1) \mid n$, so $n = \gcd(n, a-1)\gcd(n, a+1)$. □

Definition 14. Suppose that n is an odd composite number and $n - 1 = 2^s t$ with t odd. If $b \in \mathbb{Z}_n^*$ satisfies either $b^t \equiv 1(\bmod n)$ or $b^{2^r t} \equiv -1(\bmod n)$ for some r such that $0 \leq r < s$, then n is called a **strong pseudoprime to the base** b.

The next theorem is a quotation of **Proposition V.1.7** from [Ko], page 130. The reader is referred to [Ko] for the proof.

Theorem 15. *If n is an odd composite integer, then n is a strong pseudoprime to the base b for at most 25% of all $0 < b < n$.*

This suggests a probabilistic test for the primality of an odd integer n, called the **Miller-Rabin primality test**. First, write $n - 1 = 2^s t$ with t odd. This takes negligible time if n is written in binary form: s is just the number of trailing zeros in $n - 1$ and t is the number left when these zeros are dropped. Randomly select an integer b satisfying $1 < b < n$. Use the Mod Power algorithm (10) to evaluate $c \equiv b^t(\bmod n)$ in time $O((\log_2 n)^3)$. If either $c = 1$ or $c^{2^r} \equiv -1(\bmod n)$ for some r with $0 \leq r < s$, we say that n passes the test for base b. In this case, n is either a prime or is a strong pseudoprime to the base b. When $c \neq 1$ we must square and reduce the result modulo n, doing this r times with $0 \leq r < s < \log_2 n$, which can be done in time $rO((\log_2 n)^2) = O((\log_2 n)^3)$,. by (6.2) and (9). Suppose that n passes the test for k randomly chosen values of b with $1 < b < n$. It follows from Theorem (7.4) that the probability that n is composite is $\leq 1/4^k$.

If n passes the test for 50 or more values of b, we might call it an **industrial grade prime**. For a 100 digit odd number n this could take on the order of

$$50(\log_2 n)^3 \approx 50(333)^3 \approx 2 \times 10^9$$

bit operations, or 2 seconds on the very fast computer hypothesized at the beginning of this section. If we assume, perhaps more realistically, that our computer can perform 10 million bit operations per second, then it takes on the order of 3 or 4 minutes.

For each positive integer n, let $\pi(n)$ be the number of primes less than or equal to n. The **Prime Number Theorem** says thatlim$(\frac{\pi(n)\log n}{n}) = 1$ as $n \longrightarrow \infty$. Thus $\pi(n) \approx \frac{n}{\log n}$ for large values of n and the frequency of primes near a large value of n is about $\frac{1}{\log n}$, so one would expect to test $O(\log n)$ numbers in order to find a prime bigger than n. Putting all of this together gives the following result.

Lemma 16. *An industrial grade prime $p \geq n$ can be found with $O((\log_2 n)^4)$ bit operations.*

The purist may be disappointed in not having a *definitive* polynomial time primality test that actually *proves* that a number is prime. Although not polynomial time, there is a definitive primality test which in practice can prove primality of hundred digit numbers in a matter of seconds. It is described in [CL].

Great: now we can secretly produce two large primes p and q in order to set up an RSA cryptosystem. Let us assume that $p < q$. The calculation of $m = pq$ takes only $O((\log_2 p)(\log_2 q)) = O((\log_2 q)^2)$ bit operations by (6.2), and then $\varphi = \varphi(m) = (p-1)(q-1) = m - p - q + 1$ is a bargain at $O(\log_2 q)$ by (6.2)! Finally, we want to find numbers e and d such that $ed \equiv 1 (\operatorname{mod} m)$. One method is to randomly choose fairly large integer values of e and calculate the gcds until $\gcd(\varphi, e) = 1 = x\varphi + de$. Since we expect to find a prime close to φ in $\log \varphi$ tries, we should need no more to find a number relatively prime to φ. Combining this with (6.7), finding e and d can be done in $O((\log_2 \varphi)^4)$ bit operations. Another way to find an e relatively prime to φ is just to locate a prime $e > q$, which again is an $O((\log_2 \varphi)^4)$ process.

Therefore we can set up an RSA cryptosystem with modulus m in polynomial time, specifically, with $O((\log_2 m)^4)$ bit operations. But what about using it to communicate? It follows from (11) that messages can be enciphered in time $T(x^e (\operatorname{mod} m)) = O((\log_2 e)(\log_2 m)^2)$. Likewise, the deciphering time is $T(x^d (\operatorname{mod} m)) = O((\log_2 d)(\log_2 m)^2)$. This concludes the "easy" part: We have seen that it is "easy" to set up and use an RSA cryptosystem.

8 Security

Factoring products of large primes is believed to be very difficult. This belief arises from the fact that people have been trying hard to accomplish such factorizations efficiently for thousands of years without much success. The security of the RSA cryptosystem is based on the belief that breaking the cipher is equivalent to factoring the modulus m given in the public key (m, e).

Of course, if one could find a factorization of the modulus $m = pq$ used for an RSA system with public key (m, e), one would be able to find $\varphi = \varphi(m) = m - p - q + 1$ and use the extended Euclidean algorithm to find $\gcd(\varphi, e) = x\varphi + de = 1$ and d, and thereby obtain the secret deciphering key (m,d). Conversely, knowing φ is sufficient to factor m. We assume that $p<q$.

$$\begin{aligned}&A=p+q=m+1-\varphi,\\&A^2=(p+q)^2=p^2+2pq+q^2=p^2+2m+q^2,\\&A^2-4m=p^2-2m+q^2=p^2-2pq+q^2=(p-q)^2,\\&B=q-p=\sqrt{A^2-4m},\\&p=(A-B)/2 \text{ and } q=(A+B)/2.\end{aligned}$$

Hence, knowing φ is equivalent to being able to factor m.

But we still don't know that we can't break the RSA by some method that does not lead to a factorization of the enciphering modulus m. A complete solution to the cipher would mean being able to recover *every* x in $\mathbb{Z}_m$ from its encipherment $y\equiv x^e(\operatorname{mod} m)$. Could this be done without a knowledge of the unique $d\equiv e^{-1}(\operatorname{mod}\varphi)$? The answer to this question is yes, as we can see from our example with enciphering key $(m,e)=(33,7)$. Recall that the deciphering key was $(m,d)=(33,3)$. We will see that $(m,d')=(33,13)$ works just as well! For if $y\equiv x^7(\operatorname{mod} 33)$, then $y^{13}\equiv(x^7)^{13}=x^{91}\equiv x(\operatorname{mod} 33)$ by Corollary 8 because $91\equiv 1(\operatorname{mod} 2)$ and $91\equiv 1(\operatorname{mod} 10)$.

Okay, suppose that one has a number d' such that $x^{ed'}\equiv x(\operatorname{mod} m)$ for every x in $\mathbb{Z}_m$. That means that one has a number $b=ed'-1$ such that $x^b\equiv 1(\operatorname{mod} m)$ for every x in $\mathbb{Z}_m^*$. b must be even because $(-1)^b\equiv 1(\operatorname{mod} m)$. Then it turns out that there are positive integers r and a (See [Ko], p. 94.) such that $b=2^ra$, $x^{2a}\equiv 1(\operatorname{mod} m)$ for every x in $\mathbb{Z}_m^*$, but there is some x in $\mathbb{Z}_m^*$ such that $x^a\not\equiv 1(\operatorname{mod} m)$. In this case, $x^a\not\equiv 1(\operatorname{mod} m)$ for at least 50% of the values of x in $\mathbb{Z}_m^*$, and so choosing random elements of $\mathbb{Z}_m^*$ should lead to such an x fairly quickly. And for such an a, there are at least 50% of the x in $\mathbb{Z}_m^*$ for which x^a-1 is divisible by one of the primes p or q, but not both. Then $\gcd(m,x^a-1)$ is one of the two primes p or q. This gives a factorization of m.

Although we have certainly not shown that breaking the RSA algorithm is equivalent to factoring the modulus m in the public key (m,e), there is certainly a close relationship. And it is clear that the cipher is weak if m is easy to factor. One problem occurs if p is close to q. We can write $m=pq$ $=(t+s)(t-s)=t^2-s^2$ with $q=t+s$, $p=t-s$, so $t=(q+p)/2$ and $s=(q-p)/2$, where we make our usual assumption that $p<q$. If p is close to q, then $s^2=t^2-m$ is a small perfect square. So we attempt to factor m by taking $t=\lfloor\sqrt{m}\rfloor+k$ for small k. Consider an example from [Ko], p.144: Factor $m=200819$. $\lfloor\sqrt{200819}\rfloor=448$, so we try $t=448+k$. For $k=1$, $t^2-m=449^2-200819=782$ is not a perfect square. For $k=2$, $t^2-m=450^2-200819=1681=41^2$, so $s=41$ and $t=450$ gives the factorization $p=t-s=409$ and $q=t+s=491$. This method, called **Fermat factorization**, and generalizations such as the *factor base method* and the *quadratic sieve method* are especially effective when p and q are close together. There are other methods which are effective when $p-1$ and $q-1$ have many small factors.

It behooves the user of an RSA public key cipher to avoid the situations mentioned above, as well as any others which may arise because of new factoring methods. However, no known factorization methods are polynomial time algorithms, so it seems likely that the cipher can stay ahead of the factorizations by choosing larger primes. A number of additional precautions should be taken when implementing an RSA cryptosystem in order to make it secure. But if these precautions are taken, it seems that the system is here to stay! These questions are more fully addressed in an article [Bo] "Twenty Years of Attacks on the RSA Cryprosystem" by Dan Boneh in which he concludes that there have been some insightful attacks, but no devastating attack has been found, and that with proper implementation the system can be trusted to be secure.

References

[AGP] W. Alford, A. Granville, and C. Pomerance, There are infinitely many Carmichael numbers, *Annals of Math., Vol. 139 (1994), 703-722.*

[Bl] W. A. Blankinship, A new version of the Euclidean algorithm, *The American Mathematical Monthly,* September 1963, 742-745.

[Bo] Dan Boneh, Twenty years of attacks on the RSA cryptosystem, *Notices of the American Math. Soc.,* Vol. 46, No. 2 (February 1999), 203-213.

[CL] H. Cohen and H. W. Lenstra, Jr., Primality testing and Jacobi sums, *Math. Comp. Vol. 42 (1984), 297-330.*

[DH] Whitfield Diffie and Martin E. Hellman, New directions in cryptography, *IEEE Transactions on Information Theory*, Vol. IT-22, No. 6, Nov. 1976; or reprinted in *Secure Communications and Assymmetric Cryptosystems,* AAAS Selected Synposium 69, Westview Press, Boulder, CO, 1982.

[G] Martin Gardner, Mathematical Games, *Scientific American,* August 1977.

[H] Martin E. Hellman, The Mathematics of Public-Key Cryptography, *Scientific American*, August 1979, 146-157.

[K] David Kahn, **The Codebreakers**, MacMillan, New York, 1967.

[MW] Richard Maruszewski and William Wardlaw, A note on the Euclidean algorithm, *The AMATYC Journal, Vol. 17 No. 1, fall 1995, 29-32.*

[Ko] Neal Koblitz, **A Course in Number Theory and Cryptography**, 2ed, Springer, New York, 1994.

[RSA] Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman, A method for obtaining digital signatures and public key signatures, *Communications of the ACM*, Vol. 21, No. 2, Feb. 1978; or reprinted in *Secure Communications and Assymmetric Cryptosystems,* AAAS Selected Synposium 69, Westview Press, Boulder, CO, 1982.

Number Theory and Cryptography (using Maple)

John Cosgrave

Department of Mathematics, St. Patrick's College, Drumcondra, Dublin 9, IRELAND

Abstract Since 1995-96 I have taught, using Maple, a yearly course on Number Theory and Cryptography to my undergraduate students[1]. In this paper I outline some basic number theoretical topics related to cryptography, based on my experience as a teacher of those topics. I am omitting all reference to practical teaching details, but will happily forward all teaching materials (notes, examination papers, etc.) to any interested readers. Finally, several of my NT and Cryptography course Maple worksheets[2] are available on the internet [Cos].

1 Introduction

My ideal reader of this paper is

- someone familiar with elementary number theory (essentially congruences, the Euclidean Algorithm, and Fermat's 'little' theorem), who would like to know how certain number theoretic ideas relate to the basic notions of Pohlig-Hellman (private-key) and Rivest-Shamir-Adleman public-key cryptography, or
- someone who knows some number theory, has never taught any cryptography, and who is wondering if it is something he/she might undertake.

Cryptography is the study of secure communication: how can two or more persons communicate securely with each other? The subject has a long and fascinating history, the best detailing of which is undoubtedly David Kahn's monumental The Codebreakers [K]. Also, it is well recognised that the two fundamental development in cryptography took place in the 1970's when W. Diffie and M.E. Hellman [DH] proposed the idea of public-key cryptography, and shortly afterwards R.L. Rivest, A. Shamir and L.M. Adleman [RSA] gave an actual realisation of the Diffie-Hellman proposal, the now classic

[1] Most of my students are training to be primary school teachers - 38 of them in my recent class - and have chosen Mathematics (by university requirement) as one of their 'academic' subjects.

[2] The best of which, if I may say so, entitled 'Bill Clinton, Bertie Ahern, and digital signatures', covers almost all the contents of this paper in an accessible (no theorems) manner.

RSA method. Even after the passage of some twenty years, the brilliance of those path-breaking papers has not diminished, and one can still profit from re-reading them.

The manner in which elementary number theory has made an impact on private and public-key cryptography is well known , and for my purposes may well be summarised as follows:

Given two parties[3] A and B who wish to communicate, A transforms her plaintext T (*Please send more money asap*) into numerical form N (a natural number formed as a result of some agreement, e.g. that 'a' is 1, 'b' is 2, etc.), and then, using some suitable 1-to-1 function f, computes $N' = f(N)$. A then communicates N' to B, who recovers N from N' using the inverse function f^{-1}, and then recovers A's original plaintext.

So far there is no Number Theory in any of this, and the above is merely an abstract mathematical formulation of the classic problem whose various solutions are beautifully and thrillingly described in Kahn's history [K].

1.1 Number Theory makes its entrance

How does Number Theory make its contribution to a solution of the classic problem of communication? It is all so startlingly simple, and may be summarised by saying that N' is formed by modular exponentiation with a specially chosen modulus, as is the recovery of N from N'. The essential idea may be conveyed with a simple, but unrealistic, example (realism merely involves better chosen larger moduli): suppose A wishes to send B the message consisting only of the single letter 'c,' the numerically transformed form of which is '3.' A could encrypt (disguise) '3' by forming $N' = f(3) = 3^7 (\bmod 11)$ giving $N' = 9$ and then send '9' to B, who may then decrypt (recover) by forming $f^{-1}(N') \equiv 9^3 (\bmod 11)$ which produces, for general reasons which will be clear in a moment, the original '3.'

Almost everything that one needs[4] is now easily explained: with the example just given, A could send any message, and have that message recovered by B, whose numerical value was in the range 1 to 10. That is a consequence of Fermat's 'little' theorem for the prime 11, as I now briefly illustrate.

Fermat's 'little' theorem [5] is the following result.

Theorem 1. *Let p prime, and let $a \in \mathbb{Z}$ with $a \not\equiv 0 (\bmod p)$; then $a^{p-1} \equiv 1 \,(\bmod p)$.*

Returning to our example we have, for any a in the range 1 to 10, that $a^{10} \equiv 1 \,(\bmod 11)$, from which, by squaring both sides and multiplying both

[3] 'Alice and Bob.'

[4] For realistic Pohlig-Hellman (private-key) or Rivest-Shamir-Adleman (public-key) cryptography.

[5] One of the most remarkable elementary theorems, with a host of important consequences.

sides by a, we obtain $a^{21} \equiv a \,(\text{mod}\, 11)$. Thus if A used general a in the range 1 to 10, then setting $N' = f(a) = a^7 (\text{mod}\, 11)$ giving $N' = a'$, with a' chosen [6] in the range 1 to 10, and A then sends N' to B, who then decrypts by forming $f^{-1}(N') = (a')^3 \equiv (a^7)^3 \equiv a(\text{mod}\, 11)$, returning the original a.

Theorem 1 underpins the Pohlig-Hellman cryptographic system, in much the same that the following Theorem 2 (what one might call the two prime version of the Euler-Fermat theorem) underpins the Rivest-Shamir-Adleman system. Texts dealing with Theorem 2 normally first prove the full version of the **Euler-Fermat theorem** (*Let n be a natural number, $n > 1$, and let a be any integer with gcd$(a, n) = 1$; then $a^{\phi(n)} = 1(\text{mod}\, n)$*, where $\phi(n)$ - the Euler phi-function - is the number of integers between 1 and $n - 1$ that are relatively prime to n), but such an appeal can be dispensed with, as seen in the following proof.

Theorem 2. *Let p and q be distinct primes, and let $a \in \mathbb{Z}$ with $a \not\equiv 0(\text{mod}\, p)$ and $a \not\equiv 0(\text{mod}\, q)$; then $a^{(p-1)(q-1)} \equiv 1\,(\text{mod}\, pq)$.*

Proof. Since $a \not\equiv 0(\text{mod}\, p)$ then $a^{p-1} \equiv 1(\text{mod}\, p)$, and $(a^{p-1})^{q-1} \equiv 1^{q-1} \equiv 1(\text{mod}\, p)$, and so $a^{(p-1)(q-1)} \equiv 1(\text{mod}\, p)$. Similarly $a^{(p-1)(q-1)} \equiv 1(\text{mod}\, q)$, and thus $a^{(p-1)(q-1)} \equiv 1\,(\text{mod}\, pq)$ since gcd$(p,\ q) = 1$. □

2 Some technical number theory details and cryptographic applications

2.1 Relating the decryption power to the encryption power

Both the PH and RSA methods require computing the decryption power from the encryption power and the modulus, and for that one needs[7] the following result.

Theorem 3. *Let $m \in \mathbb{N}$ with $m > 1$, and let $e \in \mathbb{Z}$ with* gcd$(e,\ m) = 1$; *then there is a unique $d \in \mathbb{Z}$ with $ed \equiv 1(\text{mod}\, m)$ and $1 \leq d \leq m - 1$.*

In applications

- m is $(p - 1)$, for prime p, in the PH system,
- m is $(p - 1)(q - 1)$, for distinct primes p and q, in the RSA system.

[6] This is an *important detail* in general: the modulus must have a value greater than the value of the numerical form of the plaintext. In the above unrealistic example where we used modulus 11, had A wished to send the message ‘r’ - whose numerical value would be 18 - then it would not be clear to B whether A was transmitting the letter ‘r’ or perhaps ‘g,’ whose numerical equivalent would be 7, and $18 \equiv 7(\text{mod}\, 11)$.

[7] Using the extended Euclidean algorithm.

2.2 Cryptographic applications: Pohlig-Hellman (private-key) and Rivest-Shamir-Adleman (public-key)

Fermat's little theorem, and the above two prime version of it then form the basis for the Pohlig-Hellman and Rivest-Shamir-Adleman cryptographic methods:

Theorem 4. *(The Pohlig-Hellman case.) Let*

- *p be prime, and $e \in \mathbb{N}$ with* $\gcd(e,\ p-1) = 1$,
- *$P \in \mathbb{Z}$ with $P \not\equiv 0 (\bmod p)$, and C be defined by $C \equiv P^e (\bmod p)$, and d be chosen so that $d \in \mathbb{Z}$ with $ed \equiv 1 (\bmod\, (p-1))$ and $1 \leq d < p-1$;*

then [8] $C^d \equiv P(\bmod p)$.

Proof. Since $C \equiv P^e (\bmod p)$, then $C^d \equiv (P^e)^d \equiv P^{ed} (\bmod p)$. (We need the $d \in \mathbb{N}$ to guarantee that $C^d \in \mathbb{Z}$.) Now, since $ed \equiv 1 (\bmod\, (p-1))$ then $ed = m(p-1)+1$ for some $m \in \mathbb{Z}$, and, in fact, $m \in \mathbb{N}$ since $e,\ d \in \mathbb{N}$. Thus $C^d \equiv P^{ed} \equiv P^{m(p-1)+1} (\bmod p)$, and by Fermat's 'little' theorem, we have $P^{p-1} \equiv 1 (\bmod p)$. It follows that

$$C^d \equiv P^{m(p-1)+1} \equiv (P^{p-1})^m \times P \equiv 1^m \times P \equiv P(\bmod p),$$

i.e. $C^d \equiv P(\bmod p)$. □

Theorem 5. *(The Rivest-Shamir-Adleman case.) Let*

- *p and q be distinct primes, and $e \in \mathbb{N}$ with* $\gcd(e,\ (p-1)(q-1)) = 1$,
- *$P \in \mathbb{Z}$ with $P \not\equiv 0 (\bmod p)$, $P \not\equiv 0 (\bmod q)$, and C be defined by $C \equiv P^e (\bmod pq)$, and*
- *d be chosen so that $d \in \mathbb{Z}$ with $ed \equiv 1 (\bmod\, (p-1)(q-1))$ and $1 \leq d < (p-1)(q-1)$;*

then [9] $C^d \equiv P(\bmod pq)$.

A proof may be given along the same lines as the earlier one.

[8] This now is the guarantee that when the numerical value of the 'plaintext' P is encrypted using 'e' - thus forming the numerical form of the 'ciphertext' C - and C is then decrypted using 'd', then the upshot of all of this is to recover P, the original plaintext (rather its numerical form, which one then translates back into ordinary text).

[9] Exactly the same comment here as in the previous footnote. The plaintext just gets jumbled up by the decryption power, and gets unscrambled by the decryption power. It's as simple as that!!

Encryption, decryption, and digital signatures First I set down the details for encryption and decryption for both the Pohlig-Hellman private-key, and the Rivest-Shamir-Adleman (RSA) public-key methods. The two methods are similar, but are also quite different: the private one is based on trust between the parties, whereas the private one is based on caution.

In both systems it is understood that the plaintext (the original, text form of the message "Please send me money as quickly as possible") is transformed into numerical form according to some agreed convention ('a' is 1, 'b' is 2, ... , 'z' is 26, 'A' is 27, 'B' is 28, etc. '0' is 53, '1' is 54, '2' is 55, etc.), and then that numerical form is itself transformed in some way. In both PH and RSA, that number (or blocks of numbers) is (are) subjected to modular exponentiation:

- where the modulus is a prime p, in the PH case,
- where the modulus n is the product of two primes in the RSA case.

The Pohlig-Hellman case How do two people, A and B, communicate using the PH private-key cryptographic method?

The details. Having shared their 'private keys', namely

- prime p (the modulus),
- encryption power e, and
- decryption power d,

related by

$$e \in \mathbb{N},\ \gcd(e,\ p-1) = 1, \qquad \text{and} \qquad ed \equiv 1 (\operatorname{mod} p-1),\ d \in \mathbb{N},$$

A and B proceed as follows: A (say) converts the plaintext T into numerical form P (say) by an agreed convention, and breaks that number P into numerical blocks

1. $P_1, P_2, \ldots, P_r$, each having positive numerical value less than p.
2. A then forms the numbers $C_1, C_2, \ldots, C_r$ ('C' for cipher) as follows (this is the **encryption** of the P's):

   $$C_1 \equiv P_1^e (\operatorname{mod} p),\ C_2 \equiv P_2^e (\operatorname{mod} p), \ldots, C_r \equiv P_r^e (\operatorname{mod} p),$$

   where the values of $C_1,\ C_2,\ \ldots\ ,\ C_r$ are chosen so their numerical values are positive and less than p. A then transmits those numerical blocks $C_1,\ C_2,\ \ldots\ ,\ C_r$ to B.

On receipt of those blocks of numbers, B proceeds to decrypt (and so recover the original plaintext) by computing the numbers $c_1,\ c_2,\ \ldots\ ,\ c_r$:

$$c_1 \equiv C_1^d (\operatorname{mod} p),\ c_2 \equiv C_2^d (\operatorname{mod} p),\ \ldots\ ,\ c_r \equiv C_r^d (\operatorname{mod} p),$$

with the values of $c_1,\ c_2,\ \ldots\ ,\ c_r$ chosen so that they are positive and less than p.

Then - and this is now the whole point of all of this - those numbers $c_1,\ c_2,\ \ldots\ ,\ c_r$ are, in fact, the numbers $P_1,\ P_2,\ \ldots\ ,\ P_r$.

The Rivest-Shamir-Adleman case How do two people, A and B, communicate using the RSA public-key cryptographic method?

The details. A (say) having chosen his/her 'keys'namely

- $n = p \times q$ (n is the modulus), for distinct (and, in practice, large [10]) primes p and q,
- encryption power e, and
- decryption power d

related by

$$e \in \mathbb{N},\ \gcd(e,\ (p-1)(q-1)) = 1, \quad \text{and} \quad ed \equiv 1(\operatorname{mod}(p-1)(q-1)),\ d \in \mathbb{N}.$$

A and B proceed as follows: A (say) having made his/her 'public-key', namely $(n,\ e)$, known to B, B would then communicate with A as follows: B would convert the plaintext T into numerical form P (say) by an agreed convention, and would break that number P into numerical blocks:

1. $P_1,\ P_2,\ \ldots\ ,\ P_r$, each having positive numerical value less than n.
2. B then forms the numbers $C_1,\ C_2,\ \ldots\ ,\ C_r$ ('C' for cipher) as follows (this is the encryption of the P's):

   $$C_1 \equiv P_1^e(\operatorname{mod} n),\ C_2 \equiv P_2^e(\operatorname{mod} n),\ \ldots\ ,\ C_r \equiv P_r^e(\operatorname{mod} n),$$

   where the values of $C_1,\ C_2,\ \ldots\ ,\ C_r$ are chosen so they are positive and less than n. B then transmits those numerical blocks $C_1,\ C_2,\ \ldots\ ,\ C_r$ to A.

On receipt of those blocks of numbers, A proceeds to decrypt by computing the numbers $c_1,\ c_2,\ \ldots\ ,\ c_r$:

$$c_1 \equiv C_1^d(\operatorname{mod} n),\ c_2 \equiv C_2^d(\operatorname{mod} n),\ \ldots\ ,\ c_r \equiv C_r^d(\operatorname{mod} n),$$

with the values of $c_1,\ c_2,\ \ldots\ ,\ c_r$ chosen so that they are positive and less than n. Then - and again this is now the whole point of all of this - those numbers $c_1,\ c_2,\ \ldots\ ,\ c_r$ are, in fact, the numbers $P_1,\ P_2,\ \ldots\ ,\ P_r$.

2.3 'Signing' messages using the RSA cryptographic method.

Suppose you received a message from someone; how would you know the message really came from them? For example, suppose you received the following message: *Please call to see me on Wednesday at 3.00 P.M. John Cosgrave.*

It would be almost certain that the message came from me, especially if I signed it, and you know what my signature looks like. However, someone could have forged my signature, and you would be misled into thinking that

[10] But, and again in practice, not just 'large', but one would have to be careful about choosing the p and q so that n could not be easily factored.

I had asked you to visit me. You would turn up at my office on Wednesday at 3.00 P.M., and (possibly) find that I wasn't there ... . Of course it wouldn't really matter; the worst that would have happened is that you would have wasted your time.

Suppose, though, that an army general received a message saying something like: *At 6.00 A.M. tomorrow, send 1,000 troops to Place X ... (Signed by the) Commander-in-Chief.* How can the general be certain that the message really has come from the C.-in-C.?

In earlier times, documents or messages were authenticated by a physical signature or seal (though they could have been forged). In recent times there is increasing reliance on electronic means of communication (by government, diplomatic circles, military, business, banking, political groupings, criminal organisations, private individuals, etc.) which do not allow, of course, for a physical signature. With electronic communication, authentication is guaranteed by a 'digital signature,' and this is how it is done:

Recall the connection between e and d namely $ed \equiv 1(\bmod\,(p-1)(q-1))$ and note that it can be rewritten as $de \equiv 1(\bmod\,(p-1)(q-1))$, where the e and d have simply been interchanged. That simple interchanging has a very, very powerful consequence: it enables a user of RSA to sign a message. This is all they have to do:

To illustrate, let us return to my earlier: "For example, suppose you received a message saying: *Please call to see me on Wednesday at 3.00 P.M. John Cosgrave.*" This is what I can do (assuming I am a user of RSA, and you know my public-key (n, e)) that will convince you that the note you receive from me, really is from me:

I can encrypt a message to you by:

- using my private decryption power d (which only I know) as my encryption power.

Then, on receipt of my message, you can decrypt it by:

- using my public encryption power d (which you, and possibly others, know) as your decryption power.

Anticipating an objection. You might (rightly) say that anyone who can intercept my message, and who knows my public-key, can also decrypt my message to you. That is a simple fact (which is best illustrated in a Maple worksheet).

Fortunately public-key cryptography once again comes to our rescue. If I want to 'sign' my message to you and I don't want anyone but you to be able to read the contents of my message to you, I can then achieve my aim by performing a double encryption [11] :

[11] Assuming you are using RSA, and I know your public-key.

- First [12] I use my private-key to do an initial encryption (and in the process 'sign' my message to you),
- then I use your public-key to perform a second encryption before sending my message.

When you receive my doubly-encrypted message you can then read it by performing a double decryption:

- First you use your private-key to perform the first decryption,
- then you use my public-key to perform the second decryption, and so read my message.

Remark 6. A way of visualising this. Think of public-key cryptography in terms of paints and paint-removers. My public-key is some paint which I have made, and which, if it is used, only I can remove by applying the secret paint remover which I also have made, and which only I have access to.

You have to allow your imagination to let the paint and paint remover to be used in reverse!! By that I mean that if something is covered with my paint then not only can it be uncovered by applying my paint remover, but that the same is true in reverse: if something is first covered with my paint remover then what is now there can be uncovered by applying my paint!!

Anyone who wishes to send me a secret message simply writes a message, gets some of my paint, and paints (encryption) over my message. When I receive your message I apply my paint remover (decryption) to it, and so read your message. Everything I have said about 'I' applies to you: you have your paint and your paint remover ... .

Now form a mental image of what I have described above:

- First I use my secret paint remover to do an initial painting (encryption, and in the process 'sign' my message to you),

- then I use your public paint to perform a second encryption before sending my message.

When you receive my doubly-encrypted message you can then read it by performing a double decryption:

- First you use your private paint remover to perform the first decryption,
- then you use my public paint to perform the second decryption, and so read my message.

[12] Actually which I do first depends on whether my public modulus is smaller than your public modulus:
- If my public modulus is smaller than yours then what I have described above is in fact what I would (and should) do, but:
- If my public modulus is greater than yours then what I have described above should be done in reverse order, by me and you.

Basic understanding: We assume that A has public key $(n_A,\ e_A)$ with private key d_A and that B has public key $(n_B,\ e_B)$ with private key d_B.

Question: How can B 'sign' a message to A (equally A send one to B) so that A can have confidence that the message received has come from B?

Answer: It can be done quite easily, but it depends on which is the smaller: n_A or n_B. That is, it depends on whether:

(i) $n_B < n_A$, or
(ii) $n_A < n_B$.

The details: Let us suppose that (i) happens [13] (namely that $n_B < n_A$), and suppose that B wants to (securely) send and 'sign' a message P to A.

We make the usual understanding that P (the numerical form of the plaintext) has been put into numerical form according to some convention (a is 1, b is 2, etc.).

This, then, is what B does to (securely) send and 'sign' a message P to A.

1. B breaks P up into blocks of numbers. $P_1,\ P_2,\ \ldots\ ,\ P_r$, each having numerical value less than n_B (and so are *automatically* less than n_A since $n_B < n_A$), and each relatively prime to both n_B and n_A. B then 'signs' using his/her private key by doing this:
2. B forms the numbers $c_1,\ c_2,\ \ldots\ ,\ c_r$ as follows:

$$c_1 \equiv P_1^{d_B} (\operatorname{mod} n_B),\ c_2 \equiv P_2^{d_B} (\operatorname{mod} n_B), \ldots , c_r \equiv P_r^{d_B} (\operatorname{mod} n_B),$$

   the $c_1,\ c_2,\ \ldots\ ,\ c_r$ chosen with positive values, less than n_B.
3. B then forms the following blocks of ciphertext, and sends those to A:

$$C_1 \equiv c_1^{e_A} (\operatorname{mod} n_A), C_2 \equiv c_2^{e_A} (\operatorname{mod} n_A), \ldots , C_r \equiv c_r^{e_A} (\operatorname{mod} n_A),$$

   the $C_1,\ C_2,\ \ldots\ ,\ C_r$ chosen with positive values, less than n_A.

In summary,

1. B first signs with their own private key,
2. and then sends the newly formed ciphertexts in the usual way, using A's public key.

On receipt of the $C_1,\ C_2,\ \ldots\ ,\ C_r$ this is what A does:

1. A partially decrypts the numbers $C_1,\ C_2,\ \ldots\ ,\ C_r$ using their own private key, by forming the numbers $x_1,\ x_2,\ \ldots\ ,\ x_r$ as follows:

$$x_1 \equiv C_1^{d_A} (\operatorname{mod} n_A), x_2 \equiv C_2^{d_A} (\operatorname{mod} n_A), \ldots , x_r \equiv C_r^{d_A} (\operatorname{mod} n_A),$$

   the $x_1,\ x_2,\ \ldots\ ,\ x_r$ chosen with positive values, less than n_A. (Those $x_1,\ x_2,\ \ldots\ ,\ x_r$ are, of course, none other than $c_1,\ c_2,\ \ldots\ ,\ c_r$.)

[13] Later we will see what to do if (ii) happens.

2. A completes the decoding by forming the following blocks of ciphertext:

$$y_1 \equiv x_1^{e_B} \pmod{n_B}, y_2 \equiv x_2^{e_B} \pmod{n_B}, \dots, y_r \equiv x_r^{e_B} \pmod{n_B},$$

the $y_1, y_2, \dots, y_r$ chosen with positive values, less than n_B.

The whole point is now that those $y_1, y_2, \dots, y_r$ are none other than the numerical form of B's original plaintext message, namely $P_1, P_2, \dots, P_r$.

If, however we had $n_A < n_B$, then this is what B would do [14] : suppose that B wants to send message P to A. B breaks P up into blocks of numbers $P_1, P_2, \dots, P_r$ (each having value less than n_A (and so are *automatically* less than n_B since $n_A < n_B$), and each relatively prime to both n_A and n_B). This, then, is what B does to 'sign' the message to A:

First B forms the numbers $c_1, c_2, \dots, c_r$ by using A's public key, just as in an ordinary unsigned message, as follows:

$$c_1 \equiv P_1^{e_A} \pmod{n_A}, c_2 \equiv P_2^{e_A} \pmod{n_A}, \dots, c_r \equiv P_r^{e_A} \pmod{n_A},$$

the $c_1, c_2, \dots, c_r$ chosen with positive values, less than n_A.

Then B (and this is what 'signs' for B, the using of B's 'secret') sends to A the following blocks of ciphertext:

$$C_1 \equiv c_1^{d_B} \pmod{n_B}, C_2 \equiv c_2^{d_B} \pmod{n_B}, \dots, C_r \equiv c_r^{d_B} \pmod{n_B},$$

the $C_1, C_2, \dots, C_r$ chosen with positive values, less than n_B. In short, B first encodes in the usual way using A's public key, and then 'signs' using their own private key.

On receipt of $C_1, C_2, \dots, C_r$ this is what A does:

First A partially decodes the numbers $C_1, C_2, \dots, C_r$ using B's public key, by forming the numbers $x_1, x_2, \dots, x_r$ as follows:

$$x_1 \equiv C_1^{e_B} \pmod{n_B}, x_2 \equiv C_2^{e_B} \pmod{n_B}, \dots, x_r \equiv C_r^{e_B} \pmod{n_B},$$

the $x_1, x_2, \dots, x_r$ chosen with positive values, less than n_B. (Those $x_1, x_2, \dots, x_r$ are, of course, none other than $c_1, c_2, \dots, c_r$.

Then A completes the decoding by forming the following blocks of ciphertext:

$$y_1 \equiv x_1^{d_A} \pmod{n_A}, y_2 \equiv x_2^{d_A} \pmod{n_A}, \dots, y_r \equiv x_r^{d_A} \pmod{n_A},$$

the $y_1, y_2, \dots, y_r$ chosen with positive values, less than n_A. The whole point is, again, that those $y_1, y_2, \dots, y_r$ are none other than the numerical form of B's original plaintext message, namely $P_1, P_2, \dots, P_r$.

All of this is best illustrated in a Maple worksheet, and such details, actually carried out, may be seen in my 'Clinton ... ' Maple public lecture [Cos].

[14] Actually all that B and A do is to do what they previously did, except to do it in *reverse order.*

3 Some elementary, but non-trivial primality testing methods

Is the converse of Fermat's little theorem true? That is, if $n \in \mathbb{N}(n \geq 2)$, and $a \in \mathbb{Z}$ such that $a^{n-1} \equiv 1(\operatorname{mod} n)$, is n necessarily prime? It is well known that it isn't, as the example [15] $2^{340} \equiv 1(\operatorname{mod} 341)$ shows [16].

Lucas (starting in 1876) observed the first of a series of partial converses to Fermat's 'little' theorem. These results had the following common form: if $n \in \mathbb{N}(n \geq 2)$, and $a \in \mathbb{Z}$ such that $a^{n-1} \equiv 1(\operatorname{mod} n)$, then - *providing some extra condition is satisfied* - n is prime.

The start of a serious study of primality testing I will restrict myself to the methods of Lucas (1876-78), Proth (1878), Pocklington (1914), Lehmer (1927) and Selfridge (1967), and I begin with the following result.

Theorem 7. *(Lucas, 1876). Let $n \in \mathbb{N}(n \geq 3)$, and suppose there is an $a \in \mathbb{Z}$ such that $a^{n-1} \equiv 1(\operatorname{mod} n)$ and $a^x \not\equiv 1(\operatorname{mod} n)$ for all x with $1 \leq x < n-1$; then n is prime.*

Alternative wording of this theorem. *Let $n \in \mathbb{N}(n \geq 3)$, and suppose there is some $a \in \mathbb{Z}$ such that $ord_n a = n-1$; then n is prime.*

This theorem which appears at first sight to be so weak (but which ultimately isn't, in the sense that it can be gradually improved bit by bit to produce wonderfully effective results) marked the start of modern primality testing. As a test it is even worse than the Eratosthenes method, but Lucas himself improved upon it in 1878 by showing that the condition " $a^x \not\equiv 1(\operatorname{mod} n)$ for all x with $1 \leq x < (n-1)$ " could be replaced with the less restrictive one that "$a^x \not\equiv 1(\operatorname{mod} n)$ for all x *with* $1 \leq x < (n-1)$ with $x\,|n-1$." However, even that improvement ceases to be useful whenever $n-1$ has a lot of factors.

Example 8. A Maple computation which conveys the idea of a proof of Lucas' 1876 theorem. Here I use Maple to compute all powers of 2 modulo 101 from the 1st to the 100th power:

```
>seq(2&^x mod 101, x=1..100);  # here 'a' is 2, and 'n' is 101

    2, 4, 8, 16, 32, 64, 27, 54, 7, 14, 28, 56, 11, 22, 44, 88, 75, 49,
  98, 95, 89, 77, 53, 5, 10, 20, 40, 80, 59, 17, 34, 68, 35, 70, 39, 78,
  55, 9, 18, 36, 72, 43, 86, 71, 41, 82, 63, 25, 50, 100, 99, 97, 93, 85,
  69, 37, 74, 47, 94, 87, 73, 45, 90, 79, 57, 13, 26, 52, 3, 6, 12, 24,
  48, 96, 91, 81, 61, 21, 42, 84, 67, 33, 66, 31, 62, 23, 46, 92, 83, 65,
  29, 58, 15, 30, 60, 19, 38, 76, 51, 1
```

[15] First noted by Sarrus in 1814.

[16] 341 is an example of a **pseudoprime** to the base 2; that is it is a composite n satisfying $2^{n-1} \equiv 1(\operatorname{mod} n)$.

Noting that $2^{100} \equiv 1(\bmod 101)$ and $2^x \not\equiv 1(\bmod 101)$ for all x with $1 \le x < 100$, one should make a critical observation, namely: there are 100 outputs, and *no two of those outputs are equal*, and that, as a *consequence*, all residues between 1 and 100 must occur (exactly once), and that as a further *consequence* 101 must be prime. Why? Well, if 101 were composite then it would have as a factor some prime smaller than 101. Let's suppose it had 7 (say) as a factor, then $2^x \equiv 7(\bmod 101)$ for some x, would imply 2 is divisible by 7.

Several similar examples now reduce the proof of Lucas' theorem to a formality.

Proof. (of Lucas' 1876 theorem) [17] Suppose n is composite. We will show that is impossible, and so n must be prime. We show that $a^1,\ a^2, \dots ,\ a^{n-1}$ are congruent mod n, in some order, to $1,\ 2, \dots ,\ n-1$, and then argue that is impossible.

None of $a^1,\ a^2, \dots ,\ a^{n-1}$ is 0 mod n, because if $a^m \equiv 0(\bmod n)$ for some $m \in \mathbb{N}$, then $a^m = nX$, for some $X \in \mathbb{Z}$. Now, let p be a prime with $p|n$; we would have $p|a^m$, and so would have $p|a$. But $p|n$ and $p|a$ would conflict with $\gcd(a,\ n) = 1$, and so none of $a^1,\ a^2, \dots ,\ a^{n-1}$ is 0 mod n.

Also, if $a^v \equiv a^u(\bmod n)$ for some $u,\ v,\ 1 \le u < v \le n-1$, then $a^u(a^{v-u} - 1) \equiv 0(\bmod n)$, and from $\gcd(a^u,\ n) = 1$, it follows that $(a^{v-u} - 1) \equiv 0(\bmod n)$. Setting $x = v - u$, we have $a^x \equiv 1(\bmod n)$, where $1 \le x \le (n-2)$. That conflicts with the second condition of Lucas' theorem, and it follows that no two of $a,\ a^2, \dots ,\ a^{n-1}$ are congruent to each other mod n.

Thus $a,\ a^2, \dots ,\ a^{n-1}$ are congruent mod n, in some order, to $1,\ 2, \dots ,\ n-1$, and so for some integer r $(1 \le r \le n-2)$ we have $a^r \equiv p(\bmod n)$, where p is the earlier prime dividing n (and so $1 < p < n$). That is impossible since $a^r \equiv p(\bmod n)$ means $a^r \equiv nX' + p$, for some $X' \in \mathbb{Z}$, from which, with $p|n$ we obtain $p|a^r$. We would have $p|a$. That conflicts with $\gcd(a,\ n) = 1$, and so n cannot be composite. Thus n is prime. □

Theorem 9. *(what I call the 'Lucas-Kraitchik-Lehmer* [18] *Theorem'). Let* $n \in \mathbb{N}$ *with* $n > 1$, *and suppose there is some* $a \in \mathbb{Z}$ *with* $a^{n-1} \equiv 1(\bmod n)$ *and* $a^{\frac{n-1}{p}} \not\equiv 1(\bmod n)$ *for all primes* p *with* $p|(n-1)$*; then* n *is prime.*

[17] The original Lucas proof involved using a theorem (the famous 'Euler-Fermat theorem') whose own proof involves quite a lot of extra work. Here I give a proof which avoids such a reference.

[18] It is, of course, Lehmer's theorem of 1927. In my first couple of years of teaching a proof of this I used Lehmer's original proof, but, as anyone familiar with that proof will know, a very heavy and quite unnecessary use is made of the Euler ϕ-function, and my students had great difficulty in following it. Fortunately the proof I subsequently gave, here in this paper, was more readily understood (when properly motivated).

