Universidad de Salamanca

Facultad de Ciencias

Consejo Superior de Investigaciones Científicas

Instituto de Tecnologías Físicas y de la Información

On the hardness of the hidden subspaces problem with and without noise. Cryptanalysis of Aaronson-Christiano's quantum money scheme

Marta Conde Pena

PhD Dissertation

Salamanca, 2018

Luis Hernández Encinas, investigador científico del Instituto de Tecnologías Físicas y de la Información (ITEFI) del Consejo Superior de Investigaciones Científicas (CSIC), Raúl Durán Díaz, profesor de la Universidad de Alcalá (UAH), y Ángel Martín del Rey, profesor de la Universidad de Salamanca (USAL), certifican que la memoria titulada

On the hardness of the hidden subspaces problem with and without noise. Cryptanalysis of Aaronson-Christiano's quantum money scheme

fue realizada por Marta Conde Pena bajo la dirección de los dos primeros y la tutoría del último y que la interesada se encuentra en condiciones de optar al grado de Doctor, por lo que solicitan que sea admitida a trámite para su lectura y defensa pública.

Ámbito de conocimiento: Matemática Aplicada

Salamanca, julio de 2018

RAUL DURAN

Fdo: Luis Hernández Encinas
(Co-director)

Fdo: Raúl Durán Díaz
(Co-director)

Fdo: Ángel Martín del Rey
(Tutor)

TRIBUNAL CALIFICADOR

Tribunal nombrado por el Magfco. y Excmo. Sr. Rector de la Universidad de Salamanca el día ______ de _________________ de __________.

Presidente Dr. D. ____

Vocal Dr. D. ____

Secretario Dr. D. ____

Realizado el acto de lectura y defensa de la Tesis el día ______ de ________________ de __________ en Salamanca.

Calificación: ____

EL PRESIDENTE EL VOCAL

EL SECRETARIO

A mis padres,
y a todos los que alguna vez
emprendieron un proyecto largo y un poco loco

AGRADECIMIENTOS

Quiero empezar expresando mi agradecimiento al Consejo Superior de Investigaciones Científicas (CSIC), que en el año 2013 me dio la oportunidad de iniciar esta tesis doctoral[(*)]. Debo agradecer a Luis Hernández Encinas y a Raúl Durán Díaz el tiempo que han dedicado a la dirección de esta tesis: a Luis por su tenacidad intelectual y a Raúl por nuestras incansables reuniones de los jueves en las que tantos buenos consejos me dio; además, vuestros comentarios y correcciones fueron de gran ayuda para que esta tesis sea lo que es. Gracias también a mi tutor en la Universidad de Salamanca, Ángel Martín del Rey, por resolver amable y rápidamente todas mis dudas. No me olvido del resto de mis compañeros del Instituto de Tecnologías Físicas y de la Información, con los que compartí tres años intensos (tanto que vivimos juntos un terremoto de más de cinco grados en la escala de Ritcher un lunes de febrero): especialmente, gracias a Agus por su optimismo desbordante ante la vida, por su constante amabilidad y por todos esos partidos de pádel que jugamos y nunca gané; gracias a María Jose por los ratos que pasamos patinando en Madrid Río (si se puede llamar patinar a mi forma precaria de mantener el equilibrio); gracias a Jesús por las charlas sobre viajes (y también por poseer el gran poder de resetearme las contraseñas después de las vacaciones); gracias a Verónica por su entusiasmo contagioso; gracias a Jaime por algunas charlas sesudas sobre el teorema de Hilbert; y gracias a Carmen por todas esas conversaciones que tuvimos sobre yoga.

I would like to express my deepest gratitude to Jean-Charles Faugère and Ludovic Perret for being incredible hosts during my two short stays at Laboratoire d'Informatique de Paris 6 (LIP6): I would like to thank Jean-Charles, a meeting of fifteen minutes with you is far more effective than weeks of thinking on my own; and Ludovic, thank you for your constant support, for all those blackboard discussions and above all for your kindness, which made both of my stays really enjoyable. I also want to thank all the PhD students I met at LIP6: Alexandre, Frédéric, Thibaut, Jules and Cécile. Thank you all for the moments we shared in what I consider one of the most enriching experiences of my life, and also for all the french I learnt while I was there (oh, no, wait a minute, me learning french?, that never happened!). Outside the university sphere, I want to give a big thanks to my friend Sarah for sharing my enthusiasm about absolutely everything in Paris, for showing me many traditional places where you can eat delicious crèpes and for watching the World Cup with me in 2014; and to Nico for making me discover the best hazelnut ice cream in Paris and possibly ever.

Next, I want to thank all the people I was lucky enough to meet during my MSc in London, where the idea of starting a PhD first crossed my mind. Very specially, thanks to Gian, who became one of my best friends and taught me that a real friendship can survive not only distance but also a whole PhD (and I am not sure which of the two factors makes it harder!). Jokes aside, thank you for truly understanding me and for being always there. I also want to thank Antigoni, Britta, Sukanya (my flatmate!), Fabio, James and Chris for all the good times we had, I will treasure the memories we made all together for a long time.

Quiero terminar dando las gracias a esas personas que me acompañaron durante estos cinco años de doctorado que ahora terminan, sin las que esto habría sido aún más difícil. Gracias a Lu, a Mercedes y a Rosa por tantas experiencias compartidas durante un lustro de carrera en Santiago, por ese paso de ecuador y por ese viaje al año que tenemos que convertir en tradición. Gracias a Silvia y a Cris por otros tantos años de deporte juntas y por esa sala de escape de la que salimos por los pelos; nos quedan otras tantas. Gracias a Ge, por tantas ilusiones compartidas y porque sé que leer esto te hará la misma ilusión que a mí. Gracias a Bea por las conversaciones profundas, por las excursiones a la sierra y por los monólogos de Goyo en Madrid. Gracias a Alfonso por nuestras excursiones en bici a la casa de campo y por tus lecciones aceleradas de foto (aunque lo siento, sigo pensando que sacas las imágenes de internet). Gracias a Fer, por su paciencia y su compañía durante estos meses de escritura interminable, por prestarme otro par de ojos con los que ver bonito el mundo y por formar juntos el mejor equipo de pádel (porque "cuanto peor, mejor"). Y cómo no, gracias a mi familia: a mis padres, por verme mejor de lo que soy y por animarme a terminar en momentos en que realmente pensé que no podría; y a mi hermano Edu, por sus explicaciones de mecánica cuántica (say what?) aptas para matemáticos, y por haber sido durante estos años la prueba viviente de que se puede llegar a ser doctor y no morir en el intento.

[*] This work has been supported by the Spanish Ministry of Economy and Competitiveness through the FPI grant BES-2012-056689 (under the project with reference TIN2011-22668). The funding received from eCOST through the grant ECOST-STSM-IC1306-040515-062238, which covered a short stay at LIP6 in 2015, is also greatly acknowledged.

ABSTRACT

The boom of the internet has marked the beginning of the digital era and it has brought along a huge development of information and communication technologies, among which cryptography is the queen. Current public-key cryptography is based on two main problems that are widely accepted to be hard by the cryptographic community, namely the factorisation and the discrete logarithm problems, both of which would be compromised if efficient quantum computing was ever materialised. Since quantum computers would put modern cryptography at risk and they do not seem to be so far from becoming a reality in the not-so-distant future, the cryptographic community has begun to explore other options in order to be ready in case quantum computers appear. This has given an impulse to post-quantum cryptography, which is based on the so-called quantum-resistant problems and remains secure even for quantum computers. Post-quantum cryptography has recently attracted much attention and it is at the moment in the process of standardisation, so studying allegedly quantum-resistant problems was very relevant at the beginning of this thesis.

The core of this thesis is the study of the hardness of the *hidden subspaces problem* (HSP for short) and the *noisy hidden subspaces problem* (NHSP for short), two problems that have been claimed to be quantum-resistant. Aside from their relevance as allegedly quantum-resistant problems, they are also important because they constitute the hardness assumptions on which two versions of the very first public-key quantum money scheme with a security proof rely. This scheme is Aaronson-Christiano's, and it intends to implement quantum money — a type of money that exploits the laws of quantum mechanics to achieve unforgeability — that is verifiable by everyone. Results on the hardness of the HSP and the NHSP have a direct impact on the security of the scheme of Aaronson-Christiano, which made both problems more than worthy in our eyes to be the heart of this thesis.

Chapter 3 contains our results on the hidden subspaces problem and it is mainly based on our work [Conde Pena et al., 2015]. The HSP is originally defined by its authors over the binary field, but we extend its definition to any other finite field of prime size, always considering the instantiation proposed by its authors. After modelling the HSP with a system of equations with good properties, we use techniques of algebraic cryptanalysis to explore the system in depth. It turns out that the HSP over a field that is different from the binary one can be efficiently solved for instances meeting a certain condition, whereas the HSP over the binary field is also efficiently solvable in practice. Both our algorithms compromise the hardness of the HSP in practice as long as it is instantiated as Aaronson-Christiano propose. As a consequence, our algorithms disprove the security of the noise-free version of their scheme.

Chapter 4 contains our results on the noisy hidden subspaces problem and it is mainly based on our work [Conde Pena et al., 2018]. As we did with the HSP, we

extend the definition of the NHSP to finite fields of prime size other than two and we consider that it is instantiated as proposed by Aaronson-Christiano. It turns out that the NHSP can be reduced to the HSP over finite fields other than the binary one for instances satisfying a certain condition, whereas the NHSP over the binary field can be solved with a probability that exceeds the one stated in a conjecture assumed by the authors on which the security of the noisy version of the scheme relies. Note that while our results are obtained from a purely non-quantum perspective, concurrently to this thesis another author proved that there is a quantum reduction from the NHSP to the HSP in all cases. Therefore, the hardness of the NHSP along with the security of the noisy version of Aaronson-Christiano's scheme are fully compromised as a consequence of our findings on the HSP.

RESUMEN

El boom de internet ha marcado el comienzo de la era digital y ésta ha traído consigo un desarrollo espectacular de las tecnologías de la información y de las comunicaciones, entre las que la criptografía es la reina. La criptografía de clave pública actual está basada principalmente en dos problemas que la comunidad criptográfica asume como difíciles: la factorización y el logaritmo discreto. Sin embargo, si se llegase a construir un computador cuántico lo suficientemente potente, esta dificultad no sería tal. Así pues, la computación cuántica pondría en un grave aprieto a la criptografía moderna y, puesto que la trayectoria reciente del campo sugiere que ésta podría convertirse en una realidad en un futuro no muy lejano, la comunidad criptográfica ha comenzado a explorar otras opciones para estar lista en caso de que se logre construir un computador cuántico eficiente. Esto ha dado un impulso a lo que se conoce como criptografía post-cuántica, aquella cuya dificultad no se vería afectada por este nuevo paradigma de computación y que está basada en los llamados problemas resistentes a la computación cuántica. La criptografía post-cuántica ha suscitado mucho interés recientemente y actualmente está en proceso de estandarización, por lo que en el momento de iniciar esta tesis resultaba relevante estudiar problemas supuestamente resistentes al computador cuántico.

La parte central de esta tesis es el análisis de la dificultad del *problema de los subespacios ocultos* (HSP por sus siglas en inglés) y del *problema de los subespacios ocultos con ruido* (NHSP), dos problemas resistentes al computador cuántico según sus autores. Además de la relevancia que su supuesta resistencia a la computación cuántica les confiere, estos dos problemas son también importantes porque en su dificultad se sustenta la seguridad de las dos versiones del primer esquema de dinero cuántico de clave pública que cuenta con una prueba de seguridad. Este primer esquema es el de Aaronson-Christiano, que implementa dinero cuántico — un tipo de dinero que explota las leyes de la mecánica cuántica para crear dinero infalsificable — que cualquiera puede verificar. Los resultados obtenidos acerca de la dificultad del HSP y del NHSP tienen un impacto directo sobre la seguridad del esquema de Aaronson-Christiano, lo cual nos motivó a centrar esta tesis en estos dos problemas.

El Capítulo 3 contiene nuestros resultados acerca del problema de los subespacios ocultos y está fundamentalmente basado en nuestro trabajo [Conde Pena et al., 2015]. Los autores del HSP lo definieron originalmente sobre el cuerpo binario, pero nosotros extendemos la definición a cualquier otro cuerpo finito de orden primo, siempre considerando que la instanciación es la que los autores proponen. Después de modelar el HSP con un sistema de ecuaciones con buenas propiedades, usamos técnicas de criptoanálisis algebraico para explorar el sistema en profundidad. Para el HSP sobre cualquier cuerpo que no sea el binario diseñamos un algoritmo que resuelve de manera eficiente instancias que satisfacen una cierta condición. Utilizando técnicas distintas, construimos un algoritmo heurístico, sustentado por argumentos teóricos, que resuelve eficientemente instancias del HSP sobre el cuerpo binario. Ambos algo-

ritmos comprometen la dificultad del HSP siempre que las instancias del problema sean escogidas como Aaronson-Christiano proponen. Como consecuencia, nuestros algoritmos vulneran la seguridad de la versión del esquema sin ruido.

El capítulo 4 contiene nuestros resultados acerca del problema de los subespacios ocultos con ruido y está fundamentalmente basado en nuestro trabajo [Conde Pena et al., 2018]. Al igual que con el HSP, extendemos la definición del NHSP a cualquier otro cuerpo de orden primo y consideramos instancias generadas como especifican Aaronson-Christiano. Mostramos que el NHSP se puede reducir al HSP sobre cualquier cuerpo primo que no sea el binario para ciertas instancias, mientras que el NHSP sobre el cuerpo binario se puede resolver con una probabilidad mayor de la asumida por los autores en la conjetura sobre la que la seguridad de su esquema con ruido se sustenta. Aunque nuestros resultados se obtienen desde un punto de vista puramente no cuántico, durante el desarrollo de esta tesis otro autor demostró que existe una reducción cuántica del NHSP al HSP también en el caso binario. Por tanto, la dificultad del NHSP y la seguridad del esquema de Aaronson-Christiano con ruido se han visto comprometidas por nuestros descubrimientos acerca del HSP.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

We live in the digital era: we have a non-stop supply of news from all over the world and information from any imaginable topic just one click away; we shop online, learn online and job hunt online; we arrange video calls with friends who are thousands of kilometres away; we track our runs to monitor our progress; we use social media to interact with almost anyone we want and to raise our voices against any cause we may feel passionate about...