Remark 10. This is a very powerful theorem, whose power is only properly realised when Maple, or other similar work is performed with very large numbers.

Proof. (the strategy of the proof is to show that $ord_n a = n - 1$, and it follows from Lucas' theorem of 1876 that n is prime.) Let $r = ord_n a$, then $n - 1 = rR$, some $R \in \mathbb{N}$. If $R > 1$, then $R = pR'$, some prime p, and $R' \in \mathbb{N}$. Thus $n - 1 = rR = rpR'$, $\frac{n-1}{p} = rR' \in \mathbb{N}$, so p divides $(n - 1)$. Then $(a^r)^{R'} \equiv 1^{R'} \equiv 1 (\operatorname{mod} n)$, and thus $a^{\frac{n-1}{p}} = a^{rR'} \equiv 1 (\operatorname{mod} n)$ which conflicts with the second condition of the theorem. Thus $R \not> 1$, and $r = ord_n a = n-1$. By Lucas' 1876 theorem it follows that n is prime. □

Remark 11. The Lehmer 1927 theorem is sometimes referred to, for obvious reasons, as the '$\frac{n-1}{p}$' theorem. There is a further important improvement (dating from 1967) of D. H. Lehmer's theorem that is due to another U.S. mathematician John Selfridge. One only appreciates the value of Selfridge's improvement after one has had experience with using the Lehmer theorem with Maple computations. Selfridge's theorem is sometimes referred to as the 'change of base' theorem, for reasons which will become apparent when one uses it.

Theorem 12. *(what I call the 'Lucas-(Kraitchik)-Lehmer-Selfridge Theorem') Let $n \in \mathbb{N}$ with $n > 1$, and suppose that for each prime p_i with $p_i | (n-1)$ there is some $a_i \in \mathbb{Z}$ with $a_i^{n-1} \equiv 1 (\operatorname{mod} n)$ and $a_i^{\frac{n-1}{p_i}} \not\equiv 1 (\operatorname{mod} n)$; then n is prime.*

Another important variation is Pocklington's theorem.

Theorem 13. *(Pocklington, 1914) Let $n - 1 = UF = Up_1^{\alpha_1} p_2^{\alpha_2} \ldots p_r^{\alpha_r}$ be an incomplete factorisation of $n-1$ (where U is the 'unfactored part' of $n-1$, and $F = p_1^{\alpha_1} p_2^{\alpha_2} \ldots p_r^{\alpha_r}$ is its factored part) with $U < F$ and $\gcd(U, F) = 1$. Suppose there is an a such that $a^{n-1} \equiv 1 (\operatorname{mod} n)$ and $\gcd(a^{\frac{n-1}{p_i}} - 1,\ n) = 1$ for all i, $1 \leq i \leq r$; then n is prime.*

Example 14. Let p_r denote the *rth* prime.

- I have used LKLS to prove the primality of the 1006-digit $2^{50}(p_1 p_2 \ldots p_{20})^3 + 1$, and
- the primality of the 1405-digit $2^{37} 1!2!3!4! \ldots 48!49!50! + 1$.
- Also I have used Pocklington to establish the primality of the (serendipitously found) 2000-digit $p_1 p_2 \ldots p_{325} p_{326}^{325} + 1$ (see [Cos] for the Maple worksheet details), and also
- the primality of the 3318-digit $p_1 p_2 \ldots p_{346} p_{347}^{346} p_{348}^{346} + 1$.

Another interesting elementary result is Proth's (1878), the standard version of which is the following result.

Theorem 15. *Let $N = s \cdot 2^r + 1$, where s, $r \in \mathbb{N}$ and*[19] $s < 2^r$. *Suppose there is an $a \in \mathbb{Z}$ such that $a^{\frac{N-1}{2}} \equiv -1 (\bmod N)$; then N is prime.*

I have made the minor improvement of the $s < 2^r$ condition to $s \leq 2^r + 1$, with this proof.

Proof. First, note the standard result [20] about prime divisors of Fermat type numbers: let $x \in \mathbb{Z}$ and $m \in \mathbb{N}$, then every odd prime divisor q of $x^{2^m} + 1$ satisfies $q \equiv 1 (\bmod 2^{m+1})$. For p a prime divisor of N, we have $a^{\frac{N-1}{2}} \equiv (a^s)^{2^{r-1}} \equiv -1$ and so $p \equiv 1 (\bmod 2^r)$. If N is composite, then N is a product of at least two primes each of which has minimum value $2^r + 1$, and so $N = s \times 2^r + 1 \geq (2^r + 1)(2^r + 1) = 2^r \times 2^r + 2 \times 2^r + 1$. It follows that $s \geq 2^r + 2$, which is incompatible with $s \leq 2^r + 1$. Thus N is prime. □

4 Some elementary, but non-trivial factorisation methods

Maple has a number of factorisation commands, the default one of which [21] is the 1975 continued fraction method of Morrison and Brillhart. It also has the command `ifactor(n, pollard)` which puts into effect the Pollard $\rho-$method with the Floyd cycle algorithm improvement, but only using iterates of '2' using the function $x^2 + 1$.

I am keen that my students should have exposure to some non-trivial factorisation methods, and have narrowed myself down to just two [22]:

- Pollard's $p - 1$ method (1974), which uses Fermat's 'little' theorem,
- Pollard's $\rho-$method (1974), which uses a generalisation of the *birthday paradox.*

It is my experience that students are really fascinated by both Pollard methods, and I can assure any reader that the inclusion of these methods in such a course is a source of very great excitement in the classroom. Personally I never really appreciated these methods until I decided to teach them, and they form one of the highlights of the course. Many students are greatly

[19] Some texts add an entirely irrelevant requirement that s be *odd.*

[20] Which I prove for my students, using standard order theorems and Fermat's little theorem.

[21] The Maple command is ifactor(n).

[22] With as much details as possible. I also expose them to the elementary Fermat method, which, although it only requires high school mathematics to understand it, is nevertheless one which can't be ignored in choosing two primes for RSA usage. Many students are greatly impressed by seeing the product of two really large primes - with hundreds of digits, but which differ by only several thousands - being factored almost instantly by the Fermat method. It allows one to drive home the point that mere size is not enough when choosing primes for RSA usage.

impressed with how effective both methods are, especially the $p-1$ method when used on RSA type numbers where one of the primes has been formed by using a Lehmer-Selfridge type construction. I refer my reader to the example in my Maple public lecture 'Bill Clinton, ... ' [Cos].

My approach to teaching the Pollard methods In teaching my students the Pollard methods I abandon all reticence, and try to impress on them that in studying these methods they are considering the work of a master mathematician with an extraordinary, fertile imagination. The first point I make is that these two methods, which appear so different at first, are in fact driven by a single, apparently useless, but actually incredibly powerful, common idea. Given a composite n, known to be composite because of failing a Lucas-Fermat test to some base [23] the common idea in the methods is to attempt to find some integer M (I urge my students that they think of 'M' as being short for Magic, because in the two Pollard methods it really is magical the manner in which he creates this M) such that $\gcd(M,\ n) > 1$ and $\gcd(M,\ n) < n$.

Since Pollard's approach can - and indeed does - appear very, very strange to weaker students, then I play on that perception, and indeed I attempt to rubbish the idea before I even show them how very powerful it is. While pointing out that finding such an M would of course mean that one had found a proper factor of n, I do concede that the idea could appear completely useless because of these considerations:

- How is one going to find such an M? ...
- By trial and error? Let's see if $M = 2,\ 3,\ 5,\ 7, \ldots$ would do? Why, that would be even worse (because of the gcd computation) than using the Eratosthenes approach of trying possible factors up to $\sqrt{n}$.

However I then put it to them that they should consider it a tribute to Pollard's fertile imagination that he was able to conceive two wonderful realisations [24] of finding this elusive M. These methods will be known to my reader, and so I will only briefly describe how I attempt to convey Pollard's thinking to my students. For both methods I emphasise that Pollard's hope is to find one of the proper factors p of n (and not necessarily the least one!).

In the case of his 1974 method, his $(p-1)$ method, he attempts to find that 'p' by exploiting Fermat's little theorem in the following way.

Lemma 16. *For p any prime and $a \in \mathbb{Z}$, $a \not\equiv 0 (\mathrm{mod}\, p)$, one has $a^{k!} \equiv 1 (\mathrm{mod}\, p)$, and thus $a^{k!} - 1 \equiv 0 (\mathrm{mod}\, p)$, for all sufficiently large values of k.*

[23] Normally (but not always) that $2^{n-1} \not\equiv 1\, (\mathrm{mod}\, n)$.

[24] It is a well known joke amongst mathematicians that a trick is something that works once, while an idea is something that works twice (or more). I have often wondered if Pollard had an idea in 1974, or simply a trick.

(That ‘$(a^{k!}-1)$’, hopefully with a not too large value for k, is going to be ones ‘M’: it is divisible by p, as is n, and ones ambition is to quickly find a reasonably small k, so that the gcd of n and $(a^{k!}-1)$ comes to be greater than 1, but less than n.) That result is, of course, a trivial consequence of Fermat's little theorem, since it is certainly true for $k \geq p-1$.

Proof. From Fermat's ‘little’ theorem we have $a^{p-1} \equiv 1(\bmod p)$. Also, for sufficiently large k we have $(p-1)|k!$ (e.g., $k \geq p-1$ would do trivially), and so $k! = (p-1)K$, for some $K \in \mathbb{N}$. Then $(a^{p-1})^K \equiv 1^K \equiv 1(\bmod p)$, and so $a^{(p-1)K} \equiv 1(\bmod p)$, *i.e.* $a^{k!} \equiv 1(\bmod p)$. □

Pollard's insight was that although the latter congruence is trivial, nevertheless because of the way in which the prime factorisation structure of $k!$ behaves as k increases in size, one may have $a^{k!} \equiv 1(\bmod p)$ for substantially smaller values of k than the trivial $k = p-1$.

Example 17. For example, if $p = 97$, then the trivial $a^{96!} \equiv 1(\bmod 97)$ may be vastly improved upon with $a^{8!} \equiv 1(\bmod p)$. That is, because $97-1 = 96 = 2^5 \times 3^1$, and $8! = 1\times2\times3\times4\times5\times6\times7\times8 = 2^1\times3^1\times2^2\times5\times(2^1\times3^1)\times7\times2^3 = 2^6 \times 3^2 \times 5 \times 7$, is divisible by 96, because of the appropriate accumulation of 2's and 3's in the prime decomposition of 8!.

Thus if one had a composite n (reasonably big, say) which, unknown to one, happened to have 97 as a factor, then that fact would be quickly revealed by successively calculating the early terms of the sequence

$$\gcd(2^{1!}-1,\ n), \gcd(2^{2!}-1,\ n), \gcd(2^{3!}-1,\ n), \ldots .$$

The real work that has to be done to get the idea across may be seen in much greater detail in my related Maple worksheet [Cos]. Suffice it here to say that the key Maple programming computational steps are not to compute each $\gcd(a^{k!}-1,\ n)$ from scratch, but rather to do them recursively, and furthermore not to compute actual values of the $a^{k!}-1$, but rather their reduced values $\bmod n$.

This, then, is the final [25] version of the Maple procedure I lead my students to the following algorithm.

```
> Pollard := proc(seed, n)
  local a, k; a[1] := seed:
  for k from 2 while igcd(n, a[k-1]-1 mod n)=1
  do a[k] := a[k-1]&^k mod n od;
  if igcd(n, a[k-1]-1 mod n) < n then
  lprint(‘After‘, k-1, ‘steps we find that‘,
```

[25] With my students I deliberately build up through slower stages to make certain points, as will be seen by anyone who reads my detailed Maple worksheet. I could, of course, dispense with the ‘`lprint`’ line, and simply output a proper factor if one is found, otherwise have no output.

```
  igcd(n, a[k-1]-1 mod n),‘is a proper factor of‘, n)
  else lprint(‘No proper factor found; try some other seed‘
  fi end:
```

It is my experience that most students are greatly impressed with the effectiveness of this Pollard inspired, Maple procedure. Cryptographic examples of it in action may be read in details given in my Maple public lecture ‘Bill Clinton, Bertie Ahern, and digital signatures’ [Cos].

For example, if one performed the following Maple computations: first, create two large primes p and q, the first of which entails $p-1$ having only small prime divisors, and then formed their product n, one could verify that n is composite by showing it fails a base 2 Lucas-Fermat test, and also factor n using the above Pollard procedure.

These computations are all quickly executed:

```
>p := 2^40*3^52*7^52  1;+ #an 81-digit prime
```

p := 626041542350318657267231514757451147212314337872408353488069113603068870541705217

```
>q := nextprime(10^66  12345678910987654321) +# 67 digits
```

q := 1000000000000000000000000000000000000000000000012345678910987654447

```
>n := p*q: # value suppressed. 147 digits
>2&^(n-1) mod n; # shows ‘n’ is not prime:
```

315212795823765344257786197018813662048038750896292204273838589012443276119009530179154090994865246547633014859071848668063492151077465700747069971.

Finally, execute the following Maple command, which takes only seconds:

```
>Pollard(2, n); # choosing ‘seed’ to be ‘2.’
```

Output:

After 322 steps we find that
626041542350318657267231514757451147212314337872408353488069113603068870541705217
is a proper factor of
6260415423503186572672315147574511472123143378801372613548656271951814799412506854664231171652914999895270244213542408878176695357916559265331449999.

4.1 Some brief comments on the Pollard ρ−method

Suffice it to say that Pollard suggested another remarkable way of arriving at an M’ with the desirable properties listed earlier, except that M is now not arrived at as a consequence of constructing a sequence one of whose eventual terms is the desired M, but rather - as a consequence of the ‘generalised birthday paradox’ - so that M is arrived at by forming differences. This method is very well explained in Pomerance’s MAA notes [P1], or in Koblitz’s book [Ko], and I refer my reader to those sources. In treating this method with my own students I explain the original Pollard approach, eventually arrive at the classic ρ−figure, and discuss how the computation may be speeded up by using the Floyd cycle finding algorithm.

Any reader already familiar with Pollard’s ρ−method will know that Pollard himself suggested starting with seed ‘2’ and using iterated values of $x^2 + 1 \pmod n$ as the means of producing the random sequence. This approach, together with the Floyd cycle improvement is the one that Maple has built into its factorisation command ‘`ifactor(n,pollard).`’ The modification which I make for my own students is to allow for variable seed and iteration function, using the Floyd cycle finding method. Once again it is my experience that students are really impressed with how effective the method is.

I finish by giving a Maple procedure [26], the final version of the one I teach to my students, which incorporates the general form of Pollard-Floyd, and give two examples of the sort of output one will see on using it.

```
>PF := proc(n, f, seed) # general Pollard-Floyd
  local a, k; a[1] := seed: a[2] := f(a[1]):
  for k from 2 while igcd(n, a[2*k-2]-a[k-1])=1 do
  a[k] := f(a[k-1]) mod n;
  a[2*k] := f(f(a[2*k-2]) mod n) mod n; od:
  if igcd(a[2*k-2] - a[k-1], n) $<$ n then
  lprint(igcd(a[2*k-2]-a[k-1], n),
        ‘is a proper factor of‘, n);
  else lprint(‘Try some other seed or function.‘)
  fi; end:

>PF(1037, x-> x^2 + 1, 2);
```

```
      17 is a proper factor of 1037
```

```
>PF(2^32+1, x -> x^2 + 1, 2); # the 5th Fermat number:
```

```
      641 is a proper factor of 4294967297
```

[26] An early Maple worksheet of mine on this topic may be found on the internet [Cos].

References

[BS] Bach, E. and Shallit, J.: **Algorithmic Number Theory. Volume 1**. The MIT Press. (1996)

[B] Bressoud, D. M.: **Factorization and primality testing**. Springer-Verlag. (1989)

[BLSTW] Brillhart, J., Lehmer, D. H., Selfridge, J. L., Tuckerman, B. and Wagstaff, Jr., S. S.: Factorizations of $b^n \pm 1$ ($b = 2, 3, 5, 6, 7, 10, 11, 12$ up to high powers), AMS (Contemporary Mathematics Series), Vol. 22, 2nd edition, 1988.

[C] Cohen, H.: **A Course in Computational Algebraic Number Theory**. Springer-Verlag. (1993)

[Cos] Cosgrave, J. B.: Several of my Maple worksheets relating to my NT and Cryptography course, including the substantial [27] public lecture of 25th November 1998, Bill Clinton, Bertie Ahern [28], and digital signatures, are accessible from David Joyner's USNA Web site at this address:
`http://web.usna.navy.mil/~wdj/crypto.htm`
At the time of preparing this paper my own web site
`http://www.spd.dcu.ie/johnbcos`
is under construction, and when that is completed I will be putting up a considerable number of my Maple worksheets on that site. Alternatively, please contact me at my College using `John.Cosgrave@spd.ie`, or at home `johnbcos@iol.ie`.

[DH] Diffie, W. and Hellman, M.E.: New Directions in Cryptography. IEEE Transactions on Information Theory, v. IT-22, n. 6, (Nov 1976) 109-112

[K] Kahn, D.: **The Codebreakers (The Comprehensive History of Secret Communication from Ancient Times to the Internet)** (1996) Scribner

[Ko] Koblitz, N.: **A Course in Number Theory and Cryptography**. Springer-Verlag. (1994)

[LT] Lenstra, H. W. and Tijdeman, R. (Editors): **Computational Methods in Number Theory**. Math. Centre Tracts 154 Mathematisch Centrum. Amsterdam. (1982)

[Pol1] Pollard, J. M.: Theorems on Factorization and Primality Testing. Proc. Camb. Phil. Soc. 76 (1974) 521-528

[Pol2] Pollard, J. M.: A Monte Carlo Method for Factorization. BIT. 15 (1975) No. 3. 331-335.

[P1] Pomerance, C.: **Cryptology and Computational Number Theory**. Mathematical Association of America. MAA Notes. 4 (1984)

[P2] Pomerance, C. (Editor): **Cryptology and Computational Number Theory**. American Mathematical Society. Proceedings of Symposia in Applied Mathematics. 42 (1990)

[P3] Pomerance, C.: A Tale of Two Sieves. Notices of the American Mathematical Society. 43 No. 12. (1996) 1473-1485

[Ri] Riesel, H.: **Prime Numbers and Computer Methods for Factorization**. Birkhäuser. (1994)

[27] Over forty pages of hard copy.

[28] The Irish Prime Minister who, in September 1998, participated in an 'historic' digital signing in Ireland of a treaty on e-commerce with the US President, using software developed by the Irish company Baltimore (see their Web site for details of that signing at `http://www.baltimore.ie`)

[RSA] Rivest, R.L., Shamir, A., and Adleman, L.: A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. Communications of the ACM. 21(2), (1978) 120-126

[R] Rosen, K. H.: **Elementary Number Theory and Its Applications**. Addison-Wesley. (1988)

email: John.Cosgrave@spd.ie, johnbcos@iol.ie

A Talk on Quantum Cryptography or How Alice Outwits Eve

Samuel J. Lomonaco, Jr. *

University of Maryland Baltimore County
Baltimore, MD 21250

Abstract Alice and Bob wish to communicate without the archvillainess Eve eavesdropping on their conversation. Alice, decides to take two college courses, one in cryptography, the other in quantum mechanics. During the courses, she discovers she can use what she has just learned to devise a cryptographic communication system that automatically detects whether or not Eve is up to her villainous eavesdropping. Some of the topics discussed are Heisenberg's Uncertainty Principle, the Vernan cipher, the BB84 and B92 cryptographic protocols. The talk ends with a discussion of some of Eve's possible eavesdropping strategies, opaque eavesdropping, translucent eavesdropping, and translucent eavesdropping with entanglement.

SHORT ABSTRACT. This is a story about how Alice ingeniously devises two different quantum cryptographic communication protocols (i.e., BB84 and B92) that prevent archvillainess Eve from eavesdropping on Alice's communications with Bob. How does Alice do this? Also, how does she implement her ideas in optics?

This talk is based on the paper: Lomonaco, Samuel J., A Quick Glance at Quantum Cryptography, Cryptologia, Vol. 23, No.1, January, 1999, pp1-41. (Quant-ph/9811056)

1 Preface

1.1 The Unique Contribution of Quantum Cryptography

Before beginning our story, I'd like to state precisely what is the unique contribution of quantum cryptography.

> Quantum cryptography provides a new mechanism enabling the parties communicating with one another to:
>
> $\boxed{\text{Automatically Detect Eavesdropping}}$
>
> Consequently, it provides a means for determining when an encrypted communication has been compromised.

* Partially supported by ARL Contract #DAAL01-95-P-1884, ARO Grant #P-38804-PH-QC, the Computer Security Division of NIST, and the L-O-O-P Fund.

1.2 A Note to the Reader

This paper is based on an invited talk given at the Conference on Coding theory, Cryptology, and Number Theory held at the US Naval Academy in Annapolis, Maryland in October of 1998. It was also given as an invited talk at the Quantum Computational Science Workshop held in conjunction with the Frontiers in Computing Conference in Annapolis, Maryland in February of 1999, at a Bell Labs Colloquium in Murray Hill, New Jersey in April of 1999, at the Security and Technology Division Colloquium of NIST in Gaithersburg, Maryland, and at the Quantum Computation Seminar at the U.S. Naval Research Labs in Washington, DC.

My objective in creating this paper was to write it exactly as I had given the talk. But ... Shortly after starting this manuscript, I succumbed to the temptation of greatly embellishing the story that had been woven into the original talk. I leave it to the reader to decide whether or not this detracts from or enhances the paper.

2 Introduction

We begin our crypto drama with the introduction of two of the main characters, *Alice* ♡and♡ *Bob*, representing respectively the **sender** and the *receiver.* As in every drama, there is a triangle. The triangle is completed with the introduction the third main character, the archvillainess **Eve**, representing the **eavesdropper**.

Our story begins with Alice and Bob attending two different universities which are unfortunately separated by a great distance. Alice would like to communicate with Bob without the ever vigilant Eve eavesdropping on their conversation. In other words, how can Alice talk with Bob while at the same time preventing the evil Eve from listening in on their conversation?

3 A Course on Classical Cryptography

3.1 Alice's enthusiastic decision

Hoping to find some way out of her dilemma, Alice elects to take a course on cryptography, Crypto 351 taught by Professor Shannon with guest lecturers Diffie, Rivest, Shamir, and Adleman. Alice thinks to herself, "Certainly this is a wise choice. It is a very applied course, and surely relevant to the real world. Maybe I will learn enough to outwit Eve?"

3.2 Plaintext, ciphertext, key, and ... Catch 22

Professor Shannon begins the course with a description of classical cryptographic communication systems, as illustrated in Fig. 1. Alice, the sender, encrypts her *plaintext* P into *ciphertext* C using a *secret key* K which she shares only with Bob, and sends the ciphertext C over an *insecure channel* on which the evil Eve is ever vigilantly eavesdropping. Bob, the receiver, receives the ciphertext C, and uses the secret key K, shared by him and Alice only, to decrypt the ciphertext C into plaintext P.

Figure 1. A classical cryptographic communication system.

What is usually not mentioned in the description of a classical cryptographic communication system is that Alice and Bob must first communicate over a *secure channel* to establish a secret key K shared only by Alice and Bob before they can communicate in secret over the insecure channel. Such a channel could consist, for example, of a trusted courier, wearing a trench coat and dark sunglasses, transporting from Alice to Bob a locked briefcase chained to his wrist. In other words, we have the famous Catch 22 of classical cryptography, namely:

> **Catch 22.** There are perfectly good ways to communicate in secret, provided we can communicate in secret ...

Professor Shannon then goes on to discuss the different types of classical communication security.

3.3 Practical Secrecy

A cryptographic communication system is **practically secure** if the encryption scheme can be broken after X years, where X is determined by one's security needs and by existing technology. Practically secure cryptographic systems have existed since antiquity. One example would be the Caesar cipher used by Julius Caesar during the during the Gallic wars, a cipher that was difficult for his opponents to break at that time, but easily breakable by today's standards. A modern day example of a practically secure classical cryptographic system is the digital encryption standard (DES) which has just recently been broken[1]. For this and many other reasons, DES is to be replaced by a more practically secure classical encryption system, the Advanced Encryption Standard (AES). In turn, AES will be replaced by an even more secure cryptographic system should the advances in technology ever challenge its security.

3.4 Perfect Secrecy

A cryptographic communication is said to be **perfectly secure** if the ciphertext C gives no information whatsoever about the plaintext P, even when the design of the cryptographic system is known. In mathematical terms, this can be stated succinctly with the equation:

$$PROB(P \mid C) = PROB(P).$$

In other words, the probability of plantext P given ciphertext C, written $PROB(P|C)$, is equal to the probability of the plaintext P.

An example of a perfectly secure classical cryptographic system is the **Vernam Cipher**, better known as the **One-Time-Pad**. The plantext P is a binary sequence of zeroes and ones, i.e.,

$$P = P_1, P_2, P_3, \ \dots \ P_n, \ \dots$$

The secret key K consists of a totally random binary sequence of the same length, i.e.,

$$K = K_1, K_2, K_3, \ \dots \ , \ K_n, \ \dots$$

The ciphertext C is the binary sequence

$$C = C_1, C_2, C_3, \ \dots \ C_n, \ \dots$$

obtained by adding the sequences P and K bitwise modulo 2, i.e.,

$$C_i = P_i + K_i (\bmod 2) \text{for} i = 1, 2, 3, \ \dots$$

[1] Tim O'Reilly and the Electronic Frontier Foundation have constructed a computing device for $250,000 which does an exhaustive key search on DES in 4.5 days[15]. See also [2] and [10]. As far as I know, triple DES has not been broken.

For example,

$$\begin{array}{rl} P & = 0110 \quad 0101 \quad 1101 \\ K & = 1010 \quad 1110 \quad 0100 \\ \text{-----------} & \text{------------------} \\ C = P \oplus K & = 1100 \quad 1011 \quad 1001 \end{array}$$

This cipher is perfectly secure if key K is totally random and shared only by Alice and Bob. It is easy to encode with the key K. If, however, one succumbs to the temptation of using the same key K to encode two different plaintext $P^{(1)}$ and $P^{(2)}$ into ciphertexts $C^{(1)}$ and $C^{(2)}$, then the cipher system immediately changes from a perfectly secure cipher to one that is easily broken by even the most amateur cryptanalyst. For, $C^{(1)} \oplus C^{(2)} = P^{(1)} \oplus P^{(2)}$ is easily breakable because of the redundancy that is usually present in plaintext.

The only problem with the one-time-pad is that long bit sequences must be sent over a secure channel before it can be used. This once again leads us to the Catch 22 of classical cryptography, i.e.,

> **Catch 22.** There are perfectly good ways to communicate in secret, provided we can communicate in secret ...

... and to the:

- **Key Problem 1.** *Catch 22*: A secure means of communicating key is needed.[2]

Finally, there are two other key problems in classical cryptography in need of a solution, namely:

- **Key Problem 2.** *Authentication*: Alice needs to determine with certainty that she is actually talking to Bob, and not to an impostor such as Eve.
- **Key Problem 3.** *Intrusion Detection*: Alice needs a means of determining whether or not Eve is eavesdropping.

[2] Hired trench coats are exorbitantly expensive and time consuming.

In summary, we have the following checklist for classical cryptographic systems:

<u>**Check List**</u>forClassicalCryptoSystems	
■Catch22Solved?	**NO**
■Athentication?	**NO**
■IntrusionDetection?	**NO**

3.5 Computational Security

Relatively recently in the history of cryptography, Diffie and Hellman [4], [5] suggested a new type cryptographic secrecy. A cipher is said to be **computationaly secure** if the computational resources required to break it exceed anything possible now and into the future. For example, a cipher would be computationally secure if the number of bits of computer memory required to break it were greater than the number of atoms in the universe, or if the computational time required to break it exceeded the age of the universe. Cryptographic systems can be created in such a way that it is computationally infeasible to find the decryption key D even when the encryption key E is known. To create such a cryptographic system, all one would need is a trap-door function f.

Definition 1. if A function f is a **trap-door function** if

1) f is easy to compute, i.e., polynomial time computable, and
2) Given the function f, the inverse function f^{-1}can not be computed from f in polynomial time. I.e., such a computation is superpolynomial time, intractable, or worse.

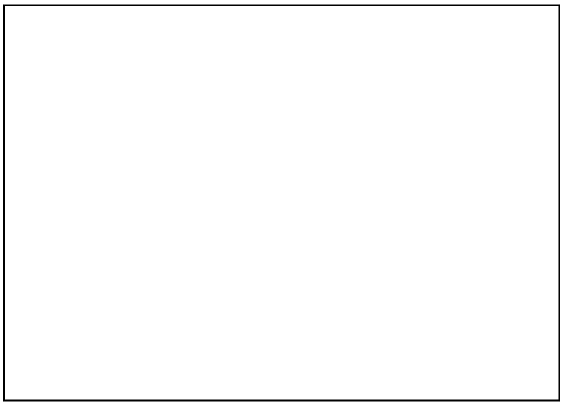


Figure 2. A public key cryptographic communication system

A trap-door function E can be used to create a **public key cryptographic system** as illustrated in Fig.2. All parties who wish to communicate in secret should choose their own trap-door function E and place it in a *public directory*, the "*yellow pages*," for all the world to see. But they should keep their decryption key $D = E^{-1}$ secret. Since E is a trap-door function, it is computationally infeasible for anyone to use the publicly known E to find the decryption key D. So D is secure in spite of the fact that its inverse E is publicly known.

If Alice wishes to send a secret communication to Bob, she first looks up in the yellow pages Bob's encryption key E_B, encrypts her plantext P with Bob's encryption key E_B to produce ciphertext $C = E_B(P)$, and the sends the ciphertext C over a public channel. Bob receives the ciphertext C, and decrypts it back into plaintext $P = D_B(C)$ using his secret decryption key D_B.

Alice can even do more than this. She can authenticate, i.e., sign her encrypted communication to Bob so that Bob knows with certitude that the message he received actually came from Alice and not from an Eve masquerading as Alice. Alice can do this by encrypting her signature $\mathcal{ALICE}$ using her secret decryption key D_A into $D_A(\mathcal{ALICE})$. She then encrypts plaintext P plus her signature $D_A(\mathcal{ALICE})$ using Bob's publicly known encryption key E_B to produce the signed ciphertext $C_{\mathcal{S}} = E_B(P + D_A(\mathcal{ALICE}))$, and then sends her signed ciphertext $C_{\mathcal{S}}$ over the public channel to Bob. Bob can then decrypt the message as he did before to produce the signed plaintext $P + D_A(\mathcal{ALICE})$. Bob can verify Alice's digital signature $D_A(\mathcal{ALICE})$ by looking up Alice's encryption key E_A in the "yellow pages," and using it to find her signature $E_A(D_A(\mathcal{ALICE})) = \mathcal{ALICE}$. In this way, he authenticates that Alice actually sent the message because only she knows her secret decryption key. Hence, only she could have signed the plaintext.[3]

The RSA cryptographic system is believed to be one example of a public key cryptographic system. There are many public software implementations of RSA, e.g., PGS (Pretty Good Security).

Thus, besides solving the authentication problem for cryptography, public key cryptographic systems appear also to solve the Catch 22 of cryptography. However, frequently the encryption and decryption keys of a public key cryptographic system are managed by a central key bank. In this case, the Catch 22 problem is still there. For that reason, we have entered 'MAYBE' in the summary given below.

[3] Because of the need for brevity, we have not discussed all the subtleties involved with digital signatures. For example, for more security, Alice should add a time stamp and some random symbols to her signature. For more information on digital signatures, please refer to one of the standard references such as [14].

Check ListforPKS

■Catch22Solved?	**MAYBE**
■Athentication?	**YES**
■IntrusionDetection?	**NO**

4 A Course on Quantum Mechanics

4.1 Alice's Reluctant Decision

In spite of Alice's many intense efforts to avoid taking a course in quantum mechanics, she was finally forced by her university's General Education Requirements (GERs) to register for the course Quantum 317, taught by Professor Dirac with guest lecturers Feynman, Bennett, and Brassard. She did so reluctantly. "Afterall," she thought, "Certainly this is an insane requirement. Quantum mechanics is not applied. It's too theoretical to be relevant to the real world. Ugh! But I do want to graduate."

4.2 The Classical World – Introducing the Shannon Bit

Professor Dirac began the course with a brief introduction to the classical world of information. In particular, Alice was introduced to the classical Shannon Bit, and shown that he/she/it is a very decisive individual. The Shannon Bit is either 0 or 1, but by no means both at the same time.

"Hmm ... ," she thought, "I bet that almost everyone I know is gainfully employed because of the Shannon Bit."

The professor ended his brief discussion of the Shannon Bit by mentioning that there is one of its properties that we take for granted. I.e., it can be copied.

4.3 The Quantum World – Introducing the Qubit

Next Professor Dirac switched to the mysterious world of the quantum. He began by introducing the runt of the Bit clan, i.e., the Quantum Bit, nicknamed **Qubit**. He began by showing the class a small dot, i.e., a quantum dot. In fact it was so small that Alice couldn't see it at all. He promptly pulled out a microscope[4], and projected a large image on a screen for the entire class to view.

[4] This is a most unusual microscope!

Professor Dirac went on to say, "In contrast to the decisive classical Shannon Bit, the Qubit is a very indecisive individual. It is both 0 and 1 at the same time! Moreover, unlike the Shannon Bit, the Qubit cannot be copied because of the no cloning theorem of Wootters and Zurek[21]. Qubits are very slippery characters, exceedingly difficult to deal with."

"One example of a qubit is a spin $\frac{1}{2}$ particle which can be in a spin-up state $|1\rangle$ which we label as 1, in a spin-down state $|0\rangle$ which we label as 0, or in a **superposition** of these states, which we interpret as being both 0 and 1 at the same time." (The term "superposition" will be explained shortly.)

"Another example of a qubit is the polarization state of a photon. A photon can be in a vertically polarized state $|\updownarrow\rangle$. We assign a label of 1 to this state. It can be in a horizontally polarized state $|\leftrightarrow\rangle$. We assign a label of 0 to this state. Or, it can be in a superposition of these states. In this case, we interpret its state as representing both 0 and 1 at the same time."

"Anyone who has worn polarized sunglasses should be familiar the polarization states of the photon. Polarized sunglasses eliminate glare because they let through only vertically polarized light while filtering out the horizontally polarized light that is reflected from the road."

4.4 Where do qubits live?

But where do qubits live? They live in a Hilbert space $\mathcal{H}$. By a Hilbert space, we mean:

Definition 2. A **Hilbert Space** is a vector space over the complex numbers $\mathbb{C}$ together with an inner product

$$\langle\ \ ,\ \ \rangle : \mathcal{H} \times \mathcal{H} \longrightarrow \mathbb{C}$$

such that

1) $\langle u_1 + u_2, v\rangle = \langle u_1, v\rangle + \langle u_2, v\rangle$ for all $u_1, u_2, v \in \mathcal{H}$
2) $\langle u, \lambda v\rangle = \langle \lambda u, v\rangle$ for all $u, v \in \mathcal{H}$ and $\lambda \in \mathbb{C}$
3) $\langle u, v\rangle^* = \langle v, u\rangle$ for all $u, v \in \mathcal{H}$, where the superscript '$*$' denotes complex conjugation.
4) For every Cauchy sequence $u_1, u_2, u_3, \ \ldots$ in $\mathcal{H}$,

$$\lim_{n\to\infty} u_n \text{existsandliesin}\mathcal{H}$$

In other words, a Hilbert space is a vector space over the complex numbers $\mathbb{C}$ with a sequilinear inner product in which sequences that should converge actually do converge to points in the space.

4.5 Some Dirac notation – Introducing kets

The elements of $\mathcal{H}$ are called **kets**, and will be denoted by

$$|label\rangle\,,$$

where '|' and '>' are left and right delimiters, and '*label*' denotes any label, i.e., name, we wish to assign to a ket.

4.6 Finally, a definition of a qubit

So finally, we can define what is meant by a qubit.

Definition 3. *A* **qubit** *is a ket (state) in a two dimensional Hilbert space $\mathcal{H}$.*

Thus, if we let $|0\rangle$ and $|1\rangle$ denote an arbitray orthonormal basis of a two dimensional Hilbert space $\mathcal{H}$, then each qubit in $\mathcal{H}$ can be written in the form

$$|qubit\rangle = \alpha_0 |0\rangle + \alpha_1 |0\rangle$$

where $\alpha_0, \alpha_1 \in \mathbb{C}$. Since any scalar multiple of a ket represents the same state of an isolated quantum system, we can assume, without loss of generality, that that $|qubit\rangle$ is a ket of unit length, i.e., that

$$|\alpha_0|^2 + |\alpha_1|^2 = 1$$

The above qubit is said to be in a **superposition** of the states $|0\rangle$ and $|1\rangle$. This is what we mean when we say that a qubit can be simutaneously both 0 and 1. However, if the qubit is observed it immediately "makes a decision." It "decides" to be 0 with probability $|\alpha_0|^2$ and 1 with probability $|\alpha_1|^2$. Some physicists call this the **"collapse"** of the wave function[5].

[5] It is very difficult, if not impossible, to find two physicists who agree on the subject of quantum measurement. The phrase "collapse of the wave function" immediately engenders a "war cry" in most physicists. For that reason, "collapse" is enclosed in quotes.

4.7 More Dirac notation – Introducing bras and bra-c-kets

Given a Hilbert space $\mathcal{H}$, let

$$\mathcal{H}^* = Hom(\mathcal{H}, \mathbb{C})$$

denote the set of all linear maps from $\mathcal{H}$ to $\mathbb{C}$. Then $\mathcal{H}^*$ is actually a Hilbert space, called the **dual** Hilbert space of $\mathcal{H}$, with scalar product and vector sum defined by:

$$\begin{cases} (\lambda \cdot f)\left(|\Psi\rangle\right) & = \lambda\left(f(|\Psi\rangle)\right), \qquad \text{forall}\lambda \in \mathbb{C}\text{andforall}f \in \mathcal{H}^* \\ (f_1 + f_2)\left(|\Psi\rangle\right) & = f_1\left(|\Psi\rangle\right) + f_2\left(|\Psi\rangle\right), \text{forall}f_1, f_2 \in \mathcal{H}^* \end{cases}$$

We call the elements of $\mathcal{H}^*$ **bra**'s, and denote them as:

$$\langle label|$$

We can now define a bilinear map

$$\mathcal{H}^* \times \mathcal{H} \longrightarrow \mathbb{C}$$

by

$$\left(\langle\Psi_1|\right)\left(|\Psi_2\rangle\right) \in \mathbb{C}$$

since bra $\langle\Psi_1|$ is a complex valued function of kets. We denote this product more simply as

$$\langle\Psi_1 \mid \Psi_2\rangle$$

and call it the **Bra-c-Ket** (or **bracket**) of bra $\langle\Psi_1|$ and ket $|\Psi_2\rangle$.

Finally, the bracket induces a dual correspondence[6] between $\mathcal{H}$ and $\mathcal{H}^*$, i.e.,

$$|\Psi_2\rangle \overset{D.C.}{\longleftrightarrow} \langle\Psi_1|$$

4.8 Activities in the quantum world – Unitary transformations

All "activities" in the quantum world are linear transformations

$$U : \mathcal{H} \longrightarrow \mathcal{H}$$

from the Hilbert space $\mathcal{H}$ into itself, called **unitary transformations** (or, **unitary operators**). If we think of linear transformations as matrices, then

[6] This is true for finite dimensional Hilbert spaces. It is more subtle for infinite dimensional Hilbert spaces.

a **unitary transformation** U is a square matrix of complex numbers such that

$$\overline{U}^T U = I = U\overline{U}^T$$

where $\overline{U}^T$ denotes the matrix obtained from U by conjugating all its entries and then transposing the matrix. We denote $\overline{U}^T$ by $U^\dagger$, and refer to it as the **adjoint** of U.

Thus, an "activity" in the quantum world would be, for example, a unitary transformation U that carries a state ket $|\Psi_0\rangle$ at time $t = 0$ to a state ket $|\Psi_1\rangle$ at time $t = 1$, i.e.,

$$U : |\Psi_0\rangle \longmapsto |\Psi_1\rangle$$

4.9 Observables in quantum mechanics – Hermitian operators

In quantum mechanics, what does an observer observe?

All **observables** in the quantum world are linear transformations

$$\mathcal{O} : \mathcal{H} \longrightarrow \mathcal{H}$$

from the Hilbert space $\mathcal{H}$ into itself, called **Hermitian operators** (or, **self-adjoint operators**). If we think of linear transformations as matrices, then a **Hermitian operator** $\mathcal{O}$ is a square matrix of complex numbers such that

$$\overline{\mathcal{O}}^T = \mathcal{O}$$

where $\overline{\mathcal{O}}^T$ again denotes the matrix obtained from $\mathcal{O}$ by conjugating all its entries, and then transposing the matrix. As before, we denote $\overline{\mathcal{O}}^T$ by $\mathcal{O}^\dagger$, and refer to it as the **adjoint** of $\mathcal{O}$.

Let $|\varphi_i\rangle$ denote the eigenvectors, called **eigenkets**, of an observable $\mathcal{O}$, and let a_i denote the corresponding eigenvalue, i.e.,

$$\mathcal{O} : |\varphi_i\rangle = a_i\,|\varphi_i\rangle$$

In the cases we consider in this talk, the eigenkets form an othonormal basis of the underlying Hilbert space $\mathcal{H}$.

Finally, we can answer our original question, i.e.,

$$\boxed{\text{What does an observer observe?}}$$

Let us suppose that we have a physical device M that is so constructed that it measures an observable $\mathcal{O}$, and that we wish to use M to measure a quantum system which just happens to be in a quantum state $|\Psi\rangle$. We assume $|\Psi\rangle$ is a ket of unit length. The quantum state $|\Psi\rangle$ can be written as a linear combination of the eigenkets of $\mathcal{O}$, i.e.,

$$|\Psi\rangle = \sum \alpha_i |\varphi_i\rangle$$

When we use the device M to measure $|\Psi\rangle$, we observe the eigenvalue a_i with probability $p_i = |\alpha_i|^2$, and in addition, after the measurement the quantum system has "collapsed" into the state $|\varphi_i\rangle$. Thus, the outcome of a measurement is usually random, and usually has a lasting impact on the state of the quantum system.

We can use Dirac notation to write down an expression for the average observed value. Namely, the **averaged observed value** is given by the expression $\langle\Psi| (\mathcal{O} |\Psi\rangle)$, which is written my succinctly as $\langle\Psi| \mathcal{O} |\Psi\rangle$, or simply as $\langle\mathcal{O}\rangle$.

4.10 The Heisenberg uncertainty principle – A limitation on what we can actually observe

There is, surprisingly enough, a limitation of what can be observed in quantum mechanics.

Two observables A and B are said to be **compatible** if they commute, i.e., if

$$AB = BA.$$

Otherwise, they are said to be **incompatible**.

Let $[A, B]$, called the **commutator** of A and B, denote the expression

$$[A, B] = AB - BA$$

In this notation, two operators A and B are compatible if and only if $[A, B] = 0$.

Finally, let

$$\triangle A = A - \langle A\rangle$$

The following principle is one expression of how quantum mechanics places limits on what can be observed:

Heisenberg's Uncertainty Principle[7]

$$\left\langle(\triangle A)^2\right\rangle \left\langle(\triangle B)^2\right\rangle \geq \frac{1}{4} \|\langle[A, B]\rangle\|^2$$

[7] We have assumed units have been chosen such that $\hbar = 1$.

where $\left\langle (\triangle A)^2 \right\rangle = \langle \varPsi | (\triangle A)^2 | \varPsi \rangle$ is the **standard deviation** of the observed eigenvalue, written in Dirac notation. It is a measure of the uncertainty in A.

This if A and B are incompatible, i.e., do not commute, then, by measuring A more precisely, we are forced to measure B less precisely, and vice versa. We can not simultaneously measure both A and B to to unlimited precision. Measurement of A somehow has an impact on the measurement of B.

4.11 Young's two slit experiment – An example of Heisenberg's uncertainty principle

For the purpose of illustrating Heisenberg's Uncertainty Principle, Professor Dirac wheeled out into the classroom a device to demonstrate Young's two slit experiment. The device consisted of an electron gun which spewed out electrons in the direction of a wall with two slits. The electrons that managed to pass through the two slits then impacted on a backstop coated with a phosphorescent material that produced light when hit by an electron. The intensity pattern of light and shadows that was produced on the backdrop was projected onto the classroom screen for all to view[8].

Professor Dirac proceeded to demonstrate what the device could do. He began by covering slit 2, allowing the incoming electrons to pass only through slit 1. The resulting intensity pattern appearing on the projection screen was a bell shaped curve, i.e., the Guassian distribution, as illustrated by curve P_1 as drawn in Fig. 3a. When the professor covered the slit 1 instead of slit 2, exactly the same patterned appeared, but only this time shifted a short distance to the right, as illustrated by the curve P_2 shown in Fig. 3a.

Figure 3a. Young's two slit experiment with one slit closed.

[8] The original Young's two slit experiment used photons rather than electrons.

Professor Dirac then asked the students in the class what pattern they thought would appear if he uncovered both of the slits. Most of the class responded by saying that the resulting light pattern would simply be the sum of the two patterns, i.e., the bell shaped curve $P_1 + P_2$, as illustrated in Fig. 3c. Most of the class was convinced that the two classical probability distributions would simply add, as many of them had learned in the probability course Prob 323.

The remainder of the class stated quite emphatically that they did not care what happened. What was being illustrated was far from an applied area, and hence not relevant to their real world. Or so they thought ...

Professor Dirac smiled, and then proceeded to uncover both slits. What appeared on the screen to almost everyone's surprise was not the pattern with the bell shape $P_1 + P_2$. It was instead a light pattern with a wavy bell shaped curve, as illustrated by the curve P_{12} in Fig. 3b.

Figure 3b. Young's two slit experiment with both slits open.

Professor Dirac explained, "Something non-classical had occured. Unlike classical probabilities, the quantum probabilities (or more correctly stated, the quantum amplitudes) had interferred with one another to produce an interference pattern. In the dark areas, one finds destructive interference. In the bright areas, one finds constructive interference. Something non-classical is happening here."

"Strangely enough, quantum mechanics is telling us that each electron is actually passing through both slits simultaneously! It is as if each electron were a wave and not a particle."

"But what happens when we actually try to observe through which slit each electron passes?"

Professr Dirac pulled out his trusty microscope[9] to observe which of the two slits each electron passed through. As soon as he started making an observation, the intereference pattern was transformed into the classical light

[9] This is a most unusual microscope.

pattern all had initially expected to see, i.e., the light pattern of the bell shaped curve $P_1 + P_2$, as shown in Fig. 3c. When observed, the electrons act as particles and not as waves!

Figure 3c. Young's 2 slit experiment when the slit through which the electron passes is determined by observation.

After a brief pause, Professor Dirac said, "This is actually an example of the Heisenberg Uncertainty Principle. We can see this as follows:"

"In the experiment, we are effectively observing two incompatible observables, the position operator X (i.e., which slit each electron passes through) and the momentum operator P (i.e., which includes the direction at which each electron leaves the slitted wall.) When we observe the momentum P, the interference pattern is present. But when we observe the position X, the interference pattern vanishes."

5 The Beginnings of Quantum Cryptography

5.1 Alice has an idea

After class on her way back to her dorm room, Alice began once again to ruminate over her dilemma in regard to Bob and Eve.

"If only her message to Bob were like the interference pattern in Young's two slit experiment. Then, if the prying Eve were to observe which of the two slits the electron emerged from (i.e., 'listen in'), Bob would immediately know of her presence. For, if Eve were to make an observation, the pattern on the screen would immediately change from the beautiful wavy interference pattern to the dull ugly guassian distribution pattern."

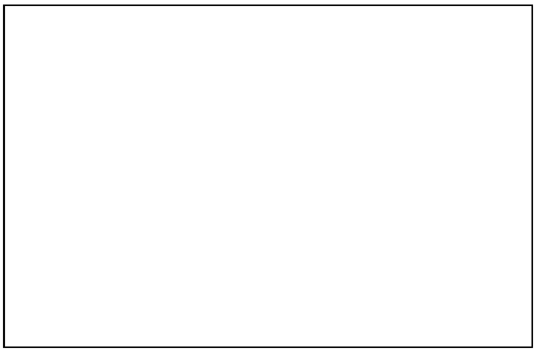


Figure 4a. Bob sees an interference pattern when Eve is not eavesdropping.

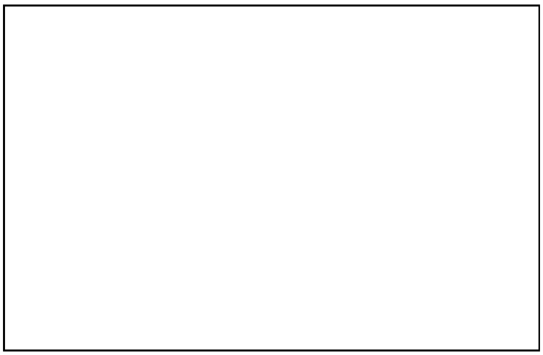


Figure 4b. Bob sees no interference pattern when Eve is eavesdropping.

"This idea has possibilities. Maybe quantum mechanics is relevant afterall!"

Her mind began to race. "Perhaps something like Young's two slit experiment could be used to communicate random key K? Then Bob could tell which key had been compromised by an intruder such as Eve. But most importantly, he could also surmise which key had not been compromised. Bob could then communicate to me over the phone (or even over any public channel available also to Eve) whether or not the key had been compromised, without, of course, revealing the key itself. Any uncompromised key could then be employed to send Bob a message by using the one-time-pad that was mentioned yesterday in Crypto 351."

"The beauty of this approach is that the one-time-pad is perfectly secure. There is no way whatsoever that Eve could get any information about our conversation. This would be true even if I used the campus radio station to send my encrypted message."

"The evil Eve is foiled! Eureka! Contrary to student conventional wisdom, both cryptography and quantum mechanics are relevant to the real world!"

"I have discovered a new kind of secrecy, i.e., quantum secrecy, which has built-in detection of eavesdropping based on the principles of quantum mechanics. I can hardly wait to tell Professor Dirac. She ran immediately to his office."

After listening to Alice's excited impromptu, and at times disjointed, explanation, Professor Dirac suggested that she present her newly found discoveries in his next class. Alice happily agreed to do so.

5.2 Quantum secrecy – The BB84 protocol without noise

Two days later, after two sleepless but productive nights of work, Alice was prepared for her presentation. She walked in the classroom for Quantum 317 carrying an overhead projector and a sizable bundle of transparencies.

After Professor Dirac had turned the large lecture hall over to her, she began as follows:

"Let us suppose that I (Alice) would like to transmit a secret key K to Bob. Let us also suppose that someone by the name of Eve intends to make every effort to eavesdrop on the transmission and learn the secret key."

Wouldn't you know it. Eve just so happens to be sitting in the classroom!

"My objective today is to show you how the principles of quantum mechanics can be used to build a cryptographic communication system in such a way that the system detects if Eve is eavesdropping, and which also gives a guarantee of no intrusion if Eve is not eavesdropping."

"A diagrammatic outline of the system I'm about to describe is shown on the screen. (Please refer to Fig. 5.) Please note that the system consists of two communication channels. One is a non-classical one-way quantum communication channel, which Iwill soon describe. The other is an ordinary run-of the-mill classical two-way public channel, such as a two way radio communication system. I emphasize that this classical two-way channel is public, and open to whomever would like to listen in. For the time being, I will assume that the two-way public channel is noise free."

Figure 5. A quantum cryptographic communication system.

"I will now describe how the polarzation states of the photon can be used to construct a quantum one-way communication channel[10]."

"From Professor Dirac's last lecture, we know that the polarization states of a photon lie in a two dimensional Hilbert space $\mathcal{H}$. For this space, there are many othonormal bases. We will use only two for our quantum channel."

"The first is the basis consisting of the vertical and horizontal polarization states, i.e, the kets $|\updownarrow\rangle$ and $|\leftrightarrow\rangle$, respectively. We will refer to this orthonormal basis as the **vertical/horizontal (V/H) basis**, and denote this basis with the symbol '⊞.' "

"The second orthonormal basis consists of the polarization states $|\nearrow\rangle$ and $|\nwarrow\rangle$, which correspond to polarizations directions formed respectively by 45% clockwise and counter-clockwise rotations off from the vertical. We call this the **oblique basis**, and denote this basis with the symbol '⊠.' "

"If I (Alice) decide to use the VH basis ⊞ on the quantum channel, then I will use the following **quantum alphabet**:

$$\begin{cases} \text{“1”} = |\updownarrow\rangle \\ \text{“0”} = |\leftrightarrow\rangle \end{cases}$$

In other words, if I use this quantum alphabet on the quantum channel, I will transmit a "1" to Bob simply by sending a photon in the polarization state $|\updownarrow\rangle$., and I will transmit a "0" by sending a photon in the polarization state $|\leftrightarrow\rangle$."

"On the other hand, if I (Alice) decide to use the oblique basis ⊠, then I will use the following **quantum alphabet**:

$$\begin{cases} \text{“1”} = |\nearrow\rangle \\ \text{“0”} = |\nwarrow\rangle \end{cases},$$

sending a "1" as a photon in the polarization state $|\nearrow\rangle$, and sending a "0" as a photon in the polarization state $|\nwarrow\rangle$."

"I have chosen these two bases because the Heisenberg Uncertainty Principle implies that observations with respect to the ⊞ basis are incompatible with observations with respect to the ⊠ basis. We will soon see how this incompatiblity can be translated into intrusion detection."

[10] Any two dimensional quantum system such as a spin $\frac{1}{2}$ particle could be used.

Alice	⊞	⊠	⊠	⊠	⊞	⊠	⊞	⊠	⊞	⊠
	↕	↖	↖	↗	↕	↖	↔	↗	↔	↗
	1	0	0	1	1	0	0	1	0	1
	*		*					*		*
Bob	⊠	⊠	⊞	⊠	⊞	⊠	⊞	⊞	⊞	⊞
	1	0	1	1	1	0	0	0	0	0
					⇕					
Raw Key⇒		0		1	1	0	0		0	

Fig. 6a. The BB84 protocol without Eve present (No noise)

Alice and Bob now communicate with one another using a two stage protocol, called the **BB84 protocol**[1]. (Please refer to Figs. 6a and 6b.)

In stage 1, Alice creates a random sequence of bits, which she sends to Bob over the quantum channel using the following protocol:

Stage 1 protocol: Communication over a quantum channel

Step 1. Alice flips a fair coin to generate a random sequence S_{Alice} of zeroes and ones. This sequence will be used to construct a secret key shared only by Alice and Bob.

Step 2. For each bit of the random sequence, Alice flips a fair coin again to choose at random one of the two quantum alphabets. She then transmits the bit as a polarized photon according to the chosen alphabet.

Step 3. Each time Bob receives a photon sent by Alice, he has no way of knowing which quantum alphabet was chosen by Alice. So he simply uses the flip of a fair coin to select one of the two alphabets and makes his measurement accordingly. Half of the time he will be lucky and choose the same quantum alphabet as Eve. In this case, the bit resulting from his measurement will agree with the bit sent by alice. However, the other half of the time he will be unlucky and choose the alphabet not used by Alice. In this case, the bit reulting from his measurement will agree with the bit sent by Alice only 50% of the time. After all these measurements, Bob now has in hand a binary sequence S_{Bob}.

Alice and Bob now proceed to communicate over the public two-way channel using the following stage 2 protocol:

Stage 2 protocol: Communication over a public channel

Phase 1. *Raw key extraction*

Step 1. Over the public channel, Bob communicates to Alice which quantum alphabet he used for each of his measurements.

Step 2. In response, Alice communicates to Bob over the public channel which of his measurements were made with the correct alphabet.

Step 3. Alice and Bob then delete all bits for which they used incompatible quantum alphabets to produce their resulting **raw keys**. If Eve has not eavesdropped, then their resulting raw keys wil be the same. If Eve has eavedropped, their resulting raw keys will not be in total agreement.

Phase 2. *Error estimation*

Step 1. Over the public channel, Alice and Bob compare small portions of their raw keys to estimate the error-rate R, and then delete the disclosed bits from their raw keys to produce their **tentative final keys**. If through their public disclosures, Alice and Bob find no errors (i.e., $R = 0$), then they know that Eve was not eavesdropping and that their tentative keys must be the same **final key**. If they discover at least one error during their public disclosures (i.e., $R > 0$), then they know that Eve has been eavesdropping. In this case, they discard their tentative final keys and start all over again[11].

Alice	⊞	⊠	⊠	⊠	⊞	⊠	⊞	⊠	⊞	⊠
	↕	↖	↖	↗	↕	↖	↔	↗	↔	↗
	1	0	0	1	1	0	0	1	0	1

Eve	⊠	⊞	⊞	⊠	⊞	⊞	⊠	⊠	⊞	⊞
	1	0	1	1	1	1	0	1	0	0

Bob	⊠	⊠	⊞	⊠	⊞	⊠	⊞	⊞	⊞	⊞
	1	0	1	1	1	1	1	0	0	0
	*	0	*	1	1	1	1	*	0	*
						E	E			

Fig 6b. The BB84 with Eve present (No noise)

[11] If Eve were to intercept each qubit received from Alice, to measure it, and then to masqurade as Alice by sending on to Bob a qubit in the state she measured, then Eve would be introducing a 25% error rate in Bob's raw key. This method of eavesdropping is called **opaque eavesdropping**. We will discuss this eavesdropping strategy as well as others at a later time.

5.3 Quantum secrecy – The BB84 protocol with noise

Alice continues her presentation by addressing the issue of noise.

"So far we have assumed that that our cryptographic communication system is noise free. But every realistic communication system has noise present. Consequently, we now need to modify our quantum protocol to allow for the presence of noise."

"We must assume that Bob's raw key is noisy. Since Bob can not distinquish between errors caused by noise and by those caused by Eve's intrusion, the only practical working assumption he can adopt is that all errors are caused by Eve's eavesdropping. Under this working assumption, Eve is always assumed to have some information about bits transmitted from Alice to Bob. Thus, raw key is always only *partially secret*."

"What is needed is a method to distill a smaller secret key from a larger partially secret key. We call this **privacy amplification**. We will now create from the old protocol a new protocol that allows for the presence of noise, a protocol that includes privacy amplification."

Stage 1 protocol: Communication over a quantum channel

This stage is exactly the same as before, except that errors are now also induced by noise.

Stage 2 protocol: Communication over a public channel

Phase 1 protocol: Raw key extraction.

This phase is exactly the same as in the noise-free protocol, except that Alice and Bob also delete those bit locations at which Bob should have received but did not receive a bit. Such "non-receptions" could be caused by Eve's intrusion or by *dark counts* in Bob's detection device. The location of dark counts are communicated by Bob to Alice over the public channel.

Phase 2 protocol: Error estimation.

Over the public channel, Alice and Bob compare small portions of their raw keys to estimate the error-rate R, and then delete the disclosed bits from their raw key to produce their **tentative final keys**. If R exceeds a certain threshold R_{Max}, then privacy amplification is not possible If so, Alice and Bob return to stage 1 to start over. On the other hand, if $R \leq R_{Max}$, then Alice and Bob proceed to phase 3.

Phase 3 protocol: Extraction of reconciled key[12].

[12] There are more efficient and elegant procedures than the procedure descibed in Stage 2 Phase 3. See [9] for references.

In this phase[13], Alice and Bob remove all errors from what remains of raw key to produce a common error-free key, called **reconciled key**.

Step 1. Alice and Bob publically agree upon a random permutation, and apply it to what remains of their respective raw keys. Next Alice and Bob partition the remnant raw key into blocks of length ℓ, where the length ℓ is chosen so that blocks of that length are unlikely to have more than one error. For each of these blocks, Alice and Bob publically compare overall parity checks, making sure each time to discard the last bit of each compared block. Each time an overall parity check does not agree, Alice and Bob initiate a binary search for the error, i.e., bisecting the block into two subblocks, publically comparing the parities for each of these subblocks, discarding the right most bit of each subblock. They continue their bisective search on the subblock for which their parities are not in agreement. This bisective search continues until the erroneous bit is located and deleted. They then continue to the next ℓ-block.
This step is repeated, i.e., a random permutation is chosen, a remnant raw key is partitioned into blocks of length ℓ, parities are compared, etc.. This is done until it becomes inefficient to continue in this fashion.

Step 2. Alice and Bob publically select randomly chosen subsets of remnant raw key, publically compare parities, each time discarding an agreed upon bit from their chosen key sample. If a parity should not agree, they employ the binary search strategy of Step 1 to locate and delete the error.

- Finally, when, for some fixed number N of consecutive repetions of Step 2, no error is found, Alice and Bob assume that to a high probability, the remnant raw key is without error. Alice and Bob now rename the remnant raw key **reconciled key**, and proceed to the next phase.

Phase 4: Privacy amplification

Alice and Bob now have a common reconciled key which they know is only partially secret from Eve. They now begin the process of **privacy amplification**, which is the extraction of a secret key from a partially secret one.

Step 1. Alice and Bob compute from the error-rate R obtained in Phase 2 of Stage 2 an upper bound k of the number of bits of reconciled key known by Eve.

[13] The procedure given in Stage 2 Phase 3 is only one of many different possible procedures. In fact, there are much more efficient and elegant procedures than the one described herein.

Let n denote the number of bits in reconciled key, and let s be a *security parameter* to be adjusted as required.

Step 2. Alice and Bob publically select $n - k - s$ random subsets of reconciled key, without revealing their contents. The undisclosed parities of these subsets become the final secret key.

It can be shown that Eve's average information about the final secret key is less than $2^{-s}/\ln 2$ bits.

The bell rang, indicating the end of the period. The entire class with two exceptions, immediately raced out of the lecture hall, almost knocking Alice down as they passed by. Professor Dirac thanked Alice for an excellent presentation.

As Alice left, she saw Eve in one of the dark recesses of the large lecture hall with her head resting on the palm of her hand as if in deep thought. She had a frown on her face. Alice left with a broad smile on her face.

6 The B92 quantum cryptographic protocol

In the next class, Alice continued her last presentation.

In thinking about about the BB84 protocol this weekend, I was surprised to find that it actually is possible to build a different quantum protocol that uses only one quantum alphabet instead of two. I'll call this new quantum protocol **B92**."

"As before, we will describe the protocol in terms of the polarization states of the photon[14]."

"As our quantum alphabet, we choose

$$\begin{cases} \text{“1”} = |\theta_+\rangle \\ \\ \text{“0”} = |\theta_-\rangle \end{cases} ,$$

where $|\theta_+\rangle$ and $|\theta_-\rangle$ denote respectively the polarization states of a photon in linearly polarized at angles θ and $-\theta$ with respect to the vertical, where $0 < \theta < \frac{\pi}{4}$."

[14] Any two dimensional quantum system such as a spin $\frac{1}{2}$ particle could be used.

"We assume that Bob's quantum receiver, called a **POVM receiver** [3], is base on the following observables[15]:

$$\begin{cases} A_{\theta_+} &= \frac{1 \ - \ |\theta_-\rangle\langle\theta_-|}{1 \ + \ \langle\theta_+|\theta_-\rangle} \\ \\ A_{\theta_-} &= \frac{1 \ - \ |\theta_+\rangle\langle\theta_+|}{1 \ + \ \langle\theta_+|\theta_-\rangle} \\ \\ A_? &= 1 - A_{\theta_+} - A_{\theta_-} \end{cases},$$

where A_{θ_+} is the observable for $|\theta_+\rangle$, A_{θ_-} the observable for $|\theta_-\rangle$ and $A_?$ is the observables for inconclusive receptions."

The **B92** quantum protocol is as follows:

Stage 1 protocol. Communication over a quantum channel.

Step 1. The same as in the BB84 protocol. Alice flips a fair coin to generate a random sequence S_{Alice} of zeroes and ones. This sequence will be used to construct a secret key shared only by Alice and Bob.

Step 2. The same as in the previous protocol, except this time Alice uses only one alphabet, the one above. So she does not have to flip a coin to choose an alphabet.

Step 3. Bob uses his POVM receiver to measure photons received from Alice.

Stage 2. Communication in four phases over a public channel.

This stage is the same as in the BB84 protocol, except that in phase 1, Bob publically informs Alice as to which time slots he received non-erasures. The bits in these time slots become Alice's and Bob's raw keys.

Alice completed her discussion of the B92 protocol with,

"Eve's presence is again detected by an unusual error rate in Bob's raw key. Moreover, for some but not all eavesdropping strategies, Eve can also be detected by an unusual erasure rate for Bob."

Alice then stepped down from the lecture hall podium and returned to her seat.

[15] The observables A_{θ_+}, A_{θ_-}, and $A_?$ form a postive operator value measure (POVM).

7 There are many other quantum cryptographic protocols

Before continuing our story about Alice, Bob, and Eve, there are a few points that need to be made:

There are many other quantum cryptographic protocols. Quantun protocols showing the greatest promise for security are those based on EPR pairs. Unfortunately, the technology for implementing such protocols is not yet available. For references on various protocols, please refer to [9].

8 A comparison of quantum cryptography with classical and public key cryptography

Quantum cryptography's unique contribution is that it provides a mechanism for eavesdropping detection. This is an entirely new contribution to cryptography. On the other hand, one of the main drawbacks of quantum cryptography is that it provides no mechanism for authentication, i.e., for detecting whether or not Alice and Bob are actually communicating with each other, and not with an intermediade Eve masquarading as each of them. Thus, the Catch 22 problem is not solved by quantum cryptography. Before Alice and Bob can begin their quantum protocol, they first need to send an authentication key over a secure channel.

Thus, quantum cryptography's unique contribution is to provide a means of expanding existing secure key. Quantum protocols are secure key expanders. First a small authentication key is exchanged over a secure channel. Then that key can be amplified to an arbitrary length through quantum cryptography.

Check ListforQ.Crypto. Sys.

■Catch22Solved?	**YES & NO**
■Athentication?	**NO**
■IntrusionDetection?	**YES**

9 Eavesdropping strategies and counter measures

Now let us resume our story:

Not a split second after Alice had seated herself, Eve raised her hand and asked for permission to make her own presentation to the class. Professor Dirac yielded the podium, not knowing exactly what to expect, but nonetheless elated that his usually phlegmatic class was beginning to show signs of something he had not seen for some time, class participation and initiative.

Eve began, "In the last two classes, Alice has suggested that I (Eve) might be eager to eavesdrop on her conversations with my ♡close♡ friend ♡Bob♡. I assure you that that simply is in no way true."

"But such innuendo really doesn't bother me."

9.1 Opaque eavesdropping

"What really irks me is that Alice suggests that, if I were to eavesdrop (which never would happen), then I (Eve) would use **opaque eavesdropping**. By **opaque eavesdropping**, I mean that I (Eve) would intercept and observe (measure) Alice's photons, and then masquerade as Alice by sending photons in the states I had measured on to Bob."

"I assure you that, if I ever wanted to eavesdrop (which would never be the case), I would not use such a simplistic form of intrusion."

Eve really wanted to use the adjective 'stupid' instead of 'simplistic,' but restrained herself.

Eve then said indignantly, "If I ever were to eavesdrop (which will never happen), I would use more sophisticated, more intellegent, and yes ... , more deliciously devious schemes!"

9.2 Translucent eavesdropping without entanglement

"I (Eve) could for example make my probe interact unitarily with the information carrier from Alice, and then let it proceed on to Bob in a slightly modified state. For B92 protocol, the interaction is given by:

$$\begin{cases} \left|\theta_+\right\rangle \left|\psi\right\rangle \longmapsto U\left|\theta_+\right\rangle \left|\psi\right\rangle = \left|\theta'_+\right\rangle \left|\psi_+\right\rangle \\ \\ \left|\theta_-\right\rangle \left|\psi\right\rangle \longmapsto U\left|\theta_-\right\rangle \left|\psi\right\rangle = \left|\theta'_-\right\rangle \left|\psi_-\right\rangle \end{cases},$$

where $\left|\psi\right\rangle$ and $\left|\psi_\pm\right\rangle$ denote respectively the state of my (Eve's) probe before and after the interaction and where $\left|\theta_\pm\right\rangle$ and $\left|\theta'_\pm\right\rangle$ denote respectively the state of Alice's photon before and after the interaction."

9.3 Translucent eavesdropping with entanglement

"Another approach, one of the most sophisticated, would be for me (EVE) to entangle my probe with the information carrier from Alice, and then let it proceed on to Bob. For the B92 protocol, the interaction is given by:

$$\begin{cases} \left|\theta_+\right\rangle\left|\psi\right\rangle \longmapsto U\left|\theta_+\right\rangle\left|\psi\right\rangle = a\left|\theta'_+\right\rangle\left|\psi_+\right\rangle + b\left|\theta'_-\right\rangle\left|\psi_+\right\rangle \\ \\ \left|\theta_-\right\rangle\left|\psi\right\rangle \longmapsto U\left|\theta_-\right\rangle\left|\psi\right\rangle = b\left|\theta'_+\right\rangle\left|\psi_-\right\rangle + a\left|\theta'_-\right\rangle\left|\psi_-\right\rangle \end{cases},$$

where $\left|\psi\right\rangle$ and $\left|\psi_\pm\right\rangle$ denote respectively the state of my (Eve's) probe before and after the entanglement and where $\left|\theta_\pm\right\rangle$ and $\left|\theta'_\pm\right\rangle$ denote respectively the state of Alice's photon before and after the entanglement."

9.4 Eavesdropping based on implementation weaknesses

"On the other hand, I could also take advantage of implementation weaknesses."

"One of the great difficulties with quantum cryptography is that technology has not quite caught up with it. Many devices, such as lasers, do not emit a single quantum, but many quanta at each emission time. The implementation of quantum protocols really requires single-quantum emitters. Such single-quantum emitters are now under development. Until such emitters become available, the quantum protocols can only be approximately implemented."

"For example, for many optical implementations of quantum protocols, the laser intensity is turned down to that of $\frac{1}{10}$-th the intensity of a photon. Thus, if anything is emitted at all (one chance out of 10), then the probability that it is a single photon is extremely high. However, when there is an emission, then there is a probability of $\frac{1}{200}$ that more than one photon is emitted. So it is conceivable (but not yet on the technological horizon) that I (Eve) could build an eavesdropping device that would detect multiple photon transmissions, and, when so detected, would divert one of the photons for measurement. In this way, I (Eve) could conceivably read $\frac{1}{200}$ of Alice's transmission without being detected. One way of countering this type of threat is to allow for it during privacy amplification."

"Finally, depending on Alice's implementation, it might also be possible for me (Eve) to gain information simply by observing Alice's transmitter without measuring its output. This may or may not be far fetched."

Eve then returned to her seat. Her face was lit up with a sinister grin of satisfaction.

10 Implementations

Before continuing our story, we should mention that quantum cryptographic protocols have been implemented over more than 30 kilometers of fiber optic cable, [16],[17],[18], [19], and most amazingly, over 100 meters of free space[7],[6] in the presence of ambient sunlight. There have been a number of ambitious proposals to demonstrate the feasibility of quantum cryptography in earth to satelite communications. And as mentioned earlier, there is a clear need for the development of single-quantum emitting devices.

11 Conclusion

Much remains to be done. There has been some work on the development of multiple-user quantum cryptographic protocols for communication networks[20]. There also have been at least two independent claims of the proof of ultimate security, i.e., a proof that quantum cryptographic protocols are impervious to all possible eavesdropping strategies [8], [11], [12], [13].

Our story continues:

As Alice sat in her seat, she happened to spy in the corner of her eye an abrupt change in Eve's demeanor. Eve suddenly became agitated, lit up with excitement, and started to frantically write on her notepad. The bell rang. Eve immediately jumped up, and raced out of the lecture hall, being pushed along by the usual frantic mass of students, equally eager to get out of the classroom.

As Eve whisked past, Alice caught just a fleeting glimpse of Eve's notepad. All Alice was able to discern in that brief moment was an illegible jumble of equations and ... yes, ... the acronym "POVM."

Alice thought to herself, "Oh, well! ... Forget it! I think I'll just visit Bob this weekend."

THE END[16]

[16] Any resemblance of the characters in this manuscript to individuals living or dead is purely coincidental.

12 Acknowledgement

I would like to thank Howard Brandt and Lov Grover for their helpful suggestions. I would also like to thank the individuals who attended my talk. Their many comments and insights were of invaluable help in writing this paper. Thanks are also due to the NIST Computer Security Division for providing an encouraging environment in which this paper could be completed.

References

1. Bennett, Charles H., and Gilles Brassard, *Quantum cryptography: Public key distribution and coin tossing*, International Conference on Computers, Systems & Signal Processing, Bagalore, India, December 10-12, 1984, pp 175 - 179.
2. Biham, Eli, and Adi Shamir, **Differential Crytanalysis of the Data Encryption Standard**, Springer-Verlag (1993).
3. Brandt, Howard E., John M. Meyers, And Samuel J. Lomonaco,Jr., *Aspects of entangled translucent eavesdropping in quantum cryptography*, Phys. Rev. A, Vol. 56, No. 6, December 1997, pp. 4456 - 4465.
4. Diffie, W., *The first ten years in public-key cryptography*, in **Contemporary Cryptology: The Science of Information Integrity**, pp 135 - 175, IEEE Press (1992).
5. Diffie, W., and M.E. Hellman, *New directions in cryptography*, IEEE Tranactions on Information Theory, **22** (1976), pp 644 - 654.
6. Franson, J.D., and H. Ilves, *Quantum cryptography using polarization feedback,* Journal of Modern Optics, Vol. 41, No. 12, 1994, pp 2391 - 2396.
7. Jacobs, B.C. and J.D. Franson, *Quantum cryptography in free space*, Optics Letters, Vol. 21, November 15, 1996, p1854 - 1856.
8. Lo, H.-K, and H.F. Chau, *Quantum computers render quantum key distribution unconditionally secure over arbitrarily long distance*, quant-ph/9803006.
9. Lomonaco, Samuel J., *A quick glance at quantum cryptography*, Cryptologia, Vol. 23, No. 1, January, 1999, pp1-41. (quant-ph/9811056)
10. Matsui, Mitsuru, *Linear crytanalysis method for DES cipher*, Lecture Notes in Computer Science, vol. 765, edited by T. Helleseth, Springer-Verlag (1994), pp386-397.
11. Mayers, Dominic, Crypto'96, p343.
12. Mayers, Dominic, and Andrew Yao, *Quantum cryptography with imperfect apparatus*, quant-ph/9809039.
13. Mayers, Dominic, *Unconditional security in quantum cryptography*, quant-ph/9802025.
14. Menezes, Alfred J., Paul C. van Oorschot, and Scott A. Vanstone, **Handbook of Applied Cryptography**, CRC Press (1977).
15. O'Reilly, Tim, and the Electronic Frontier Foundation, **Cracking DES: Secrets of Encryption Research, Wiretap Politics & Chip Design**, (1st Edition), July 1998 (US) ISBN 1-56592-520-3 (272 pages) `http://www.ora.com/catalog/crackdes/`
16. Phoenix, Simon J., and Paul D. Townsend, *Quantum cryptography: how to beat the code breakers using quantum mechanics*, Comtemporay Physics, vol. 36, No. 3 (1995), pp 165 - 195.

17. Townsend, P.D., *Secure key distribution system based on quantum cryptography*, Electronic Letters, 12 May 1994, Vol. 30, No. 10, pp 809 - 811.
18. Townsend, Paul D., and I Thompson, *A quantum key distribution channel based on optical fibre*, Journal of Modern Optics, Vol. 41, No. 12, 1994, pp 2425 - 2433.
19. Townsend, P.D., J.G. Rarity, and P.R. Tapster, *Single photon interference in 10km long optical fibre interferometer,* Electronic Letters, **29** (1993), pp 634 - 635.
20. Townsend, P.D., Nature 385, p47.
21. Wooters, W.K., and W.H. Zurek, *A single quantum cannot be cloned,*" Nature, 299 (1982), pp982-983.

`email: Lomonaco@UMBC.EDU`

The Rigidity Theorems of Hamada and Ohmori, Revisited

T. S. Michael

Mathematics Department
United States Naval Academy
Annapolis, MD 21402

Abstract Let A be a $(0,1)$-matrix of size b by v with $b \geq v$. Suppose that all rows (columns) of A are nonzero and distinct. We show that the rank of A over a field of characteristic 2 satisfies

$$\operatorname{rank}_2(A) \geq \log_2(v+1)$$

with equality if and only if A is the incidence matrix of a point-hyperplane Hadamard design. This generalizes a rigidity theorem of Hamada and Ohmori, who assumed that $v+1$ is a power of 2 and that A is already known to be the incidence matrix of a Hadamard design. Our results follow from a generalization of a rank inequality of Wallis.

1 Introduction and Definitions

In this paper we observe that some inequalities in the literature on the ranks of incidence matrices of Hadamard designs apply to a wider class of matrices and that the proofs are somewhat simpler in the more general context. In particular, we show that an analysis of the case of equality of an extension of a rank inequality of Wallis yields generalizations of two rigidity theorems of Hamada and Ohmori.

Let A be a rectangular matrix with entries in a field F. Let $\operatorname{rank}_F(A)$ denote the rank of A over F. The **p-rank** of A is the rank of A over the field Z_p of integers modulo p and is denoted by

$$\operatorname{rank}_p(A).$$

Let us say that the matrix A is **column-projective** over F provided its columns are nonzero and no column is a multiple of another column. Also, A is **projective** provided both A and its transpose A^T are column-projective. Suppose that A is $(0,1)$-matrix, that is, a matrix with each element in the set $\{0,1\}$. Then A is column-projective if and only if its columns are distinct and nonzero; this property is independent of the field F.

Combinatorial interest in p-ranks stems in part from their use in the study of linear codes associated with incidence matrices of block designs. Let F_q denote a field with q elements, where q is a power of the prime p. Let A

be the incidence matrix of a square balanced incomplete block design with parameters (v, k, λ). Thus A is a $(0,1)$-matrix of size v by v that satisfies

$$A^T A = (k - \lambda)I + \lambda J \qquad \text{and} \qquad AJ = kJ,$$

where J denotes an all 1's matrix of an appropriate size. The codewords of the q-ary linear code $\mathcal{C}_q(A)$ generated by A are the linear combinations over F_q of the row vectors of A. Clearly the length and dimension of $\mathcal{C}_q(A)$ are v and $\text{rank}_p(A)$, respectively. One may show that $\mathcal{C}_q(A)$ corrects $\lfloor k/2\lambda \rfloor$ errors [2]. In this context it is natural to assume that A is (column)-projective.

2 Two Rigidity Theorems

We now state our generalizations of two rigidity theorems of Hamada and Ohmori. The proofs appear in § 4. The point-hyperplane design $\mathbf{D}_s$ in the projective geometry $\text{PG}(s-1, 2)$ and its complement $\overline{\mathbf{D}}_s$ play special roles in these theorems. We recall some basic properties of these designs in § 3.

Theorem 1. *Let A be a projective $(0,1)$-matrix of size b by v with $b \geq v$. Then*

$$\text{rank}_2(A) \geq \log_2(v+1) \tag{1}$$

with equality if and only if $b = v$ and A is an incidence matrix of the complement $\overline{\mathbf{D}}_s$ of the point-hyperplane design, where $s = \log_2(v+1)$.

Remark. Hamada and Ohmori [1] established Theorem 1 under much stronger hypotheses; they assumed that A is the incidence matrix of a square block design with parameters of the form

$$(v, k, \lambda) = (2^s - 1, 2^{s-1}, 2^{s-2}), \tag{2}$$

that is, a **Hadamard design.** Their proofs seemingly depend on deeper results on the ranks of incidence matrices of block designs associated with projective geometries. We remove these apparent dependencies and show that a specific design-theoretic structure is *forced* when equality holds in (1). This characterization of a combinatorial structure from the value of a single parameter is the hallmark of a rigidity theorem.

Theorem 2. *Let A be a $(0,1)$-matrix of size b by v with $b \geq v$. Suppose that $J - A$ is projective, b is odd, and each row and column of A has an odd number of 1's. Then*

$$\text{rank}_2(A) \geq \log_2(v+1) \; + \; 1 \tag{3}$$

with equality if and only if $b = v$ and A is an incidence matrix of the point-hyperplane design $\mathbf{D}_s$, where $s = \log_2(v+1)$.

Remark. Hamada and Ohmori [1] established Theorem 2 under the stronger hypothesis that A is the incidence matrix of the complement of a Hadamard design with parameters of the form (2).

3 A Rank Inequality

Theorem 3. *Let A be a column-projective $(0,1)$-matrix with v columns. Then over any field F*

$$\operatorname{rank}_F(A) \geq \log_2(v+1). \tag{4}$$

Equality holds if and only if A has a column-projective submatrix of size s by $2^s - 1$, where $s = \operatorname{rank}_F(A)$.

Remark. Wallis [3] used an elaborate inductive scheme of row and column operations to prove an inequality equivalent to (4) under much stronger hypotheses; he assumed that A is a $(0,1)$-matrix of size v by v that is nonsingular over the field of rationals. (Also see his book [4], pp 168-170.) Moreover, he did not characterize equality. Our direct proof is based on a counting argument and leads to the stated characterization of equality, which in turn leads to proofs of Theorems 1 and 2.

Proof. Let $s = \operatorname{rank}_F(A)$. Without loss of generality the leading s by s submatrix of A has rank s. Then two columns of A are distinct and nonzero if and only if they are distinct and nonzero in their leading s positions. There are exactly $2^s - 1$ nonzero column vectors of 0's and 1's with s components. Thus $v \leq 2^s - 1$. This proves (4) with the stated characterization of equality. □

4 The Extremal Designs

In this section we recall properties of the extremal designs that arise in Theorems 1 and 2. Let $\overline{\mathbf{D}}_s$ denote the complement of the point-hyperplane design in the projective geometry PG$(s-1,2)$. Let $\overline{A}_s$ be the incidence matrix of $\overline{\mathbf{D}}_s$. The columns of $\overline{A}_s$ correspond to the $2^s - 1$ nonzero vectors (points) in an s-dimensional vector space over Z_2, while the rows correspond to the $2^s - 1$ complements of the $(s-1)$-dimensional subspaces (blocks). Containment defines incidence in $\overline{D}_s$. Without loss of generality

$$\overline{A}_s = \left[\begin{array}{c|c} N_s & M \\ \hline * & * \end{array}\right],$$

where the leading s by s submatrix N_s is nonsingular. The columns of the submatrix $[N_s|M]$ consist of all $2^s - 1$ nonzero linear combinations of the columns of N_s. Now the symmetric difference of two blocks of $\overline{\mathbf{D}}_s$ is also a block; this is a defining property of the design $\overline{\mathbf{D}}_s$. It follows that the rows of $\overline{A}_s$ are the nonzero linear combinations of the rows of $[N_s|M]$. One may verify that $\overline{A}_s$ is a $(0,1)$-matrix of size $2^s - 1$ by $2^s - 1$ that satisfies

$$A^T A = 2^{s-2}(I+J) \qquad \text{and} \qquad AJ = 2^{s-1}J,$$

and thus that $\overline{\mathbf{D}}_s$ is indeed a square block design with parameters

$$(v, k, \lambda) = (2^s - 1, 2^{s-1}, 2^{s-2}).$$

Clearly the incidence matrix $\overline{A}_s$ is determined by the parameter s up to row and column permutations. Also,

$$\text{rank}_2(\overline{A}_s) = s.$$

The complementary design $\mathbf{D}_s$ has incidence matrix $J - \overline{A}_s$ and parameters

$$(v, k, \lambda) = (2^s - 1, 2^{s-1} - 1, 2^{s-2} - 1).$$

Both $\overline{\mathbf{D}}_s$ and $\mathbf{D}_s$ satisfy $v = 4(k - \lambda) - 1$ and hence are Hadamard designs.

5 Proofs of Theorems 1 and 2

Proof of Theorem 1. Apply Theorem 3 with $F = Z_2$ to deduce that $\text{rank}_2(A) \geq \log_2(v + 1)$. Suppose that equality holds, say, $v = 2^s - 1$, where $s = \text{rank}_2(A)$. Then A has a column-projective submatrix of size s by $2^s - 1$. The projectivity of A and the inequality $b \geq v$ imply that $b = v = 2^s - 1$, and thus the rows and columns of A are determined up to permutations, as in the discussion of the matrix A_s in § 4; the characterization of equality follows. □

Theorem 2 follows immediately from Theorem 1 and the following lemma.

Lemma. *Let A be a $(0,1)$-matrix of size b by v. Suppose that b is odd and that each row and column of A has an odd number of 1's. Then*

$$\text{rank}_2(A) = \text{rank}_2(J - A) \ + \ 1.$$

Proof. The hypotheses imply that we may transform A as follows without altering its 2-rank: Append a column of 1's, and then append a row of 1's to the resulting matrix to obtain a bordering of A of size $b+1$ by $v+1$. Now add column $v + 1$ to each of the first v columns, and then subtract row $b + 1$ from each of the first b rows. The resulting matrix is the direct sum $(J + A) \oplus [1]$, which clearly has 2-rank equal to $\text{rank}_2(J - A) + 1$. □

6 The Smith Normal Form

Let A be an integral matrix of size b by v and rank r over the field of rationals. Then A may be transformed by elementary row and column operations to a diagonal matrix

$$S_A = \text{diag}[a_1, a_2, \ldots, a_r, 0, \ldots, 0],$$

known as the **Smith normal form** of A, with the property that a_i divides a_{i+1} for $i = 1, \dots, r-1$. The diagonal elements $a_1, \dots, a_r, 0, \dots, 0$ are the **invariant factors** of A and are uniquely determined up to sign. The p-rank of A is related to the invariant factors $a_1, \dots, a_r$ in a simple manner:

$$\mathrm{rank}_p(A) = \max\{i : \ p \text{ does not divide } a_i\}. \tag{5}$$

Our generalization in Theorem 3 of a result of Wallis leads directly to the following two theorems, which extend his work in [3], [4]. The first is an immediate consequence of (5) and Theorem 3.

Theorem 4. *The invariant factors of a column-projective* $(0,1)$*-matrix* A *with* v *columns satisfy* $a_1 = \cdots = a_s = 1$ *for some* $s \geq \log_2(v+1)$.

Theorem 5. *Let* H *be a* $(1,-1)$*-matrix with* v *columns, none of which is a multiple of any other. Then for some* $s \geq \log_2(v)$ *the Smith normal form of* H *is of the form*

$$S_H = \mathrm{diag}[1, \overbrace{2, \dots, 2}^{s}, *, \dots, *].$$

Proof. We may multiply suitable rows and columns of H by -1 so that all elements in the first row and column are 1. Now subtract column 1 from all other columns, and then subtract row 1 from all other rows to transform H to a matrix $[1] \oplus (2A)$, where A is a $(0,1)$-matrix with $v-1$ columns. The hypothesis on the columns of H implies that A is a column-projective, and the result follows from Theorem 4. □

References

1. N. Hamada and H. Ohmori, On the BIB design having the minimum p-rank, *J. Combin. Theory, Ser. A*, **18** (1975), 131-140.
2. L. D. Rudolph, A class of majority logic decodable codes, *IEEE Trans. Information Theory*, IT-**13** (1967), 305-307.
3. W. D. Wallis, Integral equivalence of Hadamard matrices, *Israel J. Math.*, **10** (1971), 349-358.
4. W. D. Wallis, **Combinatorial Designs.** Marcel Dekker, New York, (1988).

email: tsm@nadn.navy.mil

Counting Prime Divisors on Elliptic Curves and Multiplication in Finite Fields

M. Amin Shokrollahi

Bell Labs, Rm. 2C-353, 700 Mountain Ave, Murray Hill, NJ 07974, USA

Abstract. Let $K/\mathbb{F}_q$ be an elliptic function field. For every natural number n we determine the number of prime divisors of degree n of $K/\mathbb{F}_q$ which lie in a given divisor class of K.