These above are only some examples of how the internet has revolutionised our daily lives and shaped the world we live in. As a consequence of this new lifestyle, the amount of private information that users leave behind on the internet has grown to be huge, while still expected by its owners to remain private. The need of meeting this demand has driven cryptography, the discipline that studies mathematical techniques to protect and make communications secure, to a level of development that would have possibly not been anticipated in its early times (see [Singh, 2011] for a history of cryptography). Today, cryptography is present everywhere and it has reached such a peak that it seems difficult to find any technological device without any cryptography embedded into it.

Cryptography has definitely come a long way since its origins. After the times of the paradigm "security through obscurity" were over, cryptography started to be mathematically systematised [Shannon, 1948], a crucial step for it to move forward. Until around the 1970s, cryptography was based on secret-key (or symmetric) systems, which used the same key for encryption and decryption. Therefore, it was required that the two parties intending to communicate exchanged a key prior to their communication. Maintaining this key secret was crucial for the security of these schemes, which meant that there had to be a secure channel for both parties to exchange their shared key. This was not practical for mass adoption and cryptography needed to transcend.

It did in 1976 when the Diffie-Hellman key exchange [Diffie and Hellman, 1976] was proposed, showing that a secure channel was not needed in order to establish a key. This work was a breakthrough and it gave rise to public-key (or asymmetric) systems, which use one key for encryption (which is public) and a different key for decryption (which is kept secret by each user). The security of this kind of systems relies on the fact that it is hard to recover the secret key from the public key, in the sense that doing so would imply solving a mathematical problem that is assumed to be computationally intractable by the cryptographic community.

Nowadays communications are usually protected with a combination of public-key cryptography, often used for key establishment, and secret-key cryptography, often used for encryption and decryption. As of today, it is widely accepted by the community that cryptography is secure with the actual computational resources assuming appropriate choices of key lengths and standard implementations. However, this is not the whole story.

In 1997, Peter W. Shor proved in [Shor, 1997] that if a sufficiently powerful quantum computer —a type of computer that takes advantage of the capability of subatomic particles to exist in more than one state at any time— was built, then there would exist polynomial-time algorithms that would break both the factorisation problem and the discrete logarithm problem. This would compromise the two main hardness assumptions on which public-key cryptography is based [Rivest et al., 1978, ElGamal, 1985, Koblitz, 1987, Miller, 1985]: in the era of quantum computers, RSA, ElGamal and cryptography based on elliptic curves would no longer be secure. Furthermore, Grover showed in [Grover, 1997] that efficient quantum computing would also have an impact on private-key cryptography, reducing the time needed to break current private-key systems to its square root.

Efficient quantum computation may not have been any close to becoming a reality back in the nineties, but today research in quantum computing is very active —with corporate giants like Google, IBM or Microsoft getting involved— and showing some progress [Conover, 2018, Google, 2018, IBM, 2018, Institute for Quantum Computing, 2018a, Institute for Quantum Computing, 2018b, Microsoft, 2018], which puts modern cryptography in a very vulnerable position. So much that we are likely witnessing yet another time in which cryptography needs to transcend. And it has already started to do so.

Anticipating the threat that the construction of an efficient quantum computer would pose to current cryptography, cryptographers have started to explore a new kind of problems on which to base new public-key schemes. Specifically, problems that are hard even for quantum computers, which are usually referred to as quantum-resistant problems. Since the cryptographic community agrees on the fact that quantum computing may be a reality in the not-so-distant future, studying quantum-resistant problems becomes particularly relevant.

The kind of cryptography that would still be secure if an efficient quantum computer was ever built is known as post-quantum cryptography [Bernstein et al., 2009] and it has recently attracted considerable attention. At the moment, post-quantum cryptography comprises four main lines of research: lattice-based cryptography [Peikert, 2014, Güneysu et al., 2012, Zhang et al., 2015, Ducas et al., 2013], hash-based cryptography [Merkle, 1979, Lamport, 1979, Buchmann et al., 2011, Bernstein et al., 2015], code-based cryptography [McEliece, 1978, Niederreiter, 1986, Courtois et al., 2001] and multivariate-quadratic-based cryptography [Matsumoto and Imai, 1988, Patarin, 1996, Faugère and Joux, 2003, Braeken et al., 2005]. Multivariate-quadratic-based cryptography builds on the hardness of solving random multivariate quadratic systems of polynomial equations and it caught our attention at the beginning of this thesis. That gave rise to the article [Conde Pena et al., 2014] focusing on the isomorphism of polynomials problem, which essentially consists in deciding whether two families of multivariate polynomials are isomorphic via two affine mappings.

However, the core of this thesis is a related problem introduced in [Aaronson and Christiano, 2013] which we noticed some time later and whose study gave rise to our work [Conde Pena et al., 2015, Conde Pena et al., 2018]. This problem is known as the *hidden subspaces problem* and it was claimed by its authors to be quantum-resistant. The problem was new in the literature to the best of our knowledge, although it can be seen as a modification of a variant of the isomorphism of polynomials problem known as the isomorphism of polynomials with one secret problem [Patarin et al., 1998, Macario-Rat et al., 2013, Geiselmann et al., 2003, Perret, 2005]. Since post-quantum cryptography was in an early stage of development at the beginning of this dissertation (currently undergoing NIST's post-quantum cryptography standardisation process [NIST, 2018]), it seemed like the right time to explore in depth allegedly quantum-resistant problems and study their hardness to determine whether they stood or fell. Although the hidden subspaces problem did not underlie the security of any post-quantum system, thoroughly analysing its hardness was the only way to get an idea about its strength as a potential candidate to do so.

Aside from its relevance in the context of post-quantum cryptography, the hidden subspaces problem also has interest because it underlies the security of a public-key quantum money scheme designed by Scott Aaronson and Paul Christiano in the aforementioned paper [Aaronson and Christiano, 2013]. Public-key quantum money does probably not sound familiar, but a naive guess about it having something to do with money, cryptography and quantum mechanics turns out to be correct. Surprisingly, and as oblivious to technological advances as the subject of cash may appear, cryptography is having an impact over it. As of today, two areas of research on the subject of money, namely cryptocurrencies and quantum money, stand out as two different approaches to design what the money of the future could be.

Before going into quantum money, let us briefly go over the research area concerning the so-called *cryptocurrencies*, which probably do sound familiar since the advent of Bitcoin [Nakamoto, 2008, Nakamoto, 2009], the first and likely the most widely-known cryptocurrency. Bitcoin is the first decentralised electronic currency, which essentially means that it has a distributed generation and verification of money. To picture what this means, think about Wikipedia, an online encyclopedia that anyone can contribute to (by creating or modifying an entry) thanks to a group of moderators that ensure the veracity and absence of abusive language of the content. The functioning of Bitcoin is somehow analogous: transactions are from peer to peer and they are included in a public decentralised ledger (a blockchain) by a group of users who are responsible for verifying the transactions and keeping the consensus. For a more visual explanation of the foundations of a cryptocurrency, see Figure 1.1 (source: https://course-studies.corsairs.network/understanding-bitcoin-a-course-study-d919bd01730b).

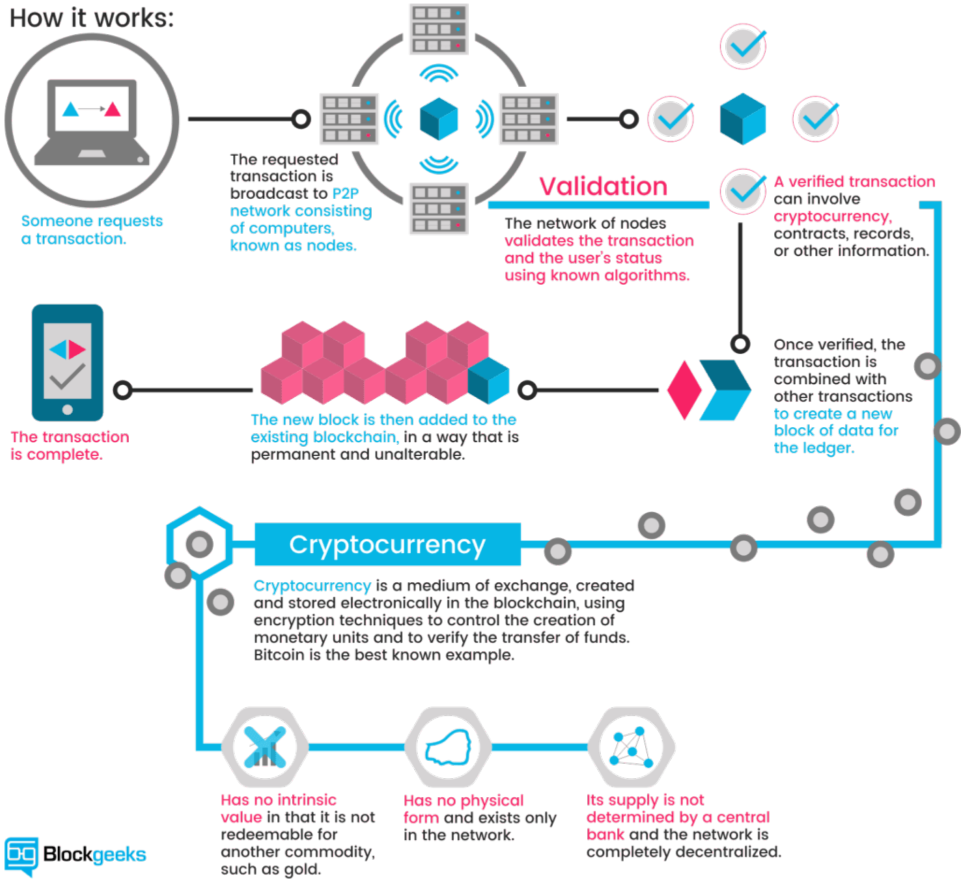
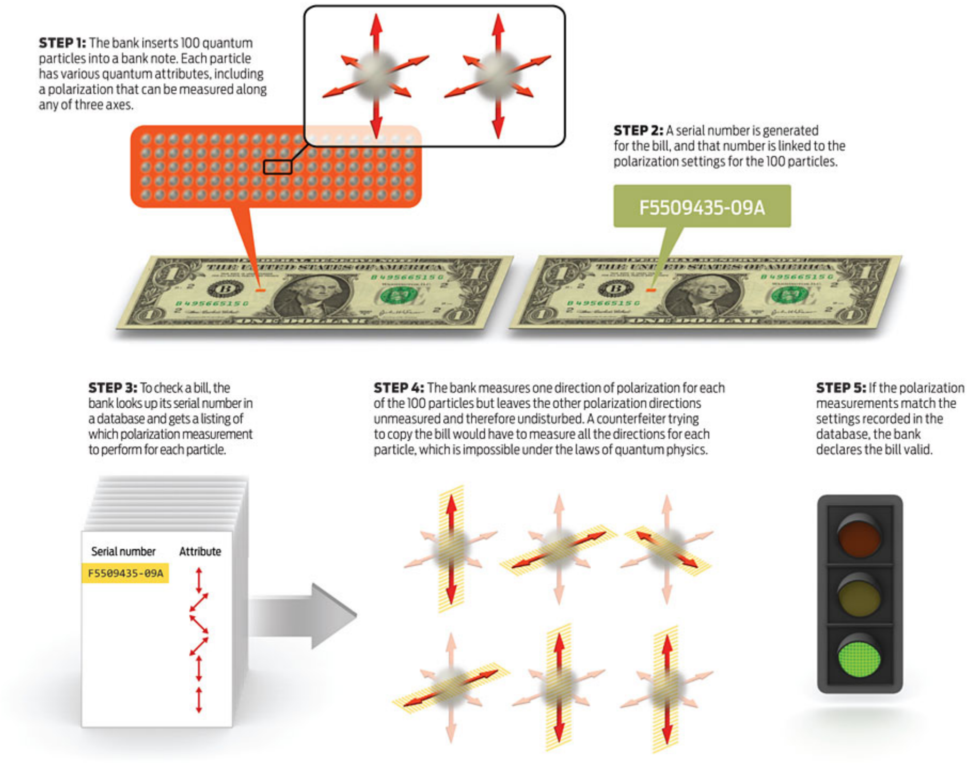

How it works:
Someone requests a transaction.
The requested transaction is broadcast to P2P network consisting of computers, known as nodes.
Validation
The network of nodes validates the transaction and the user's status using known algorithms.
A verified transaction can involve cryptocurrency, contracts, records, or other information.
Once verified, the transaction is combined with other transactions to create a new block of data for the ledger.
The new block is then added to the existing blockchain, in a way that is permanent and unalterable.
The transaction is complete.
Cryptocurrency
Cryptocurrency is a medium of exchange, created and stored electronically in the blockchain, using encryption techniques to control the creation of monetary units and to verify the transfer of funds. Bitcoin is the best known example.
Has no intrinsic value in that it is not redeemable for another commodity, such as gold.
Has no physical form and exists only in the network.
Its supply is not determined by a central bank and the network is completely decentralized.
Blockgeeks

Figure 1.1: Sketch of how a cryptocurrency works

The other research area concentrates on what is called *quantum money*, a type of money devised with the hope of constructing money that cannot be forged other than with a negligible probability. Ever since the origin of cash, forgery has been an issue to worry about and even though many efforts have been put into preventing it through history (from milled edges in Newton's time to today's embedded strips or special inks that look different depending on the angle of vision), it is impossible to completely prevent it. This is because cash is built with a physical device under the laws of classical physics, which inherently makes it theoretically possible for a counterfeiter to replicate the process. Furthermore, surpassing that physical implication and preventing forgery seems rather insurmountable.