1 Introduction

If K is a number field it is well known that there exist infinitely many prime divisors which belong to a fixed divisor class and that the (Dirichlet-)density of the set of such prime divisors is $1/h$ where h is the class number of K [6, §9]. The proof of this theorem is similar to the proof of the theorem on the number of primes in an arithmetic progression and proceeds roughly as follows: One first separates the prime divisors in the different classes by the characters of the class group and uses then orthogonality relations of the characters together with the non-vanishing of the L-series at 1 (which constitutes the deep part of the theorem). In fact, one of the approaches to class field theory is based on the investigation of the Dirichlet-density of the set of prime divisors lying in a given (ray-) class [6].

If K is an algebraic function field over a finite field, then we might similarly ask for the number of prime divisors belonging to a fixed class. In this situation there exist only finitely many prime divisors in each class since there are only finitely many integral divisors in each divisor class. However, as it will be shown in this paper, the method outlined above will prove successful in answering this question for the case K is an elliptic function field over the finite field $\mathbb{F}_q$.

Let $K/\mathbb{F}_q$ be an elliptic function field and C be a divisor class of K. Denoting by $a(C)$ the number of prime divisors in the class C, we are thus asking for the exact value of $a(C)$ for all C. If C is a class of degree one for example, $a(C) = 1$, since no two distinct prime divisors of degree one are equivalent in K.

The group $\mathcal{C}$ of divisor classes of K is the direct product of the group $\mathcal{C}_0$ of divisor classes of degree zero and the infinite cyclic group generated by $[Q]$ where $[Q]$ is the class of an arbitrary divisor Q of degree one [5, pp. 64]. Let us define $a_{n,Q}(C_0)$ for a class C_0 of degree zero and an integer n by

$$a_{n,Q}(C_0) := a(C_0[Q]^n).$$

For the ease of notation we shall suppress the dependency of $a_{n,Q}$ on Q and write simply a_n instead. We can thus equivalently ask for the value of $a_n(C_0)$ for all n and all classes C_0 of degree zero (where we can of course confine ourselves to the case $n \geq 1$). This question will be answered in this paper (Theorem 6).

Our question can be translated to the language of elliptic curves: let E be an elliptic curve over the field $\mathbb{F}_q$. Denote by $E(\mathbb{F}_{q^n})$ the group of $\mathbb{F}_{q^n}$-rational points of E. Let σ denote the Frobenius automorphism of $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$. We define the trace map by

$$\begin{aligned}\mathrm{Tr}\colon\ E(\mathbb{F}_{q^n}) &\to E(\mathbb{F}_q)\\ P &\mapsto \textstyle\sum_{i=0}^{n-1} P^{\sigma^i}\end{aligned}$$

where $\sum$ denotes the summation in the group $E(\mathbb{F}_q)$. Denote the restriction of Tr on $E(\mathbb{F}_{q^n})\setminus\cup_{d|n,\, d<n}E(\mathbb{F}_{q^d})$ by tr. Suppose that Q is the neutral element of $E(\mathbb{F}_q)$. Let C be an arbitrary class of degree 0 of the function field K of E. As a consequence of the Riemann-Roch theorem there exists a unique $P \in E(\mathbb{F}_q)$ such that $C = [P - Q]$. Then $a_n(C)$ is equal to the cardinality of the fiber of tr at P.

The method for obtaining a formula for $a_n(C)$ (which resembles Dirichlet's proof of the existence of infinitely many primes in an arithmetic progression) can be described as follows: First we introduce appropriate characters of the class group $\mathcal{C}$ of $K/\mathbb{F}_q$ which will "separate" the prime divisors belonging to different classes. Then the corresponding L-functions are constructed. Taking logarithms of the L-functions and applying the inversion formula for the characters we will be able to obtain a recursion formula for $a_n(C)$ (Section 2). Next we apply the principle of inclusion and exclusion to solve the recursion (Section 3). The final formula for $a_n(C)$ involves the numbers Π_d of prime divisors of degree d of $K/\mathbb{F}_q$ for different d (or equivalently the numbers N_d of divisors of degree one of $K\mathbb{F}_{q^d}/\mathbb{F}_{q^d}$) and the number of classes C' some power of which equal C (Theorem 6). The next sections deal with the problem of determining extremal values of a_n for given n. It turns out that a_n is constant if n and the number h of classes of degree 0 of K, are coprime (Lemma 7). Further a_n attains its minimum value at $\mathcal{H}$, the principal class of K (Theorem 16). The techniques developed in Section 5 can be utilized to prove several results on the distribution of the numbers $a_n(C_0)$ for a fixed n. We have confined ourselves to mention some of the more interesting results (compare also Theorems 27 and 29).

The interest of the author in the numbers $a_n(C)$ arouse in the context of optimal bilinear algorithms for multiplication in finite fields. For instance, the results of this paper have been used in [9] to construct an efficient randomized algorithm which produces optimal bilinear algorithms for multiplication in certain finite fields (see also [10]). An example in this direction is given in the final section of the paper.

2 L-Functions

Let $K/\mathbb{F}_q$ be an elliptic function field. By $\mathcal{C}$ we denote the class group and by $\mathcal{C}_0$ the group of classes of degree 0 of $K/\mathbb{F}_q$. Let P be a prime divisor of degree one of $K/\mathbb{F}_q$ and denote by $[P]$ its class. It is well known (see the introduction) that $\mathcal{C} = \mathcal{C}_0 \times [P]^{\mathbb{Z}}$. Any character χ of $\mathcal{C}_0$ can be extended to a character $\tilde{\chi}$ of $\mathcal{C}$ by setting $\tilde{\chi}(A) := \chi(A - \deg(A)P)$. By abuse of notation we shall denote $\tilde{\chi}$ by χ.

The *L-function* of a character χ of $\mathcal{C}$ is defined by

$$L(s,\chi) := \sum_{A} \chi(A)N(A)^{-s}$$

where the sum is over all integral divisors of $K/\mathbb{F}_q$, and $N(A)$ denotes the norm of the divisor A, i.e., $N(A) = q^{\deg(A)}$. $L(s,\chi)$ has an *Euler-product-expansion* [5, §24]

$$L(s,\chi) = \prod_{P}(1 - \chi(P)N(P)^{-s})^{-1} \tag{1}$$

where the product extends over all prime divisors of $K/\mathbb{F}_q$. If ε denotes the principal character of $\mathcal{C}$, we call $L(s,\varepsilon)$ the *ζ-function* of $K/\mathbb{F}_q$ and denote it by $\zeta(s)$.

For the rest of this paper we assume that $\mathrm{Re}(s) > 1$ which implies that the series encountered converge absolutely [5, Lecture 11].

While the ζ-function of an elliptic function field plays a great role in the arithmetic theory, the L-functions attached to extensions of non-principal characters of $\mathcal{C}_0$ to C are trivial:

Lemma 1. *Let $K/\mathbb{F}_q$ be an elliptic function field. Further let χ be a non-principal character of $\mathcal{C}_0$. Then $L(s,\chi) = 1$.*

Proof. In [5, pp. 66, §25] it is proved that

$$(q-1)L(s,\chi) = \sum_{C\in\mathcal{C}_0} \chi(C)q^{\dim(C)}.$$

Now observe that $\dim(C) = 0$ if C is not the principal class and $\dim(C) = 1$ if C is the principal class. Since χ is not the principal character, we have $\sum_{C\in\mathcal{C}_0} \chi(C) = 0$, hence

$$(q-1)L(s,\chi) = q - 1$$

which yields the assertion. □

In order to get a formula for the numbers $a_n(C)$ we first take the logarithm of $L(s,\chi)$ using formula (1):

$$\log L(s,\chi) = \sum_{P}\sum_{m\geq 1} \frac{\chi(P)^m}{mN(P)^{-sm}}.$$

Taking into account that $N(P) = q^{\deg(P)}$ this yields

$$\log L(s,\chi) = \sum_{P}\sum_{m\geq 1}\frac{\chi(P)^m}{m}u^{\deg(P)m}$$

where we have followed the customary convention $u := q^{-s}$. Now we divide the above sum into sums over prime divisors belonging to a fixed class:

$$\log L(s,\chi) = \sum_{C\in\mathcal{C}}\sum_{\substack{P\\ [P]=C}}\sum_{m\geq 1}\frac{\chi(C)^m}{m}u^{\deg(P)m}.$$

The isomorphism $\mathcal{C} = \mathcal{C}_0 \times [Q]^{\mathbb{Z}}$ allows to classify the classes according to their degree:

$$\log L(s,\chi) = \sum_{C\in\mathcal{C}_0}\sum_{n\geq 1}\sum_{m\geq 1}a_n(C)\frac{\chi(C)^m}{m}u^{nm}.$$

The above sum is a power series in u. A trivial computation yields the following normal representation of this power series:

$$\log L(s,\chi) = \sum_{\nu\geq 1}\frac{1}{\nu}\Big(\sum_{d|\nu}\sum_{C\in\mathcal{C}_0}a_d(C)\,d\,\chi(C)^{\frac{\nu}{d}}\Big)u^{\nu}. \qquad (2)$$

Now let $C_0 \in \mathcal{C}_0$ be a fixed class and X denote the character group of $\mathcal{C}_0$. We have:

$$\sum_{\chi\in X}\chi(C_0^{-1})\log L(s,\chi) = \sum_{\nu\geq 1}\frac{1}{\nu}\Big(\sum_{d|\nu}\sum_{C\in\mathcal{C}_0}a_d(C)d\sum_{\chi\in X}\chi(C_0^{-1}C^{\frac{\nu}{d}})\Big)u^{\nu}.$$

It is well known that if a is an element of a finite abelian group A and X denotes the character group of A, then $\sum_{\chi\in X}\chi(a) = 0$ if a is not equal to the identity-element of A, whereas this sum equals the cardinality of A if a is the identity element of A. Applying this we get

$$\sum_{\chi\in X}\chi(C_0^{-1})\log L(s,\chi) = \sum_{\nu\geq 1}\frac{1}{\nu}\Big(\sum_{d|\nu}\sum_{\substack{C\in\mathcal{C}_0\\ C^{\nu/d}=C_0}}a_d(C)\,d\,h\Big)u^{\nu} \qquad (3)$$

where $h = |\mathcal{C}_0|$ is the number of classes of degree 0 of $K/\mathbb{F}_q$ (i.e., the number of $\mathbb{F}_q$-rational points of the corresponding elliptic curve).

By Lemma 1, $\log L(s,\chi) = 0$ for non-principal χ. Hence, taking into account Equation (2) and the fact that $\sum_{C_0\in\mathcal{C}_0}a_n(C_0) = \Pi_n$, the above sum equals

$$\log L(s,\varepsilon) = \sum_{\nu\geq 1}\frac{1}{\nu}\Big(\sum_{d|\nu}\Pi_d d\Big)u^{\nu}. \qquad (4)$$

Since the right hand sides of Equation (3) and Equation (4) are equal power series we get

$$\forall C_0 \in \mathcal{C}_0 \, \forall \nu \geq 1 : \sum_{d|\nu} \Pi_d d = \sum_{d|\nu} \Big(\sum_{\substack{C \in \mathcal{C}_0 \\ C^{\nu/d} = C_0}} a_d(C) \Big) dh.$$

This proves the following recursion formula:

Lemma 2. *Let n be an arbitrary positive integer and C_0 be an arbitrary class of degree zero of the elliptic function field $K/\mathbb{F}_q$. We have*

$$a_n(C_0) = \frac{1}{h}\Pi_n + \sum_{\substack{d|n \\ d<n}} \Big(\frac{1}{h}\Pi_d - \sum_{\substack{C \in \mathcal{C}_0 \\ C^{n/d} = C_0}} a_d(C) \Big) \frac{d}{n}. \tag{5}$$

In the next section we shall solve this recursion.

3 Resolution of the Recursion

The principle of inclusion and exclusion is applied in this section to solve the recursion (5).

For $n \in \mathbb{N}$ and $C_0 \in \mathcal{C}_0$ define

$$A(d; n, C_0) = A(d) := \begin{cases} \Big(\frac{1}{h}\Pi_d - \sum_{\substack{C \in \mathcal{C}_0 \\ C^{n/d} = C_0}} a_d(C) \Big) \frac{d}{n} & \text{if } d|n \\ 0 & \text{otherwise.} \end{cases}$$

Further, let $f(m; n, C_0) = f(m) := \sum_{d|\gcd(m,n)} A(d)$. Our first aim is to prove the following:

Lemma 3. *If $m|n$, we have*

$$f(m) = \frac{1}{hn} \sum_{d|m} d\, \Pi_d \, (1 - \big|\{ C \mid C^{n/m} = C_0 \}\big|).$$

Proof. We have

$$f(m) = \Big(\frac{1}{h}\Pi_m - \sum_{\substack{C \in \mathcal{C}_0 \\ C^{n/m} = C_0}} a_m(C) \Big) \frac{m}{n} + \sum_{\substack{d|m \\ d<m}} \Big(\frac{1}{h}\Pi_d - \sum_{\substack{C \in \mathcal{C}_0 \\ C^{n/d} = C_0}} a_d(C) \Big) \frac{d}{n}.$$

By Equation (5) we get

$$\begin{aligned} f(m) &= \Big(\frac{1}{h}\Pi_m - \sum_{\substack{C\in\mathcal{C}_0\\ C^{n/m}=C_0}} \Big(\frac{1}{h}\Pi_m + \sum_{\substack{d\mid m\\ d<m}} \Big(\frac{1}{h}\Pi_d - \sum_{\substack{C'\in\mathcal{C}_0\\ C'^{m/d}=C}} a_d(C')\Big)\frac{d}{n}\Big)\Big)\frac{m}{n} \\ &\quad + \sum_{\substack{d\mid m\\ d<m}} \Big(\frac{1}{h}\Pi_d - \sum_{\substack{C\in\mathcal{C}_0\\ C^{n/d}=C_0}} a_d(C)\Big)\frac{d}{n} \\ &= \frac{1}{h}\sum_{d\mid m} d\,\Pi_d\,(1 - \big|\{\, C \mid C^{n/m} = C_0 \,\}\big|) \\ &\quad - \sum_{\substack{d\mid m\\ d<m}} \Big(\sum_{\substack{C\in\mathcal{C}_0\\ C^{n/m}=C_0}} \sum_{\substack{C'\in\mathcal{C}_0\\ C'^{m/d}=C}} a_d(C') - \sum_{\substack{C\in\mathcal{C}_0\\ C^{n/d}=C_0}} a_d(C)\Big). \end{aligned}$$

Now note that

$$\sum_{\substack{C\in\mathcal{C}_0\\ C^{n/m}=C_0}} \sum_{\substack{C'\in\mathcal{C}_0\\ C'^{m/d}=C}} a_d(C') = \sum_{\substack{C\in\mathcal{C}_0\\ C^{n/d}=C_0}} a_d(C).$$

Hence the assertion follows. □

Denote by N_m the number of prime divisors of degree one of $K\mathbb{F}_{q^m}/\mathbb{F}_{q^m}$. It is well known that $N_m = \sum_{d\mid m} \Pi_d d$. Thus the following corollary follows:

Corollary 4. *We have*

$$f(m) = \frac{1}{hn} N_m (1 - \big|\{\, C \mid C^{n/m} = C_0 \,\}\big|).$$

Now we apply the principle of inclusion and exclusion:

Lemma 5. *Let S be a finite set, $S_1, \dots, S_k$ subsets of S and $A\colon S \to \mathbb{Z}$ a mapping. For $T \subseteq S$ let $A_\Sigma(T) := \sum_{t\in T} A(t)$. Then we have*

$$\begin{aligned} A_\Sigma(S \setminus \cup_{i=1}^k S_i) = A_\Sigma(S) &- \sum_{i=1}^k A_\Sigma(S_i) + \sum_{1\le i<j\le k} A_\Sigma(S_i \cap S_j) \\ &- \sum_{1\le i<j<l\le n} A_\Sigma(S_i \cap S_j \cap S_l) + \cdots \\ &+ (-1)^k A_\Sigma(S_1 \cap S_2 \cap \cdots \cap S_k). \end{aligned}$$

Proof. This is a straightforward generalization of [1, Theorem 5.31]. □

Theorem 6. *Let $K/\mathbb{F}_q$ be an elliptic function field, $\mathcal{C}_0$ be the group of divisor classes of degree zero of K, $C_0 \in \mathcal{C}_0$ and n an integer greater or equal to zero.*

Then

$$a_n(C_0) = \frac{1}{h}\Big(\varPi_n - \frac{1}{n}\sum_{\substack{d|n\\ d<n}} \mu\Big(\frac{n}{d}\Big)\, N_d(1 - \big|\{\, C \mid C^{n/d} = C_0 \,\}\big|)\Big)$$
$$= \frac{1}{hn}\sum_{d|n} \mu\Big(\frac{n}{d}\Big)\, N_d\big|\{\, C \mid C^{n/d} = C_0 \,\}\big|.$$

Proof. Let $n = \prod_{i=1}^{k} p_i^{a_i}$ be the prime factor decomposition of n. Set $S := \{d \mid d|n\}$ and $S_i := \{d \mid d|\frac{n}{p_i}\}$ for $i = 1, \dots, k$. Then $S \setminus \cup_{i=1}^{k} S_i = \{n\}$. Applying Lemma 5 with $A(.) = A(.; n, C_0)$ we get

$$\frac{1}{h}\varPi_n - a_n(C_0) = \sum_{d|n} \mu\Big(\frac{n}{d}\Big)\, f(d).$$

Now note that $f(n) = 0$. So applying Lemma 3 we get the first equality. The second equality follows from the first by observing that application of Möbius-inversion to $\sum_{d|n} \varPi_d d = N_n$ yields $\varPi_n = \frac{1}{n}\sum_{d|n} N_d\, \mu(n/d)$. □

Before going into elaborate estimates of the above sums, let us derive first a simple lemma from Theorem 6. Let $C_0 \in \mathcal{C}_0$ and consider $a_1(C_0)$. As was remarked in the introduction $a_1(C_0) = 1 = \varPi_1/h$ for all C_0 in $\mathcal{C}_0$. Can we expect $a_n(C_0) = \varPi_n/h$ for all n and all $C_0 \in \mathcal{C}_0$ (at least as long as $\varPi_n$ is a multiple of h)? The following lemma gives a sufficient condition for this to be the case.

Lemma 7. *Suppose that n and h are coprime. Then $a_n(C_0) = \varPi_n/h$ for all $C_0 \in \mathcal{C}_0$.*

Proof. Since $(n, h) = 1$, the homomorphism $C_0 \mapsto C_0^n$ is an automorphism of $\mathcal{C}_0$. Hence $\big|\{\, C \mid C^{n/d} = C_0 \,\}\big| = 1$ for all $C_0 \in \mathcal{C}_0$. The assertion follows now from Theorem 6. □

The following example shows that the condition in the preceding lemma is not necessary:

Example 8. The elliptic function field $K = \mathbb{F}_4(x, y), y^2 + y = x^3 + 1$ has 9 prime divisors of degree one [2]. The group of divisors of degree 0 of K is easily computed to be the direct product of two cyclic groups of order 3 (note that $\mathcal{C}_0$ is isomorphic to the group of $\mathbb{F}_4$-rational points of the corresponding elliptic curve). K has $\varPi_6 = 648$ prime divisors of degree 6. Further, $N_2 = N_1 = 9$ and $|\{C \mid C^2 = C_0\}| = 1$ since $(2, 9) = 1$, and $|\{C \mid C^3 = C_0\}| = |\{C \mid C^6 = C_0\}|$ for all $C_0 \in \mathcal{C}_0$. Applying Theorem 6 we get

$$a_6(C_0) = \frac{1}{9}\varPi_6 = 72 \qquad \text{for all } C_0 \in \mathcal{C}_0.$$

Example 8 is actually an exception. In Section 5 we will prove a partial converse to Lemma 7 (see Theorem 29).

In the next sections we investigate the extremal values of the function a_n.

4 Some Tools

In this section we shall gather some well known results about abelian groups and elliptic function fields. This will serve as a toolbox for the computations in the next section.

To begin with, let us introduce a notation: If A is an abelian group and n an integer, we define $A^n := \{a^n \mid a \in A\}$. The following lemma is almost trivial:

Lemma 9. *Let A be a finite abelian group and n, m integers. Then $A^n \cap A^m = A^{\text{lcm}(n,m)}$ and $A^n A^m = A^{\text{gcd}(n,m)}$.*

Proof. Let A be the direct product of B and C. Then $A^n \cap A^m = (B^n \times C^n) \cap (B^m \times C^m)$, so $A^n \cap A^m = (B^n \cap B^m) \times (C^n \cap C^m)$. Analogously $A^n A^m = (B^n B^m) \times (C^n C^m)$. Since the assertion of the lemma is easily verified for cyclic groups, the general case follows by decomposition of A into cyclic factors. □

Let A be a finite abelian group, n an integer and $\pi_n^A \colon A \to A^n, a \mapsto a^n$.

Lemma 10. *If A is a finite abelian group, m, n are integers and π_n^A is as above, we have:*

$$|\ker \pi^A_{\text{lcm}(m,n)}| = \frac{|\ker \pi_n^A||\ker \pi_m^A|}{|\ker \pi^A_{\text{gcd}(n,m)}|}.$$

Proof. Application of Lemma 9 yields:

$$\begin{aligned} |\ker \pi^A_{\text{lcm}(m,n)}| &= \frac{|A|}{|A^{\text{lcm}(m,n)}|} \\ &= \frac{|A|}{|A^n|}\frac{|A^n|}{|A^{\text{lcm}(m,n)}|} \\ &= \frac{|A|}{|A^n|}\frac{|A^{\text{gcd}(m,n)}|}{|A^m|} \\ &= \frac{|A|}{|A^n|}\frac{|A|}{|A^m|}\frac{|A^{\text{gcd}(n,m)}|}{|A|} \\ &= \frac{|\ker \pi_n^A||\ker \pi_m^A|}{|\ker \pi^A_{\text{gcd}(n,m)}|}. \ \square \end{aligned}$$

We immediately get the following corollaries whose proofs are obvious.

Corollary 11. *If A is a finite abelian group and m, n are coprime integers, we have*

$$|\ker \pi^A_{mn}| = |\ker \pi_n^A||\ker \pi_m^A|.$$

Corollary 12. *Let A be a finite abelian group and n, m coprime integers. Then*

$$H_{A^{nm}} = H_{A^n} H_{A^m},$$

where H_M is denotes the characteristic function of the set M.

Now we want to investigate some elementary problems related to elliptic function fields.

Lemma 13. *Let $K/\mathbb{F}_q$ be an elliptic function field, $q \geq 5$, and n, k be positive integers, $k \geq 2$. Then*

$$q^{n(k-2)} N_n \leq N_{nk} \leq q^{nk} N_n.$$

Proof. We apply the well known *Hasse-Weil-inequality*

$$|N_n - q^n - 1| \leq 2\sqrt{q^n}$$

to get

$$\frac{(\sqrt{q^{nk}} - 1)^2}{(\sqrt{q^n} + 1)^2} \leq \frac{N_{nk}}{N_n} \leq \frac{(\sqrt{q^{nk}} + 1)^2}{(\sqrt{q^n} - 1)^2}$$

Now note that if $a \geq \sqrt{5}$ is a real number and k is as above, we have

$$\frac{(a^k - 1)^2}{(a+1)^2} \geq a^{2(k-2)}$$
$$\frac{(a^k + 1)^2}{(a-1)^2} \leq a^{2k}.$$

Putting $a = \sqrt{q^n}$ we get the assertion. □

Remarks. (1) The first inequality in the above lemma is also valid for $q = 4$. It is even sharp for $q = 4, n = 1, k = 2$ (as can be seen in the case of the function field $K = \mathbb{F}_4(x, y)$, $x^3 + y^3 = 1$).

(2) The inequalities given are very crude for big q. Nevertheless we shall not need more refined estimates for the computations in the next section.

Now let $K/\mathbb{F}_q$ be an elliptic function field and $\mathcal{C}_0$ denote the group of divisor classes of degree 0 of K. For a nonnegative integer n we denote the homomorphism $\pi_n^{\mathcal{C}_0}$ simply by π_n. The kernel of π_n (also called the *group of n-division points*) plays an important role in the formulas of Theorem 6 as is apparent from the following

Lemma 14. *Let $C_0 \in \mathcal{C}_0$ and n be a positive integer. Then*

$$|\{C \mid C^n = C_0\}| = H_{\mathcal{C}_0^n}(C_0) |\ker \pi_n|.$$

Proof. Trivial. □

The order of $\ker \pi_n$ depends on n and the structure of $\mathcal{C}_0$. However, since $\mathcal{C}_0$ is always of the type $C_l \times C_m$ with $l|m$ (See e.g. [8]), we get the following (well known) estimate:

Lemma 15. *With the above notations we have* $|\ker \pi_n| \leq n^2$.

Proof. If $\mathcal{C}_0 = C_l \times C_m$ we have $|\ker \pi_n| = \gcd(n,l)\gcd(n,m) \leq n^2$. □

With these tools at hand, we are now able to derive some lower and upper bounds for the numbers $a_n(C_0)$. This will be done in the next section.

5 Some Estimates for $a_n(C_0)$

The aim of this section is to prove the following

Theorem 16. *Let $K/\mathbb{F}_q$ be an elliptic function field, $q \geq 7$, and n be an integer satisfying $n \leq 2^{q^{12}}$. Denote by $\mathcal{C}_0$ the group of divisor classes of degree 0 and by $\mathcal{H}$ the principal class of K. For $C_0 \in \mathcal{C}_0$ let $a_n(C_0)$ be defined as above. Then we have*

(1) $a_n(\mathcal{H}) = \min_{C_0 \in \mathcal{C}_0} a_n(C_0)$.
(2) *Let $\overline{n}$ denote the squarefree part of n. Then $a_n(C_0) = a_n(\mathcal{H})$ if and only if $C_0 \in \mathcal{C}_0^{\overline{n}}$.*
(3) *Suppose there exists $C_0 \in \mathcal{C}_0$ such that $C_0 \notin \mathcal{C}_0^p$ for all $p|n$. Then $a_n(C_0) = \max_{C_0 \in \mathcal{C}_0} a_n(C_0) = \frac{1}{hn}N_n$.*

Before starting with the proof of this theorem, let us state an immediate corollary:

Corollary 17. *With the same notations as above we have $a_n(\mathcal{H}) \leq \Pi_n/h$.*

Proof. We have

$$\Pi_n = \sum_{C_0 \in \mathcal{C}_0} a_n(C_0) \geq \sum_{C_0 \in \mathcal{C}_0} a_n(\mathcal{H}) = h a_n(\mathcal{H}). \ \square$$

The proof of Theorem 16 requires some preliminary discussions. It is based on the investigation of $a_n(\mathcal{H}) - a_n(C_0)$ for arbitrary $C_0 \in \mathcal{C}_0$. Application of Theorem 6 and Lemma 14 yields the following formula for this difference

$$a_n(\mathcal{H}) - a_n(C_0) = \frac{1}{hn} \sum_{d|n} N_d \, \mu\Big(\frac{n}{d}\Big) \, |\ker \pi_{n/d}| (1 - H_{\mathcal{C}_0^{n/d}}(C_0)). \tag{6}$$

Let us agree upon the following notations for the rest of this section:

Notation 51. *$K/\mathbb{F}_q$ is always assumed to be an elliptic function field. $\mathcal{C}_0$ is the group of divisor classes of degree 0 of K and h denotes its order. For an integer m, N_m is the number of prime divisors of degree one of $K\mathbb{F}_{q^m}/\mathbb{F}_{q^m}$ and Π_m is the number of prime divisors of degree m of K; the homomorphism $\pi_m^{\mathcal{C}_0}$ of the last section is simply denoted by π_m.*

n is always a positive integer which satisfies $n \leq 2^{q^{12}}$ (for technical reasons). $\mathrm{P} = \{p_1, \dots, p_r\}$ *is the set of* distinct *prime divisors of n; for $0 \leq l \leq r-1$ we set $v_l := p_1 \dots p_l$ and $u_l := p_{l+1} \dots p_r$ (note that $v_0 = 1$). If $P \subseteq \mathrm{P}$ we denote by n_P the number $\prod_{q \in P} q$ ($n_\emptyset = 1$). For a non-negative integer i, $\binom{\mathrm{P}}{i}$ denotes the set of subsets of* P *of order i.*

The following lemma is the heart of the estimates following.

Lemma 18. *Let m be an integer such that $q^m \geq 7$. Further let l be an integer satisfying $0 \leq l \leq r-2$. Then we have*

$$\sum_{k=l+1}^{r} q^{mu_l/p_k} p_k^2 < q^{m(u_l-2)}.$$

Proof. Let us first replace q^m by t. Observe that the function $f(x) = t^{a/x}x^2$ decreases monotonically for $x < a\ln(t)/2$, hence also for $x \leq a$ (note that $t \geq 7$). The proof is divided in two cases:

Case 1. $l \leq r-3$. In this case $u_l \geq 2 \cdot 3 \cdot 5 = 30$. Further $p_k < u_l/2 < u_l$, hence $t^{u_l/p_k}p_k^2 \leq 4t^{u_l/2}$ by the above observation. So we get

$$\begin{aligned}
\sum_{k=l+1}^{r} t^{u_l/p_k} p_k^2 &\leq 4rt^{u_l/2} \\
&\leq 4\log_2(n)t^{u_l/2} \\
&\leq 4q^{12}t^{u_l/2} \\
&\leq q^{13}t^{u_l/2} \quad (q \geq 7) \\
&\leq t^{u_l-2} \quad (t \geq q, u_l \geq 30).
\end{aligned}$$

Case 2. $l = r-2$. This condition implies $u_l \geq 2 \cdot 3 = 6$. Hence we get

$$\begin{aligned}
\sum_{k=l+1}^{r} t^{u_l/p_k} p_k^2 &\leq 4t^{u_l/2} + 9t^{u_l/3} \\
&= t^{u_l/2}(4 + 9t^{-u_l/6}) \\
&\leq t^{u_l/2}(4 + \frac{9}{t}) \quad (u_l \geq 7) \\
&< t^{u_l/2+1} \quad (t \geq 7) \\
&\leq t^{u_l-2} \quad (u_l \geq 7). \; \square
\end{aligned}$$

Lemma 19. *Let $q \geq 7$.*

(1) *If* $0 \le l \le r-2$, *we have*

$$-N_{n/v_l} + \sum_{k=l+1}^{r} N_{n/(v_l p_k)} p_k^2 < 0.$$

(2) $-N_{n/v_{r-1}} + N_{n/v_r} p_r^2 < 0$.

Proof. (1) Applying Lemma 13 we get

$$-N_{n/v_l} + \sum_{k=l+1}^{r} N_{n/(v_l p_k)} p_k^2 \le N_{n/v_r}\Big(-q^{\frac{n}{v_r}(u_l-2)} + \sum_{k=l+1}^{r} q^{\frac{n u_l}{v_r p_k}} p_k^2\Big).$$

The right hand side of the inequality is less than zero by Lemma 18.

(2) Let $\mu := v_r$. Then $N_{n/v_{r-1}} = N_{\mu p_r}$. Applying the Hasse-Weil-inequality we obtain

$$\begin{aligned}
\frac{N_{\mu p_r}}{N_\mu} &\ge \frac{q^{\mu p_r} + 1 - 2q^{\mu p_r/2}}{q^\mu + 1 + 2q^{\mu/2}}\\
&> \frac{q^{\mu(p_r-1)} - 2q^{\mu(p_r/2-1)}}{1 + q^{-\mu} + 2q^{-\mu/2}}\\
&\ge \frac{2}{3} q^{\mu(p_r/2-1)}(q^{\mu p_r/2} - 2) \qquad (q \ge 7)\\
&\ge \frac{2}{3}(q^{p_r} - 2) \qquad (\mu \ge 2)\\
&\ge p_r^2 \qquad (q \ge 5). \ \square
\end{aligned}$$

Lemma 20. *Let* $q \ge 7$ *and* $0 \le l \le r-1$. *Then*

$$-N_{n/v_l} |\ker \pi_{v_l}| + \sum_{k=l+1}^{r} N_{n/(v_l p_k)} |\ker \pi_{v_l p_k}| < 0.$$

Proof. We have

$$\begin{aligned}
&-N_{n/v_l} |\ker \pi_{v_l}| + \sum_{k=l+1}^{r} N_{n/(v_l p_k)} |\ker \pi_{v_l p_k}| =\\
&= |\ker \pi_{v_l}|\Big(-N_{n/v_l} + \sum_{k=l+1}^{r} N_{n/(v_l p_k)} |\ker \pi_{p_k}|\Big) \quad \text{(by Corollary 11)}\\
&\le |\ker \pi_{v_l}|\Big(-N_{n/v_l} + \sum_{k=l+1}^{r} N_{n/(v_l p_k)} p_k^2\Big) \quad \text{(by Lemma 15)}\\
&< 0 \quad \text{(by Lemma 19)}. \ \square
\end{aligned}$$

Corollary 21. *Let i be an integer satisfying $0 \leq i \leq r-1$ and $\emptyset \neq \mathcal{P} \subseteq \binom{\mathrm{P}}{i}$ (note that $\mathcal{P} = \{\emptyset\} \neq \emptyset$ if $i = 0$). Then we have*

$$\sum_{P \in \mathcal{P}} \Big(-N_{\frac{n}{n_P}} |\ker \pi_{n_P}| + \sum_{\substack{P' \in \binom{\mathrm{P}}{i+1} \\ P \subseteq P'}} N_{\frac{n}{n_{P'}}} |\ker \pi_{n_{P'}}| \Big) < 0.$$

Proof. Let $P \in \binom{\mathrm{P}}{i}$. Rearranging P if necessary, we can assume that $v_l = n_P$. Further, for every $P' \in \binom{\mathrm{P}}{i+1}$ with $P \subseteq P'$ there exists a unique p_k with $k \geq l+1$ such that $n_{P'} = v_l p_k$. Now the assertion follows from Lemma 20. □

Corollary 22. *Assumptions being as in Lemma 20, let $C_0 \in \mathcal{C}_0$ we have*

$$-\sum_{\substack{P \in \binom{\mathrm{P}}{i} \\ H_{\mathcal{C}_0^{n_P}}(C_0)=1}} N_{\frac{n}{n_P}} |\ker \pi_{n_P}| + \sum_{\substack{P' \in \binom{\mathrm{P}}{i+1} \\ H_{\mathcal{C}_0^{n_{P'}}}(C_0)=1}} N_{\frac{n}{n_{P'}}} |\ker \pi_{n_{P'}}| \leq 0$$

with equality holding if and only if both sums are empty.

Proof. First of all note that by Corollary 12 we have

$$H_{\mathcal{C}_0^{n_{P'}}}(C_0) = 1 \Rightarrow \forall\, P \subseteq P' :\ H_{\mathcal{C}_0^{n_P}}(C_0) = 1.$$

Hence, if the left sum is empty, both sums are empty and so the given term equals 0. If the left sum is not empty and the right sum is empty, the given term is trivially < 0. So assume that the right sum is not empty (which implies that the left sum is non-empty as well). We get

$$\text{Given term} \leq \sum_{\substack{P \in \binom{\mathrm{P}}{i} \\ H_{\mathcal{C}_0^{n_P}}(C_0)=1}} \Big(-N_{\frac{n}{n_P}} |\ker \pi_{n_P}| + \sum_{\substack{P' \in \binom{\mathrm{P}}{i+1} \\ P \subseteq P'}} N_{\frac{n}{n_{P'}}} |\ker \pi_{n_{P'}}| \Big) < 0$$

by the previous corollary. □

Lemma 23. *For all $1 \leq i \leq r-1$ we have*

$$\begin{gathered} -sum_{P \in \binom{\mathrm{P}}{i}} N_{\frac{n}{n_P}} |\ker \pi_{n_P}| (1 - H_{\mathcal{C}_0^{n_P}}(C_0)) \\ + \textstyle\sum_{P' \in \binom{\mathrm{P}}{i+1}} N_{\frac{n}{n_{P'}}} |\ker \pi_{n_{P'}}| (1 - H_{\mathcal{C}_0^{n_{P'}}}(C_0)) \leq 0 \end{gathered}$$

with equality holding if and only if $C_0 \in \mathcal{C}_0^p$ for all $p|n$.

Proof. Of course the given sum equals 0 if $C_0 \in \mathcal{C}_0^p$ for all $p|n$. So suppose that there exists $p|n$ such that $C_0 \notin \mathcal{C}_0^p$. It follows that for all k there exists $P \in \binom{\mathrm{P}}{k}$ such that $C_0 \notin \mathcal{C}_0^{n_P}$. Let

$$\{P'_1, \ldots, P'_m\} = \{P' \subseteq \mathrm{P} \mid |P'| = i+1, C \notin \mathcal{C}_0^{n_{P'}}\}.$$

So by Corollary 12 there exist pairwise distinct $P_1, \dots, P_k$ such that $|P_l| = i$ for $1 \le l \le k$ and such that each P_l is a subset of at least one of the P'_t, $1 \le t \le m$. Hence the sum in question is less or equal to

$$\sum_{l=1}^{k} \Bigl(-N_{\frac{n}{n_{P_l}}} |\ker \pi_{n_{P_l}}| + \sum_{\substack{P' \in \binom{\mathrm{P}}{i+1} \\ P_l \subseteq P'}} N_{\frac{n}{n_{P'}}} |\ker \pi_{n_{P'}}|\Bigr) < 0$$

by Corollary 21. □

Remark. Note that the condition $i \ge 1$ is crucial in this proof. It is easily seen that the assertion of Lemma 23 is false for $i = 0$.

Lemma 24. *With the assumptions of Lemma 23 we have*

$$\begin{aligned} -\sum_{P \in \binom{\mathrm{P}}{i}} N_{\frac{n}{n_P}} |\ker \pi_{n_P}| (1 - H_{\mathcal{C}_0^{n_P}}(C_0)) &+ \sum_{P' \in \binom{\mathrm{P}}{i+1}} N_{\frac{n}{n_{P'}}} |\ker \pi_{n_{P'}}| (1 - H_{\mathcal{C}_0^{n_{P'}}}(C_0)) \\ \ge -\sum_{P \in \binom{\mathrm{P}}{i}} N_{\frac{n}{n_P}} |\ker \pi_{n_P}| &+ \sum_{P' \in \binom{\mathrm{P}}{i+1}} N_{\frac{n}{n_{P'}}} |\ker \pi_{n_{P'}}| \end{aligned}$$

with equality holding if and only if $C_0 \notin \mathcal{C}_0^{n_P}$ *for all* $P \in \binom{\mathrm{P}}{i} \cup \binom{\mathrm{P}}{i+1}$.

Proof. The left hand side of the inequality equals

$$\sum_{\substack{P \in \binom{\mathrm{P}}{i} \\ H_{\mathcal{C}_0^{n_P}}(C_0) = 0}} N_{\frac{n}{n_P}} |\ker \pi_{n_P}| + \sum_{\substack{P' \in \binom{\mathrm{P}}{i+1} \\ H_{\mathcal{C}_0^{n_{P'}}}(C_0) = 0}} N_{\frac{n}{n_{P'}}} |\ker \pi_{n_{P'}}| =: A.$$

By Corollary 22 we have

$$\begin{aligned} A &\ge A - \overbrace{\sum_{\substack{P \in \binom{\mathrm{P}}{i} \\ H_{\mathcal{C}_0^{n_P}}(C_0) = 1}} N_{\frac{n}{n_P}} |\ker \pi_{n_P}| + \sum_{\substack{P' \in \binom{\mathrm{P}}{i+1} \\ H_{\mathcal{C}_0^{n_{P'}}}(C_0) = 1}} N_{\frac{n}{n_{P'}}} |\ker \pi_{n_{P'}}|} \\ &= -\sum_{P \in \binom{\mathrm{P}}{i}} N_{\frac{n}{n_P}} |\ker \pi_{n_P}| + \sum_{P' \in \binom{\mathrm{P}}{i+1}} N_{\frac{n}{n_{P'}}} |\ker \pi_{n_{P'}}| \end{aligned}$$

with equality holding if and only if the sums under the bracket are empty, i.e., if and only if $C_0 \notin \mathcal{C}_0^{n_P}$ for all $P \in \binom{\mathrm{P}}{i} \cup \binom{\mathrm{P}}{i+1}$. □

Lemma 25. *Assumptions being as above, we have*

$$\sum_{d|n} N_d \, \mu\Bigl(\frac{n}{d}\Bigr) |\ker \pi_{n/d}| (1 - H_{\mathcal{C}_0^{n/d}}(C_0)) \le 0$$

with equality holding if and only if $H_{\mathcal{C}_0^p}(C_0) = 1$ *for all* $p|n$.

Proof. The above sum equals

$$\sum_{i=1}^{\lfloor r/2\rfloor}\Big(-\sum_{P\in\binom{\mathrm{P}}{2i-1}} N_{\frac{n}{n_P}}|\ker\pi_{n_P}|(1-H_{\mathcal{C}_0^{n_P}}(C_0))$$

$$+\sum_{P\in\binom{\mathrm{P}}{2i}} N_{\frac{n}{n_P}}|\ker\pi_{n_P}|(1-H_{\mathcal{C}_0^{n_P}}(C_0))\Big)-\delta$$

where $\delta = N_{n/v_r}|\ker\pi_{v_r}|(1-H_{\mathcal{C}_0^{n/v_r}})$ if r is odd and $\delta = 0$ if r is even. So by Lemma 23

$$\text{Given sum} \le -\delta \le 0$$

with equality if and only if $\delta = 0$ and for all $\emptyset \ne P \subseteq \mathrm{P}$ we have $C_0 \in \mathcal{C}_0^{n_P}$, i.e., if and only if $C_0 \in \mathcal{C}_0^p$ for all $p|n$. □

Lemma 26. *Assumptions being as above we have*

$$\sum_{d|n} N_d\,\mu\Big(\frac{n}{d}\Big)\,|\ker\pi_{n/d}|(1-H_{\mathcal{C}_0^{n/d}}(C_0)) \ge \sum_{\substack{d|n\\ d<n}} N_d\,\mu\Big(\frac{n}{d}\Big)\,|\ker\pi_{n/d}|$$

with equality holding if and only if $C_0 \notin \mathcal{C}_0^p$ *for all* $p|n$.

Proof. Resolving the sum on the left hand side of the above inequality as in the proof of the preceding lemma and applying Lemma 24 we get

$$\text{Given sum} \ge \sum_{i=1}^{\lfloor r/2\rfloor}\Big(-\sum_{\substack{P\\ |P|=2i-1}} N_{\frac{n}{n_P}}|\ker\pi_{n_P}|$$

$$+\sum_{\substack{P\\ |P|=2i-1}} N_{\frac{n}{n_P}}|\ker\pi_{n_P}|\Big)-\delta$$

$$=\sum_{\substack{d|n\\ d<n}} N_d\,\mu\Big(\frac{n}{d}\Big)\,|\ker\pi_{n/d}|$$

with equality holding if and only if

$$\forall\,\emptyset\ne P\subseteq\mathrm{P}:\quad C_0\notin\mathcal{C}_0^{n_P}\iff C_0\notin\mathcal{C}_0^p \text{ for all } p|n.\ \square$$

Now we are able to prove Theorem 16:

Proof. (Of Theorem 16) (1) and (2) follow from Lemma 25 and Equation (6), (3) follows from Lemma 26, Equation (6) and Theorem 6. □

Remark. The assertion of Theorem 16 can be extended to $q = 4, 5$ by more careful estimations.

With the tools developed in this section we are able to prove several properties of the function a_n. The next two theorems serve as examples in this direction.

Theorem 27. *Notations and conditions being as in 51 we have $a_n(C_0) > 0$.*

Proof. In view of Theorem 16 it suffices to show that $a_n(\mathcal{H}) > 0$ for the principal class $\mathcal{H}$ of K. Now

$$a_n(\mathcal{H}) = \frac{1}{hn} \sum_{d|n} N_d \, \mu\Big(\frac{n}{d}\Big) \, |\ker \pi_{n/d}|.$$

The above sum equals

$$\frac{1}{hn} \sum_{i=0}^{\lfloor (r-1)/2 \rfloor} \Big(\sum_{P \in \binom{\mathrm{P}}{2i}} N_{\frac{n}{n_P}} |\ker \pi_{n_P}| - \sum_{P \in \binom{\mathrm{P}}{2i+1}} N_{\frac{n}{n_P}} |\ker \pi_{n_P}| \Big) + \frac{1}{hn} \delta'$$

where $\delta' = 0$ if r is odd and $\delta' = N_{n/v_r} |\ker \pi_{v_r}|$ if r is even. But

$$\sum_{P \in \binom{\mathrm{P}}{2i}} N_{\frac{n}{n_P}} |\ker \pi_{n_P}| - \sum_{P \in \binom{\mathrm{P}}{2i+1}} N_{\frac{n}{n_P}} |\ker \pi_{n_P}|$$

is greater than zero by Lemma 23 (if $i > 0$) and Lemma 21 (if $i = 0$). □

Remark. The above theorem states in other words that under the conditions stated the mapping tr defined in the introduction is surjective.

The following example shows that the assertion of Theorem 27 need not be true for $q \leq 5$:

Example 28. We consider again the elliptic function field

$$K = \mathbb{F}_4(x, y), \quad y^2 + y = x^3 + 1.$$

Let $n = 3$ and $\mathcal{H}$ be the principal class of K. An easy computation shows that $N_3 = 24$. Application of Theorem 6 yields

$$a_3(\mathcal{H}) = \frac{1}{9}\Big(24 - \frac{1}{3} \cdot 9 \cdot 8\Big) = 0.$$

The next theorem is a partial converse to Lemma 7.

Theorem 29. *Notations and conditions being as in 51 we have: $a_n(C_0) = \Pi_n/h$ for all $C_0 \in \mathcal{C}_0$ if and only if* $\gcd(n, h) = 1$.

Proof. In view of Lemma 7 we have to prove that for $\gcd(n, h) \neq 1$ there exists a class $C_0 \in \mathcal{C}_0$ such that $a_n(C_0) \neq a_n(\mathcal{H})$ where $\mathcal{H}$ is as usual the principal class of K. Now if $\gcd(n, h) \neq 1$, there exists a prime number p such that $p| \gcd(n, h)$. Hence there exists a class $C_0 \notin \mathcal{C}_0^p$. Lemma 25 implies now $a_n(\mathcal{H}) - a_n(C_0) < 0$. □

Example 8 shows that the assertion of Theorem 29 need not be true for $q \leq 5$.

6 An Optimal Algorithm for Multiplication in $\mathbb{F}_{27}/\mathbb{F}_3$

This section gives an application of the results of this paper to the problem of determining optimal bilinear multiplication algorithms for finite extensions of finite fields. For a background on the bilinear complexity theory, we refer the reader to [3, Chap. 14].

A bilinear algorithm of length r for the multiplication in a finite dimensional k-algebra A consists of r triples (f_i, g_i, w_i) where f_i and g_i are k-linear forms on the vector space A, and $w_i \in A$, such that

$$\forall a, b \in A\colon \quad a \cdot b = \sum_{i=1}^{r} f_i(a) g_i(b) w_i.$$

($a \cdot b$ is the product of a and b in A.) The aim is to obtain for an algebra A a bilinear algorithm of minimal length.

As an example, consider the algebra $A := k[x]/(x^2 - a)$, for some $a \in k$. A basis for this algebra is given by $(1, x)$, where we identify polynomials with their residue classes modulo $x^2 - a$. The naive way of multiplying elements in this algebra is by implementing the following formula:

$$(A + Bx)(C + Dx) = (AC + aBD) + (AD + BC)x.$$

Let $f_1(\alpha + \beta x) = g_1(\alpha + \beta x) = \alpha$, $f_2(\alpha + \beta x) = g_2(\alpha + \beta x) = \beta$, $w_1 := 1$, $w_2 := x$, and $w_3 := a$. Then, it is easily verified that

$$(f_1, g_1, w_1), (f_1, g_2, w_2), (f_2, g_1, w_2), (f_2, g_2, w_3)$$

is a bilinear computation for A of length 4. Another, more efficient algorithm is derived from

$$(A + Bx)(C + Dx) = (AB + aBD) + ((A + B)(C + D) - AC - BD)x$$

which gives rise to a bilinear algorithm of length 3: let f_i, g_i, $i = 1, 2$ be as above, and let $f_3(\alpha + \beta x) = g_3(\alpha + \beta x) := \alpha + \beta$. Further, let $w_1 := 1 - x$, $w_2 := a - x$, and $w_3 := x$. Then

$$(f_1, g_1, w_1), (f_2, g_2, w_2), (f_3, g_3, w_3)$$

is a bilinear computation for A of length 3.

The bilinear complexity does not measure the number of additions/subtractions, or scalar multiplications. (This is expressed by the fact that the additions and scalar multiplications necessary for evaluating f_i and g_i are not counted.) However, for many important problems like the matrix multiplication, the asymptotic complexity can be measured in terms of the bilinear complexity only [3, Chap. 15]. Furthermore, in some situations, using bilinear algorithms recursively leads to overall savings in the running time. For instance, the Toom-Karatsuba method of multiplication [7, Chap. 4.3.3] can be seen as recursively using the multiplication algorithm of length 3 in the algebra A above (for suitable A).

An important class of k-algebras are simple field extensions. Multiplication in these algebras can be reduced to polynomial multiplication, which in turn can be accomplished using Lagrange interpolation. One can prove that if $|k| \geq 2n-2$, then the bilinear complexity of multiplication in a simple field extension of degree n over k is exactly $2n-1$, and that it is larger than $2n-1$ otherwise [3, Th. 17.29 and Rem. 17.30].

In [4] the authors describe an algorithm for multiplication in extensions of small finite fields, i.e., in extensions of degree n of a finite field k with $|k| < 2n-2$. In a nutshell, Goppa's idea is used to replace the Lagrange interpolation by interpolation on algebraic curves. The algorithm was slightly modified in [10] for elliptic curves. In particular, it was proved there that the bilinear complexity of multiplication in $\mathbb{F}_{q^n}$ is $2n$ if $\frac{1}{2}q+1 < n < \frac{1}{2}m(q)$ where $m(q)$ is the maximum number of points of an elliptic curve over $\mathbb{F}_q$. In a subsequent work [9], it was described how to obtain these algorithms using arithmetic properties of elliptic curves.

In this section, we apply by way of an example some of the results of this paper to obtain an optimal algorithm for multiplication in the field extension $\mathbb{F}_{27}$ of $\mathbb{F}_3$. Note that this case is not covered by [9]. The similar case of $\mathbb{F}_{256}/\mathbb{F}_4$ was solved in [2].

For the following computations we present $\mathbb{F}_3$ as $\mathbb{F}_3 = \{0,1,2\}$. We assume familiarity with [9].

In order to compute the optimal algorithm we are looking for, we follow [9, Algorithm IV–B] with minor modifications. In particular, we will compute two matrices $A \in \mathbb{F}_3^{3\times 6}$ and $B \in \mathbb{F}_3^{6\times 3}$ and a basis (f_0, f_1, f_2) of $\mathbb{F}_{27}/\mathbb{F}_3$ with the following properties: the multiplication of $x_0f_0 + x_1f_1 + x_2f_2$ and $y_0f_0 + y_1f_1 + y_2f_2$ is given as $z_0f_0 + z_1f_1 + z_2f_2$ where

$$(z_0, z_1, z_2) = (X_0Y_0, \dots, X_5Y_5)B,$$

and

$$\begin{pmatrix} X_0 \dots X_5 \\ Y_0 \dots Y_5 \end{pmatrix} = \begin{pmatrix} x_0, x_1, x_2 \\ y_0, y_1, y_2 \end{pmatrix} A.$$

Hence, A and B completely determine the multiplication in $\mathbb{F}_{27}$.

To obtain these two matrices, we first compute an elliptic curve E over $\mathbb{F}_3$ having 7 $\mathbb{F}_3$-rational and compute its set of points $E(\mathbb{F}_3)$. Next we determine prime divisors $\mathfrak{D}$ and $\mathfrak{p}$, both of degree 3, such that $L(\mathfrak{D} - \mathfrak{p}) = 0$, where by $L(A)$ we denote the linear space of the divisor A. Notice that this is equivalent to requiring $[\mathfrak{D}] \neq [\mathfrak{p}]$, where $[A]$ denotes the class of the divisor A. Afterwards we compute a basis $\{f_1, \dots, f_6\}$ of $L(2\mathfrak{D})$ such that $\{f_1, f_2, f_3\}$ is a basis of $L(\mathfrak{D})$. In order to compute the matrices Γ and A of Step (v) of [9, Algorithm IV–B], we first need to compute a set of points $P_1, \dots, P_6$ of $E(\mathbb{F}_3)$ such that

$$[P_1 \oplus \cdots \oplus P_6 - Q] \neq [\mathfrak{D} - 3Q]$$

where Q is the neutral element of $E(\mathbb{F}_3)$. Next we perform Steps (vi) and (vii) of [9, Algorithm IV–B] to obtain the matrices A and B which are the final outputs.

It is easily verified that the curve $E : y^2 = x^3 - x + 1$ has 7 $\mathbb{F}_3$-rational points and $E(\mathbb{F}_3) = \{Q, P_1, \dots, P_6\}$, where

$$\begin{aligned} &P_1 := (0,2), P_2 := (0,1), P_3 := (1,2),\\ &P_4 := (1,1), P_5 := (2,2), P_6 := (2,1). \end{aligned}$$

For the sake of simplicity we choose for $\mathfrak{D}$ a prime divisor of degree 3. The representation of prime divisors is the same as described in [9, Section 3.6], i.e., a prime divisor is given by (g, h) where g is an irreducible polynomial of degree 3 and h is a polynomial of degree at most 2 such that $h^2 \equiv x^3 - x + 1 \bmod g$. We use [9, Algorithm III–F] to compute two random prime divisors $\mathfrak{D}$ and $\mathfrak{p}$. This is the place where we use the results of this paper. Namely, for a random choice of $\mathfrak{D}$ and $\mathfrak{p}$, the probability that they belong to the same divisor class is $\frac{1}{7}$ by Theorem 29. Hence, with high probability, we will pick two prime divisors that do not belong to the same class. In fact, after one guess we obtain

$$\mathfrak{D} = (x^3 + 2x^2 + 2x + 2, 2x^2 + x), \qquad \mathfrak{p} = (x^3 + 2x^2 + 1, 2x^2 + 2x).$$

The class finding algorithm [9, Algorithm III–E] gives

$$[\mathfrak{D} - 3Q] = [P_5 - Q], \qquad [\mathfrak{p} - 3Q] = [P_3 - Q], \tag{7}$$

which shows that $\mathfrak{D}$ and $\mathfrak{p}$ belong to different classes. The function $(x + 1)$ has the divisor $P_5 + P_6 - 2Q$. Hence (7) implies that $\mathfrak{D} + P_6 - 4Q$ is a principal divisor, i.e., the divisor of a function, say u. Observe first that $u \in L(4Q)$. Since $1, x, y, x^2$ is a basis of $L(4Q)$, u is a linear combination of these functions. Further, $\operatorname{ord}_Q(u) = -4$, hence w.l.o.g. we may assume that $u = x^2 + ax + by + c$ with some constants $a, b, c \in \mathbb{F}_3$. Now $u(\mathfrak{D}) = 0$, hence computing with x as $X \bmod (X^3 + 2X^2 + 2X + 2)$ and with y as $(2X^2 + X) \bmod (X^3 + 2X^2 + 2X + 2)$, we obtain

$$(1 + 2b)X^2 + (a + b)X + C = 0.$$

This gives $u = x^2 + 2x + y$. We claim that

$$\mathsf{B} := \left\{1, \frac{x+1}{u}, \frac{(x+1)^2}{u}, \frac{(x+1)^2}{u^2}, \frac{(x+1)^3}{u^2}, \frac{(x+1)^4}{u^2}\right\}$$

is a basis of $L(2\mathfrak{D})$ and that the first three elements of B form a basis of $L(\mathfrak{D})$: first, $\mathsf{B} \subset L(2\mathfrak{D})$ and the first three elements of B belong to $L(\mathfrak{D})$ (simply compute their divisors!). In order to prove linear independence, we evaluate the representation matrix of the linear morphism

$$\begin{aligned} \gamma\colon\ \langle \mathsf{B} \rangle &\to \mathbb{F}_3^6\\ v &\mapsto (v(P_1), \dots, v(P_6)). \end{aligned}$$

We will show that $\mathrm{rk}(\gamma) = 6$, which implies that B is a basis of $L(2\mathfrak{D})$. For this, it is sufficient to show that

$$\Gamma := (v(P_i))_{v\in\mathsf{B},1\le i\le 6}$$

has full rank. First, we have to explain how to evaluate the function in B at the point P_6, since $u(P_6) = 0$. Observe that

$$\mathrm{ord}_{P_6}((x+1)^2/u), \mathrm{ord}_{P_6}((x+1)^3/u^2), \mathrm{ord}_{P_6}((x+1)^4/u^2) \ge 1,$$

which implies that these functions vanish at P_6. Setting $v := (x+1)/u$, we thus have to compute $v(P_6)$. Note that $t := (x+1)$ is a local parameter for P_6, i.e., $\mathrm{ord}_{P_6}(t) = 1$. We obtain the following power series expansion for v in $\mathbb{F}_3[[t]]$:

$$v = \frac{t}{t^2+2+\sqrt{t^3+2t+1}} = 1 + t + 2t^2 + \cdots .$$

Hence $v(P_6)$ (which equals the constant term of $v \in \mathbb{F}_3[[t]]$) is 1. Thus,

$$\Gamma = \begin{pmatrix} 1&1&1&1&1&1\\ 2&1&1&2&0&1\\ 2&1&2&1&0&0\\ 1&1&1&1&0&1\\ 1&1&2&2&0&0\\ 1&1&1&1&0&0 \end{pmatrix}.$$

It is easily seen that Γ is invertible. Hence B is a basis of $L(2\mathfrak{D})$.

Let us denote the elements of the basis B by $v_1 \ldots, v_6$ (in the order given above). We first compute a matrix T such that

$$\begin{pmatrix} v_1 \bmod \mathfrak{p} \\ \vdots \\ v_6 \bmod \mathfrak{p} \end{pmatrix} = T \begin{pmatrix} 1 \\ x \\ x^2 \end{pmatrix} \bmod (x^3+2x^2+1).$$

Now $u \bmod \mathfrak{p} = x \bmod (x^3+2x^2+1)$, hence $1/u \bmod \mathfrak{p} = 2x^2 + x \bmod (x^3 + 2x^2+1)$. Hence,

$$T = \begin{pmatrix} 1&0&0\\ 1&1&2\\ 2&2&2\\ 2&1&1\\ 1&0&0\\ 1&1&0 \end{pmatrix}.$$

Let T' be the matrix consisting of the first three rows of T. As is shown in [9, Section 4.4] T' is non-singular and $C = T(T')^{-1}$. Further, $B = \Gamma^{-1}C$. This yields

$$B = \begin{pmatrix} 2\,2\,1 \\ 0\,2\,1 \\ 1\,2\,2 \\ 0\,2\,0 \\ 0\,0\,1 \\ 1\,1\,1 \end{pmatrix}.$$

Finally, the matrix A is given by the first three rows of Γ, i.e.,

$$A = \begin{pmatrix} 1\,1\,1\,1\,1\,1 \\ 2\,1\,1\,2\,0\,1 \\ 2\,1\,2\,1\,0\,0 \end{pmatrix}.$$

The multiplication algorithm is thus as follows: to compute the product of (x_0, x_1, x_2) and (y_0, y_1, y_2), first compute

$$\begin{array}{lll} X_0 := x_0 + 2x_1 + 2x_2 & X_1 := x_0 + x_1 + x_2 & X_2 := x_0 + x_1 + 2x_2 \\ X_3 := x_0 + 2x_1 + x_2 & X_4 := x_0 & X_5 := x_0 + x_1 \\ Y_0 := y_0 + 2y_1 + 2y_2 & Y_1 := y_0 + y_1 + y_2 & Y_2 := y_0 + y_1 + 2y_2 \\ Y_3 := y_0 + 2y_1 + y_2 & Y_4 := y_0 & Y_5 := y_0 + y_1. \end{array}$$

The product is given by (z_0, z_1, z_2), where

$$\begin{aligned} z_0 &:= 2X_0Y_0 + X_2Y_2 + X_5Y_5, \\ z_1 &:= 2X_0Y_0 + 2X_1Y_1 + 2X_2Y_2 + 2X_3Y_3 + X_5Y_5, \\ z_2 &:= X_0Y_0 + X_1Y_1 + 2X_2Y_2 + X_4Y_4 + X_5Y_5. \end{aligned}$$

Acknowledgement

The research on this paper was done while the author was visiting the department of mathematics of the university of Brasília. The author wants to thank the GMD for a travel grant, the FINEP for financial support, the department of mathematics of the university of Brasília, especially Prof. S. Shokranian, for their hospitality during his visit in Brasília.

References

1. T.M. Apostol. *Introduction to Analytic Number Theory.* Undergraduate Texts in Mathematics. Springer Verlag, 1979.
2. U. Baum and M.A. Shokrollahi. An optimal algorithm for multiplication in $\mathbb{F}_{256}/\mathbb{F}_4$. *AAECC*, 2:15–20, 1991.
3. P. Bürgisser, M. Clausen, and M.A. Shokrollahi. *Algebraic Complexity Theory*, volume 315 of *Grundlehren der Mathematischen Wissenschaften.* Springer Verlag, Heidelberg, 1996.

4. D.V. Chudnovsky and G.V. Chudnovsky. Algebraic complexity and algebraic curves over finite fields. *J. Compl.*, 4:285–316, 1988.
5. M. Deuring. *Lectures on the Theory of Algebraic Functions of One Variable*, volume 314 of *Lecture Notes in Mathematics*. Springer Verlag, 1973.
6. H. Hasse. *Vorlesungen über Klassenkörpertheorie*. Physica-Verlag, Würzburg, 1967.
7. D.E. Knuth. *The Art of Computer Programming*, volume 2: semi-numerical algorithms. Addison-Wesley, 1981.
8. H. G. Rück. A note on elliptic curves over finite fields. *Math. Comp.*, 49:301–304, 1987.
9. M.A. Shokrollahi. Efficient randomized generation of algorithms for multiplication in certain finite fields. *Comp. Compl.*, 2:67–96, 1992.
10. M.A. Shokrollahi. Optimal algorithms for multiplication in certain finite fields using elliptic curves. *SIAM J. Comp.*, 21:1193–1198, 1992.

On Cyclic MDS-Codes

M. Amin Shokrollahi

Bell Labs, Rm. 2C-353, 700 Mountain Ave, Murray Hill, NJ 07974, USA

Abstract We investigate the question when a cyclic code is maximum distance separable (MDS). For codes of (co-)dimension 3, this question is related to permutation properties of the polynomial $(x^b-1)/(x-1)$ for a certain b. Using results on these polynomials we prove that over fields of odd characteristic the only MDS cyclic codes of dimension 3 are the Reed-Solomon codes. For codes of dimension $O(\sqrt{q})$ we prove the same result using techniques from algebraic geometry and finite geometry. Further, we exhibit a complete q-arc over the field $\mathbb{F}_q$ for even q. In the last section we discuss a connection between modular representations of the general linear group over $\mathbb{F}_q$ and the question of whether a given cyclic code is MDS.

1 Introduction

A linear code C is called **cyclic** if $(c_{n-1}, c_0, c_1, \ldots, c_{n-2})$ is in C whenever $(c_0, c_1, \ldots, c_{n-1})$ is. Let ϕ denote the morphism from $\mathbb{F}_{q^n}$ to $\mathbb{F}_q[x]/(x^n-1)$ sending $(a_0, \ldots, a_{n-1})$ to $\sum_i a_i x^i \bmod (x^n-1)$. Then it is easy to see that a subspace C of $\mathbb{F}_{q^n}$ is a cyclic code if and only if its image under ϕ is an ideal of $\mathbb{F}_q[x]/(x^n-1)$ [7, Chap. 6]. Since every ideal in this ring is principal, the image of C is generated by a polynomial $g(x)$ dividing x^n-1, unique up to scalar multiples. This polynomial is called the **generator polynomial** of C and the image of any codeword under ϕ is divisible by this polynomial. If n and q are co-prime, then $g(x)$ is uniquely determined by the set of its roots, which are also called the **zeros of** C.

It is easily seen that C is of dimension $n-\deg(g)$. The determination of the minimum distance of C from the set of zeros of g is much harder, though exact results on the complexity of this problem are not known. Typically, most of the research on this problem has concentrated on obtaining good lower bounds for the minimum distance in terms of the set of roots [8]. In this paper, we concentrate on aspects of the problem of bounding the minimum distance from above. Specifically, we discuss problems related to $[q-1,k,d]_q$-cyclic codes. (Here and in the following, an $[n,k,d]_q$-code is a code of block-length n, dimension k, and minimum distance d over the field $\mathbb{F}_q$.) Since q and $q-1$ are obviously co-prime, the code is uniquely identified by the set $\{\omega^{a_0}, \ldots, \omega^{a_r}\}$ of its zeros, where ω is a generator of $\mathbb{F}_q^\times$ which we assume to be fixed throughout the paper. We also assume thoughout, except in a brief remark in §2, that $r > 1$. We may hence identify the code with its **exponent set** $\{a_0, \ldots, a_r\}$. The question we want to investigate is whether the code is maximum distance separable, i.e., whether it has the maximum possible minimum distance $r+2$. If this is the case, then we call the exponent set $\{a_0,\ldots,a_r\}$ q-**MDS**, or **MDS** for short, if q is obvious from the context.

If the exponent set $\{a_0,\ldots,a_r\}$ is not MDS, then there exists a polynomial $f = f_0x^{i_0} + \cdots + f_rx^{i_r} \in \mathbb{F}_q[x]$ with pairwise different nonnegative integers $i_0,\ldots,i_r$ less than n such that $f(\omega^{a_0}) = \cdots = f(\omega^{a_r}) = 0$. This is equivalent to

$$\begin{pmatrix} \omega^{a_0 i_0} & \cdots & \omega^{a_0 i_r} \\ \vdots & \ddots & \vdots \\ \omega^{a_r i_0} & \cdots & \omega^{a_r i_r} \end{pmatrix} \begin{pmatrix} f_0 \\ \vdots \\ f_r \end{pmatrix} = 0.$$

The existence of a nonzero f with this property is equivalent to the vanishing of the determinant of the above matrix. Let now

$$X := \begin{pmatrix} X_0^{a_0} & \cdots & X_r^{a_0} \\ \vdots & \ddots & \vdots \\ X_0^{a_r} & \cdots & X_r^{a_r} \end{pmatrix},$$

where $X_0,\ldots,X_r$ are indeterminates over $\mathbb{F}_q$. Then it is easily seen that C is MDS if and only if

$$\det(X) \in \mathbb{F}_q[X_0,\ldots,X_r]$$

has no zeros in $(\mathbb{F}_q^\times)^{r+1}\backslash\Delta$, where $\Delta \subset \mathbb{F}_q^{r+1}$ is the zeroset of the discriminant $\prod_{i<j}(X_i - X_j)$. It follows that for any a and b, b co-prime to $q-1$, all exponent sets of the form $\{a, b+a, 2b+a, \ldots, rb+a\} \bmod (q-1)$ are q-MDS. Indeed, for these sets the above determinant is essentially Vandermonde. The corresponding cyclic codes are equivalent to *Reed-Solomon codes* [7, §6.8]. In the sequel we call these sets "trivial." One of the results of this paper is that in many cases the only q-MDS exponent sets are the trivial ones, i.e., the corresponding cyclic codes are essentially Reed-Solomon codes. We note in passing that our results also solve some cases of a problem of Nick Reingold and Dan Spielman posed by Andrew Odlyzko in [10, p. 399].

We start our investigation in the next section by studying exponent sets of size three. We show that these sets are MDS if and only if the polynomial $x^{b-1} + \cdots + x + 1$ is a permutation polynomial over $\mathbb{F}_q$, where b is an integer obtained from the exponent set in question. This problem has been investigated by Matthews [9] in case of odd q. Using his results, we show that q-MDS exponent sets of size three are trivial for odd q. In Section 3 we investigate exponent sets whose sizes are "small" relative to q , and use some algebraic geometry as well as results about arcs in projective spaces to show that they are MDS if and only if they are trivial. Section 4 deals with the special exponent set $\{0, 1, \ldots, r-1, m\}$ for some m satisfying $r \le m \le q-2$. We show that if r is not large compared to q, then these exponent sets are MDS only if they are trivial. Then we will proceed by exhibiting an explicit family of complete q-arcs over fields of even characteristic. The last section of the paper deals with an unexpected connection between the minimum distance of cyclic codes and certain modular representations of $\mathrm{GL}_n(\mathbb{F}_p)$.

Many thanks go to E.F. Assmus, D. Spielman, and M. Zieve for pointing out to me the references [12], [10], and [9], respectively.

2 Small Exponent Sets

Exponent sets if size two are easy to handle: obviously, $\{0, a\}$ is q-MDS iff $\gcd(a, q-1) = 1$ and $\{a, b\}$ is q-MDS iff $\gcd(a-b, q-1) = 1$.

Exponent sets of size three are slightly more difficult to investigate. Let $I := \{0, a, b\}$ be an exponent set. We may without loss of generality assume that a divides $q-1$ and that $a \leq d := \gcd(b, q-1)$. I is q-MDS iff for every $x, y \in \mathbb{F}_q^\times \setminus \{1\}$, $x \neq y$ we have

$$\det \begin{pmatrix} 1 & 1 & 1 \\ 1 & x^a & y^a \\ 1 & x^b & y^b \end{pmatrix} = (x^a - 1)(y^b - 1) - (x^b - 1)(y^a - 1) \neq 0.$$

If $a \geq 3$, then we may take for x and y two different ath roots of unity in $\mathbb{F}_q^\times$, both unequal to one, to see that I is not MDS. The same argument works if $d \geq 3$. If $a = 2$, then necessarily $d = 2$ and we may take $x = -1$ to see that I is not MDS. Hence, we are left with the case $a = 1$. We may without loss of generality assume that $b \leq q/2$, since we may replace $\{0, 1, b\}$ by $\{q-0, q-1, q-b\} = \{1, 0, q-b\}$. Hence $\{0, 1, b\}$ is MDS if and only if the polynomial $(x^b - 1)/(x-1) = x^{b-1} + \ldots + 1$ is injective on $\mathbb{F}_q \setminus \{0, 1\}$. This implies that the size of the image of this polynomial considered as a polynomial function over $\mathbb{F}_q$ is at least $q-2$ which is larger than $q - (q-1)/(b-1)$. Hence, we deduce by Wan's Theorem [15] that $x^{b-1} + \ldots + 1$ is a permutation polynomial. A result of Matthews' [9] yields that $b = 2$ if q is odd.

Proposition 1. *For odd q, q-MDS exponent sets of size three are trivial. Equivalently, a cyclic code of block length $q-1$ and co-dimension three over $\mathbb{F}_q$ is MDS if and only if it is equivalent to a Reed-Solomon code.*

The above assertion does not hold for even q. For instance, the exponent set $\{0, 1, 8\}$ is not trivial but it is 32-MDS. To see the latter, note that the polynomial $(x^8 + 1)/(x + 1)$ is a permutation polynomial over $\mathbb{F}_{32}$, since the change of variable $y := x + 1$ transforms it into y^7. (Table 1 gives all values of b such that $(x^b + 1)/(x + 1)$ is a permutation polynomial over $\mathbb{F}_q$ for some small values of q.) Further, a small calculation shows that existence of a and b such that $\{0, 1, 8\} = \{a, a+b, a+2b\}$ leads to a contradiction; hence the exponent set is nontrivial. (Details are left to the reader.) In general, MDS exponent sets of size three over finite fields of characteristic two correspond to certain ovals in finite Desarguesian planes of even order, for which a complete description has not yet appeared. (See [9, Section 4].)

In the next section we will derive similar assertions for other exponent sets of small size. The method is different from the one used in this section, as it

q	b
4	2
8	$2, 4, 6$
16	$2, 8, 14$
32	$2, 4, 6, 8, 10, 16, 22, 24, 26, 28, 30$
64	$2, 32, 63$
128	$2, 4, 6, 8, 16, 20, 22, 32, 42, 52, 64, 76$ $86, 96, 106, 108, 112, 120, 122, 124, 126$
256	$2, 8, 32, 74, 128, 182, 224, 248, 254$

Table1. Values of b such that $(x^b + 1)/(x + 1)$ is a permutation polynomial over $\mathbb{F}_q$.

employs techniques from the theory of finite geometries and some algebraic geometry.

3 Arcs and Normal Rational Curves

We denote the r-dimensional projective space over a field K by $\mathbb{P}^r(K)$. A point P with projective coordinates $x_0, \ldots, x_r$ is denoted by $P = (x_0 : \cdots : x_r)$. We start by introducing some definitions and recalling some basic facts about projective spaces over finite fields. A good reference for these subjects is Hirschfeld's book [4].

A **k-arc** in $\mathbb{P}^r(\mathbb{F}_q)$ is a set S of $k \geq r+1$ points such that no $r+1$ of them lie on a hyperplane. For any point in S we consider a representative in $\mathbb{F}_q^{r+1}$ and form the $(r+1) \times k$-matrix G_S whose columns are these points. Obviously S is an arc if and only if any $(r+1) \times (r+1)$-submatrix of G_S is invertible. (This condition is independent of the choice of the representatives for the points.) So, for $q \geq r+2$ the subset $S(\mathbb{F}_q^\times)$ of $\mathbb{P}^r(\mathbb{F}_q)$ consisting of the points $(1 : \alpha^{a_1} : \cdots : \alpha^{a_r})$, $\alpha \in \mathbb{F}_q^\times$, is a $(q-1)$-arc if and only if $\{0, a_1, \ldots, a_r\}$ is q-MDS.

A standard example of arcs is given by the set of points of a **normal rational curve**. A **rational curve** C_n of order n in $\mathbb{P}^r(\mathbb{F}_q)$ is the set of points $(g_0(t_0, t_1) : \cdots : g_r(t_0, t_1))$ where $t_0, t_1 \in \mathbb{F}_q$ and each g_i is a binary form of degree n and a highest common factor of $g_0, \ldots, g_r$ is 1. The curve C_n may also be written as the set of points $(f_0(t) : \cdots : f_n(t))$, where $f_i(t) := g_i(1, t)$, $t \in \mathbb{F}_q^+ := \mathbb{F}_q \cup \{\infty\}$, and $f_i(\infty)$ is by definition the coefficient of t^n in f_i. As the g_i have no nontrivial common factor, at least one f_i has degree n. The curve C_n is called **normal** if it is not a projection of a rational curve C'_n in $\mathbb{P}^{r+1}(\mathbb{F}_q)$, where C'_n is not contained in any r-dimensional hyperplane of $\mathbb{P}^{r+1}(\mathbb{F}_q)$. A **projective equivalence** in $\mathbb{P}^r(\mathbb{F}_q)$ is a self-mapping of $\mathbb{P}^r(\mathbb{F}_q)$ which associates to a point $(x_0 : \cdots : x_r)$ the point $(y_0 : \cdots : y_r)$ where

$$(y_0, \ldots, y_r)^\top = A \cdot (x_0, \ldots, x_r)^\top$$

for a nonsingular $(r+1)\times(r+1)$-matrix A. The basic facts about normal rational curves can be summarized as follows, see [5, Chapter 21].

Theorem 2. *Let C_n be a normal rational curve in $\mathbb{P}^r(\mathbb{F}_q)$ not contained in a hyperplane. Then*

(i) $q \geq r$;
(ii) $n = r$;
(iii) *C_r is projectively equivalent to*

$$\{(t^r, t^{r-1}, \ldots, t, 1) \mid t \in \mathbb{F}_q^+\};$$

(iv) *C_r consists of $q+1$ points no $r+1$ of which lie on a hyperplane.*
(v) *If $q \geq r+2$ then there is a unique C_r through any $r+3$ points of $\mathbb{P}^r(\mathbb{F}_q)$ no $r+1$ of which lie on a hyperplane.*

Much of the research on arcs has concentrated on the following three problems posed by B. Segre in 1955 [11]: (1) For given r and q what is the maximum value of k for which there exists a k-arc in $\mathbb{P}^r(\mathbb{F}_q)$? (2) For what values of r and q, with $q > r+1$, is every $(q+1)$-arc of $\mathbb{P}^r(\mathbb{F}_q)$ the point set of a normal rational curve? (3) For given r and $q > r+1$, what are the values of k for which every k-arc of $\mathbb{P}^r(\mathbb{F}_q)$ is contained in a normal rational curve of this space?

Theorem 3. (1) (Thas [14]) *For odd q every k-arc in $\mathbb{P}^r(\mathbb{F}_q)$ with $k > q - \sqrt{q}/4 + r - 7/16$ is contained in a unique normal rational curve of this space.*
(2) (Bruen et al. [1], Storme and Thas [13]) *For even $q \geq 4$ and $r \geq 4$ every k-arc of $\mathbb{P}^r(\mathbb{F}_q)$ with $k \geq q+r-\sqrt{q}/2-3/4$ is contained in a unique normal rational curve of this space.*

We remark that the the bound in Part (1) of the above theorem can be improved considerably if q is a prime, see [13].

Using the above results and the Bézout Inequality we will be able to prove that certain MDS exponent sets are essentially trivial. For the proof of the following lemma we assume familiarity with the concept of degree of an algebraic variety, see, e.g., [3, Lecture 18].

Lemma 4. *Let $a_1, \ldots, a_r$ be pairwise different positive integers, and K be an algebraically closed field. Suppose that $d := \gcd(a_1, \ldots, a_r)$ is not divisible by the characteristic of K. The Zariski-closure X of the image of the map $K \to K^r$, $t \mapsto (t^{a_1}, \ldots, t^{a_r})$ is a rational curve of degree A/d, where $A := \max_i a_i$.*

Proof. Obviously X is a rational curve. Further, as d is not divisible by the characteristic of K, X is the closure of the image of the map $t \mapsto (t^{a_1/d}, \ldots, t^{a_r/d})$. So we may suppose that $d = 1$. In addition, we may assume that $a_1 < a_1 < \cdots < a_r$. The degree of X is the maximum of the numbers $|X \cap H|$, where H runs over all hyperplanes of $\mathbb{P}^r(K)$ such that $X \cap H$ is finite. (For this and other characterizations of degree see, e.g., [3, Lecture 18].) Let $x_0, \ldots, x_r$ be the coordinates of $\mathbb{P}^r(K)$, and let H be the zeroset of $\alpha_0 x_0 + \ldots + \alpha_r x_r$. Then

$$X \cap H = \left\{ (1\colon \tau^{a_1}\colon \cdots \colon \tau^{a_r}) \;\middle|\; \alpha_0 + \sum_{i=1}^{r} \alpha_i \tau^{a_i} = 0 \right\}.$$

In particular, $|X \cap H| \leq a_r$. We thus need to show that there is some H such that $|X \cap H| = a_r$. Suppose first that $\gcd(\operatorname{char} K, a_r) = 1$, and let H be the zeroset of $x_0 - x_r$. Then $X \cap H$ consists of the points $(1\colon \zeta^{a_1}\colon \cdots \colon \zeta^{a_r})$, where ζ runs over all the a_rth roots of unity. These points are all different, as $\gcd(a_1, \ldots, a_r) = 1$, so $|X \cap H| = a_r$. Suppose now that $\gcd(\operatorname{char} K, a_r) \neq 1$. Then there is some a_i such that $\operatorname{char} K$ does not divide a_i. The polynomial $X^{a_r} + X^{a_i} + 1$ has $\ell := a_r$ different roots $\tau_1, \ldots, \tau_\ell$ in K, as it is relatively prime to its derivative. Since $\gcd(a_1, \ldots, a_r) = 1$, each of these roots gives rise to a different point $(1\colon \tau_i^{a_1}\colon \cdots \colon \tau_i^{a_r})$ in $X \cap H$, where H is the zeroset of $x_0 + x_i + x_r$. □

The main theorem of this section is now as follows.

Theorem 5. *Let $I := \{0, a_1, \ldots, a_r\}$ be q-MDS, where the a_i are pairwise different positive integers, and suppose that a_1 divides $q - 1$. Further, suppose that $r(\max_i a_i) < q - 1$. If $r < \sqrt{q}/4 + 9/16$ and q is odd, then $I = \{0, 1, 2, \ldots, r\}$. If $4 \leq r \leq \sqrt{q}/2 - 1/4$ and $q \geq 4$ is even, then $I = \{0, 1, 2, \ldots, r\}$.*

Proof. We may suppose that $r \geq 1$. Let $d := \gcd(a_1, \ldots, a_r)$. By assumption, the cyclic code over $\mathbb{F}_q$ with the zeroset $\{1, \omega^{a_1}, \ldots, \omega^{a_r})$ is MDS, hence has minimum distance $r + 2$. But this is not possible if $d \neq 1$, as this code contains the codeword $x^{(q-1)/d} - 1$ of weight $2 < r + 2$. So $d = 1$. Further, $S := \{(1\colon \alpha^{a_1}\colon \cdots \colon \alpha^{a_r}) \mid \alpha \in \mathbb{F}_q^\times\}$ is a $(q-1)$-arc. By Theorem 3 we deduce that S is contained in a normal rational curve C_r of $\mathbb{P}^r(\mathbb{F}_q)$. On the other hand, S is contained in the set of $\mathbb{F}_q$-rational points of the curve $X := \{(1\colon t^{a_1}\colon \cdots \colon t^{a_r}) \mid t \in K^+\}$, K being the algebraic closure of $\mathbb{F}_q$. By the Bézout Inequality and the last lemma we have $\deg(X \cap C_r) \leq r(\max_i a_i) < q - 1$, hence $X = C_r$, as C_r is irreducible. We thus obtain $\max_i a_i = r$, which gives $I = \{0, 1, \ldots, r\}$. □

4 The Special Exponent Set $\{0, 1, \ldots, r-1, m\}$

Consider a cyclic code with exponent set $\{0, 1, \ldots, r-1, m\}$. Its minimum distance is at least $r + 1$ since it is contained in a Reed-Solomon code of dimension $n - r$. Hence, if the code is not MDS, then its minimum distance is $r + 1$. The result of Theorem 5 can be somewhat sharpened for this special exponent set in the following way.

Theorem 6. *Let r and m be positive integers satisfying $r \leq m \leq q-2$. Further, suppose that $r < \sqrt{q}/4 + 7/16$ if q is odd, and $4 \leq r \leq \sqrt{q}/2 + 3/4$ if $q \geq 4$ is even. Then $\{0, 1, \ldots, r-1, m\}$ is q-MDS if and only if $m = r$ or $m = q-2$.*

The if-part being clear, we continue with the only-if-part. Let

$$S_{r,m} := \{(1\colon \alpha\colon \alpha^2\colon \cdots\colon \alpha^{r-1}\colon \alpha^m) \mid \alpha \in \mathbb{F}_q^\times\}.$$

We need to show that under the above conditions on r the set $S_{r,m}$ is an arc if and only if $m = r$ or $m = q-2$. Obviously $S_{r,m}$ is an arc if and only if $\mathcal{K} = \mathcal{K}_{r,m} := S_{r,m} \cup \{P\}$ is, where $P = (0\colon \cdots\colon 0\colon 1)$. Suppose that $\mathcal{K}$ is an arc. Using Theorem 3 we deduce that $\mathcal{K}$ lies on a normal rational curve. For the rest of this section we concentrate on proving that $m = r$ or $m = q-2$, or, equivalently, that $\mathcal{K}$ does not lie on a normal rational curve if $r < m < q-2$. This would complete the proof of Theorem 6. To proceed with the proof, we need some notation and some auxiliary results.

Let C_r be a normal rational curve of $\mathbb{P}^r(\mathbb{F}_q)$ given by

$$C_r = \{(g_0(t_0, t_1)\colon \cdots\colon g_r(t_0, t_1)) \mid t_0, t_1 \in \mathbb{F}_q\}.$$

Let ∂_i denote the differential operator $\partial/\partial T_i$ of the bivariate polynomial ring $\mathbb{F}_q[T_0, T_1]$. The line ℓ_R through the points $R := (g_0(t_0, t_1)\colon \cdots\colon g_r(t_0, t_1))$ and $(\partial_0 g_0(t_0, t_1)\colon \cdots\colon \partial_0 g_r(t_0, t_1))$ is called **the tangent line** to C_r at R. Let $x_0, \ldots, x_r$ be the coordinates of $\mathbb{P}^r(\mathbb{F}_q)$ and let $\mathbb{P}^{r-1}(\mathbb{F}_q) = \Pi$ be the hyperplane given by $x_r = 0$. The projection of C_r from P onto Π together with the point $R^* := \ell_R \cap \Pi$ is a normal rational curve C_r^* of $\mathbb{P}^{r-1}(\mathbb{F}_q)$, see [6, Lemma 7]. Now let C_r be a normal rational curve containing $\mathcal{K}$. Then $C_r^* = \{(1\colon t\colon \cdots\colon t^{r-1}\colon 0) \mid t \in \mathbb{F}_q^+\}$, since the projection of $\mathcal{K}$ is clearly contained in C_r^* and this normal rational curve of Π is uniquely determined by $r+2 < q$ of its point by Theorem 2, Part (v).

Proposition 7. *Let C be a normal rational curve of $\mathbb{P}^r(\mathbb{F}_q)$ containing $P = (0\colon \cdots\colon 0\colon 1)$. Suppose that the projection of C from P onto Π is the curve $C^* = \{(1\colon t\colon \cdots\colon t^{r-1}\colon 0) \mid t \in \mathbb{F}_q^+\}$. Then C is one of the following curves:*

- (Type ∞) $C = \{(1\colon t\colon t^2\colon \cdots\colon t^{r-1}\colon \mu(t)) \mid t \in \mathbb{F}_q^+\}$ *for some* $\mu \in \mathbb{F}_q[X]$ *with* $\deg(\mu) = r$.
- (Type β, $\beta \in \mathbb{F}_q$) $C = \{(t\colon t(t+\beta)\colon \cdots\colon t(t+\beta)^{r-1}\colon \eta(t)) \mid t \in \mathbb{F}_q^+\}$ *for some* $\eta \in \mathbb{F}_q[X]$ *with* $\deg(\eta) \leq r$ *and* $\eta(0) \neq 0$.