Even so, one can think out of the box and wonder if provably unforgeable money can be created in any other way. This is what Stephen Wiesner wondered in his article [Wiesner, 1983], where he proposed to break the mould and construct money that obeyed the laws of quantum mechanics instead. In the quantum mechanical subatomic world, it turns out to be impossible to clone a quantum particle whose state is unknown [Park, 1970, Wootters and Zurek, 1982, Dieks, 1982, Wootters and Zurek, 2009]. With that in mind, Wiesner proposed that the bank embedded several quantum particles into each banknote and stored their quantum descriptions in a secret database. This way, every time a banknote needed to be validated, the bank would need to look up the database and do appropriate quantum measurements to check that the quantum particles of the banknote matched the corresponding description stored in the database. However, a counterfeiter intending to replicate a banknote would not know with certainty which were the states of the quantum particles embedded into it, and so she would not be able to replicate them as a consequence of the laws of quantum mechanics. The details of quantum money are slightly more complex and will be explained in Chapter 2, but we have oversimplified it now for the sake of a better understanding. See Figure 1.2 (source: `https://futureofmoney2025.weebly.com/quantum-money.html`) to obtain an anticipation of the detailed idea.



STEP 1: The bank inserts 100 quantum particles into a bank note. Each particle has various quantum attributes, including a polarization that can be measured along any of three axes.

STEP 2: A serial number is generated for the bill, and that number is linked to the polarization settings for the 100 particles.

STEP 3: To check a bill, the bank looks up its serial number in a database and gets a listing of which polarization measurement to perform for each particle.

STEP 4: The bank measures one direction of polarization for each of the 100 particles but leaves the other polarization directions unmeasured and therefore undisturbed. A counterfeiter trying to copy the bill would have to measure all the directions for each particle, which is impossible under the laws of quantum physics.

STEP 5: If the polarization measurements match the settings recorded in the database, the bank declares the bill valid.

Figure 1.2: Sketch of how quantum money works

1.1 Justification and objectives

Choosing the hidden subspaces problem to be the focus of this thesis has a double motivation, in the sense that this problem is relevant regarding two different areas.

First, having a good understanding of the hardness of alleged quantum-resistant problems seemed important in the context of increasing efforts being dedicated to research on post-quantum cryptography at the start of this thesis. Analysing in depth the hardness of the hidden subspaces problem, something no one had yet accomplished to the best of our knowledge, felt then like an enticing challenge.

Second, studying the hardness of the hidden subspaces problem finds applications in the area of quantum money, as it underlies the security of the quantum money scheme of Aaronson-Christiano. Explaining why this scheme is particularly relevant needs further details: let us try to explain why it marked a milestone in the field of quantum money.

Although it was not well understood at the time of publication, the idea of Wiesner [Wiesner, 1983] was promising and his work brought along the hope of unforgeable money. However, as it usually happens with papers that are groundbreaking, it presented several drawbacks. Perhaps the most urgent issue of Wiesner's quantum money scheme was the size of the bank's database, which was huge due to it having to store the quantum description of each banknote in circulation. This was solved in [Bennett et al., 1982]. Several other papers followed the work of Wiesner [Mosca and Stebila, 2010, Gavinsky, 2012] and tried to overcome some of its disadvantages in different manners.

Nevertheless, what might be the most profound issue of all is that Wiesner's money can only be verified by the issuing entity, jeopardising its usability. Some partial solutions to this problem have been found in [Mosca and Stebila, 2010, Gavinsky, 2012, Molina et al., 2012, Pastawski et al., 2011], but the main efforts in quantum money research today are directed towards achieving a more ambitious objective: constructing quantum money that can be verified by anyone rather than only by the authority that issued it. This concept of *publicly-verifiable quantum money*, referred to as public-key quantum money in a clear analogy to public-key cryptography, was first introduced by Scott Aaronson in [Aaronson, 2009].

In that very same paper Aaronson himself proposed the first *public-key quantum money scheme*, subsequently followed by several other proposals by other authors [Lutomirski, 2011, Farhi et al., 2012]. However, all proposed schemes were either broken or lacked a systematic way for evaluating their security. In this sense, Scott Aaronson and Paul Christiano were the first authors to propose a public-key quantum money scheme [Aaronson and Christiano, 2013] that was proved to be cryptographically secure under a certain hardness assumption. The problem which was assumed to be hard and quantum-resistant was precisely the hidden subspaces problem.

To sum up, as an allegedly quantum-resistant problem that underlay the security of the first public-key quantum money scheme with a security proof, the hidden subspaces problem appeared like a perfect problem to focus on. Additionally, quantum money —which aspires to eliminate the traditional problem of counterfeit money and constitutes an exciting long-term challenge that could end up replacing standard money— seemed like a motivating background. We chose the objectives of this thesis to be:

- The analysis of the hardness of the hidden subspaces problem over any finite field of prime order, and the impact that the findings have on the security of the quantum money scheme of Scott Aaronson and Paul Christiano [Aaronson and Christiano, 2013].

- The analysis of the hardness of a noisy version of the hidden subspaces problem over any finite field of prime order, and the impact that the findings have on the security of the noisy version of the quantum money scheme of Aaronson-Christiano, proposed in the same paper as the scheme without noise.

1.2 Methodology and work plan

The hidden subspaces problem and its noisy version, which constitute the heart of this thesis, are strongly based on polynomial theory. To approach the analysis of their hardness we first need to study the existing techniques for solving non-linear multivariate polynomial systems, which is hard in general. Still, algorithms for solving non-linear systems turn out to be efficient in practice in some occasions. In this sense, we have to study the core details of two of the main algorithms for that purpose, namely the F_4 and the F_5 [Faugère, 1999, Faugère, 2002], along with the complexity results they achieve. These algorithms are the main tool we use to deal with polynomial systems throughout this thesis.

Before proceeding to study the hidden subspaces problem, we need to find a suitable polynomial system that models it. This is a relevant step because the system should be constructed so that it has good properties that can translate into a better performance of the algorithms on it. After constructing the model, we need to implement the generation of instances of the hidden subspaces problem —that should be as generic as possible— and the system of equations that the model particularised to an instance yields. We use the Magma [Bosma et al., 1997] software for this step. We should then try to see if we can identify any structure in the system, maybe dependent on some characteristic of the instances, that makes the process of solving it efficient. If that is the case, we will move on to design an algorithm to find a solution of the hidden subspaces problem.

As for the hidden subspaces with noise, the natural approach is to investigate if there is a reduction from the noisy hidden subspaces to the noise-free hidden subspaces problem. If so, then we can apply the results we attained on the noise-free case to the case with noise. If this approach does not succeed, we should then try to see if the noisy hidden subspaces problem presents any intrinsic weakness that can be exploited.

1.3 Contents of the chapters

We close the introduction with a short overview of the chapters ahead.

1.3.1 Chapter 2. Preliminaries and notation

This chapter gathers all the information needed to read the rest of this manuscript smoothly. We start by fixing the notation used throughout and by contextualising our area of research, connecting it to previous and related work. We then go along to formally define the hidden subspaces and the noisy hidden subspaces problems, which constitute the core of this thesis, and to explain how they underlie the security of the public-key quantum money scheme proposed by Scott Aaronson and Paul Christiano in [Aaronson and Christiano, 2013]. Lastly, we dedicate considerable time to explain the concept of a Gröbner basis and its application to non-linear multivariate polynomial system solving. We give details about the existing algorithms to compute a Gröbner basis, with special emphasis on F_4 and F_5, and we discuss the complexity results that they achieve.

1.3.2 Chapter 3. Cryptanalysis of Aaronson and Christiano's scheme: The noise-free case

Along this chapter we construct a polynomial system that models the hidden subspaces problem over a finite field of prime size. Once this model is presented, we go on to study if there is any structure in it that can be exploited. In this sense, we present an algorithm that solves instances of the hidden subspaces problem over a finite field other than the binary one provided that a certain condition is met. Our algorithm is randomised and it runs in polynomial time, with a complexity that we characterise. This is complemented with experimental results that confirm the efficiency of the algorithm in practice. Moreover, we present a probabilistic algorithm that heuristically solves the hidden subspaces problem over the binary field in polynomial time. We give a theoretical result that supports our heuristic algorithm and run experiments that turn out to be very efficient in practice. Both of these algorithms yield a cryptanalysis of the noise-free quantum money scheme of Aaronson-Christiano and the extension of it to any other field as long as a certain condition is satisfied.

This chapter is mainly based on our work [Conde Pena et al., 2015].

1.3.3 Chapter 4. Cryptanalysis of Aaronson and Christiano's scheme: The noisy case

In this chapter we focus on the noisy hidden subspaces problem, designed to allegedly enhance the hardness of the hidden subspaces problem. In the case that the finite field is not the binary one and as long as it satisfies a certain condition, we show that there is a polynomial-time reduction from the noisy hidden subspaces problem to the noise-free version of the problem. In combination with our results of Chapter 3, this yields a probabilistic polynomial-time solving the noisy hidden subspaces problem in this scenario. This is complemented with experimental results on the performance of the algorithm, which proves to be as efficient in practice as expected. As for the case of the noisy hidden subspaces over the binary field, we show that exhaustive search together with our algorithm that solves the noise-free problem is enough to break a hardness conjecture made by the authors. The first algorithm yields a cryptanalysis of the extension of the noisy scheme of Aaronson-Christian to any field other than the binary one, whereas the second one breaks a conjecture on its hardness over the binary field.

This chapter is mainly based on our work [Conde Pena et al., 2018].

1.3.4 Chapter 5. Conclusions, contributions and future work

In this chapter we summarise our contributions and we discuss the impact that our findings have on the hardness of the hidden subspaces problem with and without noise. We also comment on several areas of research that arose during this thesis (some of them were initiated but unfinished and other were not explored) that could be interesting for future work.

1.3.5 Appendix A. Magma codes

In this appendix we include the source code in the Magma software (also available online in GitHub, see https://github.com/Marta-PhD/solving-HSP-NHSP) to generate and solve instances of the hidden subspaces problem and its noisy version. This way, our experiments can be replicated and further ones can be performed if desired.

Chapter 2

Preliminaries and notation

2.1 State of the art

Choosing the hidden subspaces problem with and without noise to be the centre of this thesis is partially motivated by the fact that they underlie the security of the noisy and noise-free version of Aaronson-Christiano's public-key quantum money scheme. In order to understand how these problems relate to the security of the former quantum money schemes, we first need to explain more in detail what is quantum money. As we sketched in the introduction, the idea of Wiesner's quantum money is to construct money that does not obey the laws of classical physics but those of quantum mechanics instead, with the hope of preventing forgery by doing so. Indeed, under the laws of quantum mechanics things are quite different: new game, new rules.

Let us very briefly explain some basic notions of quantum mechanics that are required to understand the idea of quantum money. First note that every quantum particle has attributes that can be measured (called observables), and that the value of these observables at a given time determines what is called the quantum state of the particle. A particle can have many observables, but we are interested in one that is called the spin projection, which can be measured along an (oriented) axis. In particular, we are interested in quantum particles whose spin projection can take only two distinct possible values, either a number that is greater than zero or its opposite. Finding a classical analogy of what the concept of spin projection means is not easy. Often, the result of measuring the spin projection of a particle along a given axis is interpreted as if the quantum particle rotates (*spins*) either clockwise or anticlockwise around that axis. This is, as if the quantum particle is polarised along the axis. For simplicity, we refer to the spin projection as the polarisation and we make use of this classical analogy in what follows.

The polarisation attribute of particles that can only take two distinct values (either greater or less than zero) is important in the context of unforgeable money due to the following. The laws of quantum mechanics guarantee that if the polarisation is measured along the axis around which the quantum particle is indeed spinning, then the result is always greater than zero. However, if the polarisation is measured along an axis which is different from the axis around which the quantum particle is spinning, then the quantum state gets altered and the result of measuring the polarisation is random.

Wiesner thought about using the above property of quantum particles in his favour essentially as follows: whenever a banknote needs to be created, the bank assigns a serial number to it and prepares n quantum particles with random polarisations, which are embedded into the banknote. More concretely, in Wiesner's scheme each of the quantum particles that form the banknote are chosen to be either spinning clockwise or anticlockwise around the axis X or spinning clockwise or anticlockwise around the axis Z (see Figure 1.2). Once the banknote is generated, its serial number and the polarisation of the particles embedded into it are stored in a secret database owned by the bank.

This way, every time that a banknote needs to be validated, the bank can look up its secret database and measure the polarisation of each quantum particle along the appropriate axis, which should be greater than zero. However, this measurement will not be positive with certainty if it is carried out by a counterfeiter intending to replicate the banknote, since he ignores the polarisations of the quantum particles. Recall that if the counterfeiter measures the polarisation of a particle along an incorrect axis, then the quantum state changes and the measurement of its polarisation is randomly positive or negative. Still, the counterfeiter may be lucky enough to obtain the correct polarisation measurements if he happens to measure along all the appropriate axes, but the probability of copying a banknote successfully is proved in [Molina et al., 2012] to be at most $(3/4)^n$ (where n is the number of particles embedded into the banknote), which means that it decreases exponentially with the number of quantum particles embedded into the banknote.