Moreover, C is of type γ, $\gamma \in \mathbb{F}_q^+$, if and only if the tangent line to C at P intersects C^* at the point corresponding to $t = \gamma$.

Proof. Suppose that the tangent line to C at P intersects C^* in the point $(0\colon \cdots\colon 0\colon 1\colon 0)$. For every $t \in \mathbb{F}_q$ there exists $\tau \in \mathbb{F}_q$ such that $(1\colon t\colon \cdots\colon t^{r-1}\colon \tau) \in C$. Hence, $C = \{(1\colon t\colon t^2\colon \cdots\colon t^{r-1}\colon \mu(t)) \mid t \in$

$\mathbb{F}_q\} \cup \{P\}$, where μ is a polynomial of degree $\le q-1$. As C is an arc, $\deg(\mu) \ge r$. Hence, $C = \{(1\colon t\colon t^2\colon \cdots\colon t^{r-1}\colon \mu(t)) \mid t \in \mathbb{F}_q^+\}$. Since C is normal, $\deg(\mu) = r$.

Suppose now that the tangent line to C at P intersects C^* at the point $(1\colon \beta\colon \beta^2\colon \cdots\colon \beta^{r-1}\colon 0)$, for some $\beta \in \mathbb{F}_q$. Notice that

$$C^* = \{(\tau^{r-1}\colon (1+\beta\tau)\tau^{r-2}\colon (1+\beta\tau)^2\tau^{r-3}\colon \cdots\colon (1+\beta\tau)^{r-1}\colon 0) \mid \tau \in \mathbb{F}_q^+\}.$$

The tangent line at P intersects C^* in the point corresponding to $\tau = \infty$. Hence,

$$C = \{(\tau^{r-1}\colon (1+\beta\tau)\tau^{r-2}\colon \cdots\colon (1+\beta\tau)^{r-1}\colon \mu(\tau)) \mid \tau \in \mathbb{F}_q\} \cup \{P\},$$

for some polynomial $\mu \in \mathbb{F}_q[X]$. As before, we obtain $\deg(\mu) = r$, and hence $C = \{(\tau\colon (1+\beta\tau)\tau^{r-2}\colon \cdots\colon (1+\beta\tau)^{r-1}\colon \mu(\tau)) \mid \tau \in \mathbb{F}_q^+\}$. Thus

$$\begin{aligned} C &= \left\{ \left(\frac{1}{t^{r-1}}\colon \frac{1+\beta/t}{t^{r-2}}\colon \cdots\colon (1+\beta/t)^{r-1}\colon \mu(1/t)\right) \,\middle|\, t \in \mathbb{F}_q^\times \right\} \\ &\quad \cup \{P\} \cup \{(0\colon 0\colon \cdots\colon 1\colon \mu(0))\} \\ &= \left\{ (t\colon (t+\beta)t\colon \cdots\colon (t+\beta)^{r-1}t\colon t^r\mu(1/t)) \,\middle|\, t \in \mathbb{F}_q^\times \right\} \\ &\quad \cup \{P\} \cup \{(0\colon 0\colon \cdots\colon 1\colon \mu(0))\} \\ &= \left\{ (t\colon (t+\beta)t\colon \cdots\colon (t+\beta)^{r-1}t\colon \eta(t)) \,\middle|\, t \in \mathbb{F}_q^+ \right\}, \end{aligned}$$

where $\eta(X) = X^r\mu(1/X)$ is the reversal of μ. Note that $\eta(0) \ne 0$ as $\deg(\mu) = r$, and that $\deg(\eta) \le r$. □

The last step in the proof of Theorem 6 is the following result.

Proposition 8. *Suppose that $r < m < q-2$. Then the set $\mathcal{K}_{r,m}$ does not lie on a normal rational curve.*

Proof. Suppose that $\mathcal{K} = \mathcal{K}_{r,m}$ lies on a normal rational curve C. By Proposition 7, C is of type γ for some $\gamma \in \mathbb{F}_q^+$.

Assume first that $\gamma = \infty$. Then there exists a polynomial μ of degree r over $\mathbb{F}_q$ such that $C = \{(1\colon t\colon \cdots\colon t^{r-1}\colon \mu(t)) \mid t \in \mathbb{F}_q^+\}$. As $\mathcal{K}$ lies on C, we deduce that the polynomial $X^m - \mu(X)$ has $q-1$ different roots over $\mathbb{F}_q$, hence is zero. But this implies that $X^m = \mu(X)$, hence $m = r$, a contradiction.

Suppose now that $\gamma = \beta$. Then there exists a polynomial η over $\mathbb{F}_q$ of degree $\le r$, and for all $\tau \in \mathbb{F}_q^\times$ there exists $t \in \mathbb{F}_q^\times$ such that

$$(1\colon \tau\colon \cdots\colon \tau^{r-1}\colon \tau^m) = (1\colon (t+\beta)\colon \cdots\colon (t+\beta)^{r-1}\colon \eta(t)/t).$$

Hence, $\tau = t+\beta$ and $(t+\beta)^m = \eta(t)/t$ for all $t \in \mathbb{F}_q^\times$. Thus, the polynomial $X(X+\beta)^m - \eta(X)$ has $q-1$ zeros in $\mathbb{F}_q$. Since $\deg(\eta) \le r < m$, this polynomial is not zero, and is of degree $m+1$. Hence, $m+1 \ge q-1$, which is a contradiction to $m < q-2$. This proves the proposition and completes the proof of Theorem 6. □

5 Complete q-Arcs over $\mathbb{F}_q$, q Even

In this section we will prove that the set

$$K_q := \{(1 \colon \alpha \colon \cdots \colon \alpha^{q-5} \colon \alpha^{q-3}) \mid \alpha \in \mathbb{F}_q^+ \setminus \{0\}\}$$

is a **complete** q-arc in $\mathbb{P}^{q-4}(\mathbb{F}_q)$, i.e., it is a q-arc which cannot be extended to a $q+1$-arc. We remark that Storme and Thas [12] have determined all values for k for which there exists a complete k-arc in $\mathbb{P}^r(\mathbb{F}_q)$, $q-2 \geq r > q - \sqrt{q} - 11/4$.

The exponent set corresponding to this arc is $\{0, 1, \ldots, q-5, q-3\}$ which turns out to be the set $\{2j+1 \mid j = 0, \ldots, q-3\}$ which is clearly trivial. Hence, the corresponding set of $(q-1)$ points in the projective space lies on a normal rational curve. However, the particular one-point extension of this set given by K_q does not lie on a normal rational curve even though it is an arc.

Theorem 9. *For $q \geq 8$ a power of two the set K_q is a complete q-arc in $\mathbb{P}^{q-4}(\mathbb{F}_q)$.*

Proof. We first prove that $K := K_q$ is a q-arc. Let $P := (0 \colon 0 \colon \cdots \colon 0 \colon 1)$. K is a q-arc iff $K' := K \setminus \{P\}$ is. Suppose that there exist pairwise different $\alpha_1, \ldots, \alpha_{q-3} \in \mathbb{F}_q^\times$ such that the corresponding points in K' lie on a hyperplane, i.e., such that the matrix $M := (\alpha_{ij})$, $\alpha_{ij} := \alpha_i^j$ for $i = 1, \ldots, q-3$, $j = 0, \ldots, q-5$, and $\alpha_{q-3,j} = \alpha_j^{q-3}$, is singular. Let V denote the Vandermonde matrix $V = (\alpha_i^j)$, $i = 1, \ldots, q-3$, $j = 0, \ldots, q-4$. Then $0 = \det M / \det V = \alpha_1 + \cdots + \alpha_{q-3}$, which is a contradiction, as the sum of all the elements of $\mathbb{F}_q$ is zero. Hence, K' and K are arcs.

Let us now show that K is complete. Suppose not, and assume that there is a point $\Gamma := (\gamma_0 \colon \gamma_1 \colon \cdots \colon \gamma_{q-5} \colon \gamma_{q-4})$ such that $K'' := K \cup \{\Gamma\}$ is a $(q+1)$-arc in $\mathbb{P}^{q-4}(\mathbb{F}_q)$. The dual of K'' is a $(q+1)$-arc in $\mathbb{P}^3(\mathbb{F}_q)$, which by a result of Casse and Glynn [2] is projectively equivalent to $\{P_t \mid t \in \mathbb{F}_q^+\}$, where $P_t := (1 \colon t \colon t^\theta \colon tt^\theta)$, θ being an $\mathbb{F}_2$-automorphism of $\mathbb{F}_q$. Hence, there exists $j \in \{1, \ldots, q+1\}$ such that

$$\begin{pmatrix} 1 & 1 & \cdots 1 & \gamma_0 & 0 \\ \alpha_1 & \alpha_2 & \cdots \alpha_{q-1} & \gamma_1 & 0 \\ \vdots & \vdots & \ddots \vdots & \vdots & \vdots \\ \alpha_1^{q-5} & \alpha_2^{q-5} & \cdots \alpha_{q-1}^{q-5} & \gamma_{q-5} & 0 \\ \alpha_1^{q-3} & \alpha_2^{q-3} & \cdots \alpha_{q-1}^{q-3} & \gamma_{q-4} & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & \beta_1 & \beta_1^\theta & \beta_1 \beta_1^\theta \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \beta_{j-1} & \beta_{j-1}^\theta & \beta_{j-1}\beta_{j-1}^\theta \\ 0 & 0 & 0 & 1 \\ 1 & \beta_{j+1} & \beta_{j+1}^\theta & \beta_{j+1}\beta_{j+1}^\theta \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \beta_q & \beta_q^\theta & \beta_q \beta_q^\theta \\ 1 & \beta_j & \beta_j^\theta & \beta_j \beta_j^\theta \end{pmatrix} = 0^{(q-3)\times 4}, \tag{1}$$

where $\mathbb{F}_q = \{\alpha_1, \ldots, \alpha_{q-1}, 0\} = \{\beta_1, \ldots, \beta_q\}$. Considering the $(1,1)$-component of the product in (1) we see that $j \neq q$. Suppose that $j < q$.

Considering the $(1,1)$-component we see that $q-2+\gamma_0=0$, hence $\gamma_0=0$. Considering the $(1,2)$-component we obtain $\sum_{i<q,i\neq j}\beta_i=0$, which is a contradiction, since this yields $\beta_q+\beta_j=0$, i.e., $\beta_q=\beta_j$. Suppose now that $j=q+1$. Considering the $(j,1)$-component of (1), $j=1,\dots,q-4$, we obtain $\sum_{i=1}^{q-1}\alpha_i^{j-1}+\gamma_{j-1}=0$, which yields $\gamma_0=1$, $\gamma_1=\cdots=\gamma_{q-5}=0$. Considering the $(q-3,1)$-component gives $\sum_{i=1}^{q-1}\alpha_i^{q-3}+\gamma_{q-4}=0$, hence $\gamma_{q-4}=0$. So, $\Gamma=(1\colon 0\colon \cdots : 0)$. But the following argument shows that $K\cup\{\Gamma\}$ is not an arc, and this gives us the desired contradiction: choose pairwise different $\alpha_1,\dots,\alpha_{q-4}\in\mathbb{F}_q^\times$ which sum up to zero, and let V be the Vandermonde determinant of the α_i. Then

$$\det\begin{pmatrix} 1 & 1 & \cdots & 1 & 1\\ \alpha_1 & \alpha_2 & \cdots & \alpha_{q-4} & 0\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ \alpha_1^{q-5} & \alpha_2^{q-5} & \cdots & \alpha_{q-4}^{q-5} & 0\\ \alpha_1^{q-3} & \alpha_2^{q-3} & \cdots & \alpha_{q-4}^{q-3} & 0\end{pmatrix}=\Big(\sum_i\alpha_i\Big)\Big(\prod_i\alpha_i\Big)V=0.$$

This completes the proof. □

6 Relationship to Modular Representations of $\mathbf{GL}_n(\mathbb{F}_q)$

In this section we are going to point out a somewhat unexpected relationship between the classification problem for certain cyclic MDS-codes and certain modular representations of the general linear group over a finite field.

For $m\geq r$ the exponent set $\{0,1,\dots,r-1,m)$ is q-MDS if and only if the polynomial

$$\frac{\det\begin{pmatrix} X_0^0 & X_1^0 & \cdots & X_{r-1}^0 & X_r^0\\ X_0^1 & X_1^1 & \cdots & X_{r-1}^1 & X_r^1\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ X_0^{r-1} & X_1^{r-1} & \cdots & X_{r-1}^{r-1} & X_r^{r-1}\\ X_0^m & X_1^m & \cdots & X_{r-1}^m & X_r^m\end{pmatrix}}{\det\begin{pmatrix} X_0^0 & X_1^0 & \cdots & X_{r-1}^0 & X_r^0\\ X_0^1 & X_1^1 & \cdots & X_{r-1}^1 & X_r^1\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ X_0^{r-1} & X_1^{r-1} & \cdots & X_{r-1}^{r-1} & X_r^{r-1}\\ X_0^r & X_1^r & \cdots & X_{r-1}^r & X_r^r\end{pmatrix}} \tag{2}$$

does not have any roots outside $(\mathbb{F}_q^\times)^{r+1}\setminus\Delta$ where Δ is the zeroset of the discriminant (see Sect. 1). The above quotient is easily seen to be equal to the sum of all monomials in $r+1$ variables of degree $m-r$. This is a special case of a **Schur-function**. Such functions arise as characters of polynomial representations of GL_n over fields of characteristic 0. This classical result has a certain analogue in our case: the special Schur function derived as the quotient of the determinants above appear as characters of certain modular representations of $\mathrm{GL}_{r+1}(\mathbb{F}_q)$.

To be more specific, let ρ denote the representation of $\mathrm{GL}_{r+1}(\mathbb{F}_q)$ given by the action of this group on the space of homogeneous $r+1$-variate polynomials of degree $m-r$, and let ϕ be the character of ρ. Suppose that $A \in \mathrm{GL}_{r+1}(\mathbb{F}_q)$ is a diagonal matrix with entries $\alpha_0, \ldots, \alpha_r$. It acts on the space of homogeneous $r+1$-variate polynomials of degree $m-r$ by sending a monomial $\mu = \mu(X_0, \ldots, X_r)$ into $\mu(\alpha_0, \ldots, \alpha_r)\mu$. Hence, the value of $\phi(A)$ is given by $S(\alpha_0, \ldots, \alpha_r)$ where S is the sum of all monomials of degree $m-r$ in $r+1$ variables. In other words, S is equal to the expression given in (2).

Proposition 10. *Assumptions and notation being as above, the exponent set* $\{0, 1, \ldots, r-1, m\}$ *is q-MDS if and only if the character* ϕ *has no zeros in the union of those conjugacy classes of* $\mathrm{GL}_{r+1}(\mathbb{F}_q)$ *which have* $r+1$ *different eigenvalues in* $\mathbb{F}_q$.

Proof. The assertion is essentially proved above. If the exponent set is q-MDS, then $S(\alpha_0, \ldots, \alpha_r)$ is nonzero for any setting of pairwise different nonzero α_i in $\mathbb{F}_q$. Hence, since S is the value of ϕ at the conjugacy class of the diagonal matrix having $\alpha_0, \ldots, \alpha_r$ as diagonal entries, the assertion follows. Conversely, if ϕ does not have a zero on the union of the given conjugacy classes, then $S(\alpha_0, \ldots, \alpha_r)$ is nonzero for any setting of pairwise different nonzero α_i in $\mathbb{F}_q$, which implies that the given exponent set is MDS. □

As of yet, we do not know of any methods in modular representation theory which would resolve the question of whether or not the exponent set $\{0, 1, \ldots, r-1, m\}$ is MDS.

References

1. A.A. Bruen, J.A. Thas, and A. Blokhius. On MDS codes, arcs in $PG(n, q)$ with q even, and a solution of three fundamental problems of B. Segre. *Invent. math.*, 92:441–459, 1988.
2. L.R.A. Casse and D.G. Glynn. The solution to Beniamino Segre's problem $I_{r,q}$, $r = 3$, $q = 2^h$. *Geom. Ded.*, 13:157–163, 1982.
3. J. Harris. **Algebraic Geometry**. Number 133 in Graduate Texts in Mathematics. Springer-Verlag, New York, 1992.
4. J.W.P. Hirschfeld. **Projective Geometries over Finite Fields**. Clarendon Press, Oxford, 1979.
5. J.W.P. Hirschfeld. **Finite Projective Spaces of Three Dimensions**. Clarendon Press, Oxford, 1985.
6. H. Kaneta and T. Maruta. An elementary proof and an extension of Thas' theorem on k-arcs. *Math. Proc. Camb. Phil. Soc.*, 105:459–462, 1989.
7. J.H. van Lint. **Introduction to Coding Theory**, volume 86 of *Graduate Texts in Mathematics.* Springer Verlag, third edition, 1998.

8. J.H. van Lint and R.M. Wilson. On the minimum distance of cyclic codes. *IEEE Trans. Inform. Theory*, 32:23–40, 1986.
9. R. Matthews. Permutation properties of the polynomials $1 + x + \ldots + x^k$ over a finite field. *Proc. Amer. Math. Soc.*, 120:47–51, 1994.
10. G.L. Mullen and P.Jau-Shyong Shiue (editors). **Finite Fields: Theory, Applications, and Algorithms**. American Mathematical Society, Providence, Rhode Island, 1994.
11. B. Segre. Curve razionali normali e k-archi negli spazi finite. *Ann. Mat. Pura Appl. IV*, 39:357–379, 1955.
12. L. Storme and J.A. Thas. Complete k-arcs in $PG(n, q)$, q even. *Disc. Math.*, 106/107:455–469, 1992.
13. L. Storme and J.A. Thas. MDS codes and arcs in $PG(n, q)$ with q even: an improvement on the bounds of Bruen, Thas, and Blokhius. *J. Comb. Theory, Series A*, 62:139–154, 1993.
14. J.A. Thas. Normal rational curves and k-arcs in Galois spaces. *Rend. Mat.*, (6)1:331–334, 1968.
15. D. Wan. A p-adic lifting lemma and its applications to permutation polynomials. In **Proceedings of the International Conference on Finite Fields, Coding Theory, and Advances in Communication and Computing**, volume 141 of *Lecture Notes in Pure and Appl. Math.*, pages 209–216. Marcel Dekker, New York, 1992.

Computing Roots of Polynomials over Function Fields of Curves

Shuhong Gao[1] and M. Amin Shokrollahi[2]

[1] Department of Mathematical Sciences, Clemson University, Clemson, SC 29634 USA
[2] Bell Labs, Rm. 2C-353, 700 Mountain Ave, Murray Hill, NJ 07974, USA

Abstract We design algorithms for finding roots of polynomials over function fields of curves. Such algorithms are useful for list decoding of Reed-Solomon and algebraic-geometric codes. In the first half of the paper we will focus on bivariate polynomials, i.e., polynomials over the coordinate ring of the affine line. In the second half we will design algorithms for computing roots of polynomials over the function field of a nonsingular absolutely irreducible plane algebraic curve. Several examples are included.

1 Introduction

In this paper we will study the following problem: given a nonsingular absolutely irreducible plane curve $\mathcal{X}$ over the finite field $\mathbb{F}_q$, a divisor G on $\mathcal{X}$, and a polynomial H defined over the function field of $\mathcal{X}$, compute all zeros of H that belong to $L(G)$. Our interest in this problem stems mainly from recent list decoding algorithms [5,9,11] for Reed-Solomon and algebraic geometric codes. Originally, those algorithms found the roots of H by completely factoring it and looking for factors of degree one. This method is, however, not very efficient, especially if $\mathcal{X}$ is not a rational curve.

We will design more efficient algorithms by utilizing the fact that we are interested in *roots* of H rather than a complete factorization. For instance, suppose that $\mathcal{X}$ is the projective line, $L(G)$ is the space of univariate polynomials of degree $\leq k$ over $\mathbb{F}_q$, and $H(x, y)$ is a bivariate polynomial over $\mathbb{F}_q$. The problem is then that of finding polynomials f of degree $\leq k$ in the variable x such that $H(x, f) = 0$. For this problem we will design an algorithm that runs in time $O(k^2 b^3)$ where b is the degree of H in the variable y.

In the next section we review some well-known facts on the running times for certain operations on polynomials over finite fields and introduce an algorithm for computing roots of bivariate polynomials and demonstrate its use by means of several examples. In Section 3 we will attack the more general problem stated at the beginning of the introduction.

2 Roots of Polynomials over Rational Function Fields

In this paper we will mainly deal with probabilistic algorithms. The measure of an algorithm, usually called "time," will be the (expected) number of operations in $\mathbb{F}_q$, and usually we will use the "Soft O" notation to ignore logarithmic factors: $g = \tilde{O}(n)$ means that $g = O(n \log^c n)$ for some constant c. The term "deterministic time" of an algorithm is meant to imply that the algorithm in question is deterministic.

We briefly recall some well-known results. Two polynomials of degree $< n$ over $\mathbb{F}_q$ can be multiplied in deterministic time $\tilde{O}(n)$ [2, Th. 2.13]. The same is true for computing the division with remainder [2, Cor. 2.26], and the gcd of two such polynomials [2, Th. 3.13]. In particular, arithmetic operations in a given extension field $\mathbb{F}_{q^d}$ of $\mathbb{F}_q$ can be done in deterministic time $\tilde{O}(d)$. Furthermore, given two polynomials f and g, both of degree $< n$, and an integer ℓ, one can compute $f^\ell \bmod g$ in deterministic time $\tilde{O}(n \log \ell)$ using the "binary method" [2, pp. 3–4]. The roots of a polynomial of degree $< n$ over $\mathbb{F}_q$ can be computed in time $\tilde{O}(n \log q)$ [1, Theorem 5]. Without using fast algorithms, the running time for this task is $O(n^2 \log n \log q)$. Moreover, for any given d one can find an irreducible polynomial of degree d over $\mathbb{F}_q$ and hence construct the field $\mathbb{F}_{q^d}$ via an algorithm that runs in time $\tilde{O}(n^2 \log q)$ [1, Theorem 3].

In this section we present an algorithm which solves the following problem: given a polynomial $H(x, y)$ in two variables of degree m in x and degree b in y over a finite field $\mathbb{F}_q$ and a positive integer k, find all polynomials f in x of degree at most k such that $H(x, f(x)) \equiv 0 \bmod x^{k+1}$. For simplifying assertions on the running time we will assume the following.

Assumption 1. *H is a bivariate polynomial whose degree b in y satisfies $b \leq k$. We further assume that $\log q \leq k$, and that H is not divisible by x.*

Our algorithm is a modification of Kaltofen's [6] and is based on the following simple idea: let $H = \sum_{i=0}^{m} H_i(y)x^i$. We are looking for $f_0, \ldots, f_k \in \mathbb{F}_q$ and $\psi_0, \ldots, \psi_k \in \mathbb{F}_q[y]$ such that

$$\begin{aligned}(y - f_0 - f_1x - \cdots - f_kx^k)(\psi_0 + \psi_1x + \cdots + \psi_kx^k) &= \\ H_0 + H_1x + \cdots + H_kx^k \bmod x^{k+1}.\end{aligned} \tag{1}$$

f_0 is found by factoring H_0 over $\mathbb{F}_q$. If H_0 is squarefree, then multiplying out and comparing the "coefficients" of x^i for $i = 0, \ldots, k$ will successively reveal $f_1, \ldots, f_k$.

Algorithm 2. *On input a bivariate polynomial $H = \sum_{i=0}^{m} H_i(y)x^i$ over the field $\mathbb{F}_q$ such that $H_0(y)$ is squarefree, and an integer $k \geq 1$, the algorithm outputs a list $f^{(1)}, \ldots, f^{(s)}$ of polynomials of degree $\leq k$ such that $H(x, f^{(j)}(x)) \equiv 0 \bmod x^{k+1}$ for $j = 1, \ldots, s$.*

```
(1) Find all roots of $H_0$ in $\mathbb{F}_q$. Call them $\beta_1, \ldots, \beta_s$. If $s = 0$, then terminate
    the algorithm and output the empty set.
(2) For $\ell = 1, \ldots, s$ do
    (a) Set $\beta := \beta_\ell$.
    (b) For $i = 0, \ldots, k$ compute $h_i := H_i(\beta)$.
```

(c) *Set* $f_0 := \beta$, $\varphi_0 := (y-\beta)$, $\psi_0 := H_0/(y-\beta)$, $\eta_0 := H_0'(\beta)$.

(d) *For* $i = 1, \ldots, k$ *compute*

$$\begin{aligned}
\varphi_i &:= \frac{h_i - \varphi_1\eta_{i-1} - \cdots - \varphi_{i-1}\eta_1}{\eta_0}, \\
\psi_i &:= \frac{H_i - \varphi_i\psi_0 - \cdots - \varphi_1\psi_{i-1}}{\varphi_0}, \\
\eta_i &:= \psi_i(\beta), \\
f_i &:= -\varphi_i.
\end{aligned}$$

Theorem 3. *The above algorithm computes its output in time* $O(k^2b^2)$.

Proof. Let us first prove correctness. Fix ℓ. We will show by induction on i that

$$\left(\sum_{j=0}^{i}\varphi_j x^j\right)\left(\sum_{j=0}^{i}\psi_j x^j\right) \equiv H(x,y) \bmod x^{i+1}.$$

The assertion is true for $i = 0$: $\varphi_0\psi_0 = H_0$. Suppose now that the assertion is true for $i-1$. We only need to show that $\varphi_0\psi_i + \cdots + \varphi_i\psi_0 = H_i$. But, since $\eta_0 \neq 0$ by the assumption of squarefreeness of H_0, this is equivalent to

$$\begin{aligned}
\varphi_i &= \frac{H_i(\beta) - \varphi_1\eta_{i-1}(\beta) - \cdots - \varphi_{i-1}\eta_1(\beta)}{\eta_0(\beta)} \\
\psi_i &= \frac{H_i - \varphi_i\psi_0 - \cdots - \varphi_1\psi_{i-1}}{\varphi_0}
\end{aligned}$$

which is exactly what is computed in the most inner loop of the algorithm. Stated in terms of f, this result shows that

$$(y - f_0 - f_1x - \cdots - f_ix^i)(\psi_0 + \psi_1x + \cdots + \psi_ix^i) \equiv H(x,y) \bmod x^{i+1}.$$

Hence, $H(x,f) \equiv 0 \bmod x^{i+1}$.

For assessing the running time, note first that computing the β_i uses $O(b^2 \log b \log q)$ operations. Computation of the h_i takes at most kb operations using Horner's rule. In the inner loop (d) computing φ_i uses $O(i)$ operations, computing ψ_i uses $O(bi)$ operations (note that each H_i and each ψ_j is of degree at most b) and computing η_i uses another $O(b)$ operations. Hence, steps (a) to (d) use $O(k^2b)$ operations, which shows that the cost of Step (2) is $O(k^2b^2)$. Since $\log q \leq k$ by our general assumption, the result follows. □

Remark 4. Even if H_0 is not squarefree, the above algorithm works for a particular root β of H_0 as long as β is a simple root. In that case, the algorithm finds a solution f of $H(x,f) \equiv 0 \bmod x^{k+1}$ with $f(0) = \beta$ in time $O(bk^2)$.

We proceed with an example. Let

$$\begin{aligned}H(x,y) &= x^7+y^3x^5+y^3x^4+(y^4+y^2+y+1)x^3+(y^3+y^2+1)x^2+\\ &\quad (y^2+y)x+y^5+y^4+y^3+y\\ &=: H_7x^7+H_6x^6+H_5x^5+H_2x^2+H_1x+H_0\end{aligned}$$

over the base field $\mathbb{F}_2$. As $H_0(y)$ is squarefree, we can apply the foregoing algorithm. We set $k := 3$, i.e., we are looking for those polynomials $f \in \mathbb{F}_2[x]$ such that $H(x,f) \equiv 0 \bmod x^4$. One root of $H_0(y)$ over $\mathbb{F}_2$ is $\beta := 0$. Applying our algorithm we then obtain

$$\begin{array}{lll}\varphi_0 = y,\ \psi_0 = y^4+y^3+y^2+1, & \eta_0 = 1, & f_0 = 0,\\ \varphi_1 = 0,\ \psi_1 = y+1, & \eta_1 = 1, & f_1 = 0,\\ \varphi_2 = 1,\ \psi_2 = y^3, & \eta_2 = 0, & f_2 = 1,\\ \varphi_3 = 0, & & f_3 = 0.\end{array}$$

Hence, this setting of β yields the polynomial $f = x^2$. Another root of H_0 is 1. For this root we obtain

$$\begin{array}{llll}\varphi_0 = y+1, & \psi_0 = y^4+y^2+y, & \eta_0 = 1, & f_0 = 1,\\ \varphi_1 = 0, & \psi_1 = y, & \eta_1 = 1, & f_1 = 0,\\ \varphi_2 = 1, & \psi_2 = y^3+1, & \eta_2 = 0, & f_2 = 1,\\ \varphi_3 = 1, & & & f_3 = 1,\end{array}$$

which yields $f = x^3+x^2+1$. In both these cases we have in fact $H(x,f) = 0$. Polynomial division results in the factorization

$$H(x,y) = (y+x^2)\left(y+(x^3+x^2+1)\right)\left(y^3+y+(x^2+x+1)\right).$$

Let us now consider the case when H_0 is not squarefree. We will use the method of Newton polygons. Since we seek solutions modulo an arbitrary power of x, it is natural to work in the ring $\mathbb{F}_q[[x]]$ of formal power series in x. Denote by $\mathbb{F}_q[[x]][y]$ the ring of polynomials in y with coefficients in $\mathbb{F}_q[[x]]$, and by $\mathbb{F}_q[[x,y]]$ the ring of formal power series in x and y. Note that $\mathbb{F}_q[x,y] \subset \mathbb{F}_q[[x]][y] \subset \mathbb{F}_q[[x,y]]$, and they are all unique factorization domains. For any $H \in \mathbb{F}_q[[x,y]]$, its **Newton polygon** is defined to be the lower convex hull of all points (i,j) with $c_{ij} \neq 0$ and the point $(+\infty,+\infty)$ at infinity in the real Euclidean plane. For example, the Newton polygon of $H = y^5+(y^4+y^3)x^2+y^5x^4+y^3x^6+x^8+x^9$ is shown in Figure 1. A Newton polygon consists of (finite) line segments with nonzero slopes, called edges of the polynomial. For the above example, H has two edges with slopes 1 and $1/2$ respectively. The definition of Newton polygon given here is equivalent to that of Cassels [3, Sections 6.3 and 6.4] (note that $\mathbb{F}_q[[x]]$ is a complete local ring in which x is a prime). Denote each edge by a pair (ρ,ℓ) where ρ is its slope and ℓ its length on the x-axis, and denote a Newton polygon by a list of pairs $[(\rho_1,\ell_1),\ldots,(\rho_t,\ell_t)]$ of all the edges. So the above Newton

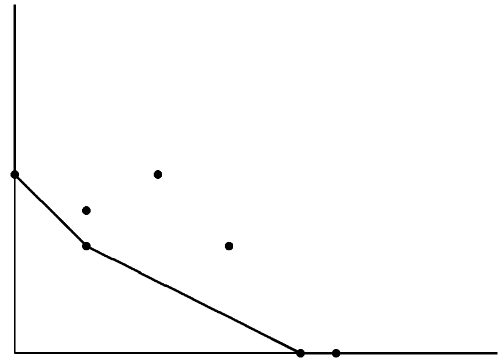


Figure1. Newton Polygon of H

polygon is denoted by $[(1,2),(1/2,6)]$. The notation also implicitly implies that we are interested in the Newton polygon only up to a translation in the real Euclidean plane.

The Newton polygon of a power series H carries a lot of information about the factors of H. The following result is from Cassels [3].

Lemma 5. *Let $H \in \mathbb{F}_q[[x]][y]$.*

(i) *If $G \in \mathbb{F}_q[[x]][y]$ divides H then the slope of every edge of G is also a slope of H.*

(ii) *Suppose that the Newton polygon of H is of the form $[(\rho_1, \ell_1), \ldots, (\rho_t, \ell_t)]$. Then there exist $G_i \in \mathbb{F}_q[[x]][y]$ with Newton polygon of the form $[(\rho_i, \ell_i)]$, $1 \le i \le t$, such that $H = G_1 \cdots G_t$.*

In particular, each edge of H corresponds to a distinct factor of H. These factors may still be reducible. To describe how they factor, we follow McCallum [7]. We need some more terminology. Let w be a rational number. For a monomial $x^i y^j$, we define its w-degree to be $i + wj$. For a power series $H \in \mathbb{F}_q[[x,y]]$, its w-order is defined to be the minimum w-degree of all nonzero terms of H, denoted by $o_w(H)$. For a polynomial $H \in \mathbb{F}_q[x,y]$, its w-degree is defined to be the maximum w-degree of all nonzero terms of H, denoted by $d_w(H)$. A polynomial is called a w-form if all of its nonzero terms have the same w-degree. Obviously, any polynomial can be written as a sum of w-forms. The initial w-form of H is the w-form in H that has the smallest w-degree. It is straightforward to see that if ρ is a slope of H and $w = 1/\rho$, then the edge with slope ρ corresponds exactly to the initial w-form of H (see below). The next result from McCallum [7, Theorem 2.2] says that the factors of the initial w-form give factors of H itself.

Lemma 6. *Let $w > 0$ be a rational number and $H \in \mathbb{F}_q[[x]][y]$. Suppose that the initial w-form h_0 of H is not divisible by x. If $h_0 = f_0 g_0$ for $f_0, g_0 \in$*

$\mathbb{F}_q[x,y]$ and $\gcd(f_0, g_0) = 1$ *then there exist $F, G \in \mathbb{F}_q[[x]][y]$ such that $H = FG$, $\deg_y F = \deg_y f_0$, and f_0 (resp. g_0) is the initial w-form of F (resp. G).*

We describe below more explicitly how an initial w-form factors. Let $H \in \mathbb{F}_q[[x]][y]$. Consider a typical edge of H, say from $A = (t, h)$ to $B = (t-u, h+v)$ where $t \geq u > 0$, $h \geq 0$ and $v > 0$ are integers. The slope of the edge is $\rho = v/u$. Let $\ell = \gcd(u, v)$, $u_1 = u/\ell$, $v_1 = v/\ell$ and $w = 1/\rho = u/v = u_1/v_1$. Any integral point on the edge AB must be of the form $A + (-u_1 i, v_1 i)$ for some $0 \leq i \leq \ell$. All the terms of H that lie on the edge have the same w-degree $t + wh$. Any point above the egde has higher w-degree. Thus the initial w-form of H is

$$H_0 = \sum_{i=0}^{\ell} c_i x^{t-u_i i} y^{h+v_1 i} = x^{t-u} \cdot y^h \cdot x^u \sum_{i=0}^{\ell} c_i \left(\frac{y^{v_1}}{x^{u_1}}\right)^i = x^{t-u} \cdot y^h \cdot x^u \tilde{H}_0(z)$$

for some $c_i \in \mathbb{F}_q$, where $z = y^{v_1}/x^{u_1}$ and $\tilde{H}_0(z) = \sum_{i=0}^{\ell} c_i z^i$. Note that $x^u \tilde{H}_0(z) \in \mathbb{F}_q[[x]][y]$. In the following, we call $\tilde{H}_0$ the reduced polynomial of H_0. Note that $\tilde{H}_0$ has degree ℓ and $\tilde{H}_0(0) \neq 0$, since H must have two nonzero terms corresponding to the vertices A and B on its Newton polygon. If $\gcd(u, v) = 1$ then there is no integral point on the edge AB except the end points, and so AB is the shortest line segment with slope $\rho = v/u$. By Lemma 5 (i), $x^u \tilde{H}_0$ can not factor (in $\mathbb{F}_q[[x]][y]$). In this case $x^u \tilde{H}_0$ must be (absolutely) irreducible and can be lifted to a factor of H by Lemma 5. Now suppose $\ell = \gcd(u, v) > 1$. Since $\tilde{H}_0$ is a univariate polynomial, it factors into linear factors over an extension field of $\mathbb{F}_q$. Each linear factor $z - \beta$ of $\tilde{H}_0$ gives an absolutely irreducible factor $y^{v_1} - \beta x^{u_1}$ of H_0.

Lemma 7. *Let $H \in \mathbb{F}_q[[x]][y]$ with $H(0,0) = 0$ and H not divisible by y. Then any factor $y - f(x)$ of H, where $f(x) \in \mathbb{F}_q[[x]]$ and $f(0) = 0$, is of the form*

$$y - (\beta x^w + \textit{terms of higher degrees in } x)$$

where $w > 0$ is an integer and $\beta \in \mathbb{F}_q$ such that $1/w$ is a slope of the Newton polygon of H and β is a root of the reduced polynomial of the initial w-form of H.

Proof. Suppose $f(x) = f_1 x + f_2 x^2 + \cdots \in \mathbb{F}_q[[x]]$ and $y - f(x)$ divides H. Let $w > 0$ be the smallest integer such that $f_w \neq 0$. Then the Newton polygon of $y - f(x)$ has only one edge starting at $(0,1)$ and ending at $(w, 0)$ whose slope is obviously $1/w$. By Lemma 5 (i), $1/w$ is also a slope of H. Note that the initial w-forms are multiplicative, i.e., $(FG)_0 = F_0 G_0$ for $F, G \in \mathbb{F}_q[[x]][y]$. Let H_0 be the initial w-form of H. As $y - f_w x^w$ is the initial w-form of $y - f(x)$, we see that $y - f_w x^w$ divides H_0. By the above argument, f_w is a root of the reduced polynomial $\tilde{H}_0$ of H_0. □

Lemma 7 shows clearly how to find solutions for our problem. Let $H = H_0 + H_1 x + \cdots + H_m x^m \in \mathbb{F}_q[x, y]$. We want to find all solutions $f(x) \in \mathbb{F}_q[x]$

for (1). Suppose that $y = \beta$ is a root of H_0 of multiplicity $v > 1$. Make a change of variables $y_1 = y - \beta$ and $G = H(x, y_1 + \beta)$. To lift $y - \beta$, we need to find all factors of G of the form $y_1 - f_1 x - \cdots - f_k x^k$. If $y_1 \mid G$ then $y_1 = y - \beta$ is a solution. In this case, we can divide out y_1 in G and denote the resulting polynomial by G'. If $G'(0,0) \neq 0$ then G has no other factor of the form $y_1 - f_1 x - \cdots - f_k x^k$. So we may assume that G is not divisible by y_1 and $G(0,0) = 0$. Thus G is of the form of H as in Lemma 7 with respect to x and y_1. Compute the Newton polygon of G (with respect to x and y_1). We find all the egdes of G with slopes of the form $1/w$ for some integers w. For each such edge, find all the linear factors $y_1 - \beta_1 x^w$ of the initial w-form G_0 of G where $\beta_1 \in \mathbb{F}_q$ is a root of the reduced polynomial $\tilde{G}_0$ of g_0. When G has no such edges or $\tilde{G}_0$ has no roots in $\mathbb{F}_q$ then $y_1 = y - \beta$ can not be lifted to a factor $y - f(x)$ of H with $f_0 = \beta$. If β_1 is a simple root of $\tilde{G}_0$, we will show below how to lift such a partial solution. So suppose that β_1 is a multiple root. Make another change of variables $y_2 = y_1 - \beta_1 x^w$ and let $G_1 = G(x, y_2 + \beta_1 x^w)$. We can compute the Newton polygon again for G_1 and repeat the above procedure. For G_1, we need only to consider the edges of slopes $1/w_1$ with $w_1 > w$, so that higher powers of x will be added in the changes of variables. Since the w's increase at least by 1 each time and we only need powers of x up to k, this procedure will stop after at most k iterations. As is described below, all such partial solutions can be lifted to true solutions.

We illustrate this method by means of an example. Let H be as in the above example and we compute over $\mathbb{F}_2$. The first change of variables is not needed. H has two edges of slopes 1 and 1/2 respectively. For the edge of slope 1, $w = 1$ and the initial w-form of H equals $H_0 = y^5 + y^3x^2 = y^3(y^2 + x^2) = y^3(y + x)^2$. So $\beta = 1$ is a multiple root. Let $y_1 = y - x$ and

$$G = H(x, y_1 + x) = x^9 + x^8 + y^2x^7 + (y^3 + 1)x^6 + y^4x^5 + y^5x^4 + y^2x^3 + \\ (y^4 + y^3)x^2 + y^4x + y^5$$

The Newton polygon of G is shown in Figure 2. Note that G has two edges of slopes 1 and 2/3 respectively, none of them is of the form $1/w_1$ with w_1 an integer $> w = 1$. Hence $y - x$ can not be lifted to a factor $y - f(x)$ of H with $f_0 = 0$ and $f_1 = 1$. Consider the edge of H with slope 1/2. Then $w = 2$ and the intial w-form is $G_0 = y^3x^2 + x^8 = x^8(z^3 + 1) = x^8(z + 1)(z^2 + z + 1)$ where $z = y/x^2$. As $\beta = 1$ is a simple root of $z^3 + 1$, $y + x^2$ can be lifted to a factor of H. Therefore H has only one solution which is a lift of $y + x^2$.

Algorithm 8. *(Finding partial solutions) On input $H \in \mathbb{F}_q[x, y]$ and an integer $k > 1$, this algorithm compute a list L of all triples (w, β, g) where w is ∞ or an integer > 0, $\beta \in \mathbb{F}_q$, and $g \in \mathbb{F}_q[x]$ is a polynomial such that if $w = \infty$ then $H(x, g) \equiv 0 \bmod x^{k+1}$, and if $w < \infty$ then β is a simple root of the reduced initial w-form of $H(x, y + g)$.*

(0) *Initialization: $w := 0$, $g := 0$, and $L = \{\}$.*

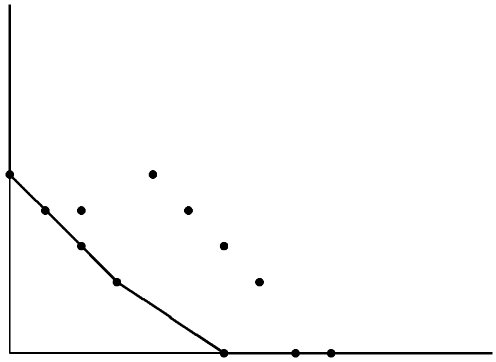


Figure2. Newton Polygon of G

(1) *Compute the initial w-form H_0 of H and write it in the reduced form $\tilde{H}_0 \in \mathbb{F}_q[z]$. Find all the roots β of $\tilde{H}_0$ in $\mathbb{F}_q$.*

(2) *For all roots β from Step 1, do the following:*

  (a) *If β is a simple root then $L := L \cup \{(w, \beta, g)\}$;*

  (b) *If β is a multiple root then compute*

$$H := H(x, y + \beta x^w) \bmod x^{k+1}, \textit{ and } g := g + \beta x^w.$$

  If y divides H then $L := L \cup (\infty, \beta, g)$ and set $H := H/y^a$ where a is the largest integer such that $y^a \mid H$.

  (i) *Compute the Newton polygon of H and the slopes of the edges.*

  (ii) *For each slope of the form $1/d$ with $d > w$ where d is a positive integer, set $w := d$ and go to Step (1).*

(3) *Return the list L.*

It is important to note that Step (2) of the algorithm is executed in parallel for all roots β. This means that the algorithm traverses the computation tree in a breadth-first fashion. A partial solution is built up on each path separately. One can implement the algorithm more efficiently in a depth-first fashion.

The algorithm returns two types of partial solutions (w, β, g). For $w < \infty$, we will show below how to lift g to a solution f modulo any power of x. For $w = \infty$, g is already a solution modulo x^{k+1}. In the latter case, however, g may not in general be liftable to a solution modulo a higher power of x.