We already said in the introduction that Wiesner's proposal for quantum money was not perfect and presented some drawbacks: let us detail this a bit more now. The issue concerning the huge size of the bank's secret database was solved in [Bennett et al., 1982], where the authors propose a variant of Wiesner's scheme (the BBBW scheme) in which the quantum particles are generated using a pseudorandom function depending on a key only known by the bank. Of course, this reduces the information-theoretical security of Wiesner's original scheme to computational security, but reducing the size of the database was essential to bringing Wiesner's scheme any closer to being practical. Besides, all modern cryptography is constructed over computational security anyway.

Another issue for both Wiesner's scheme and the BBBW scheme was pointed out in [Aaronson, 2009, Lutomirski, 2010]. It turns out that the former schemes can be broken in linear time if a counterfeiter can submit an alleged banknote to the bank for verification and get back not only the validity or invalidity of the banknote but also the post-measurement quantum state. This allows the counterfeiter to successively guess the polarisation of each of the quantum particles embedded into the banknote. Nonetheless, note that it is not mandatory to return the post-measurement quantum states to the counterfeiter.

But as we already sketched in the introduction, perhaps the most serious drawback of Wiesner's scheme is that money can only be verified by the issuer bank, which somehow boycotts its usability. This is referred to in the literature as the verifiability problem and it has attracted quite a lot of attention recently. Some solutions have been found: in [Mosca and Stebila, 2010] the authors suggest for the bank to delegate the verification process on the retailer through a blind quantum computing protocol, and in [Gavinsky, 2012] Gavinsky proposed a variant of Wiesner's scheme which requires only classical communication between the vendor and the bank (see also followup work [Molina et al., 2012, Pastawski et al., 2011]). These solutions heavily rely on a trusted third party.

However, many efforts in the field of quantum money today are precisely directed towards constructing money whose verification is less dependent on a third party. More in detail, the aim is constructing quantum money such that:

- The bank can create banknotes, which means that there is an efficient algorithm to generate the random quantum states.
- Anyone can verify the validity of a banknote, which means that there is an efficient and public algorithm to verify, with high probability, if the banknote was generated by the bank
- No one can copy it, this is, no one other than the bank can efficiently produce quantum states that are accepted by the verification procedure with more than an exponentially small probability.

A scheme that fulfils these requirements is called a *public-key quantum money scheme* (see [Aaronson, 2009]), in analogy with public-key cryptosystems.

At the moment of starting this thesis, the public-key quantum money schemes that had been proposed had either been broken or their security could not have been proved. Indeed, Aaronson gave in his paper [Aaronson, 2009] the first proposal for a public-key quantum money scheme, but it was broken soon after in [Lutomirski et al., 2010] using techniques from [Alon et al., 1998]. Another public-key quantum money scheme based on knot theory was suggested in [Farhi et al., 2012], and an abstract version of it in the followup work [Lutomirski, 2011]. However, characterising which quantum states were accepted by the verification procedure of [Farhi et al., 2012] seemed to require major advances in knot theory, and similarly for the case of [Lutomirski, 2011]. Therefore, all the public-key quantum money schemes proposed in the literature suffered from the same problem at the beginning of this dissertation: the lack of a systematic way to evaluate their security.

In this sense, Scott Aaronson and Paul Christiano were the first authors to propose a public-key quantum money scheme [Aaronson and Christiano, 2013] with a proof of security assuming the resistance to quantum computing of a new problem in the literature, which they called the *hidden subspaces problem*. In short, this problem consists in recovering two mutually orthogonal subspaces, each of them encoded (*hidden*) as the common zeros of a set of polynomials. In this thesis we are the first ones to address the question of how hard the hidden subspaces problem is.

After this brief contextualisation we are ready to define the hidden subspaces problem with and without noise. Before proceeding to do so, let us fix the notation that will be used throughout this manuscript.

2.2 Notation

In this dissertation, $\mathbb{Z}_{\geq 0}$ denotes the set of integers greater than or equal to zero and $\mathbb{R}^+$ denotes the set of real positive numbers.

Note that we follow the standard Big-O and Big-Omega notation to describe asymptotic behaviour in the context of computational complexity theory. In particular, if f and g are real-valued functions defined on the set of natural numbers, we write

$$\begin{aligned} f\left(x\right) = \mathcal{O}\left(g\left(x\right)\right) &\Longleftrightarrow \exists c \in \mathbb{R}^+, \exists x_0 \in \mathbb{N}, \text{ such that } f\left(x\right) \leq cg\left(x\right), \quad \text{for } x \geq x_0 \\ &\Longleftrightarrow \lim_{x\to\infty} \frac{f(x)}{g(x)} = 0 \end{aligned}$$

and

$$f\left(x\right) = \Omega\left(g\left(x\right)\right) \Longleftrightarrow g\left(x\right) = \mathcal{O}\left(f\left(x\right)\right).$$

In the same context, we denote the matrix multiplication exponent by ω. Elaborating a bit more, the matrix multiplication algorithm that results from the definition requires n^3 multiplications and $(n-1)n^2$ additions in order to multiply two square matrices of order n. The naive computational complexity of matrix multiplication is therefore $\mathcal{O}(n^3)$. However, this complexity is not optimal and in [Strassen, 1969] a complexity of

$$\mathcal{O}\left(n^{\log_2 7}\right) \approx O\left(n^{2.807}\right)$$

was achieved. The exponent appearing in the complexity of matrix multiplication has been improved several times, and the lowest bound for it is generally denoted by ω. It occurs that, up to date, $2 \leq \omega < 2.373$ [Le Gall, 2014].

Throughout this thesis, $\mathbb{F}$ denotes a finite field of any prime order and n always denotes an even integer unless otherwise specified.

In what concerns polynomials, we set $\mathbf{x} = (x_1, \ldots, x_v)$ to be a v-vector of symbolic variables over $\mathbb{F}$ and $\mathbb{F}[\mathbf{x}]$ denotes then the polynomial ring in v variables over $\mathbb{F}$. Given a certain subspace $A \subset \mathbb{F}^v$ and $d \in \mathbb{N}$, we maintain the notation of [Aaronson and Christiano, 2013] and we take $I_{d,A}$ to denote the set of degree-d polynomials in $\mathbb{F}[\mathbf{x}]$ that vanish on A. Furthermore, given a subspace $A \subset \mathbb{F}^v$, the subspace $A^\perp$ denotes its orthogonal complement. Recall that given a subspace $A \subset \mathbb{F}^v$, its orthogonal complement is defined as

$$A^\perp = \{x \in \mathbb{F}^v : x \cdot a = 0, \quad \forall a \in A\},$$

where $\cdot$ denotes the standard scalar product in $\mathbb{F}^v$.

Given $m \in \mathbb{N}$ and certain polynomials $p_1, p_2, \ldots, p_m, q_1, q_2, \ldots, q_m$ in $\mathbb{F}[\mathbf{x}]$, we write $(\mathbf{p}, \mathbf{q}) = \left((p_1, \ldots, p_m), (q_1, \ldots, q_m)\right) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$. Finally, we denote by

$$\mathrm{M}(\mathbb{F}[\mathbf{x}]), \quad \mathrm{M}_d(\mathbb{F}[\mathbf{x}])$$

the set of monomials in $\mathbb{F}[\mathbf{x}]$ and the set of monomials of degree $d \in \mathbb{N}$ in $\mathbb{F}[\mathbf{x}]$. Since the cardinality of the set $\mathrm{M}_d(\mathbb{F}[\mathbf{x}])$ will be often used in Chapter 4, we denote it as

$$N_{d,\mathbf{x}}^{|\mathbb{F}|}$$

for the sake of shortness.

As for matrices, the sets

$$\mathcal{M}_{k,\ell}(\mathbb{F}), \quad \mathcal{M}_k(\mathbb{F}), \quad \mathrm{GL}_k(\mathbb{F})$$

denote the set of $k \times \ell$ matrices with entries in $\mathbb{F}$, the set of square matrices of order k with entries in $\mathbb{F}$ and the set of invertible matrices in $\mathcal{M}_k(\mathbb{F})$, respectively. For a matrix $G = \left(g_{i,j}\right) \in \mathcal{M}_{k,\ell}(\mathbb{F})$, the matrix $G^T = \left(g_{j,i}\right) \in \mathcal{M}_{\ell,k}(\mathbb{F})$ denotes the transpose of the matrix G as usual. Given two matrices $A_1 \in \mathcal{M}_{k,\ell}(\mathbb{F})$, $A_2 \in \mathcal{M}_{k,h}(\mathbb{F})$, the matrix $(A_1|A_2) \in \mathcal{M}_{k,\ell+h}(\mathbb{F})$ denotes its concatenation. Finally, we express by $\gamma_{|\mathbb{F}|}(k)$ the probability that a square matrix of order k over $\mathbb{F}$ is invertible, which is dependent on the size $|\mathbb{F}|$ of the ground field.

2.3 The hidden subspaces problem (with and without noise)

Aaronson and Christiano introduced the hidden subspaces problem (with and without noise) in [Aaronson and Christiano, 2013], along with two versions of a public-key quantum money scheme based upon the latter problems. They claimed both versions of the problem to be quantum-resistant and, considering that the noise-free problem is somehow similar to other hard problems used as a basis for multivariate cryptography it is not unreasonable to believe a priori that it may be.

Briefly, the hidden subspaces problem consists in recovering a subspace whose elements, along with those from its orthogonal, have been encoded as the set of zeros of certain random multivariate polynomials over a finite field. It is formally defined as follows.

The Hidden Subspaces Problem ($\mathrm{HSP}_{|\mathbb{F}|}$ for short)
Input: $(\mathbf{p}, \mathbf{q}) = ((p_1, \ldots, p_m), (q_1, \ldots, q_m)) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ of degree $d \geq 3$, with $\mathbf{x} = (x_1, \ldots, x_n)$ and $n \leq m \leq 2n$.
Find: a subspace $A \subset \mathbb{F}^n$ of dimension $n/2$ such that

$$p_i(A) = 0 \text{ and } q_i(A^{\perp}) = 0, \quad \forall i \in \{1, \ldots, m\}.$$

Remark 2.1 *The hidden subspaces problem was defined by the authors over $\mathbb{F}_2$, but we extended its definition to any finite field of prime size in order to study the widest possible version of the problem. Besides, the hardness of the hidden subspaces problem over a finite field of cardinality other than two is left as an open problem by the authors themselves in* [Aaronson and Christiano, 2013].

Even though the authors claimed that the hidden subspaces problem was already quantum-resistant, they proposed a variant of the former problem with a supposedly enhanced hardness due to the addition of noise. Adding noise to a problem is a common technique used in cryptography to increase the difficulty of problems. The learning with errors problem (LWE problem for short) introduced by Regev in [Regev, 2009] is an example of a hard problem used in cryptography derived from adding noise to a problem that is not hard at all [Goldreich et al., 1997, Kawachi et al., 2007, Peikert, 2009, Güneysu et al., 2012, Bos et al., 2015, Peikert, 2014, Alkim et al., 2015].

More concretely, given a prime $p \geq 2$, if we have a system of linear equations over $\mathbb{Z}_p$ as follows

$$\begin{aligned}
c_{11}x_1 + c_{12}x_2 + \ldots + c_{1v}x_v &= b_1 \mod p, \\
c_{21}x_1 + c_{22}x_2 + \ldots + c_{2v}x_v &= b_2 \mod p, \\
&\vdots \\
c_{m1}x_1 + c_{m2}x_2 + \ldots + c_{mv}x_v &= b_m \mod p,
\end{aligned}$$

with $(c_{ij}) \in \mathcal{M}_{m,v}(\mathbb{Z}_p)$ and $(b_i) \in \mathbb{Z}_p^m$, a solution of the system can be found in polynomial time via Gaussian elimination. The LWE problem consists in solving a system of equations derived from the former one by adding small perturbations on the right hand side of the equations, this is, it consists in finding a solution $(x_1, \ldots, x_v) \in \mathbb{Z}_p^v$ of a linear system of the form

$$\begin{aligned}
c_{11}x_1 + c_{12}x_2 + \ldots + c_{1v}x_v &= b_1 + \epsilon_1 \mod p, \\
c_{21}x_1 + c_{22}x_2 + \ldots + c_{2v}x_v &= b_2 + \epsilon_2 \mod p, \\
&\vdots \\
c_{m1}x_1 + c_{m2}x_2 + \ldots + c_{mv}x_v &= b_m + \epsilon_m \mod p,
\end{aligned}$$

where $\epsilon = (\epsilon_1, \ldots, \epsilon_m) \in \mathbb{Z}_p^m$ is called the error and it is chosen so as to be small (± 1 according to [Regev, 2009]). It turns out that whereas solving a system of linear equations is easy, the LWE problem is hard, which is why it has attracted considerable attention and it has proved itself to be an interesting basis for cryptographic constructions (see for example [Kawachi et al., 2007, Peikert et al., 2008, Bos et al., 2015, Alkim et al., 2015]).

A naive approach to add noise to the hidden subspaces problem that resembles the way in which it is done in the LWE problem would be adding noise to the right-hand side of the equations as follows:

Input: $(\mathbf{p}, \mathbf{q}) = ((p_1, \ldots, p_m), (q_1, \ldots, q_m)) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ of degree $d \geq 3$, with $\mathbf{x} = (x_1, \ldots, x_n)$ and $n \leq m \leq 2n$.

Find: a subspace $A \subset \mathbb{F}^n$ of dimension $n/2$ such that

$$p_i(A) \approx 0 \text{ and } q_i(A^{\perp}) \approx 0, \quad \forall i \in \{1, \dots, m\}.$$

However, the polynomials $p_1, \dots, p_m, q_1, \dots, q_m$ have a very well-structured set of zeros (bear in mind that A and $A^{\perp}$ are subspaces) and this way of adding noise does not make sense. Elaborating a bit more, any polynomial vanishing on either A or $A^{\perp}$ should also vanish on the n-vector $(0, \dots, 0)$. Therefore, given the approximate equation $p_i\,(A) \approx 0$, the left-hand side would equal exactly zero if the polynomial p_i did not have an independent term and it would equal exactly one if p_i had an independent term.