Theorem 9. *Algorithm 8 correctly returns all partial solutions with $O(b^3k^2)$ operations.*

Proof. The correctness follows from the discussion above. On the running time, the dominant cost is at Step (2b) for updating H and computing Newton polygons. Since H has at most bk nonzero terms, $H(x, y+\beta x^w) \bmod x^{k+1}$

can be computed by Horner's rule (on y) in time $O(b^2k)$, and the Newton polygon of H can also be computed in this time. Each β here represents a term in a potential solution. Since H has at most b solutions and each one has at most k terms, Step (2b) is executed at most bk times. So the whole algorithm runs in time as claimed. □

Remark 10. When the degree b in y of H is large, Algorithm 8 can be improved by the following strategy. By Lemma 5, each edge of H corresponds to a factor of H. Only the factors of edges with slopes of the form $1/w$ with w an integer can have factors linear in y. Hence one can factor H at each stage according to the edges of the Newton polygon. Then for each factor of an edge with slope $1/w$, make a translation of variables and repeat the same procedure to the new polynomial. Since the degree in y of the factors decreases at each stage, this modified version of Algorithm 8 will be faster.

Finally, we show how to lift the partial solutions (w, β, g), $w < \infty$, returned by Algorithm 8. McCallum [7] discusses a more general case. Since we are dealing with linear factors, the algorithm here will be much simpler and in fact it will be exactly Algorithm 2.2. Let $G = H(x, y + g)$, which is computed at Step 2.b in Algorithm 8. Write G into a sum of w-forms

$$G = G_u + G_{u+1} + \cdots + G_{u+v}$$

where $u = o_w(G)$, $u + v = d_w(G)$, and G_i is either zero or a w-form of w-degree i for $u \le i \le u + v$. So G_u is the initial w-form of G. Note that G_i is of the form

$$G_i = x^i \sum_{j=0}^{\ell} c_j \left(\frac{y}{x^w}\right)^j$$

for some integer ℓ, where $c_j \in \mathbb{F}_q$. Let

$$\tilde{G}_i = \sum_{j=0}^{\ell} c_j z^j \in \mathbb{F}_q[z]$$

where $z = y/x^w$. Then

$$G = x^u(\tilde{G}_u + \tilde{G}_{u+1}x + \cdots + \tilde{G}_{u+v}x^v).$$

Note that $\tilde{G}_u$ is equal to the reduced polynomial of G_u up to a factor of a power of y. By the design of Algorithm 8, β is a simple root of $\tilde{G}_u$ and $y - \beta x^w$ is a factor of G_u. We want to find $f_0 = \beta, f_1, \ldots, f_{k-w} \in \mathbb{F}_q$ such that

$$(y - f_0x^w - f_1x^{w+1} - \cdots - f_{k-w}x^k)\psi \equiv G \bmod x^{k+1} \qquad (2)$$

for some $\psi \in \mathbb{F}_q[x, y]$. Since $o_w(G) = u$ and $o_w(y - f(x)) = w$, we have $o_w(\psi) = u - w$. If we write ψ into a sum of w-forms and use the reduced form as we did for G, then we have

$$\psi = x^{u-w}(\tilde{\psi}_0 + \tilde{\psi}_1x + \cdots + \tilde{\psi}_{k-w}x^{k-w})$$

where $\tilde{\psi}_i \in \mathbb{F}_q[z]$ of appropriate degrees. Now divide the equation (2) on both sides by x^u, we have

$$(z - f_0 - f_1x - \cdots - f_{k-w}x^{k-w})(\tilde{\psi}_0 + \tilde{\psi}_1x + \cdots + \tilde{\psi}_{k-w}x^{k-w}) \equiv$$
$$\tilde{G}_u + \tilde{G}_{u+1}x + \cdots + \tilde{G}_{u+v}x^v \bmod x^{k+1-u}.$$

This is exactly the type of the equation (1) we started with. Since β is a simple root of $\tilde{G}_u$, Algorithm 2.2 can be applied to find a solution $f(x) = f_0 + f_1x + \cdots + f_{k-w}x^{k-w} \in \mathbb{F}_q[x]$ with $f_0 = \beta$ for the above equation. Then $g + x^w f(x)$ is a solution for the equation (1).

Theorem 11. *Algorithms 2 and 8 find all solutions of the Equation (1) in time $O(b^3k^2)$.*

Proof. Since Algorithm 2 lifts a partial solution in time $O(bk^2)$ and there are at most b solutions, all the solutions of (1) can be found in time $O(b^3k^2)$. □

Example. Consider the polynomial

$$H = x^6 + (y+1)x^5 + x^4 + x^3 + (y^3 + y)x^2 + y^2x + (y^4 + y^3)$$

over $\mathbb{F}_2$. $H_0 = y^3 + y^4 = y^3(y+1)$ has a simple root $y = 1$ and a triple root $y = 0$. The first one can be lifted to a true solution by Algorithm 2.2. To lift the second one, we need to find the Newton polygon of H, which happens to have only one edge of slope 1. So let $w = 1$. The initial w-form of H is

$$h_0 = y^3 + y^2x + yx^2 + x^3 = x^3(\frac{y}{x} + 1)^3.$$

Thus $\beta = 1$ is a triple root of $\tilde{h}_0$. Make a translation of variables $y_1 = y - x$. Then

$$\begin{aligned} G = H(x, y_1 + x) &= x^5y_1 + x^2y_1^3 + y_1^2x^3 + x^4y_1 + y^3 + y_1^4 \\ &= y_1(x^5 + x^2y_1^2 + y_1x^3 + x^4 + y_1^2 + y_1^3). \end{aligned}$$

Hence $y_1 = y + x$ is a solution of H (modulo any power of x). Let $G_1 = G/y_1$. Its Newton polygon has one edge of slope 1/2. Let $w = 2$. Then the initial w-form of G_1 is

$$g_0 = y_1^2 + x^4 = x^4(y_1/x^2 + 1)^2.$$

So $\beta = 1$ is a double root of $\tilde{g}_0$. Make another translation of variables $y_2 = y_1 - x^2$ and

$$G_2 = G_1(x, y_2 + x^2) = x^4y_2 + y_2x^3 + y_2^2 + y_2^3 = y_2(x^4 + x^3 + y_2 + y_2^2).$$

So $y_2 = y_1 - x^2 = y - x - x^2$ is solution (modulo any power of x). The Newton polygon of $G_3 = G_2/y_2$ has an edge of slope 1/3. Let $w = 3$. Then the initial w-form of G_3 is $y_2 + x^3 = x^3(y^2/x^3 + 1)$ for which $\beta = 1$ is a simple root. So $y_2 - x^3 = y - x - x^2 - x^3$ can be lifted to a solution modulo any power of x. In total there are four solutions: $y - x$, $y - x - x^2$, and lifts of $y - 1$ and $y - x - x^2 - x^3$.

3 The General Case

In this section we assume familiarity with basic concepts from the theory of algebraic curves. (See, e.g., [10].) Let $\mathcal{X}$ be a nonsingular curve given as the zeroset of an absolutely irreducible polynomial $F \in \mathbb{F}_q[x, y]$ and let $R := \mathbb{F}_q[x, y]/(F)$ denote its coordinate ring. Let G be a divisor on $\mathcal{X}$ defined over $\mathbb{F}_q$ and let $L(G)$ denote the linear space of G. Assume that we are given a basis $\varphi_1, \ldots, \varphi_\ell$ of $L(G)$ such that each $\varphi_i \in R$. We are interested in computing the roots in $L(G)$ of a polynomial

$$H(T) = u_b T^b + \cdots + u_1 T + u_0$$

with coefficients $u_i \in R$. The algorithm we will present below is a generalization and simplification of that stated in [9] for polynomials of degree $b = 2$.

Assumption 12. *For the rest of this section we will assume that* $\deg F =: D \geq 3$*, that* $k := \deg G \geq 2D^2$*, that the basis functions* φ_i *of* $L(G)$ *are represented modulo* F *as bivariate polynomials of degree* $\leq B$*, and that the functions* u_i *are represented modulo* F *as bivariate polynomials of degree* $\leq C$*. Furthermore, we assume that* $b, \log q \leq k$*.*

The first step of the algorithm to be presented below consists of finding an $\mathbb{F}_{q^d}$-rational solution $\mathfrak{p} = (a, b)$ of $F(x, y) = 0$ where $d > k$ and where either a or b is a primitive element of the extension $\mathbb{F}_{q^d}/\mathbb{F}_q$. We call $\mathfrak{p}$ an (affine) point of $\mathcal{X}$ (or of F) of degree d over $\mathbb{F}_q$.

Algorithm 13. *On input an irreducible nonsingular bivariate polynomial* $F(x, y)$ *over* $\mathbb{F}_q$ *of degree* D *and an integer* $d \geq 2D^2$ *the algorithm computes an affine point* $\mathfrak{p}$ *of the zeroset of* F *of degree* d *over* $\mathbb{F}_q$*.*

(1) *Construct the field* $\mathbb{F}_{q^d}$*.*
(2) *Randomly select an element* ζ *of* $\mathbb{F}_{q^d}$ *until a primitive element of* $\mathbb{F}_{q^d}/\mathbb{F}_q$ *is found.*
(3) *Test whether* $g(\zeta, y)$ *has a zero* y_0 *in* $\mathbb{F}_{q^d}$*. If yes, then output* $\mathfrak{p} = (\zeta, y_0)$*. If not, then test whether* $g(x, \zeta)$ *has a zero* x_0 *in* $\mathbb{F}_{q^d}$*. If yes, then output* $\mathfrak{p} = (x_0, \zeta)$*. If not, then go back to Step* (2)*.*

Theorem 14. *The above algorithm correctly computes its output in time*

$$\tilde{O}(d^2 D \log q + d^3).$$

Proof. Let N_i denote the number of solutions in $\mathbb{F}_{q^i}^2$ of $F(x, y) = 0$. We first prove that

$$|N_i - q^i| \leq D^2 q^{i/2}. \tag{3}$$

Since F is nonsingular, the genus g of the zeroset of F is $(D-1)(D-2)/2$ [4, Chap. 8, Prop.5]. The number $\tilde{N}_i$ of $\mathbb{F}_{q^i}$-rational points of the zeroset $\mathcal{X}$ of F

in the projective plane over $\mathbb{F}_q$ satisfies $|\tilde{N}_i - q^i - 1| \le 2gq^{i/2}$ by the Hasse-Weil inequality. Let $\tilde{F}(X,Y,Z)$ be the homogenized version of F. The number of $\mathbb{F}_{q^i}$-rational points of $\mathcal{X}$ in the projective plane over $\mathbb{F}_{q^i}$ which have $Z=0$ is obviously upper bounded by $2D$. As a result we have $\tilde{N}_i \le N_i + 2D$, which gives us

$$q^i+1-(D-1)(D-2)q^{i/2}-2D \le N_i \le q^i+1+(D-1)(D-2)q^{i/2} \le q^i+D^2q^{i/2}.$$

It remains to show that $q^i+1-(D-1)(D-2)q^{i/2}-2D \ge q^i - D^2q^{i/2}$. A simple manipulation leads to the equivalent condition $q^{i/2} \ge (2D-1)/(3D-1)$ which is trivially true, as $D \ge 3$ by Assumption 12.

Next, we compute a lower bound for the number N of those solutions (a,b) of $F(a,b)=0$ such that a or b is primitive. Obviously, $N = N_d - \sum_{\ell|d,\ell<d} N_\ell \ge N_d - \sum_{\ell=1}^{\lfloor d/2 \rfloor} N_\ell$, since an element of $\mathbb{F}_{q^d}$ is primitive iff it does not belong to any proper subfield of $\mathbb{F}_{q^d}$. Use of (3) yields

$$\begin{aligned} N &\ge q^d - D^2q^{d/2} - q\frac{q^{d/2}-1}{q-1} - \sqrt{q}D^2\frac{q^{d/4}-1}{\sqrt{q}-1} \\ &\ge q^d - q^{d/2}(D^2 + q + \sqrt{q}D^2q^{-d/4}). \end{aligned}$$

The number of primitive elements of $\mathbb{F}_{q^d}$ is $q^d - \sum_{\ell|d,\ell<d} q^\ell \ge q^d - \sum_{\ell=1}^{\lfloor d/2 \rfloor} q^\ell \ge q^d - q^{d/2+1}$. As a result, a random element in $\mathbb{F}_{q^d}$ is primitive over $\mathbb{F}_q$ with at most a constant probability. This shows that Step (2) is performed, on average a constant number of times. After this step we will have obtained a uniform randomly chosen primitive element of $\mathbb{F}_{q^d}$. The probability p that a random primitive element of $\mathbb{F}_{q^d}$ is either the x- or the y-coordinate of a solution of $F(x,y)=0$ satisfies

$$p \ge \frac{1-q^{-d/2}(D^2+q+\sqrt{q}D^2q^{-d/4})}{1-q^{-d/2+1}} \ge \frac{1-q^{-d/2}(2D^2+q)}{1-q^{-d/2+1}}.$$

(Note that $q^{-d/4} < q^{-1/2}$.) Now observe that $2D^2 \le d \le q^{d/4}$ and that $(2D^2+q) \le 2q^{d/4}$. This implies

$$p \ge \frac{1-2q^{-d/4}}{1-q^{-d/2+1}}.$$

Hence, Step (3) is performed on average a constant number of times.

Let us now focus on the running time. Step (1) uses $\tilde{O}(d^2 \log q)$ operations. Testing primitivity of an element ζ is done by computing $1, \zeta, \ldots, \zeta^{d-1}$ in the polynomial basis ($\tilde{O}(d^2)$ operations), and testing linear independence of these elements as vectors ($O(d^3)$ operations). So, Step (2) uses $O(d^3)$ operations. Each iteration of Step (3) consists of computing $F(\zeta,y)$ (or $F(x,\zeta)$) which uses $O(D^2)$ operations over $\mathbb{F}_{q^d}$, i.e., $\tilde{O}(dD^2)$ operations over $\mathbb{F}_q$, and of computing the roots of a univariate polynomial of degree $\le D$ over $\mathbb{F}_{q^d}$ which takes $\tilde{O}(D \log q^d)$ operations over $\mathbb{F}_q$, i.e., $\tilde{O}(d^2 D \log q)$ $\mathbb{F}_q$-operations. □

Remark 15. The assumption $d \geq 2D^2$ in Algorithm 13 is related to applications in coding theory where one assumes that $k = \deg G > 2g - 2$, where g is the genus of the curve. It can be weakened at the expense of a more tedious analysis. However, we remark that points of degree d may not exist for all values of d. For instance, the Hermitian curve $x^3 = y^2 + y$ does not have any points of degree 2 over $\mathbb{F}_4$.

The final algorithm now follows.

Algorithm 16. *Given an irreducible algebraic nonsingular bivariate polynomial F, a divisor G of the zeroset of F defined over $\mathbb{F}_q$, basis functions $\varphi_1, \ldots, \varphi_\ell$ of $L(G)$ and functions $u_0, \ldots, u_b \in \mathbb{F}_q[x,y]/(F)$ satisfying the conditions in Notation 12, the algorithm computes a list $f^{(1)}, \ldots, f^{(s)}$ of at most b functions in $L(G)$ which includes any $f \in L(G)$ such that $H(f) = 0$, where $H = \sum_{i=0} u_i T^i$.*

(1) *Using Algorithm 13 compute an affine point $\mathfrak{p}$ of degree $d = k+1$ over $\mathbb{F}_q$ of the zeroset of F.*
(2) *Compute $\varphi_1(\mathfrak{p}), \ldots, \varphi_\ell(\mathfrak{p})$ and represent them as d-dimensional vectors $\mathbf{v}_1, \ldots, \mathbf{v}_\ell$ over $\mathbb{F}_q$.*
(3) *Compute the values $u_0(\mathfrak{p}), \ldots, u_b(\mathfrak{p})$.*
(4) *Compute the zeros $\beta_1, \ldots, \beta_s$ of the polynomial $u_0(\mathfrak{p}) + u_1(\mathfrak{p})x + \cdots + u_b(\mathfrak{p})x^b$ in the field $\mathbb{F}_{q^d}$ and represent them as d-dimensional vectors $\mathbf{b}_1, \ldots, \mathbf{b}_s$ over $\mathbb{F}_q$.*
(5) *Compute vectors $\mathbf{h}_i = (h_{1,i}, \ldots, h_{\ell,i})^\perp \in \mathbb{F}_q^\ell$, $i = 1, \ldots, s$ such that*

$$(\mathbf{v}_1 \mid \cdots \mid \mathbf{v}_\ell)\,(\mathbf{h}_1 \mid \cdots \mid \mathbf{h}_s) = (\mathbf{b}_1 \mid \cdots \mid \mathbf{b}_s)$$

over $\mathbb{F}_q$ and output $f^{(i)} = h_{1,i}\varphi_1 + \cdots + h_{\ell,i}\varphi_\ell$.

Theorem 17. *The above algorithm correctly computes its output in time*

$$\tilde{O}(k^2 D \log q + k^3 + k^2 B^2 + kbC^2 + k^2 b \log q).$$

Proof. We first prove correctness. If $f \in L(G)$ is such that $H(f) = 0$, then we have that $\sum_{i=0}^b u_i(\mathfrak{p}) f(\mathfrak{p}) = 0$, i.e., $f(\mathfrak{p})$ is one of the β_i. Writing $f = \sum_i h_i \varphi_i$, we see that $h_1, \ldots, h_\ell$ satisfy the equations in Step (5). We now prove that, for each i, the solution to this system is unique. Indeed, two solutions would give rise to functions $f, g \in L(G)$ defined over $\mathbb{F}_q$ such that $(f-g)(\mathfrak{p}) = 0$. But then $(f-g)(\mathfrak{p}^\sigma) = 0$ for all the d different automorphisms σ of $\mathbb{F}_{q^d}/\mathbb{F}_q$. This shows that $f - g$ has more zeros than poles, which implies that $f = g$. We infer that f is one of the $f^{(i)}$'s.

Step (1) of the algorithm uses $\tilde{O}(d^2 D \log q)$ operations in $\mathbb{F}_q$ by Theorem 14. Each φ_i is represented by a bivariate polynomial of degree $\leq B$. So, computing $\varphi_i(\mathfrak{p})$ uses, in the worst case, $O(B^2)$ operations in $\mathbb{F}_{q^d}$, i.e., $\tilde{O}(dB^2) = \tilde{O}(kB^2)$ operations in $\mathbb{F}_q$. Since there are ℓ of these functions and $\ell \leq k+1$, Step (2) requires $O(k^2 B^2)$ time. Similarly, computing the $u_i(\mathfrak{p})$

uses $\tilde{O}(bkC^2)$ $\mathbb{F}_q$-operations. Step (4) uses $\tilde{O}(b\log q^d) = \tilde{O}(db\log q)$ operations in $\mathbb{F}_{q^d}$, i.e., it requires $\tilde{O}(k^2 b\log q)$ $\mathbb{F}_q$-operations. The cost of Step (5) is $O(b^2k)$: it consists of reducing a $d \times 2s$-matrix to echelon form using row operations, and $s \leq b$. □

Remark 18. (1) In applications to coding theory one usually has a fixed divisor G (corresponding to fixing the code) and one wants to compute zeros in $L(G)$ for different polynomials H. In this case one can compute the point $\mathfrak{p}$ and the evaluation of the φ_i at $\mathfrak{p}$ in advance. Neglecting the cost of this preconditioning, the running time of the algorithm would then be $\tilde{O}(kbC^2 + b^2k + k^2b\log q)$. Assuming that b is a constant and that $C, \log q \leq k$ (both reasonable assumptions in list decoding scenarios), this gives a running time of $\tilde{O}(k^3)$.

(2) If the functions u_i and φ_i are not polynomials in x and y, it is still possible (though tedious) to analyze the running time of the algorithm. The only major change in the algorithm is to ensure that the point $\mathfrak{p}$ found has the property that the functions u_i can be evaluated at it.

(3) The assumption that the curve $\mathcal{X}$ has a *nonsingular* plane model was only needed to bound the number of solutions of $F(x,y)$ over extensions of $\mathbb{F}_q$. One can also bound these numbers without this assumption [8] and can obtain similar (though a little worse) results.

As was pointed out in Remark 15, the assumption $d \geq 2D^2$ for the existence of points of degree d can be weakened. In the next example, we will find a point of degree 6 of the degree 5 Hermitian curve over $\mathbb{F}_2$ given by the equation $x^5 = y^4 + y$. Let Q be the common pole of x and y. We are interested in zeros of the polynomial

$$\begin{aligned} H(T) &= T^3 + (x+y+1)T^2 + (x^2+y)T + (x^2y + x^3 + xy + x^2) \\ &=: T^3 + u_2T^2 + u_1T + u_0 \end{aligned}$$

among the elements of $L(5Q) = \langle 1, x, y\rangle$. The first step consists of finding a point of degree 6. We represent $\mathbb{F}_{2^6}$ as $\mathbb{F}_2(\zeta)$ with $\zeta^6 + \zeta + 1 = 0$. Applying Algorithm 13, we find $\mathfrak{p} = (\zeta^2 + \zeta, \zeta^4 + \zeta^2)$.

The next step of the algorithm is to find the zeros of the polynomial

$$T^3 + u_2(\mathfrak{p})T^2 + u_1(\mathfrak{p})T + u_0(\mathfrak{p}) = T^3 + (\zeta^4 + \zeta + 1)T^2 + \zeta^3$$

in $\mathbb{F}_{2^6}$. They turn out to be $\beta_1 = \zeta^2 + \zeta$, $\beta_2 = \zeta^2 + \zeta + 1$, and $\beta_3 = \zeta^4 + \zeta$. Now we represent elements of $\mathbb{F}_{2^6}$ with respect to the $\mathbb{F}_q$-basis $1, \zeta, \ldots, \zeta^5$ and solve the system of equations given in Step (5):

$$\begin{pmatrix} 1\,0\,0 \\ 0\,1\,0 \\ 0\,1\,1 \\ 0\,0\,0 \\ 0\,0\,1 \\ 0\,0\,0 \\ 0\,0\,0 \end{pmatrix} \begin{pmatrix} h_{1,1} & h_{1,2} & h_{1,3} \\ h_{2,1} & h_{2,2} & h_{2,3} \\ h_{3,1} & h_{3,2} & h_{3,3} \end{pmatrix} = \begin{pmatrix} 0\,1\,0 \\ 1\,1\,0 \\ 1\,1\,0 \\ 0\,0\,0 \\ 0\,0\,1 \\ 0\,0\,0 \\ 0\,0\,0 \end{pmatrix}.$$

Solving this system yields the solutions $(0,1,0)^\perp$, $(1,1,0)^\perp$, and $(0,1,1)^\perp$ for $(h_1,h_2,h_3)^\perp$ which leads to the functions $f^{(1)} = x$, $f^{(2)} = x+1$, and $f^{(3)} = x+y$.

References

1. M. Ben-Or. Probabilistic algorithms in finite fields. In *Proceedings of the 22nd IEEE Symposium on Foundations of Computer Science*, pages 394–398, 1981.
2. P. Bürgisser, M. Clausen, and M.A. Shokrollahi. **Algebraic Complexity Theory**, volume 315 of *Grundlehren der Mathematischen Wissenschaften.* Springer Verlag, Heidelberg, 1996.
3. J.W.S. Cassels. **Local Fields**. Number 3 in London Mathematical Society Student Texts. Cambridge University Press, London, 1986.
4. W. Fulton. **Algebraic Curves**. Addison-Wesley, 1989.
5. V. Guruswami and M. Sudan. Improved decoding of Reed-Solomon and algebraic-geometric codes. In *Proceedings of the 39th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 28–37, 1998.
6. E. Kaltofen. Polynomial-time reductions from multivariate to bi- and univariate integral polynomial factorization. *SIAM J. Comput.*, 14:469–489, 1985.
7. S. McCallum. On testing a bivariate polynomial on analytic reducibility. *J. Symb. Comp.*, 24:509–535, 1997.
8. W. M. Schmidt. **Equations over Finite Fields: An Elementary Approach**. Number 536 in Lecture Notes in Mathematics. Springer-Verlag, Berlin, 1976.
9. M.A. Shokrollahi and H. Wasserman. List decoding of algebraic-geometric codes. *IEEE Trans. Inform. Theory*, 45:432–437, 1999.
10. H. Stichtenoth. **Algebraic Function Fields and Codes**. Universitext. Springer Verlag, 1993.
11. M. Sudan. Decoding of Reed-Solomon codes beyond the error-correction bound. *J. Compl.*, 13:180–193, 1997.

Remarks on codes from modular curves: MAPLE applications

David Joyner and Salahoddin Shokranian

August 12, 1999

Abstract

This paper is an exposition of some aspects of geometric coding theory and Goppa codes on modular curves.

1 Introduction

Suppose that V is a smooth projective variety over a finite field k. An important problem in arithmetical algebraic geometry is the calculation of the number of k-rational points of V, $|V(k)|$. The work of Goppa and others have shown its importance in geometric coding theory as well. We refer to this problem as the **counting problem**. In most cases it is very hard to find an explicit formula for the number of points of a variety over a finite field.

When the variety is a "Shimura variety" defined by certain group theoretical conditions (see §2 below), methods from non-abelian harmonic analysis on groups can be used to find an explicit solution for the counting problem. The Arthur-Selberg trace formula [S], provides one such method. Using the Arthur-Selberg trace formula, an explicit formula for the counting problem has been found for Shimura varieties, thanks primarily to the work of Langlands and Kottwitz ([Lan1], [K1]) [1]. Though it may be surprising and indeed very interesting that the trace formula allows one (with sufficient skill and expertise) to relate, when V is a Shimura variety, the geometric numbers

[1] For some introductions to this highly technical work of Langlands and Kottwitz, the reader is referred to Labesse [Lab], Clozel [Cl], and Casselman [Cas2].

$|V(\mathbf{F}_q)|$ to orbital integrals from harmonic analysis ([Lab], for example), or to a linear combination of coefficients of automorphic forms ([Gel], for example), or even to representation-theoretic data ([Cas2], for example), these formulas do not yet seem to be helping the coding theorist in any practical way that we know of.

However, another type of application of the trace formula is very useful. Moreno [M] first applied the trace formula in the context of Goppa codes to obtaining a new proof of a famous result of M. Tsfasman , S. Vladut, T. Zink, and Y. Ihara. (Actually, Moreno used a formula for the trace of the Hecke operators acting on the space of modular forms of weight 2, but this can be proven as a consequence of the Arthur-Selberg trace formula, [DL], §II.6.) This will be discussed below. We are going to restrict our attention in this paper to the interplay between Goppa codes of modular curves and the counting problem, and give some examples using MAPLE, where the programs using for the calculation is written in MAPLE by the first named author. In coding theory, curves with many rational points over finite fields are being used for construction of codes with some good specific characteristics. We discuss the Goppa codes, first from an abstract general perspective then turning to concrete examples associated to modular curves. We will try to explain these extremely technical ideas using a special case at a level to a typical graduate student with some background in modular forms, number theory, group theory, and algebraic geometry. For an approach similar in spirit, though from a more classical perspective, see the book of C. Moreno [M].

2 Shimura curves

In this section we study arithmetic subgroups, arithmetical quotients, and their rational compactifications. Ihara first introduced Shimura curves, a rational compactification of $\Gamma\backslash\mathbf{H}$ where Γ is a particular discrete subgroup, from a classical perspective. We shall recall them from both the classical and group-theoretical point of view. The latter perspective generalizes to higher dimensional Shimura varieties [Del].

2.1 Arithmetic subgroups

We assume that $G = SL(2)$ is the group of 2×2 matrices with entries from an algebraically closed field Ω. In particular the group of R-points of $SL(2)$ for a subring $R \subseteq \Omega$, with unit element 1 is defined by

$$SL(2, R) = \{g \in M(2, R) \mid \det(g) = 1\},$$

where $M(2, R)$ is the space of 2×2 matrices with entries from R. We now define congruence subgroups in $SL(2, \mathbf{Z})$. Let $SL(2, \mathbf{Z})$ be the subgroup of $SL(2, \mathbf{R})$ with integral matrices. Consider a natural number N, and let

$$\Gamma(N) = \left\{ \begin{bmatrix} a & b \\ c & d \end{bmatrix} \in SL(2, \mathbf{Z}) \mid \begin{array}{c} a, d \equiv 1(\mathrm{mod}\, N) \\ b, c \equiv 0(\mathrm{mod}\, N) \end{array} \right\},$$

We note that the subgroup $\Gamma(N)$ is a discrete subgroup of $SL(2, \mathbf{R})$, which is called the *principal congruence subgroup of level* N. Any subgroup of $SL(2, \mathbf{Z})$ that contains the principal congruence subgroup is called a *congruence subgroup.*

In general an *arithmetic subgroup* of $SL(2, \mathbf{R})$ is any discrete subgroup Γ that is commensurable with $SL(2, \mathbf{Z})$, where *commensurability* means that the intersection $\Gamma \cap SL(2, \mathbf{Z})$ is of finite index in both Γ and $SL(2, \mathbf{Z})$. The group $\Gamma(N)$ has the property of being commensurable with $SL(2, \mathbf{Z})$.

2.2 Riemann surfaces as algebraic curves

Let us recall that the space $\mathbf{H} = \{z \in \mathbf{C} \mid Im(z) > 0\}$ is called *the Poincaré upper half plane.* This space plays fundamental rôle in the definition of the modular curves. Note that the group $SL(2, \mathbf{R})$ acts on $\mathbf{H}$ by

$$g \cdot z = (az + b)(cz + d)^{-1} = \frac{az + b}{cz + d},$$

where $z \in \mathbf{H}$, $g = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \in SL(2, \mathbf{R})$.

We emphasize that the action of $SL(2, \mathbf{R})$ on $\mathbf{H}$ is *transitive*, i.e., for any two points $w_1, w_2 \in \mathbf{H}$ there is an element $g \in SL(2, \mathbf{R})$ such that $w_2 = g \cdot w_1$. This can easily be proved. We also emphasize that there are subgroups of $SL(2, \mathbf{R})$ for which the action is not transitive, among them the class of arithmetic subgroups are to be mentioned. For example, the group

$SL(2,\mathbf{Z})$ does not act transitively on $\mathbf{H}$, and the set of orbits of the action of $SL(2,\mathbf{Z})$ on $\mathbf{H}$, and similarly any arithmetic subgroup, is infinite. We call the *arithmetic quotient* $\Gamma\backslash\mathbf{H}$ the set of orbits of the action of an arithmetic subgroup Γ on $\mathbf{H}$.

Example 1 *Take* Γ *to be the* Hecke subgroup $\Gamma_0(N)$ *defined by*

$$\Gamma_0(N)=\left\{\left[\begin{array}{cc} a & b \\ c & d \end{array}\right]\in SL(2,\mathbf{Z})\mid c\equiv 0(\operatorname{mod} N)\right\},$$

for a natural number N*. This is a congruence subgroup and* $Y_0(N)=\Gamma_0(N)\backslash\mathbf{H}$ *is an arithmetic quotient. Such a quotient is not a compact subset, nor a bounded one, it is however a subset with finite measure (volume) under the non-Euclidean measure induced on the quotient from the group* $SL(2,\mathbf{R})$ *which is a locally compact group and induces the invariant volume element* $\frac{dx\wedge dy}{y^2}$*, where* x,y *are the real and the complex part of an element* $z\in\mathbf{H}$*.*

We now recall the basic ideas that turns an arithmetic quotient of the form $\Gamma\backslash\mathbf{H}$ into an algebraic curve. Let $\Gamma\subset SL(2,\mathbf{Q})$ be an arithmetic subgroup. The topological boundary of $\mathbf{H}$ is $\mathbf{R}$ and a point ∞. For the rational compactification of $\mathbf{H}$ we do not need to consider all the boundaries $\mathbf{R}$ and $\{\infty\}$. In fact we need only to add to $\mathbf{H}$ the cusps of Γ (a **cusp** of Γ is a rational number (an element of $\mathbf{Q}$) that is fixed under the action of an element γ with the property that $|tr(\gamma)|=2$). Any two cusps x_1,x_2 such that $\delta\cdot x_2=x_1$ for an element $\delta\in\Gamma$ are called **equivalent**. Let $C(\Gamma)$ be the set of inequivalent cusps of Γ. Then $C(\Gamma)$ is finite. We add this set to $\mathbf{H}$ and form the space $\mathbf{H}^*=\mathbf{H}\cup C(\Gamma)$. This space will be equipped with certain topology such that a basis of the neighborhoods of the points of $\mathbf{H}^*$ is given by three type of open sets; if a point in $\mathbf{H}^*$ is lying in $\mathbf{H}$ then its neighborhoods consists of usual open discs in $\mathbf{H}$, if the point is ∞, i.e., the cusp ∞, then its neighborhoods are the set of all points lying above the line $Im(z)>\alpha$ for any real number α, if the point is a cusp different than ∞ which is a rational number, then the system of neighborhoods of this point are the union of the cusp and the interior of a circle in $\mathbf{H}$ tangent to the cusp. Under the topology whose system of open neighborhoods we just explained, $\mathbf{H}^*$ becomes a Hausdorff non-locally compact space. The quotient space $\Gamma\backslash\mathbf{H}^*$ with the quotient topology is a compact Hausdorff space. We refer to this compact quotient as the **rational compactification** of $\Gamma\backslash\mathbf{H}$. For a detailed discussion we refer the reader to [Shim].

When the arithmetic group is a congruence subgroup of $SL(2,\mathbf{Z})$ the resulting algebraic curve is called a **modular curve**. For example, the rational compactification of $Y(N) = \Gamma(N)\backslash\mathbf{H}$ is denoted by $X(N)$ and the compactification of $Y_0(N) = \Gamma_0(N)\backslash\mathbf{H}$ by $X_0(N)$.

Example 2 *Let $N = 1$. Then $\Gamma = \Gamma(1) = SL(2,\mathbf{Z})$. In this case $C(\Gamma) = \{\infty\}$, since all rational cusps are equivalent to the cusp ∞. So $\mathbf{H}^* = \mathbf{H}\cup\{\infty\}$, and $\Gamma\backslash\mathbf{H}^*$ will be identified by $\Gamma\backslash\mathbf{H}\cup\{\infty\}$. This may be seen as adding ∞ to the fundamental domain $\mathbf{F}_1 = \mathbf{F}$ of $SL(2,\mathbf{Z})$, that consists of all complex numbers in $z \in \mathbf{H}$ with $|z| \geq 1$ and $|Re(z)| \leq \frac{1}{2}$.*

The rational compactification of $\Gamma\backslash\mathbf{H}$ turns the space $\Gamma\backslash\mathbf{H}^*$ into a compact Riemann surface (cf. [Shim]) and so into an algebraic curve (cf. [Nara], or [SS]).

In general it is easiest to work with those arithmetic subgroups which are torsion free and we shall assume from this point on that the arithmetic subgroups we deal with have this property. For example $\Gamma(N)$ and $\Gamma_0(N)$ for $N \geq 3$ are torsion free.

2.3 An adelic view of arithmetic quotients

Consider the number field $\mathbf{Q}$, the field of rational numbers. Let $\mathbf{Q}_p$ be the completion of $\mathbf{Q}$ under the p-adic absolute value $|...|_p$, where $|a/b|_p = p^{-n}$ whenever a, b are integers and $a/b = p^n \prod_{\ell \neq p \text{ prime}} \ell^{e_\ell}$, $n, e_\ell \in \mathbf{Z}$. Recall that under the ordinary absolute value the completion of $\mathbf{Q}$ is $\mathbf{R}$. The ring of adeles of $\mathbf{Q}$ is the locally compact commutative ring $\mathbf{A}$ that is given by:

$$\mathbf{A} = \{(x_\infty, x_2, \cdots) \in \mathbf{R} \times \prod_p \mathbf{Q}_p \mid \text{ all but a finite number of } x_p \in \mathbf{Z}_p\},$$

where $\mathbf{Z}_p$ is the ring of integers of $\mathbf{Q}_p$ (as it is well known $\mathbf{Z}_p$ is a maximal compact open subring of $\mathbf{Q}_p$). An element of $\mathbf{A}$ is called an **adele**. If $\mathbf{A}_f$ denotes the set of adeles omitting the $\mathbf{R}$-component x_∞, then $\mathbf{A}_f$ is called the **ring of finite adeles** and we can write $\mathbf{A} = 3D\mathbf{R} \times \mathbf{A}_f$. Under the diagonal embedding $\mathbf{Q}$ is a discrete subgroup of $\mathbf{A}$.

We now consider the group $G = GL(2)$. For a choice of an open compact subgroup $K_f \subset G(\mathbf{A}_f)$, it is known that we can write the arithmetic quotient (which was originally attached to an arithmetic subgroup of $\Gamma \subset SL(2,\mathbf{Q})$) as the following quotient

$$Y(K_f) = G(\mathbf{Q})\backslash[\mathbf{H} \times (G(\mathbf{A}_f)/K_f)] = \Gamma\backslash H, \tag{1}$$

where

$$\Gamma = G(\mathbf{Q}) \cap G(\mathbf{R})K_f. \tag{2}$$

Thus our arithmetic subgroup Γ is completely determined by K_f. From now on we assume K_f has been chosen so that Γ is torsion free.

Definition 3 *Let $G = GL(2)$. To G is associated the Shimura variety $Sh(G)$ as follows. Let $N \geq 3$ be a natural number. Let $\Gamma(N)$ be the congruence subgroup of level N of $SL(2,\mathbf{Z})$, and $K = SO(2,\mathbf{R})$ the orthogonal group of 2×2 real matrices A with determinant 1 satisfying ${}^tAA = I_2$. Then*

$$Y(N) = \Gamma(N)\backslash\mathbf{H} \cong \Gamma(N)\backslash G(\mathbf{R})/K.$$

We call this the modular space of level N. *Let*

$$K_f(N) = \{g \in G(\prod_p \mathbf{Z}_p) \mid g \equiv I_2(\mathrm{mod}\, N)\}$$

be the **open compact subgroup of $G(\mathbf{A}_f)$ of level** N. *Then the modular space of level N can be written as:*

$$Y(N) \cong G(\mathbf{Q})\backslash G(\mathbf{A})/KK_f(N) = G(\mathbf{Q})\backslash[\mathbf{H} \times (G(\mathbf{A}_f)/K_f(N))].$$

Thus

$$X(K_f(N)) \cong Y(N).$$

Taking the projective limit over $K_f(N)$ by letting N gets large (which means $K_f(N)$ gets small), we see that $\lim_N Y(N) = G(\mathbf{Q})\backslash[\mathbf{H} \times G(\mathbf{A}_f)]$. *Then the (complex points of the)* **Shimura curve** $Sh(G)$ *associated to $G = SL(2)$ is defined by*

$$Sh(G)(\mathbf{C}) = G(\mathbf{Q})\backslash[\mathbf{H} \times G(\mathbf{A}_f)]. \tag{3}$$

Many mathematicians have addressed the natural questions

- What field are the curves $X(N)$, $X_0(N)$ defined over?

- How can they be described explicitly using algebraic equations?

Regarding the first question, by the general theory of Shimura varieties we know that for each reductive group G defined over $\mathbf{Q}$ satisfying the axioms of §2.1.1 in [Del], there is an algebraic number field $E = E_G$ over which a Shimura variety $Sh(G)$ is defined [Del]. In fact, the Shimura curves $X(N)$ and $X_0(N)$ are regular schemes proper over $\mathbf{Z}[1/N]$ (more precisely over $Spec(\mathbf{Z}[1/N])$) [2].

Regarding the second question, it is possible to find a **modular polynomial** $H_N(x,y)$ of degree

$$\mu(N) = N\prod_{p|N}(1+\frac{1}{p})$$

for which $H_N(x,y)=0$ describes (an affine patch of) $X_0(N)$. Let

$$G_k(q) = 2\,\zeta(k) + 2\,\frac{(2\,i\pi)^k}{(k-1)!}\sum_{n=1}^{\infty}\sigma_{k-1}(n)q^n,$$

where $q = e^{2\pi i z}, z \in \mathbf{H}$, $\sigma_r(n) = \sum_{d|n} d^r$, and let

$$\Delta(q) = 60^3 G_4(q)^3 - 27\cdot 140^2 G_6(q)^2 = q\prod_{n=1}^{\infty}(1-q^n)^{24}.$$

Define the **j-invariant** by

$$j(q) = 1728{\cdot}60^3 G_4(q)^3/\Delta(q) = q^{-1}+744+196884q+21493760q^2+864299970q^3+\dots.$$

(More details on Δ and j can be found for example in [Shim].) The key property satisfied by H_N is $H_N(j(q), j(q^N)) = 0$. It is interesting to note in passing that when N is such that the genus of $X_0(N)$ equals 0 (i.e., $N \in \{1,3,4,5,6,7,8,9,12,13,16,18,25\}$ [Kn]) then this implies that $(x,y) = (j(q), j(q^N))$ parameterizes $X_0(N)$. In general, comparing q-coefficients allows one to use a computer algebra system such as MAPLE to compute H_N

[2] This result was essentially first proved by Igusa [Ig] (from the classical perspective). See also [TV], Theorem 4.1.48, [Cas1] for an interesting discussion of what happens at the "bad primes", and Deligne's paper in the same volume as [Cas1].

for relatively small values of N. However, even for $N = 11$, some of the coefficients can involve one hundred digits or more. The cases $N = 2, 3$ are given in Elkies [E], for example. The paper by P. Cohen [Co] determines the asymptotic size of the largest coefficient of H_N (normalized to have leading coefficient equal to 1). She shows that the largest coefficient grows like $N^{c\mu(N)}$, where $c > 0$ is a constant. More practical equations for (some of) the $X_0(N)$ are given in T. Hibino and N. Murabayashi [HM], M. Shimura [ShimM], J. Rovira [R], G. Frey and M. Müller [FM], Birch [B], and the table in §2.5 below.

For deeper study of Shimura varieties and the theory of canonical models we refer the reader to [Del], [Lan2], and [Shim].

2.4 Hecke operators and arithmetic on $X_0(N)$

In this section we recall some well-known though relatively deep results on $X_0(N)(\mathbf{F}_p)$, where p is a prime not dividing N. These shall be used in the discussion of the Tsfasman, Vladut, Zink, and Ihara result later.

First, some notation: let $S_2(\Gamma_0(N))$ denote the space of holomorphic automorphic forms of weight 2 on $\Gamma_0(N)\backslash H$. Let $T_p : S_2(\Gamma_0(N)) \to S_2(\Gamma_0(N))$ denote the **Hecke operator** defined by

$$T_pf(z) = f(pz) + \sum_{i=0}^{p-1} f(\frac{z+i}{p}), \qquad z \in H.$$

Define T_{p^k} inductively by

$$T_{p^k} = T_{p^{k-1}}T_p - pT_{p^{k-2}}, \quad T_1 = 1,$$

and define the modified Hecke operators U_{p^k} by

$$U_{p^k} = T_{p^k} - pT_{p^{k-2}}, \quad U_p = T_p,$$

for $k \geq 2$. The Hecke operators may be extended to the positive integers by demanding that they be multiplicative.