The next idea that comes to mind is to add noise by somehow adding perturbations on the left-hand side of the equations instead, this is, by including in the system decoy polynomials that look like those vanishing on the subspaces A or $A^{\perp}$ but actually do not. It is not a very good idea to add polynomials that are chosen fully at random as roughly half of them would have an independent term and so they would be easily detected as fake ones. A convenient idea to avoid this issue is to add to the system polynomials that vanish on a structured set as well, this is, to add polynomials vanishing on subspaces other that A or $A^{\perp}$. This is what Aaronson and Christiano proposed to do in the noisy variant of their problem, which is formally defined as follows.

The Noisy Hidden Subspaces Problem (NHSP$_{|\mathbb{F}|}$ for short)
Input: $(\mathbf{p}, \mathbf{q}) = ((p_1, \dots, p_m), (q_1, \dots, q_m)) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ of degree $d \geq 3$, with $\mathbf{x} = (x_1, \dots, x_n)$, $m = \lceil \beta n \rceil$ where $\beta \geq 3/(1-2\varepsilon)^2$, and $0 < \varepsilon < 1/2$.
Find: a subspace $A \subset \mathbb{F}^n$ of dimension $n/2$ such that

$$p_i(A) = 0, \ \forall i \in I_{\mathbf{p}}, \quad \text{and} \quad q_j(A^{\perp}) = 0, \ \forall j \in I_{\mathbf{q}},$$

for some $I_{\mathbf{p}}, I_{\mathbf{q}} \subset \{1, \dots, m\}$ with $\# I_{\mathbf{p}} = \# I_{\mathbf{q}} = \lceil (1-\varepsilon)m \rceil$, and such that

$$p_i(A_i^{\mathbf{p}}) = 0, \ \forall i \in \{1, \dots, m\} \setminus I_{\mathbf{p}},$$

and

$$q_i(A_i^{\mathbf{q}}) = 0, \ \forall i \in \{1, \dots, m\} \setminus I_{\mathbf{q}},$$

where $A_i^{\mathbf{p}} \subset \mathbb{F}^n$ and $A_i^{\mathbf{q}} \subset \mathbb{F}^n$ are subspaces of dimension $n/2$ and

$$\begin{aligned} A_i^{\mathbf{p}} &\neq A, \quad \forall i \in \{1, \dots, m\} \setminus I_{\mathbf{p}}, \\ A_i^{\mathbf{q}} &\neq A^{\perp}, \quad \forall i \in \{1, \dots, m\} \setminus I_{\mathbf{q}}. \end{aligned}$$

Remark 2.2 *Note that in the* NHSP$_{|\mathbb{F}|}$ *there is no orthogonality relation between* $A_i^{\mathbf{p}}$ *and* $A_j^{\mathbf{q}}$ *for any* $i \in \{1, \dots, m\} \setminus I_{\mathbf{p}}, j \in \{1, \dots, m\} \setminus I_{\mathbf{q}}$.

Remark 2.3 *Note that* β *is chosen to be greater than or equal to* $3/(1-2\varepsilon)^2$ *because the authors prove that by doing so an instance* $(\mathbf{p}, \mathbf{q})$ *of the* NHSP$_{|\mathbb{F}|}$ *uniquely defines the subspace* A *with an overwhelming probability* [Aaronson and Christiano, 2013, Lemma 6.5].

Aaronson and Christiano claimed the hidden subspaces problem with and without noise to be quantum-resistant, so we state here their result for future reference.

Conjecture 2.4 [Aaronson and Christiano, 2013, Conjecture 6.7] *Given a degree-d instance of the* $\mathrm{HSP}_{|\mathbb{F}|}$ *(respectively the* $\mathrm{NHSP}_{|\mathbb{F}|}$*), no polynomial-time quantum algorithm can find a complete list of generators for* A *with success probability* $\Omega(2^{-n/2})$*.*

2.4 Aaronson-Christiano's public-key quantum money scheme

2.4.1 Definition of a general quantum money scheme

Before giving details about the scheme of Aaronson-Christiano, we formally define the general concept of a public-key quantum money scheme. We already said that one such scheme must satisfy three requirements, namely: there must be an efficient algorithm for the bank to prepare the quantum states, there must be an efficient and public algorithm for anyone to verify whether a certain banknote is valid or not, and no one other than the bank should be able to produce valid banknotes with a non-negligible probability. The cryptographic realisation of the concept of a public-key quantum money scheme is the following.

Definition 2.5 *A public-key quantum money scheme* $\mathcal{S}$ *consists of three polynomial-time algorithms:*

- ***KeyGen**, which takes as input a security parameter* λ *and probabilistically generates a key pair* $(k_{private}, k_{public})$*. The key* $k_{private}$ *is called the private key and it is kept secret by the authority issuing the money and* k_{public} *is called the public key and it is publicly accessible by anyone.*

- ***Bank**, which takes as input* $k_{private}$ *and probabilistically generates a quantum state \$ called a banknote. Usually, although depending on each public-key quantum money scheme, \$ is an ordered pair* (s, ρ_s) *consisting of a classical serial number* s *and a quantum state* ρ_s*.*

- ***Ver**, which takes as input* k_{public} *and an alleged banknote ¢, and either accepts it or rejects it.*

Remark 2.6 *In cryptography, the security parameter is a variable that measures the input size of the computational problem. The resource requirements of the cryptographic protocol as well as the adversary's probability of breaking its security are expressed in terms of the security parameter.*

An attacker who wants to produce a fake banknote can take two different approaches: he can either try to replicate an existing banknote created by **Bank** by preparing exactly the same quantum particles, or he can try to generate a banknote that is accepted by **Ver**. The first option is not feasible due to the laws of quantum mechanics, as it is impossible to clone a quantum state which is unknown. The second option could be carried out if the counterfeiter gained knowledge about $k_{private}$, as he would then be able to apply himself the algorithm **Bank**. However, the problem of inferring the private key from the public key is chosen to be computationally intractable and the security of the scheme relies on it, analogously to what happens in standard public-key cryptography.

The hidden subspaces problem with and without noise are the problems chosen by Aaronson-Christiano to underlie the noise-free and noisy versions of their scheme.

2.4.2 Instantiation of the $\mathrm{HSP}_{|\mathbb{F}|}$ and the $\mathrm{NHSP}_{|\mathbb{F}|}$. Key Generation in the Scheme of Aaronson-Christiano

Aaronson and Christiano define in their paper how the hidden subspaces problem with and without noise should be instantiated to generate key parameters for their scheme. Given an appropriate degree-d instance $(\mathbf{p}, \mathbf{q})$ of the $\mathrm{HSP}_{|\mathbb{F}|}$ (respectively the $\mathrm{NHSP}_{|\mathbb{F}|}$), the key pair of their scheme with and without noise is set to be:

$$\begin{aligned} k_{\text{private}} &= A, \\ k_{\text{public}} &= (\mathbf{p}, \mathbf{q})\,. \end{aligned}$$

First, we detail how Aaronson and Christiano decided to instantiate the $\mathrm{HSP}_{|\mathbb{F}|}$ to generate key pairs for their noise-free scheme. It is as follows: the $n/2$-dimensional subspace $A \subset \mathbb{F}^n$ is chosen uniformly at random, whereas for each $i \in \{1, \ldots, m\}$, the polynomial p_i is chosen uniformly at random from $I_{d,A}$.

Remark 2.7 *The choice of $A^\perp$ and the polynomials q_i, with $i \in \{1, \ldots, m\}$, is analogous.*

The instantiation of the $\mathrm{NHSP}_{|\mathbb{F}|}$ to generate key pairs for the noisy scheme is as follows. On the one hand, for every $i \in I_{\mathbf{p}}$ the polynomial p_i is chosen uniformly at random from $I_{d,A}$, while the $n/2$-dimensional subspace $A \subset \mathbb{F}^n$ is chosen uniformly at random too. On the other hand, for every $i \in \{1, \ldots, m\} \backslash I_{\mathbf{p}}$ the polynomial p_i is chosen uniformly at random from $I_{d,A_i^{\mathbf{p}}}$, where $A_i^{\mathbf{p}} \subset \mathbb{F}^n$ is an $n/2$-dimensional subspace different from A.

Remark 2.8 *The choice of $A^\perp$ and $A_i^{\mathbf{q}}$ ($i \in \{1, \ldots, m\} \backslash I_{\mathbf{q}}$), as well as of the polynomials q_i ($i \in \{1, \ldots, m\}$), is analogous.*

Remark 2.9 *Note that the subspaces of each of the sets*

$$\{A_i^{\mathbf{p}}\}_{i \in \{1,\ldots,m\} \backslash I_{\mathbf{p}}}, \quad \{A_i^{\mathbf{q}}\}_{i \in \{1,\ldots,m\} \backslash I_{\mathbf{q}}}$$

are pair-wise different with overwhelming probability.

Remark 2.10 *From now on, whenever we say that*

$$(\mathbf{p}, \mathbf{q}) = ((p_1, \ldots, p_m), (q_1, \ldots, q_m)) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$$

is a degree-d instance of the $\mathrm{HSP}_{|\mathbb{F}|}$ *or of the* $\mathrm{NHSP}_{|\mathbb{F}|}$*, we are referring to an instance satisfying these conditions.*

Now that the instantiation of both problems is set, we should focus on how to generate such instances, this is, on how to generate key pairs. The generation of the subspace A, as well as of the subspaces $A_i^{\mathbf{p}}$ if the $\mathrm{NHSP}_{|\mathbb{F}|}$ is being considered, is straightforward: it suffices to choose a full-rank matrix in $\mathcal{M}_{n/2,n}(\mathbb{F})$ uniformly at random.

Remark 2.11 *The generation of the subspace $A^{\perp}$ (and the subspaces $A_i^{\mathbf{q}}$ when the* $\mathrm{NHSP}_{|\mathbb{F}|}$ *is being considered) is analogous.*

The generation of uniformly random polynomials that vanish on a given subspace is slightly more complicated. The key result to produce uniformly random polynomials vanishing on a certain subspace is the following.

Lemma 2.12 [Aaronson and Christiano, 2013] *Denote by $e_i \in \mathbb{F}^n$ the vector whose i-th component equals* 1 *and whose other components equal* 0*, and by E the subspace generated by the vectors $e_1, \ldots, e_{n/2}$. It holds that:*

1. *A polynomial belongs to $I_{d,E}$ if and only if each of its monomials is divisible by an element in the set $\{x_{n/2+1}, \ldots, x_n\}$.*
2. *If L is an invertible linear transformation on $I_{d,A}$, the function*

$$p\left(\boldsymbol{x}\right) \rightarrow p\left(\boldsymbol{x}L\right)$$

maps $I_{d,A}$ to $I_{d,AL^{-1}}$.

Indeed, Lemma 2.12 allows to devise a way to generate an uniformly random polynomial vanishing on a certain subspace as follows.

Proposition 2.13 (Vanishing polynomial) *The generation of an uniformly random polynomial in $I_{d,A}$ consists of the following two steps:*

1. *Generate a polynomial $p' \in I_{d,E}$: as a consequence of* Lemma 2.12(1)*, this is done by including each monomial of degree d or lower independently and with probability* 1/2 *if it is divisible by an element in the set $\{x_{n/2+1}, \ldots, x_n\}$.*
2. *Transform $p' \in I_{d,E}$ into $p \in I_{d,A}$: considering the matrix L of change of basis (i.e., $E = AL$), the polynomial $p = p'\left(\mathbf{x}L\right)$ vanishes on A by* Lemma 2.12(2)*.*

The generation of a polynomial in $I_{d,A}$ takes polynomial time. The cost of the first step in Proposition 2.13 is of the order of the total number of monomials in $\mathbb{F}[x_1, \dots, x_n]$ of degree less than or equal to d, which is $\mathcal{O}(n^d)$. The cost of the second step in Proposition 2.13 is that of a matrix multiplication, which is $\mathcal{O}(n^\omega)$, where $\omega = 2$ since $\mathbf{x}L$ is a vector-matrix multiplication, plus that of evaluating a polynomial of degree d with $\mathcal{O}(n^d)$ terms, which yields a total complexity of $\mathcal{O}((d+1)n^d)$.

Remark 2.14 *Note that the size of the field influences the time that it takes to generate a polynomial in $I_{d,A}$, as multiplications over $\mathbb{F}$ are slower as $|\mathbb{F}|$ increases.*

2.4.3 Security of the scheme of Aaronson-Christiano

To precisely describe the scheme of Aaronson-Christiano it is necessary to have some further knowledge of quantum mechanics, which is out of the scope of this thesis. For our purposes, it is enough with knowing that the three polynomial-time algorithms that constitute the scheme of Aaronson-Christiano are essentially as follows.

- **KeyGen**(λ) generates a degree-d instance $(\mathbf{p}, \mathbf{q})$ of the HSP$_{|\mathbb{F}|}$ or the NHSP$_{|\mathbb{F}|}$ depending on the version of the scheme considered, outputting
$$(k_{private}, k_{public}) = (A, (\mathbf{p}, \mathbf{q}))\,.$$
- **Bank** (A) prepares a banknote from a classical description of A (this is, a list of $n/2$ elements that generate it).
- **Ver** $(\mathbf{p}, \mathbf{q})$ verifies whether a banknote is valid or not by querying the polynomials of $\mathbf{p}$ (respectively $\mathbf{q}$) to test membership in A (respectively $A^{\perp}$) and by carrying out certain quantum measurements.

As we have been saying from the beginning, the two versions of the scheme of Aaronson-Christiano are secure provided that the HSP$_{|\mathbb{F}|}$ and the NHSP$_{|\mathbb{F}|}$ are quantum-resistant. However, we have not formally defined yet the notion of a secure quantum money scheme. Indeed, the formal notion of a secure quantum money scheme is the following.

Definition 2.15 [Aaronson and Christiano, 2013, Definition 3.1] *(Security of a Quantum Money Scheme) We say that a public-key quantum money scheme $\mathcal{S} = (KeyGen, Bank, Ver)$ has completeness error ε if*

$$Ver(k_{public})$$

accepts a valid banknote \$ with probability at least $1 - \varepsilon$ for all public keys k_{public} and all valid banknotes \$. If $\varepsilon = 0$, then we say that the scheme $\mathcal{S}$ has perfect completeness.

Now let Count take as input k_{public} and a collection of alleged banknotes $\cent_1, \dots, \cent_r$ and output the number of indices $i \in \{1, \dots, r\}$ such that $Ver(k_{public})$ accepts the banknotes $\cent_1, \dots, \cent_r$. Let

$$C\left(k_{public}, \cent_1, \dots, \cent_q\right)$$

map $q = poly(n)$ valid banknotes $\$_1, \ldots, \$_q$ to $r = poly(n)$ alleged banknotes $¢_1, \ldots, ¢_r$ (note that q is polynomial in n). We say that $\mathcal{S}$ has soundness error δ if

$$Pr\left[Count\left(k_{public}, C\left(k_{public}, \$_1, \ldots, \$_q\right)\right) > q\right] \leq \delta,$$

where Pr denotes the probability of an event.

We say that a public-key quantum money scheme $\mathcal{S}$ is secure if it has completeness error less or equal than $1/3$ and negligible soundness error.

Indeed, according to this notion of security the authors claim that their scheme with noise and without noise is secure provided that the HSP$_{|\mathbb{F}|}$ and the NHSP$_{|\mathbb{F}|}$ are quantum-resistant, as the following result expresses.

Theorem 2.16 [Aaronson and Christiano, 2013, Theorem 6.9] **(Security Reduction)** *Assuming Conjecture 2.4, the public-key quantum money scheme of Aaronson and Christiano with and without noise has perfect completeness and soundness error $2^{-\Omega(n)}$.*

2.5 Gröbner bases: A tool for solving non-linear systems

Since the main building blocks of the HSP$_{|\mathbb{F}|}$ and the NHSP$_{|\mathbb{F}|}$ are sets of polynomials in several variables that are not linear, trying to solve algebraic systems with these properties will be a constant in this thesis. In this section we focus on the existing techniques to compute what is called a Gröbner basis, with an emphasis on the F_4 and the F_5 algorithms. A Gröbner basis is a mathematical tool that enables to find solutions for multivariate non-linear systems, aside from having other applications in computational algebraic geometry (as for example ideal membership testing [Cox et al., 2007, §2.8]). The purpose of this section is not to exhaustively explain the core of the algorithms, but instead to give some details about the way they operate and the complexity results they achieve so it is then possible to follow our results of Chapter 3 and Chapter 4. The theory of this section is mainly based on [Cox et al., 2007].

Let us first give some context. Solving linear systems is so easy that we learn how to do it in high school. It is well known that one can solve them in polynomial time via Gaussian elimination with a complexity of $\mathcal{O}(n^3)$ in the number of variables. The idea behind Gaussian elimination is simple: transform a given linear system into an equivalent one which happens to have better properties, in the sense that the latter turns out to be somehow easier to solve than the original one. For example, if we consider a linear system over $\mathbb{F}$ with a unique solution, say

$$\begin{array}{rcl} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1v}x_v & = & b_1, \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2v}x_v & = & b_2, \\ & \vdots & \\ a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mv}x_v & = & b_m, \end{array}$$

the process of Gaussian elimination produces a linear system over $\mathbb{F}$ of the form

$$\begin{array}{lcl} a'_{11}x_1 + a'_{12}x_2 + \ldots + a'_{1v}x_v & = & b'_1, \\ a'_{22}x_2 + \ldots + a'_{2v}x_v & = & b'_2, \\ & \vdots & \\ a'_{mv}x_v & = & b'_m, \end{array}$$

in such a way that it is now possible to solve the univariate equation $a'_{mv}x_v = b'_m$ and substitute backwards until a solution of the whole system is found.

This idea of finding an equivalent and better system of equations is also applied in the non-linear setting as the key idea to solve a system. At high level, this better set is called a Gröbner basis and the most efficient algorithm for its computation is essentially a generalisation of Gaussian elimination to higher degrees. Unfortunately, finding a Gröbner basis is not nearly as easy as Gaussian elimination is in the linear scenario.

Since finding a Gröbner basis allows to find a solution of non-linear systems, the complexity of computing such a basis is at least that of solving non-linear systems. In this sense, note that NP-complete problems, a class of problems that are widely considered to be hard and not efficiently solvable (see [Cook, 1971, Cook, 2000]), can all be modelled with non-linear systems of polynomials. See for example the knapsack problem, which was proved to be NP-complete in [Karp, 1972].

The Knapsack Problem
Input: $(b_1, \ldots, b_v, c) \in \mathbb{N}^{v+1}$
Find: a solution of the (overdetermined) system

$$\sum_{i=1}^{v} x_i b_i = c, \quad x_i\,(1 - x_i) = 0, \quad i = 1, \ldots, v.$$

The fact that NP-complete problems can be modelled with non-linear systems strongly suggests that the worst-case complexity of computing a Gröbner basis cannot be good. Indeed, it was proved in [Mayr and Meyer, 1982, Giusti and Lazard, 1983, Dubé, 1990] that it is at least doubly exponential in the number of variables. Nonetheless, worst-case is called worst-case for a reason, and the behaviour of computing a Gröbner basis for generic (as in random) systems is usually better.

In what follows we give the basic background that is necessary to introduce the concept of a Gröbner basis in the way that seems the most intuitive to us. Throughout Section 2.5.1 and Section 2.5.2, we set $\mathbf{x} = (x_1, \ldots, x_v)$.

2.5.1 Basic notions

We start by introducing the concept of an algebraic variety. Varieties are the geometric manifestations of the solutions of a system of polynomial equations and thus the main object of study of algebraic geometry. Recall first the concept of an ideal and a fundamental theorem of commutative algebra.

Definition 2.17 *A (polynomial) ideal $\mathcal{I} \subset \mathbb{F}[\mathbf{x}]$ is a non-empty $\mathbb{F}$-subspace that is closed under multiplication by elements of $\mathbb{F}[\mathbf{x}]$, this is,*

$$f\mathcal{I} = \{fp \mid p \in \mathcal{I}\} \subseteq \mathcal{I}, \quad p \in \mathbb{F}[\mathbf{x}].$$

Furthermore, given $m \in \mathbb{N}$ and a set of polynomials $p_1, \ldots, p_m \in \mathbb{F}[\mathbf{x}]$, the ideal generated by $p_1, \ldots, p_m$, denoted by $\langle p_1, \ldots, p_m \rangle$, is defined as

$$\langle p_1, \ldots, p_m \rangle = \left\{ \sum_{i \in \{1,\ldots,m\}} f_i p_i, \quad f_i \in \mathbb{F}[\mathbf{x}] \, \forall i \in \{1, \ldots, m\} \right\}.$$

The Hilbert basis theorem guarantees that every ideal in a finite field is finitely generated.

Theorem 2.18 (Hilbert basis theorem) *Let $\mathcal{I} \subset \mathbb{F}[\mathbf{x}]$ be an ideal. There exists $m \in \mathbb{N}$ and $p_1, \ldots, p_m \in \mathbb{F}[\mathbf{x}]$ such that*

$$\mathcal{I} = \langle p_1, \ldots, p_m \rangle.$$

Remark 2.19 *Note that* Theorem 2.18 *is a particularisation of the Hilbert basis theorem to finite fields, which is the setting we are working on throughout this thesis.*

The concept of a variety is the following.

Definition 2.20 *Let $\mathcal{I} = \langle p_1, \ldots, p_m \rangle$, with $p_1, \ldots, p_m \in \mathbb{F}[\mathbf{x}]$. The variety associated to $\mathcal{I}$ (or associated to $p_1, \ldots, p_m$), denoted by either $V(\mathcal{I})$ or $V(p_1, \ldots, p_m)$, is defined as follows:*

$$V(\mathcal{I}) = \left\{ (a_1, \ldots, a_v) \in \bar{\mathbb{F}}^v \colon p_i(a_1, \ldots, a_v) = 0, \quad \forall i \in \{1, \ldots, m\} \right\},$$

where $\bar{\mathbb{F}}$ denotes the algebraic closure of $\mathbb{F}$. Furthermore, we define:

$$V_{\mathbb{F}}(\mathcal{I}) = \{ (a_1, \ldots, a_v) \in \mathbb{F}^v \colon p_i(a_1, \ldots, a_v) = 0, \quad \forall i \in \{1, \ldots, m\} \}.$$

Let us introduce the notion of a zero-dimensional ideal.

Definition 2.21 *(Zero-dimensional ideal) Let $\mathcal{I} \subset \mathbb{F}[\mathbf{x}]$ be an ideal. The ideal $\mathcal{I}$ is said to be zero-dimensional if $V(\mathcal{I})$ is finite. In this case, the quotient*

$$\mathbb{F}[\mathbf{x}]/\mathcal{I}$$

is a vector space over $\mathbb{F}$ of finite dimension.

All throughout this thesis we are going to encounter ideals that are of this type. Indeed, we are going to be looking for solutions of systems over a finite field $\mathbb{F}$ of prime size, so the polynomials

$$x_1^{|\mathbb{F}|} - x_1, \ldots, x_v^{|\mathbb{F}|} - x_v$$

will be present in the ideals whose associated varieties we will be trying to find. As a consequence, the ideals on which we will focus are always zero-dimensional.

Prior to defining a Gröbner basis, we need to establish some way to order elements in $\mathbb{F}[\mathbf{x}]$. Let us first recall some preliminary concepts.

Definition 2.22 *A monomial is an element of* $\mathbb{F}[\mathbf{x}]$ *of the form*

$$x_1{}^{\alpha_1} \ldots x_v{}^{\alpha_v} \quad (\mathbf{x}^\alpha \textit{ for short}),$$

where $\alpha_1, \ldots, \alpha_v$ *are integers greater than or equal to zero. The degree of a monomial is denoted and defined as*

$$\deg(\mathbf{x}^\alpha) = \alpha_1 + \ldots + \alpha_v,$$

and a term is a monomial with a coefficient, that is, an element of the form

$$c\mathbf{x}^\alpha, \quad c \in \mathbb{F}.$$

Definition 2.23 *A monomial ordering* $\prec_{mon}$ *on* $\mathbb{F}[\mathbf{x}]$ *is a relation on* ${\mathbb{Z}_{\geq 0}}^v$ *(i.e., the monomial exponents) such that*

1. *The relation* $\prec_{mon}$ *is a total order, which means that for any* $\alpha, \beta \in {\mathbb{Z}_{\geq 0}}^v$*, exactly one of the following statements*

$$\alpha \prec_{mon} \beta, \quad \alpha =_{mon} \beta, \quad \beta \prec_{mon} \alpha$$

   holds.

2. *If* $\alpha \prec_{mon} \beta$ *and* $\gamma \in {\mathbb{Z}_{\geq 0}}^v$*, then* $\alpha + \gamma \prec_{mon} \beta + \gamma$*.*
3. *The relation* $\prec_{mon}$ *is a well ordering, this is, every non-empty subset has a smallest element.*

Given two monomials $\mathbf{x}^\alpha, \mathbf{x}^\beta \in \mathbb{F}[\mathbf{x}]$*, we say that* $\mathbf{x}^\alpha \prec_{mon} \mathbf{x}^\beta$ *if and only if* $\alpha \prec_{mon} \beta$*.*

Note that in the univariate case the only order satisfying these conditions is the degree, yielding the ordered sequence

$$1 \prec x \prec x^2 \prec x^3 \prec \ldots$$

In the multivariate case, however, there are many different ways to order monomials. We detail the two monomial orderings that are the most relevant ones regarding the computation of a Gröbner basis.

Definition 2.24 *We define the following orderings on* $\mathbb{F}[\mathbf{x}]$*.*

1. *The lexicographical ordering. For* $\alpha, \beta \in {\mathbb{Z}_{\geq 0}}^v$*, we say that* $\alpha \prec_{lex} \beta$ *(and* $\mathbf{x}^\alpha \prec_{lex} \mathbf{x}^\beta$*) if and only if the leftmost non-zero entry in the vector difference* $\beta - \alpha$ *is positive.*

2. *The graded reverse lexicographical ordering. For $\alpha, \beta \in {\mathbb{Z}_{\geq 0}}^v$, we say $\alpha \prec_{grevlex} \beta$ (and $\mathbf{x}^\alpha \prec_{grevlex} \mathbf{x}^\beta$) if and only if $\sum \beta_i = \deg(x^\beta) < \sum \alpha_i = \deg(x^\alpha)$ or $\deg(x^\alpha) = \deg(x^\beta)$ and the rightmost non-zero entry in the vector difference $\beta - \alpha$ is negative.*

Example 2.25 *Consider the polynomial ring $\mathbb{F}[x_1, x_2]$. The lexicographic ordering yields the ordered sequence*

$$1 \prec x_2 \prec {x_2}^2 \prec \ldots \prec x_1 \prec x_1x_2 \prec x_1{x_2}^2 \prec \ldots \prec {x_1}^2 \prec {x_1}^2x_2 \prec {x_1}^2{x_2}^2 \ldots,$$

while the graded reverse lexicographic yields the ordered sequence

$$1 \prec x_2 \prec x_1 \prec {x_2}^2 \prec x_1x_2 \prec {x_1}^2 \prec {x_2}^3 \prec x_1{x_2}^2 \prec {x_1}^2x_2 \prec {x_1}^2 \prec \ldots.$$

Once a monomial ordering is fixed the following notions, which will be useful later on, can be defined.