Theorem 4 *("Congruence relation" of Eichler-Shimura [M], §5.6.7) Let $q = p^k$, $k > 0$ an integer. If p is a prime not dividing N then*

$$|X_0(N)(\mathbf{F}_q)| = q + 1 - a_q,$$

where

$$a_q = Tr(U_q).$$

Example 5 *One may try to compute the trace of the Hecke operators T_p acting on the space of holomorphic cusp forms of weight* 2, $S_2(\Gamma_0(N))$, *by using either the Eichler-Shimura trace formula, which we give below (see Theorem 6), or by using some easier but ad hoc ideas going back to Hecke which work in special cases. One simple idea is to note that $S_2(\Gamma_0(N))$ is spanned by simultaneous eigenforms of the Hecke operators (see for example, Proposition 51 in chapter III of [Ko]). In this case, it is known that the Fourier coefficient a_p, p prime not dividing N, of a normalized (to have leading coefficient $a_1 = 1$) eigenform is the eigenvalue of T_p (see for example, Proposition 40 in chapter III of [Ko]). If $S_2(\Gamma_0(N))$ is one-dimensional then any element in that space $f(z)$ is such an eigenform.*

The modular curve $X_0(11)$ is of genus 1, so there is (up to a non-zero constant factor) only one holomorphic cusp form of weight 2 *in $S_2(\Gamma_0(11))$ (see Theorem 8 below). There is a well-known construction of this form (see [O2] or [Gel], Example 5.1), which we recall below. As we noted above, the p-th coefficient a_p (p a prime distinct from* 11*) of its Fourier expansion is known to satisfy $a_p = Tr(T_p)$. These will be computed using MAPLE.*

Let $q = e^{2\pi i z}$, $z \in$ **H**, *and consider* **Dedekind's η-function***:*

$$\eta(z) = e^{2\pi i z/24}\prod_{n=1}^{\infty}(1-q^n).$$

Then

$$f(z) = \eta(z)^2\eta(11z)^2 q\prod_{n=1}^{\infty}(1-q^n)^2(1-q^{11n})^2,$$

is an element of $S_2(\Gamma_0(11))$ [3]. *MAPLE gives*

$$\begin{gathered}f(z) = q - 2\,q^2 - q^3 + 2\,q^4 + q^5 + 2\,q^6 - 2\,q^7 - 2\,q^9 - 2\,q^{10}\\ +q^{11} - 2\,q^{12} + 4\,q^{13} + 4\,q^{14} - q^{15} - 4\,q^{16} - 2\,q^{17} + 4\,q^{18} + 2\,q^{20}\\ +2\,q^{21} - 2\,q^{22} - q^{23} - 4\,q^{25} - 8\,q^{26} + 5\,q^{27} - 4\,q^{28}...\end{gathered}$$

[3] In fact, if we write $f(z) = \sum_{n=1}^{\infty} a_n q^n$ then

$$\zeta_E(s) = (1-p^{-s})^{-1}\prod_{p\neq 11}(1-a_p p^{-s}+p^{1-2s})^{-1},$$

is the global Hasse-Weil zeta function of the elliptic curve E of conductor 11 with Weierstass model $y^2 + y = x^3 - x^2$ [Gel], page 252.

For example, the above expansion tells us that $Tr(T_3) = Tr(U_3) = -1$. The curve $X_0(11)$ is of genus 1 and is isogenous to the elliptic curve E with Weierstrass model $y^2+y = x^3-x^2$. Over the field with $p = 3$ elements, there are $|X_0(11)(\mathbf{F}_3)| = p+1-Tr(T_p) = 5$ points in $E(\mathbf{F}_3)$, including ∞:

$$E(\mathbf{F}_3) = \{[0,0],[0,2],[1,0],[1,2],\infty\}.$$

For a representation-theoretic discussion of this example, see [Gel], §14.

For an example of an explicit element of $S_2(\Gamma_0(32))$, see Koblitz [Ko], chapter II §5 and (3.40) in chapter III. For a remarkable theorem which illustrates how far this η-function construction can be extended, see Morris' theorem in §2.2 of [R].

To estimate a_{p^k}, one may appeal to an explicit expression for $Tr(T_{p^k})$ known as the "Eichler-Selberg trace formula", which we discuss next.

2.5 Eichler-Selberg trace formula

In this subsection, we recall the version of the trace formula for the Hecke operators due to Duflo-Labesse [DL], §6.

Let k be an even positive integer and let Γ be a congruence subgroup as in (2). Let S denote a complete set of representatives of $G(\mathbf{Q})$-conjugacy classes of $\mathbf{R}$-elliptic elements in Γ ($\mathbf{R}$-elliptic elements are those that are conjugate to an element of $SO(2,\mathbf{R})$, the orthogonal group). For $\gamma \in S$, let $w(\gamma)$ denote the cardinality of the centralizer of γ in Γ. If $r(\theta) = \begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix}$ then let $\theta_\gamma \in (0,2\pi)$ denote the element for which $\gamma = r(\theta_\gamma)$. Let τ_m denote the image in $G(\mathbf{A}_f)$ of the set of matrices in $GL(2,\mathbf{A}_f)$ having coefficients in $\hat{\mathbf{Z}} = \prod_{p<\infty} \mathbf{Z}_p$ and determinant in $m\hat{\mathbf{Z}}$. Consider the subspace $S_k(\Gamma) \subset L^2(\Gamma\backslash H)$ formed by functions satisfying

- $f(\gamma z) = (cz+d)^k f(z)$, for all $\gamma = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \Gamma$, $x \in H$,

- f is a holomorphic cusp form.

This is the space of holomorphic cusp forms of weight k on $\mathbf{H}$.

Let

$$\epsilon(\sqrt{m}) = \begin{cases} 1, & m \text{ is a square,} \\ 0, & \text{otherwise.} \end{cases}$$

and let

$$\delta_{i,j} = \begin{cases} 1, & i = j, \\ 0, & \text{otherwise.} \end{cases} .$$

Theorem 6 *("Eichler-Selberg trace formula") Let $k > 0$ be an even integer and $m > 0$ an integer. The trace of T_m acting on $S_k(\Gamma)$ is given by*

$$Tr(T_m) = \delta_{2,k} \sum_{d|m} b + \epsilon(\sqrt{m})(\frac{k-1}{12} m^{(k-2)/2} - \frac{1}{2} m^{(k-1)/2})$$
$$- \sum_{\gamma \in S \cap \tau_m} w(\gamma)^{-1} m^{(k-2)/2} \frac{\sin((k-1)\theta_\gamma)}{\sin(\theta_\gamma)} - \sum_{d|m,\ d^2<m} b^{k-1}.$$

Remark 7 *Let $k = 2$, $m = p^2$, $\Gamma = \Gamma_0(N)$ and $N \to \infty$ in the above formula. It is possible to show that the Eichler-Selberg trace formula implies*

$$Tr(T_{p^2}) = g(X_0(N)) + O(1), \tag{4}$$

as $N \to \infty$. The proof of this estimate (see [M], chapter 5, or [LvdG], § V.4) uses the explicit formula given below for $g(X_0(N)) = \dim(S_2(\Gamma_0(N))$, which we shall also make use of later.

Theorem 8 *("Hurwitz-Zeuthen formula" [Shim])* [4] *The genus of $X_0(N)$ is given by*

$$g(X_0(N)) = \dim(S_2(\Gamma_0(N)) = 1 + \frac{1}{12}\mu(N) - \frac{1}{4}\mu_2(N) - \frac{1}{3}\mu_3(N) - \mu_\infty(N),$$

where

$$\mu(N) = [SL(2,\mathbf{Z})/\Gamma_0(N)] = N \prod_{p|N} (1 + \frac{1}{p}),$$

$$\mu_2(N) = \begin{cases} \prod_{p|N \text{ prime}} (1 + (\frac{-4}{p})), & gcd(4,N) = 1, \\ 0, & 4|N, \end{cases}$$

[4] The genus formula for $X_0(N)$ given in [Shim] and [Kn] both contain a (typographical?) error. The problem is in the μ_2 term, which should contain a Legendre symbol $(\frac{-4}{n})$ instead of $(\frac{-1}{n})$. See for example [Ei] for a correct generalization.

$$\mu_3(N) = \begin{cases} \prod_{p|N \text{ prime}} (1 + (\frac{-3}{p})), & gcd(2,N) = 1 \text{ and } gcd(9,N) \neq 9, \\ 0, & 2|N \text{ or } 9|N, \end{cases}$$

and

$$\mu_\infty(N) = \sum_{d|N} \phi(gcd(d, N/d)),$$

where ϕ is Euler's totient function and $(\frac{\cdot}{p})$ is Legendre's symbol.

The estimate (4) and the Eichler-Shimura congruence relation imply

$$|X_0(N)(\mathbf{F}_{p^2})| = p^2 + 1 - Tr(T_{p^2} - pI) = p^2 + 1 - Tr(T_{p^2}) + p\dim(S_2(\Gamma_0(N))$$
$$= p^2 + 1 - (g(X_0(N)) + O(1))) + pg(X_0(N)) = (p-1)g(X_0(N)) + O(1),$$

as $N \to \infty$.

2.6 The curves $X_0(N)$ of genus 1

It is known (see for example [Kn]) that a modular curve of level N, $X_0(N)$, is of genus 1 if and only if

$$N \in \{11, 14, 15, 17, 19, 20, 21, 24, 32, 36, 49\}.$$

In these cases, $X_0(N)$ is birational to an elliptic curve E having Weierstrass model of the form

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6,$$

with a_1, a_2, a_3, a_4, a_6. If E is of above form then the **discriminant** is given by

$$\Delta = -b_2^2b_8 - 8b_4^3 - 27b_6^2 + 9b_2b_4b_6,$$

where

$$b_2 = a_1^2 + 4a_2, \qquad b_4 = 2a_4 + a_1a_3, \qquad b_6 = a_3^2 + 4a_6,$$
$$b_8 = a_1^2a_6 + 2a_2a_6 - a_1a_3a_4 + a_2a_3^2 - a_4^2.$$

The conductor [5] N of E and its discriminant Δ have the same prime factors. Furthermore, $N|\Delta$ ([Kn], [Gel]).

Some examples, which we shall use later, are collected in the following table.

[5] The conductor is defined in Ogg [O1], but see also [Gel], §I.2, or [Kn], P. 390.

level	discriminant	Weierstrass model	reference
11	-11	$y^2 + y = x^3 - x^2$	[BK], table 1, p. 82
14	-28	$y^2 + xy - y = x^3$	p. 391, table 12.1 of [Kn]
15	15	$y^2 + 7\,xy + 2\,y = x^3 + 4\,x^2 + x$	p. 65, table 3.2 of [Kn]
17	17	$y^2 + 3\,xy = x^3 + x$	p. 65, table 3.2 of [Kn]
19	-19	$y^2 + y = x^3 + x^2 + x$	[BK], table 1, p. 82
20	80	$y^2 = x^3 + x^2 - x$	p. 391, table 12.1 of [Kn]
21	-63	$y^2 + xy = x^3 + x$	p. 391, table 12.1 of [Kn]
24	-48	$y^2 = x^3 - x^2 + x$	p. 391, table 12.1 of [Kn]
27	-27	$y^2 + y = x^3$	p. 391, table 12.1 of [Kn]
32	64	$y^2 = x^3 - x$	p. 391, table 12.1 of [Kn]
36		(see below)	§4.3 in [R]
49		(see below)	§4.3 in [R]

When $N = 36$, §4.3 in Rovira [R] gives $y^2 = x^4 - 4x^3 - 6x^2 - 4x + 1$, which is a hyperelliptic equation but not in Weierstrass form. To put it in Weierstrass form, we use the Maple **algcurve** package [6]. This produces a cubic equation in which the coefficient of x^3 is not one. Using the change-of-variable $y \longmapsto 2y$ (which maps the curve $X_0(36)$ to an isogenous one), we obtain the Weierstrass form $y^2 = x^3 + 64$, provided $p \neq 2$. This has conductor $\Delta = -1769472$.

When $N = 49$, §4.3 in Rovira [R] gives $y^2 = x^4 - 2x^3 - 9x^2 + 10x - 3$, which is a hyperelliptic equation but not in Weierstrass form. As in the previous case, we use the Maple **algcurve** package, which produces a cubic equation in which the coefficient of x^3 is not one. Again, the change-of-variable $y \longmapsto 2y$ (which maps the curve $X_0(49)$ to an isogenous one), yields the Weierstrass form $y^2 = x^3 - 35x - 98$, provided $p \neq 2$. This has conductor $\Delta = -1404298$.

3 Codes

To have an idea of how the points of a curve over finite fields are used in the coding theory we first recall the definition of a code.

Let A be a finite set, which we regard as an alphabet. Let A^n be the n-fold Cartesian product of A by itself. In A^n we define the **Hamming**

[6] More precisely, we use the **WeierstrassForm** command written by Mark van Hoeij.

metric $d(x,y)$ by:

$$d(x,y) = d((x_1,\cdots,x_n),(y_1,\cdots,y_n)) := |\{i \mid x_i \neq y_i\}|.$$

We now assume that A^n is equipped with the Hamming metric. Then by definition a subset $C \subseteq A^n$ is called an $|A|$**-ary code**. An important case arises when we let A to be a finite field. Suppose that $q = p^m$ and $\mathbf{F}_q$ is a finite field with q elements. In this case we may put $A = \mathbf{F}_q$ and $y = (0,\cdots,0)$. Then the **weight** of x is the Hamming length $||x|| = d(x,0) = |\{i \mid x_i \neq 0\}|$. In particular a subset C of $\mathbf{F}_q^n$ is a code, and to it is associated two basic parameters: $k = \log_q |C|$, the **number of information bits** and $d = min\{||x-y|| \mid x,y \in C, y \neq 0\}$ the **minimum distance**. (A code with minimum distance d can correct $[\frac{d-1}{2}]$ errors.) Let

$$R = R(C) = \frac{k}{n},$$

which measures the information rate of the code, and

$$\delta = \delta(C) = \frac{d}{n},$$

which measures the error correcting ability of the code.

3.1 Basics on linear codes

If the code $C \subset \mathbf{F}_q^n$ is a vector space over $\mathbf{F}_q$ then we call C a **linear code**. The **parameters** of a linear code C are

- the **length** n,
- the **dimension** $k = \dim_{\mathbf{F}_q}(C)$,
- the minimum distance d.

Such a code is called an (n,k,d)**-code**. Let Σ_q denote the set of all $(\delta, R) \in [0,1]^2$ such that there exists a sequence C_i, $i = 1,2,...$, of (n_i,k_i,d_i)-codes for which $\lim_{i\to\infty} \delta_i = \delta$ and $\lim_{i\to\infty} R_i = R$.

The following theorem describes information-theoretical limits on how "good" a linear code can be.

Theorem 9 *(Manin [SS], chapter 1) There exists a continuous decreasing function*

$$\alpha_q : [0,1] \to [0,1],$$

such that

- α_q *is strictly decreasing on* $[0, \frac{q-1}{q}]$,
- $\alpha_q(0) = 1$,
- *if* $\frac{q-1}{q} \leq x \leq 1$ *then* $\alpha_q(x) = 0$,
- $\Sigma_q = \{(\delta, R) \in [0,1]^2 \mid 0 \leq R \leq \alpha_q(\delta)\}$.

Not a single value of $\alpha_q(x)$ is known for $0 < x < \frac{q-1}{q}$! It is not known whether or not the maximum value of the bound, $R = \alpha_q(\delta)$ is attained by a sequence of linear codes. It is not known whether or not $\alpha_q(x)$ is differentiable for $0 < x < \frac{q-1}{q}$, nor is it known if $\alpha_q(x)$ is convex on $0 < x < \frac{q-1}{q}$. However, the following estimate is known.

Theorem 10 *(Gilbert-Varshamov [MS], [SS] chapter 1) We have*

$$\alpha_q(x) \geq 1 - x\log_q(q-1) - x\log_q(x) - (1-x)\log_q(1-x).$$

In other words, for each fixed $\epsilon > 0$*, there exists an* (n,k,d)*-code* C *(which may depend on* ϵ*) with*

$$R(C)+\delta(C) \geq 1-\delta(C)\log_q(\frac{q-1}{q})-\delta(C)\log_q(\delta(C))-(1-\delta(C))\log_q(1-\delta(C))-\epsilon.$$

The curve $(\delta, 1-\delta\log_q(\frac{q-1}{q}) - \delta\log_q(\delta) - (1-\delta)\log_q(1-\delta)))$ is called the **Gilbert-Varshamov curve**. This theorem says nothing about constructing codes satisfying this property! Nor was it known, until the work of Tsfasman, Valdut, Zink and Ihara, how to do so.

3.2 Some basics on Goppa codes

We begin with Goppa's basic idea boiled down to its most basic form. Let R denote a commutative ring with unit and let $m_1, m_2, ..., m_n$ denote a finite number of maximal ideals such that for each $1 \leq i \leq n$, we have $R/m_i \cong \mathbf{F}_q$. Define $\gamma : R \to \mathbf{F}_q^n$ by

$$\gamma(x) = (x + m_1, x + m_2, ..., x + m_n), \qquad x \in R.$$

Of course, in this level of generality, one cannot say much about this map. However, when R is associated to the coordinate functions of a curve defined over $\mathbf{F}_q$ then one can often use the machinery of algebraic geometry to obtain good estimates on the parameters (n, k, d) of the code associated to γ.

Let V be an irreducible smooth projective algebraic variety defined over the finite field $\mathbf{F}_q$. Let $\mathbf{F}_q(V)$ denote the field of rational functions on V. Let $\mathcal{P}(V)$ denote the set of prime divisors of V, which we may identify with the closed irreducible subvarieties of $V(\overline{\mathbf{F}_p})$ of codimension 1. For each $P \in \mathcal{P}(V)$, there is a valuation map $ord_P : \mathbf{F}_q(V) \to \mathbf{Z}$ (see Hartshorne [Ha], §II.6, page 130). Let $\mathcal{D}(V)$ denote the group of divisors of V, the free abelian group generated by $\mathcal{P}(V)$.

If $A = \sum_P a_P P, B = \sum_P b_P P \in \mathcal{D}(V)$ are divisors then we say $A \leq B$ if and only if $a_P \leq b_P$ for all $P \in \mathcal{P}(V)$. If $f \in \mathbf{F}_q(V)$ is a non-zero function then let

$$(f) = \sum_{P \in \mathcal{P}(V)} ord_P(f) P,$$

where $ord_P(f)$ is the order of the zero (pole) at P (as above). This is well-defined (since the above sum is finite by Lemma 6.1 in [Ha], §II.6, page 131). For $B \in \mathcal{D}(V)$, define $\mathcal{L}(B)$ to be the vector space

$$\mathcal{L}(B) = \{0\} \cup \{f \in \mathbf{F}_q(V) \mid f \neq 0, (f) \geq -B\}.$$

Pick n different points $P_1, P_2, ..., P_n$ in $V(\mathbf{F}_q)$ and choose a divisor $A = \sum_{P \in \mathcal{P}(V)} a_P P \in \mathcal{D}(V)$ disjoint from these points. Quite often, one does *not* want A to be rational. The **Goppa code** associated to $(V(\mathbf{F}_q), A)$ is the linear code $G = G(A, V)$ defined to be the subspace of $\mathbf{F}_q^n$ which is the image of the map

$$\gamma : \mathcal{L}(A) \to \mathbf{F}_q^n, \tag{5}$$

defined by $\gamma(f) = (f(P_1), ..., f(P_n))$. (In the case of curves, this code is called the **dual Goppa code** or **Goppa function code** in [P]. Goppa gave another geometric construction of codes using differentials for which we refer the reader to [P] or [TV].)

To specify a Goppa code, one must

- choose a smooth variety V over $\mathbf{F}_q$,
- pick rational points $P_1, P_2, ..., P_n$ of V,
- choose a divisor A disjoint from the P_i's,
- determine a basis for $\mathcal{L}(A)$,
- compute the matrix for γ.

3.3 Some estimates on Goppa codes

Let g be the genus of a curve $V = C$ and let $G = G(A, P, C)$ denote the Goppa code as constructed above. If G has parameters (n, k, d) and if we then the following lemma is a consequence of the Riemann-Roch theorem.

Lemma 11 *Assume G is as above and A satisfies $2g - 2 < deg(A) < n$. Then $k = dim(G) = deg(A) - g + 1$ and $d \geq n - deg(A)$.*

Consequently, $k + d \geq n - g + 1$. Because of Singleton's inequality[7], we have

- if $g = 0$ then G is an MDS code,
- if $g = 1$ then $n \leq k + d \leq n + 1$.

The previous lemma also implies the following lower bound.

Proposition 12 *([SS] §3.1, or [TV]) With G as in the previous lemma, we have $\delta + R = \frac{d}{n} + \frac{k}{n} \geq 1 - \frac{g-1}{n}$.*

[7] It is known that $n \geq d + k - 1$ for any linear (n, k, d)-code (this is the **Singleton inequality**), with equality if and only if the code is a so-called **MDS code** (MDS=minimum distance separable).

Theorem 8 above is an explicit formula for the genus of the modular curve $X_0(N)$. It may be instructive to plug this formula into the estimate in Proposition 12 to see what we get. The formula for the genus g_N of $X_0(N)$ is relatively complicated, but simplifies greatly when N is a prime number which is congruent to 1 modulo 12, say $N = 1 + 12m$, in which case $g_N = m - 1$. For example, $g_{13} = 0$. In particular, we have the following

Corollary 13 *Let $C = X_0(N)$, where N is a prime number which is congruent to* 1 *modulo* 12 *and which has the property that C is smooth over $\mathbf{F}_q$. Then the parameters (n, k, d) of a Goppa code associated to C must satisfy*

$$\frac{d}{n} + \frac{k}{n} \geq 1 - \frac{\frac{N-1}{12} - 2}{n}.$$

Based on the above Proposition, if one considers a family of curves X_i with increasing genus g_i such that

$$\lim_{i \to \infty} \frac{|X_i(\mathbf{F}_q)|}{g_i} = \alpha \tag{6}$$

one can construct a family of codes C_i with $\delta(C_i) + R(C_i) \geq 1 - \frac{1}{\alpha}$. It is known that $\alpha \leq \sqrt{q} - 1$ (this is the so-called **Drinfeld-Vladut bound**, [TV], Theorem 2.3.22).

The following result says that the Drinfeld-Vladut bound can be attained in case $q = p^2$.

Theorem 14 *(Tsfasman, Valdut, Zink [TV], Theorem 4.1.52) Let g_N denote the genus of $X_0(N)$. If N runs over a set of primes different than p then the quotients $g_N/|X_0(N)(\mathbf{F}_{p^2})|$ associated to the modular curves $X_0(N)$ tend to the limit $\frac{1}{p-1}$.*

More generally, if $q = p^{2k}$, then there is a family of Drinfeld curves X_i over $\mathbf{F}_q$ yielding $\alpha = \sqrt{q} - 1$ ([TV], Theorem 4.2.38, discovered independently by Ihara [I] at about the same time). In other words, the Drinfeld-Vladut bound is attained in case $q = p^{2k}$.

As a corollary to the above theorem, if $p \geq 7$ then there exists a sequence of Goppa codes G_N over $\mathbf{F}_{p^2}$ associated to a sequence of modular curves $X_0(N)$ for which $(R(G_N), \delta(G_N))$ eventually (for suitable large N) lies above the Gilbert-Varshamov bound in Theorem 10. This follows from comparing the Gilbert-Varshamov curve

$$(\delta, 1-\delta\log_q(\frac{q-1}{q})-\delta\log_q(\delta)-(1-\delta)\log_q(1-\delta)))$$

with the curve $(\delta, \frac{1}{\sqrt{q}-1})$, $q=p^2$.

4 Examples

Example 15 *Let C denote the elliptic curve of conductor 32 (and birational to $X_0(32)$) with Weierstrass form $y^2=x^3-x$. If p is a prime satisfying $p\equiv 3 (\mathrm{mod}\, 4)$ then*

$$|C(\mathbf{F}_p)|=p+1$$

(Theorem 5, §18.4 in Ireland and Rosen [IR]). Let $C(\mathbf{F}_p)=\{P_0,P_1,P_2,...,P_n\}$, where P_0 is the identity, and if $A=kP_0$, for some $k>0$. The parameters of the corresponding code $G=G(A,P,C)$ satisfy $n=p$, $d+k\geq n$, since $g=1$, by the above Proposition. In this case, G is not an MDS code. As we observed above, a Goppa code constructed from an elliptic curve satisfies either $d+k-1=n$ (i.e., is MDS) or else $d+k=n$. Thus in this case,

$$n=p, \qquad d+k=p.$$

Let C be an elliptic curve. This is a projective curve for which $C(\mathbf{F}_q)$ has the structure of an algebraic group. Let $P_0\in C(\mathbf{F}_q)$ denote the identity. Let $P_1,P_2,...,P_n$ denote *all* the other elements of $C(\mathbf{F}_q)$ and let $A=aP_0$, where $0<a<n$ is an integer. The following result is an immediate corollary of the results in [Sh] (see also §5.2.2 in [TV] for weaker but closely related results).

Theorem 16 *(Shokrollahi) Let $C,P_0,P_1,...,P_n,D,A$, be as above.*

- *If $a=2$ and $C(\mathbf{F}_q)\cong C_2\times C_2$ (where C_n denotes the cyclic group of order n) then the code $G(A,D)$ is a $[n,k,d]$-code (n is the length, k is the dimension, and d is the minimum distance) with*

$$d=n-k+1, \quad \text{and} \quad k=a.$$

- *Assume $gcd(n, a!) = 1$. If $a \neq 2$ or $C(\mathbf{F}_q)$ is not isomorphic to the Klein four group $C_2 \times C_2$ then $G(A, D)$ is a $[n, k, d]$-code (n is the length, k is the dimension, and d is the minimum distance) with*

$$k = a$$

and weight enumerator polynomial (see for example [MS] for the definition)

$$W_G(x) = x^n + \sum_{i=0}^{a-1} \binom{n}{i} (q^{a-1} - 1)(x-1)^i + B_a(x-1)^a,$$

where

$$B_a = \frac{1}{n}(q-1)(\binom{n-1}{a} + (-1)^a(n-1)).$$

4.1 Weight enumerators (aprés des Shokrollahi)

In the case where E is given by the level 19, discriminant -19 Weierstrass model

$$y^2 + y = x^3 + x^2 + x,$$

and $p = 13$, we have

$$E(\mathbf{F}_p) = \begin{array}{l} \{[0,0],[0,12],[1,6],[3,0],[3,12],[4,2],[4,10],[5,3],[5,9], \\ [8,3],[8,9],[9,0],[9,12],[11,4],[11,8],[12,3],[12,9],\infty\} \end{array}.$$

Write $E(\mathbf{F}_p) = \{P_0, P_1, ..., P_{17}\}$, where P_0 denotes the identity element of the group law for E, let $A = kP_0$, and let $D = P_1 + ... + P_{17}$. The hypotheses of the above theorem are satisfied when we take $n = 17$ and $2 \leq k = a < 17$. The above construction associates to this data a Goppa code $G = G(A, D, E)$ which is a 7-error correcting code of length $n = 17$ over $\mathbf{F}_{13}$. Some of the weight enumerator polynomials W_G and the number of errors these codes G can correct are given in the following table.

$a = k$	weight enumerator W_G	number of errors G corrects
2	$x^{17} + 96x^2 + 12x + 60$	7
3	$x^{17} + 384x^3 + 480x^2 + 744x + 588$	6
4	$x^{17} + 1296x^4 + 2976x^3 + 6144x^2 + 10932x + 7212$	6
5	$x^{17} + 3072x^5 + 13200x^4 + ... + 95676$	5
6	$x^{17} + 5664x^6 + 40272x^5 + ... + 1236972$	5
7	$x^{17} + 8064x^7 + 92064x^6 + ... + 16095036$	4
8	$x^{17} + 9096x^8 + 160608x^7 + ... + 209212116$	4
9	$x^{17} + 8064x^9 + 219144x^8 + ... + 2719785636$	3
10	$x^{17} + 5664x^{10} + 235080x^9 + ... + 35357186484$	3

These were computed using a computer implementation of Shokrollahi's formula in the software package MAPLE. The number of codewords of minimum weight $n - k$ is the coefficient of the second highest term in $W_G(x)$. For example, when $k = 3$ the number of codewords of minimum weight $n - k = 14$ is 384.

A smaller example using the same elliptic curve E as above: taking $p = 3$, we find that

$$E(\mathbf{F}_p) = \{[0,0],[0,2],[1,0],[1,2],[2,1],\infty\}.$$

The hypotheses of the above theorem are satisfied when we take $n = 5$ and $2 \leq k = a < 5$. The weight enumerator when $a = 2$ is

$$W_G(x) = x^5 + 4x^2 + 2x + 2,$$

and there are 4 codewords of minimum weight 3 in the corresponding elliptic (Goppa) code. This is a 1-error correcting code of length 5 (over $\mathbf{F}_3$).

4.2 The generator matrix (aprés des Goppa)

The method used in Goppa's Fermat cubic code example of [G], pages 108-109, can be easily modified to yield analogous quantities for certain elliptic Goppa codes.

Example 17 *Let E denote the elliptic curve (of conductor $N = 11$) which we write in homogeneous coordinates as*

$$y^2z + yz^2 = x^3 - x^2z.$$

Let $\phi(x,y,z) = xy + yz + xz$*, let* F *denote the projective curve defined by* $\phi(x,y,z) = 0$*, and let* D *denote the divisor obtained by intersecting* E *and* F*. By Bezout's theorem (see for example, [G], page 80),* D *is of degree* 6*. To find the matrix of the linear transformation* γ *in (5), we must specify a basis for* $\mathcal{L}(D)$*. As in [G] page 109, a basis for* $\mathcal{L}(D)$ *is provided by the functions in the set*

$$\mathcal{B}_D = \{1, x^2/\phi(x,y,z), y^2/\phi(x,y,z), z^2/\phi(x,y,z), xy/\phi(x,y,z), yz/\phi(x,y,z)\}.$$

(This is due to the fact that dim $\mathcal{L}(D) = \deg(D) = 6$ *and the functions* $f \in \mathcal{B}_D$ *"obviously" satisfy* $(f) \geq -D$*.) We have*

$$E(\mathbf{F}_{11}) = \begin{array}{c} \{[0,0,1],[0,1,0],[0,1,1],[1,1,1],[1,3,2],[1,5,3],[1,8,3], \\ [1,8,6],[1,9,2],[1,9,8],[1,10,1],[1,10,10]\} \end{array},$$

which we write as P_1*,* P_2*, ...,* P_{10}*. For comparison,*

$$E(\mathbf{F}_{11}) \cap \mathbf{P}^2(\mathbf{F}_{11})_{z=1} = \begin{array}{l} \{[0,0],[0,1],[1,1],[1,10],[2,5],[4,9], \\ [4,10],[6,7],[6,10],[7,8],[10,1],\infty\}, \end{array}$$

where $\mathbf{P}^2(\mathbf{F}_{11})_{z=1}$ *denotes the affine patch of projective* 2*-space with* $z = 1$*. Consider the matrix*

$$G = \begin{bmatrix} 0 & 4 & 1 & 6 & 8 & 8 & 1 & 10 & 10 \\ 1 & 4 & 3 & 10 & 6 & 10 & 4 & 10 & 10 \\ 1 & 4 & 9 & 10 & 2 & 10 & 9 & 10 & 10 \\ 0 & 4 & 5 & 4 & 9 & 6 & 9 & 1 & 1 \\ 1 & 4 & 4 & 1 & 10 & 1 & 6 & 1 & 10 \\ 0 & 4 & 3 & 7 & 4 & 5 & 8 & 10 & 1 \end{bmatrix}.$$

The first row of G *gives the values of* $x^2/\phi(x,y,z)$ *at* $\{P_i \mid 1 \leq i \leq 10,\ \phi(P_i) \neq 0\} = \{P_3, P_4, P_6, P_7, ..., P_{12}\}$ *(in other words, we simply throw out all the rational prime divisors corresponding to a pole of* ϕ*). The other rows are obtained similarly from the other functions corresponding to the basis elements of* $\mathcal{L}(D)$*:* $y^2/\phi(x,y,z)$*,* $z^2/\phi(x,y,z)$*,* $xy/\phi(x,y,z)$*,* $yz/\phi(x,y,z)$*. Performing*

Gauss reduction mod 11 puts this in canonical form:

$$G' = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 2 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 5 \\ 0 & 0 & 1 & 0 & 0 & 0 & 2 & 0 & 6 \\ 0 & 0 & 0 & 1 & 0 & 0 & 6 & 6 & 2 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 2 & 5 \end{bmatrix},$$

so this code has minimum distance 3*, hence is only* 1*-error correcting. The corresponding check matrix is*

$$H = \begin{bmatrix} 1 & 1 & 2 & 6 & 0 & 1 \\ 0 & 0 & 0 & 6 & 0 & 2 \\ 2 & 5 & 6 & 2 & 2 & 5 \end{bmatrix}.$$

Example 18 *Let E denote the elliptic curve (of conductor $N = 19$) which we write in homogeneous coordinates as*

$$y^2z + yz^2 = x^3 + x^2z + xz^2$$

Let $\phi(x,y,z) = x^2 + y^2 + z^2$, let F denote the projective curve defined by $\phi(x,y,z) = 0$, and let D denote the divisor obtained by intersecting E and F. By Bezout's theorem, D is of degree 6*. As in the previous example, a basis for $\mathcal{L}(D)$ is provided by the functions in the set*

$$\mathcal{B}_D = \{1, x^2/\phi(x,y,z), y^2/\phi(x,y,z), z^2/\phi(x,y,z), xy/\phi(x,y,z), yz/\phi(x,y,z)\}.$$

(Again, this is due to the fact that $\dim \mathcal{L}(D) = \deg(D) = 6$ *and the functions $f \in \mathcal{B}_D$ "obviously" satisfy $(f) \geq -D$.) We have*

$$E(\mathbf{F}_7) = \begin{array}{c} \{\ [0,0,1],[0,1,0],[0,1,6],[1,0,2],[1,0,4], \\ [1,3,4],[1,3,6],[1,5,2],[1,5,6]\ \} \end{array},$$

which we write as P_1, P_2, ..., P_9. Consider the matrix

$$G = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 2 & 2 & 4 & 4 \\ 1 & 0 & 1 & 4 & 2 & 2 & 1 & 4 & 1 \\ 0 & 0 & 0 & 0 & 0 & 3 & 3 & 5 & 5 \\ 0 & 0 & 6 & 0 & 0 & 5 & 4 & 3 & 2 \\ 0 & 0 & 0 & 2 & 4 & 4 & 6 & 2 & 6 \end{bmatrix}.$$

The first row of G gives the values of $x^2/\phi(x,y,z)$ at $\{P_i \mid 1 \leq i \leq 9\}$. The other rows are obtained similarly from the other functions corresponding to the basis elements of $\mathcal{L}(D)$: $y^2/\phi(x,y,z)$, $z^2/\phi(x,y,z)$, $xy/\phi(x,y,z)$, $yz/\phi(x,y,z)$. Performing Gauss reduction mod 7 puts this in canonical form:

$$G' = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 4 \\ 0 & 1 & 0 & 0 & 0 & 0 & 6 & 0 & 6 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 3 & 4 \\ 0 & 0 & 0 & 1 & 0 & 0 & 6 & 1 & 6 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 4 & 4 \end{bmatrix},$$

so this code also has minimum distance 3*, hence is only* 1*-error correcting. The corresponding check matrix is*

$$H = \begin{bmatrix} 0 & 6 & 1 & 6 & 1 & 1 \\ 4 & 0 & 3 & 1 & 3 & 4 \\ 4 & 6 & 4 & 6 & 5 & 4 \end{bmatrix}.$$

For an example of the generating matrix of the elliptic code associated to $x^3 + y^3 = 1$ over $\mathbf{F}_4$ has been worked out in several places (for example, see Goppa's book mentioned above, or the books [SS], §3.3, [P], SS5.3, 5.4, 5.7, or [M], §5.7.3).

5 Concluding comments

We end this note by making some comments:

(1) The algebraic geometric relation between the number of points over a finite field for a variety is related to the Betti numbers. However, an equivalent notion of genus for higher dimensional varieties is the "arithmetic genus". Can one develop a relation between the number of points over finite fields of a variety andits arithmetic genus, useful in coding theory?

(2) Can one construct "good" codes associated to the higher dimensional Shimura varieties?

Acknowledgements The first-named author is very grateful to both Pablo Legarraga and Will Traves for very useful and informative discussions. The second author wishes to thank the hospitalities and the financial supports of the Max-Planck Institut für Mathematik (Bonn) while he was involved with this paper. We also thank Amin Shokrollahi for helpful suggestions.

References

[B] B. J. Birch, *Some calculations of modular relations*, in **Modular forms of one variable, I**, (W. Kuyk, ed.) Proc. Antwerp Conf. 1972, Springer Lecture Notes in Math., 320, Springer-Verlag, NY, 1973.

[BK] B. J. Birch and W. Kuyk (eds), **Modular forms of one variable, IV**, Proc. Antwerp Conf. 1972, Springer Lecture Notes in Math., 476, Springer-Verlag, NY, 1975.

[BBGLMS] J.-F. Boutot, L. Breen, P. Gŕardin, J. Giraud, J.-P. Labesse, J. S. Milne, C. Soulé, **Variétés de Shimura et fonctions** L Publications Mathématiques de l'Université Paris VII [Mathematical Publications of the University of Paris VII], 6. Universite de Paris VII, U.E.R. de Mathematiques, Paris, 1979.

[Cas1] W. Casselman. *On representations of* GL_2 *and the arithmetic of modular curves*, in **Modular functions of one variable, II** (Proc. Internat. Summer School, Univ. Antwerp, Antwerp, 1972), pp. 107–141. Lecture Notes in Math., Vol. 349, Springer, Berlin, 1973.

(**Correction to: "On representations of GL_2 and the arithmetic of modular curves"**, in **Modular functions of one variable, IV** (Proc. Internat. Summer School, Univ. Antwerp, Antwerp, 1972), pp. 148–149. Lecture Notes in Math., Vol. 476, Springer, Berlin, 1975.)

[Cas2] ——, *The Hasse-Weil ζ-function of some moduli varieties of dimension greater than one*, in **Automorphic forms, representations and L-functions** (Proc. Sympos. Pure Math., Oregon State Univ., Corvallis, Ore., 1977), Part 2, pp. 141–163, Proc. Sympos. Pure Math., XXXIII, Amer. Math. Soc., Providence, R.I., 1979.

[Cl] L. Clozel, *Nombre de points des variétés de Shimura sur un corps fini (d'aprés R. Kottwitz)*, Seminaire Bourbaki, Vol. 1992/93. Asterisque No. 216 (1993), Exp. No. 766, 4, 121–149.

[Co] P. Cohen, *On the coefficients of the transformation polynomials for the elliptic modular function*, Math. Proc. Cambridge Philos. Soc. 95 (1984), 389–402.

[Del] P. Deligne, *Variétés de Shimura.* In **Automorphic Forms, Representations and L-functions**, Proc. Sympos. Pure Math. 33, part 2, (1979), 247-290.

[DL] M. Duflo and J.-P. Labesse, *Sur la formule des traces de Selberg*, Ann. Sci. Ecole Norm. Sup. (4) 4 (1971), 193–284.

[Ei] M. Eichler, *The basis problem for modular forms and the traces of Hecke operators*, in **Modular functions of one variable, I** (Proc. Internat. Summer School, Univ. Antwerp, Antwerp, 1972), pp. 1–36. Lecture Notes in Math., Vol. 320, Springer, Berlin, 1973.

[E] N. Elkies, *Elliptic and modular curves over finite fields and related computational issues*, in **Computational perspectives on number theory**, (ed. D. Buell, J. Teitelbaum), AMS/IP Studies in Adv. Math., 7, 1998, 21–76.

[FM] G. Frey and M. Müller, *Arithmetic of modular curves and applications*, preprint, 1998, available at

`http://www.exp-math.uni-essen.de/zahlentheorie/preprints/Index.html`

[Gel] S. Gelbart, *Elliptic curves and automorphic representations*, Advances in Math. 21 (1976), no. 3, 235–292.

[G] V. D. Goppa, **Geometry and codes**, Kluwer, 1988.

[Ha] R. Hartshorne, **Algebraic geometry**, Springer-Verlag, 1977.

[HM] T. Hibino and N. Murabayashi, *Modular equations of hyperelliptic* $X_0(N)$ *and an application*, Acta Arithmetica 82 (1997)279-291.

[Ig] J. Igusa, *On the transformation theory of elliptic functions*, Amer. J. Math. 81 (1959)436–452.

[I] Y. Ihara, *Some remarks on the number of rational points of algebraic curves over finite fields*, J. Fac. Sci. Univ. Tokyo, 28 (1981)721–724.

[IR] K. Ireland and M. Rosen, **A classical introduction to modern number theory**, Grad Texts 84, Springer, 1982.

[Kn] A. Knapp, **Elliptic curves**, Mathematical Notes, Princeton Univ. Press, 1992.

[Ko] N. Koblitz, **Introduction to elliptic curves and modular forms**, Grad. Texts 97, Springer, 1984.

[K1] R. Kottwitz, *Shimura varieties and* λ*-adic representations*, In **Automorphic Forms, Shimura Varieties, and L-functions, Vol. 1**, Academic Press (1990), 161-209.

[K2] R. Kottwitz, *Points on Shimura varieties over finite fields*, Journal of A. M. S. 5 (1992), 373-444.

[Lab] J.P. Labesse, *Exposé VI*, in [BBGLMS].

[Lan1] R.P. Langlands, *Shimura varieties and the Selberg trace formula*, Can. J. Math. Vol. XXIX, No. 5 (1977), 1292-1299.

[Lan2] R.P. Langlands, *On the zeta function of some simple Shimura varieties*, Can. J. Math. Vol. XXXI, No. 6 (1979), 1121-1216.

[LvdG] J. Lint and G. van der Geer, **Introduction to coding theory and algebraic geometry**, Birkhäuser, Boston, 1988.

[MS] F. MacWilliams and N. Sloane, **The theory of error-correcting codes**, North-Holland, 1977.

[M] C. Moreno, **Algebraic curves over finite fields: exponential sums and coding theory**, Cambrdge Univ. Press, 1994.

[Nara] R. Narasimhan, **Complex analysis of one variable**, Basel, 1985.

[O1] A. Ogg, *Elliptic curves with wild ramification*, Amer. J. Math. 89(1967)1-21.

[O2] A. Ogg, **Modular forms and Dirichlet series**, Benjamin, 1969 (see also his paper in **Modular functions of one variable, I** (Proc. Internat. Summer School, Univ. Antwerp, Antwerp, 1972), pp. 1–36. Lecture Notes in Math., Vol. 320, Springer, Berlin, 1973.)

[P] O. Pretzel, **Codes and algebraic curves**, Oxford Lecture Series, vol 9, Clarendon Press, Oxford, 1998.

[R] J. G. Rovira, *Equations of hyperelliptic modular curves*, Ann. Inst. Fourier, Grenoble 41 (1991)779-795.

[Shim] G. Shimura, **Introduction to the arithmetic theory of automorphic functions**, Iwanami Shoten and Princeton University Press, 1971.

[ShimM] M. Shimura, *Defining equations of modular curves*, Tokyo J. Math. 18 (1995)443-456.

[Sh] M. A. Shokrollahi, *Kapitel 9* of **Beitraege zur algebraischen Codierungs- und Komplexitaetstheorie mittels algebraischer Funkionenkoerper**, Bayreuther mathematische Schriften, volume 30, pp. 1-236, 1991.

[S] S. Shokranian, **The Selberg-Arthur trace formula**, Springer-Verlag, Lecture Note Series 1503, 1992.

[SS] S. Shokranian and M.A. Shokrollahi, **Coding theory and bilinear complexity**, Scientific Series of the International Bureau, KFA Jülich Vol. 21, 1994.

[TV] M. A. Tsfasman and S. G. Vladut, **Algebraic-geometric codes**, Mathematics and its Applications, Kluwer Academic Publishers, Dordrechet 1991.

`email:wdj@gwmail.usna.edu, sash@Ipe.mat.unb.br`