Definition 2.26 *Let p be a polynomial in $\mathbb{F}[\mathbf{x}]$ and let $\prec_{mon}$ be a monomial ordering on $\mathbb{F}[\mathbf{x}]$. Then,*

- *The leading monomial of p with respect to $\prec_{mon}$, denoted by $LM_{\prec_{mon}}(p)$, is the maximal monomial of p with a non-zero coefficient.*
- *The leading coefficient of p with respect to $\prec_{mon}$, denoted by $LC_{\prec_{mon}}(p)$, is the coefficient associated to the leading monomial of p.*
- *The leading term of p with respect to $\prec_{mon}$, denoted by $LT_{\prec_{mon}}(p)$, is the term $LC_{\prec_{mon}}(p)\, LM_{\prec_{mon}}(p)$.*

The last step before introducing the concept of a Gröbner basis is to define a concept called reduction, which is a generalisation of the Euclidean division. It is defined as follows.

Definition 2.27 *Given a set of polynomials $G = \{p_1, \ldots, p_m\} \subset \mathbb{F}[\mathbf{x}]$ and a polynomial $p \in \mathbb{F}[\mathbf{x}]$, we say that p reduces to r by G, and we write $p \to_G r$, if p can be written as*

$$p = c_1p_1 + \ldots + c_mp_m + r, \quad c_i \in \mathbb{F}[\mathbf{x}], \forall i \in \{1, \ldots, m\}, \quad r \in \mathbb{F}[\mathbf{x}],$$

and the leading monomial of r is not divisible by the leading monomials of $p_1, \ldots, p_m$.

After this preliminary background, we are finally in a position to give a definition of a Gröbner basis.

Definition 2.28 *Let $\mathcal{I} \subset \mathbb{F}[\mathbf{x}]$ be an ideal and $\prec_{mon}$ a monomial ordering. A finite set $G = \{g_1, \ldots, g_s\} \subset \mathcal{I}$ is a Gröbner basis of the ideal $\mathcal{I}$ with respect to $\prec_{mon}$ if*

$$\forall g \in \mathcal{I}, \quad g \to_G 0.$$

Furthermore, a Gröbner basis is said to be reduced if for all $i = 1, \ldots, s$ it occurs that

1. $LC(g_i) = 1$.

2. $LM(g_i)$ *does not divide any term of any* g_j *for any* $j \neq i$, $j \in \{1, \ldots, s\}$.

Note that the notion of a reduced Gröbner basis is introduced to guarantee uniqueness, as the following theorem ensures.

Theorem 2.29 *Let* $\mathcal{I} \subset \mathbb{F}[\mathbf{x}]$ *be an ideal and let* $\prec_{mon}$ *be a monomial ordering on* $\mathbb{F}[\mathbf{x}]$. *Then,* $\mathcal{I}$ *admits a unique reduced Gröbner basis.*

We gave this definition of a Gröbner basis because we thought that it is a natural way to introduce the concept. In fact, this is the characterisation used in the first algorithm that computes a Gröbner basis [Buchberger, 1965] as we will see later. However, there are many equivalent characterisations. One of them, which is usually taken as the classic definition of a Gröbner basis, is stated in the following proposition.

Proposition 2.30 *Let* G *be a Gröbner basis of the ideal* $\mathcal{I} \subset \mathbb{F}[\mathbf{x}]$. *Then it occurs that*

$$\forall p \in \mathcal{I}, \quad \exists g \in G \, \textit{such that} \, LM(f) \textit{ is divisible by} \, LM(g) \, .$$

In what follows, we elaborate on how to compute a Gröbner basis of an ideal and why computing a Gröbner basis of an ideal suffices to solve the corresponding non-linear system.

2.5.2 Algorithms to compute a Gröbner basis

In [Buchberger, 1965, Buchberger, 1970], Bruno Buchberger introduced the notion of a Gröbner basis of an ideal and proposed the first algorithm to compute it. For that, he devised a characterisation of a Gröbner basis known as the Buchberger's criterion, which relies on a special kind of polynomials called S-polynomials. The definition of an S-polynomial is the following.

Definition 2.31 *Let* $g, h \in \mathbb{F}[\mathbf{x}]$ *be two non-zero polynomials. The S-polynomial of* g *and* h*, denoted by* $S(g, h)$*, is*

$$S(g, h) = \frac{L}{LT(g)} h - \frac{L}{LT(h)} g,$$

where L *is the least common multiple of* $LM(g)$ *and* $LM(h)$.

Buchberger characterised a Gröbner basis in terms of a property that the S-polynomials of all pairs of polynomials in the Gröbner basis satisfy, which is stated in the next theorem.

Theorem 2.32 (Buchberger's criterion) *Let $G \subset \mathbb{F}[\mathbf{x}]$ be a set of non-zero polynomials. Then, G is a Gröbner basis if and only if*

$$\forall g, h \in G,\ g \neq h, \quad S(g,h) \to_G 0.$$

Buchberger's constructive algorithm to compute a Gröbner basis of an ideal builds on the criterion of Theorem 2.32, and it basically consists in successively adding to the initial generating set of the ideal the S-polynomials of pairs of elements that do not reduce to zero. In particular, the algorithm is as follows.

Buchberger's algorithm

```
Input: $F = \{f_1, \dots, f_m\} \subset \mathbb{F}[\mathbf{x}]$ and $\prec_{\text{mon}}$ a monomial ordering on $\mathbb{F}[\mathbf{x}]$
$G = F$
$S = \{\{p,q\} : p,q \in G, p \neq q\}$
while $S \neq \emptyset$ do
    Select $\{p,q\} \in S$
    $S := S \setminus \{\{p,q\}\}$
    Compute $h$ such that $S(p,q) \to_G h$
    if $h \neq 0$ then
        $S := S \cup \{\{g,h\} : g \in G\}$
        $G := G \cup \{h\}$
    end if
end while
return $G$
```

Buchberger proved that his algorithm terminates and that it indeed outputs a Gröbner basis.

Remark 2.33 *Buchberger's algorithm as written above does not return a unique Gröbner basis. Note that it can be forced to do so by checking some extra conditions to guarantee that the Gröbner basis is reduced.*

However, a disadvantage of Buchberger's algorithm is that it carries out many useless operations, in the sense that it computes many reductions of S-polynomials that will be zero. Note that those S-polynomials that reduce to zero make no contribution to the final Gröbner basis, so computing their reductions only slows down the algorithm. Later, Buchberger's algorithm was improved in several ways [Kollreider and Buchberger, 1978, Buchberger, 1979] and some results on its complexity were achieved [Buchberger, 1983].

A different approach was taken by Lazard [Lazard, 1983] in 1983 , when he noticed that successively performing Gaussian elimination without any row or column swapping over a certain Macaulay matrix [Macaulay, 1994] was equivalent to performing Buchberger's algorithm. A Macaulay matrix in a certain degree of a given set of polynomials in $\mathbb{F}[\mathbf{x}]$ is essentially a matrix whose rows can be interpreted as the multiples of those polynomials by all monomials up to a given degree. Let us introduce the concept formally.

Definition 2.34 *Let $p_1, \ldots, p_m \in \mathbb{F}[\mathbf{x}]$ be polynomials with $d_i = \deg(p_i)$, and let $\prec$ be a monomial ordering on $\mathbb{F}[\mathbf{x}]$. Denote by*

$$\mathrm{M}_d^{\prec}(\mathbb{F}[\mathbf{x}])$$

the ordered set whose elements are those from the set $\mathrm{M}_d(\mathbb{F}[\mathbf{x}])$ ordered decreasingly with respect to $\prec$. Denote its cardinality by $N_{d,\mathbf{x},\prec}^{|\mathbb{F}|}$.

For the polynomials $p_1, \ldots, p_m$, the Macaulay matrix in degree d for $\prec$, denoted by $Mac_{d,\prec}(p_1, \ldots, p_m)$, is a matrix with

$$N_{d-d_1,\mathbf{x},\prec}^{|\mathbb{F}|} + \ldots + N_{d-d_m,\mathbf{x},\prec}^{|\mathbb{F}|}$$

rows, $N_{d,\mathbf{x},\prec}^{|\mathbb{F}|}$ columns and entries in $\mathbb{F}$ in such a way that:

- *The rows of $Mac_{d,\prec}(p_1, \ldots, p_m)$ are indexed by the polynomials*

  $$m_j p_i, \quad \text{where } m_j \in \mathrm{M}_{d-d_i}^{\prec}(\mathbb{F}[\mathbf{x}]), \quad j \in N_{d-d_i,\mathbf{x},\prec}^{|\mathbb{F}|}, \quad i \in \{1, \ldots, m\}.$$

- *The columns of $Mac_{d,\prec}(p_1, \ldots, p_m)$ are indexed by the monomials in $\mathrm{M}_d^{\prec}(\mathbb{F}[\mathbf{x}])$.*
- *The entry of $Mac_{d,\prec}(p_1, \ldots, p_m)$ corresponding to the row indexed by $m_j p_i$, with $m_j \in \mathrm{M}_{d-d_i}^{\prec}(\mathbb{F}[\mathbf{x}])$ for some $j \in N_{d-d_i,\mathbf{x},\prec}^{|\mathbb{F}|}$ and some $i \in \{1, \ldots, m\}$, and to the column indexed by $m_k \in \mathrm{M}_d^{\prec}(\mathbb{F}[\mathbf{x}])$, is the coefficient of the monomial m_k in the polynomial $m_j p_i$.*

More precisely, if

$$p_i = \sum_{\alpha} c_\alpha \mathbf{x}^\alpha, \text{ and } m_j = \mathbf{x}^\beta \in \mathcal{M}_{d-d_i}^{\prec}(\mathbb{F}[\mathbf{x}]) \text{ for some } \beta \in \mathbb{N}^v,$$

then the entry of $Mac_{d,\prec}(p_1, \ldots, p_m)$ in the row indexed by $\mathbf{x}^\beta p_i$ and in the column indexed by $m_k = \mathbf{x}^{\alpha+\beta}$ is c_α, this is,

$$\begin{array}{c} \\ \cdots \\ \mathbf{x}^\beta p_i \\ \cdots \end{array} \begin{array}{c} \begin{array}{ccc} & \mathbf{x}^{\alpha+\beta} & \end{array} \\ \begin{pmatrix} \cdots & \cdots & \cdots \\ \cdots & c_\alpha & \cdots \\ \cdots & \cdots & \cdots \end{pmatrix} \end{array}.$$

To clarify the notion of a Macaulay matrix see the following example.

Example 2.35 *Consider the polynomials*

$$f_1 = 1 + x_2^2 + 2x_1 + x_1 x_2^2, \quad f_2 = 4x_1 x_2^2 + 3x_1^2 + x_1^2 x_2$$

in $\mathbb{F}_5[x_1, x_2]$. Under the lexicographic order $\prec$, the Macaulay matrix in degree 3, $Mac_{3,\prec}(f_1, f_2)$, equals

$$\begin{array}{c|cccccccccc} & x_1^3 & x_1^2x_2 & x_1^2 & x_1x_2^2 & x_1x_2 & x_1 & x_2^3 & x_2^2 & x_2 & 1 \\ \hline f_1 & 0 & 0 & 0 & 1 & 0 & 2 & 0 & 1 & 0 & 1 \\ f_2 & 0 & 1 & 3 & 4 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}$$

Since the Macaulay matrix constructed, $Mac_{3,\prec}(f_1, f_2)$, is in the same degree as that of the polynomials f_1, f_2, the matrix only represents the polynomials f_1, f_2.

The Macaulay matrix in degree four would represent the multiplications of f_1, f_2 by monomials of degree one in $\mathbb{F}[x_1, x_2]$, and so on for subsequent degrees.

Indeed, given an ideal $\mathcal{I} \subset \mathbb{F}^n$, Lazard proved [Lazard, 1983] that there exists a certain degree $D \in \mathbb{N}$ such that Gaussian elimination (with no row or column swapping) performed on the Macaulay matrix of the generating set of $\mathcal{I}$ in degree D yields a Gröbner basis of $\mathcal{I}$. This is stated in the following result.

Theorem 2.36 [Lazard, 1983, Lazard's theorem] *Let $\mathcal{I} = \langle p_1, \ldots, p_m \rangle$ be an ideal of $\mathbb{F}[\mathbf{x}]$. There exists a degree $D \in \mathbb{N}$ such that the rows of the row echelon form (with Gaussian elimination performed with no row or column swapping) of the Macaulay matrix*

$$Mac_{D,\prec}(p_1, \ldots, p_m)$$

represent a Gröbner basis of the ideal $\mathcal{I}$.

Lazard's theorem ensures then that by successively performing Gaussian elimination on Macaulay matrices in increasing degree a Gröbner basis is eventually contained in a Macaulay matrix. Note that the Macaulay matrix that contains the Gröbner basis is in a degree that equals at least that of the generating polynomials of $\mathcal{I}$ of highest degree.

Remark 2.37 *Note that testing whether a certain Macaulay matrix contains a Gröbner basis can be done efficiently using any of the characterisations of a Gröbner basis (see for example* Proposition 2.30*).*

Several algorithms that build on this idea of successively computing row echelon forms of Macaulay matrices in increasing degrees have been proposed. Among these, we focus on the algorithms F_4 [Faugère, 1999] and F_5 [Faugère, 2002], both introduced by Jean-Charles Faugère with the latter being one of the fastest known algorithms to compute Gröbner bases. The F_4 algorithm presents an analogous issue to Buchberger's, in the sense that many linear combinations of rows of the Macaulay matrices that will end up being zero rows are computed anyway, yielding very sparse matrices. In [Faugère, 2002] Faugère gave a criterion to detect some linear combinations of rows that will lead to zero and avoid its computation, which gave rise to the F_5 algorithm.

Finally, and as we have been saying from the beginning, computing the reduced Gröbner basis of an ideal suffices to solve the associated system of equations. This is guaranteed by the following theorem, which gives the power of variable elimination to Gröbner bases, analogously to what happens with Gaussian elimination in the linear case.

Theorem 2.38 (Elimination property of Gröbner bases) *Let G be a Gröbner basis of an ideal $\mathcal{I} \subset \mathbb{F}[\mathbf{x}]$ with respect to the lexicographic ordering, with $x_1 \prec \ldots \prec x_v$. Then, for $i \in \{1, \ldots, m\}$, it holds that*

$$\mathcal{I} \cap \mathbb{F}[x_1, \ldots, x_i] = \langle G \cap \mathbb{F}[x_1, \ldots, x_i] \rangle,$$

where the ideal on the right hand side is generated over $\mathbb{F}[\mathbf{x}]$.

A consequence of Theorem 2.38 is that a Gröbner basis G of an ideal $\mathcal{I} \subset \mathbb{F}[\mathbf{x}]$ with respect to the lexicographic ordering is of the form

$$G = \begin{cases} g_{1,1}(x_1, \ldots, x_v), \\ \quad \vdots \\ g_{1,\ell_1}(x_1, \ldots, x_v), \\ g_{2,1}(x_2, \ldots, x_v), \\ \quad \vdots \\ g_{2,\ell_2}(x_2, \ldots, x_v), \\ \quad \vdots \\ g_{v-1,\ell_{v-1}}(x_{v-1}, x_v), \\ \quad \vdots \\ g_v(x_v). \end{cases}$$

for some $\ell_1, \ldots, \ell_{v-1} \in \mathbb{N}$. In particular, the polynomial g_v is univariate and so the equation $g_v(x_v) = 0$ can be solved in polynomial time. Substituting the solutions of the former equation in the equations of the form $g_{v-1,j} = 0$, with $j \in \{1, \ldots, \ell_{v-1}\}$, yield univariate equations, which can again be solved in polynomial time. Successively substituting backwards, the solutions of the system of equations associated to the ideal is found in polynomial time.

Let us briefly mention that this nice property of elimination only holds for a Gröbner basis with respect to the lexicographic ordering. However, the lexicographic ordering is usually less efficient than the graded reverse lexicographic ordering regarding the computation of a Gröbner basis. We do not extend on this, let us only say that there are efficient algorithms to transform a Gröbner basis with respect to one ordering into a Gröbner basis with respect to another ordering [Faugère et al., 1993, Faugère et al., 2014].

2.5.3 Complexity asymptotics

According to Theorem 2.36, the complexity of computing a Gröbner basis coincides with that of performing Gaussian elimination over a Macaulay matrix of a certain degree $D \in \mathbb{N}$ that is known to exist. As a consequence, this degree D at which the Gröbner basis is found is crucial to the overall efficiency of the algorithms that follow this approach. The concept of degree of regularity is introduced to formalise this idea as follows.

Definition 2.39 *Let $p_1, \ldots, p_m \in \mathbb{F}[\mathbf{x}]$ be homogeneous polynomials and let $\mathcal{I} = \langle p_1, \ldots, p_m \rangle \subset \mathbb{F}[\mathbf{x}]$ be an ideal. We define the degree of regularity of $\mathcal{I}$, denoted by $d_{reg}(\mathcal{I})$, as*

$$d_{reg}(\mathcal{I}) = min\left\{ d \in \mathbb{N} \colon dim_{\mathbb{F}}\left(p \in \mathbb{F}[\mathbf{x}] \text{ with } \deg(p) = d\right) = \binom{v+d-1}{d} \right\}.$$

If $p_1, \ldots, p_m \in \mathbb{F}[\mathbf{x}]$ are not homogeneous, then denoting by $\widetilde{p_i}$ the homogeneous part of highest degree of p_i for each $1 \leq i \leq m$, we define:

$$d_{reg}(\langle p_1, \ldots, p_m \rangle) = d_{reg}(\langle \widetilde{p_1}, \ldots, \widetilde{p_m} \rangle).$$

The degree of regularity bounds the maximal degree reached in the successive computations of the echelon form of the Macaulay matrices, so the efficiency of computing a Gröbner basis that the algorithms based on Theorem 2.36 display heavily depends on it. In particular, it has been proved that the complexity of computing a Gröbner basis with the F_5 algorithm depends on the degree of regularity as follows.

Proposition 2.40 [Bardet et al., 2015] *Let $\mathcal{I} = \langle p_1, \ldots, p_m \rangle \subset \mathbb{F}[\mathbf{x}]$ be an homogeneous zero-dimensional ideal and let d_{reg} be its degree of regularity. The complexity of computing a Gröbner basis for the graded reverse lexicographic ordering with the F_5 algorithm is, in field operations, bounded when d_{reg} grows to infinity by*

$$\mathcal{O}\left(m d_{reg} \left(\binom{v + d_{reg} - 1}{d_{reg}}^{\omega} \right) \right),$$

which is very roughly $\mathcal{O}\left(m v^{\omega d_{reg}}\right)$.

Determining the degree of regularity of a given system is not easy in general, as it depends on the inner structure of the ideal. However, there is a certain class of square systems (this is, those with as many equations as unknowns) whose degree of regularity is well understood, as well as the behaviour of computing a Gröbner basis of their associated ideals [Lazard, 1983, Giusti, 1984]. This class of systems are those associated to an ideal whose generators form what is called a *regular* sequence. The notion of regularity is the following.

Definition 2.41 *An homogeneous sequence of polynomials $(p_1, \ldots, p_m)$ is regular if for all $i \in \{1, \ldots, m\}$ and $g \in \mathbb{F}[\mathbf{x}]$ such that $g p_i \in \langle f_1, \ldots, f_{i-1} \rangle$, then g is also in $\langle f_1, \ldots, f_{i-1} \rangle$.*

A non-homogeneous sequence of polynomials $(p_1, \ldots, p_m)$ is regular if the homogeneous sequence $(\widetilde{p_1}, \ldots, \widetilde{p_m})$ is regular, where $\widetilde{p_i}$ denotes the homogeneous component of highest degree of p_i, $i \in \{1, \ldots, m\}$.

A particularly interesting property of homogeneous regular systems is that the F_5 algorithm does not perform any useless reductions to zero when performing the F_5 algorithm on the associated ideals. The smooth behaviour of the F_5 algorithm with respect to ideals generated by regular systems is ensured by the following result.

Theorem 2.42 [Faugère, 2002] *The homogeneous sequence $(p_1, \ldots, p_m)$ is regular if and only if there are no reductions to* 0 *in the F_5 algorithm.*

For non-homogeneous regular systems, a degree fall is said to occur during the computation of a Gröbner basis with the algorithm F_5 if a linear combination of rows of a Macaulay matrix produces a row that represents a polynomial of a degree that is strictly lower than those that were combined. Degree falls are analogous to reductions to zero in the homogeneous regular case and they should not occur in general during the computation of a Gröbner basis with the F_5 algorithm of an ideal generated by a non-homogeneous regular sequence [Faugère et al., 2013, Bardet, 2004].

The notion of regularity applies only to square systems. However, throughout this thesis we will encounter systems that have more equations than unknowns (i.e., that are overdefined). This is why it is important to mention that the notion of regularity was extended to overdefined systems in [Bardet, 2004], giving rise to the next notion of semi-regularity.

Definition 2.43 *An homogeneous sequence of polynomials $(p_1, \ldots, p_m)$ is semi-regular if for all $i \in \{1, \ldots, m\}$ and g such that*

$$gp_i \in \langle f_1, \ldots, f_{i-1} \rangle \ \text{ and } \ \deg(gp_i) < d_{reg},$$

then g is also in $\langle f_1, \ldots, f_{i-1} \rangle$.

A non-homogeneous sequence of polynomials $(p_1, \ldots, p_m)$ is semi-regular if the homogeneous sequence $(\widetilde{p_1}, \ldots, \widetilde{p_m})$ is semi-regular, where $\widetilde{p_i}$ denotes the homogeneous component of highest degree of p_i.

Analogously to what happened for regular systems, the behaviour of computing a Gröbner basis of semi-regular systems is also well-understood. Analogously to what occurs in the regular case, when performing the F_5 algorithm on semi-regular systems there are no unnecessary reductions to zero or degree falls occurring. Besides, the degree of regularity can be characterised for semi-regular systems as we detail in this proposition.

Proposition 2.44 *Let $(p_1, \ldots, p_m) \in \mathbb{F}[\mathbf{x}]$ be a semi-regular sequence and let $d_i = \deg(p_i)$ for $1 \leq i \leq m$. The degree regularity of $\langle p_1, \ldots, p_m \rangle$ is the index i of the first non-positive coefficient of the series*

$$\frac{\prod_{i=1}^{m} \left(1 - z^{d_i}\right)}{(1-z)^{v}} = \sum_{i \geq 0} c_i z^i.$$

Chapter 3

Cryptanalysis of Aaronson and Christiano's scheme: The noise-free case

This chapter is dedicated to analyse the hardness of the so-called hidden subspaces problem, proposed by Scott Aaronson and Paul Christiano at STOC$'$12.

Using techniques of polynomial modelling and algebraic cryptanalysis, we design a randomised polynomial-time algorithm that solves degree-d instances of the hidden subspaces problem over fields $\mathbb{F}$ of prime size that satisfy the condition $|\mathbb{F}| > d$. Over the field $\mathbb{F}_2$, we present a heuristic randomised polynomial-time algorithm that solves degree-d instances of the hidden subspaces problem with no constraints on the degree. We report experimental results on the performance of both algorithms that confirm that they are efficient in practice.

Since the security of the noise-free version of the public key quantum money scheme of Aaronson-Christiano relies on the hardness of the hidden subspaces problem over $\mathbb{F}_2$, our results yield a cryptanalysis of their scheme and of any of its extensions to a field $\mathbb{F}$ satisfying the condition $|\mathbb{F}| > d$.

3.1 Modelling the $\mathrm{HSP}_{|\mathbb{F}|}$

The process of modelling a problem essentially consists of transitioning from a concrete problem to an abstract system of polynomial equations that represents it, in the sense that finding a solution of the concrete problem is equivalent to finding a solution of the polynomial system that models it.

Along this chapter we focus on analysing the hardness of the $\mathrm{HSP}_{|\mathbb{F}|}$ and the first step towards that aim is finding a convenient model for it. The relevance of the model is not minor: the more optimal it is, the bigger the chances are of obtaining favourable results from its analysis. In light of the complexity performance exhibited by the algorithms to compute Gröbner bases that we described in Section 2.28, it is convenient to construct a model for the $\mathrm{HSP}_{|\mathbb{F}|}$ taking into account the following observations.

- The number of variables affects the complexity of a Gröbner basis computation (the more the variables, the higher the complexity, see Proposition 2.40), so we are interested in a model having as few variables as possible.
- It is better if the model has a unique solution. In general, finding an infinite number of solutions for a problem that models the private key recovery in a public-key system indicates that the model is not optimal and can be refined.

In the first part of this section we describe a straightforward model, which we refer to as the naive one, and observe that it has two main drawbacks: first, the system of equations has multiple solutions and second, it does not actively exploit the relation of orthogonality between the subspaces A and $A^{\perp}$. These flaws make the naive model unappealing for our purposes, but it sets the ground to design an optimised model, which is what we do in the second part of this section.

3.1.1 The naive model

A subspace is a particularly well-structured set that is determined by any basis of it, as every element of a subspace can be expressed as a linear combination of the elements of such a basis. Therefore, the first thing that comes to mind in order to recover a subspace that has been encoded as the zeros of a certain set of polynomials is to recover a basis of it.

Therefore, given a degree-d instance $(\mathbf{p},\mathbf{q}) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ of the $\mathrm{HSP}_{|\mathbb{F}|}$, the straightforward idea to recover the subspaces $A \subset \mathbb{F}^n$ and $A^{\perp} \subset \mathbb{F}^n$ of dimension $n/2$ is to define two matrices of unknowns that represent a basis of A and $A^{\perp}$ respectively, and go from there. By expressing the elements of A and $A^{\perp}$ as linear combinations of the elements in the corresponding basis, and by forcing the polynomials of the m-tuple $\mathbf{p} \in \mathbb{F}[\mathbf{x}]^m$ to vanish on all the elements of A and the polynomials of the m-tuple $\mathbf{q} \in \mathbb{F}[\mathbf{x}]^m$ to vanish on all the elements of $A^{\perp}$, we obtain a first and naive approach to construct a system of equations that models the $\mathrm{HSP}_{|\mathbb{F}|}$.

To make it more visual we write some equations of the model constructed as explained above, derived from a toy example of a degree-2 instance of the HSP_2, say $(\mathbf{p},\mathbf{q}) \in \mathbb{F}_2[\mathbf{x}]^4 \times \mathbb{F}_2[\mathbf{x}]^4$, where $\mathbf{x} = (x_1,\ldots,x_4)$ and

$$\begin{aligned} p_1 &= x_1x_2 + x_2x_4 + x_2,\\ q_1 &= x_1x_2 + x_1 + x_2x_3 + x_2x_4 + x_3 + x_4. \end{aligned}$$

Remark 3.1 *Note that the polynomials are chosen to be quadratic only to make the example short, since the degree of the polynomials in the* $\mathrm{HSP}_{|\mathbb{F}|}$ *is at least three.*

To construct the naive model we define a matrix of unknowns in $\mathcal{M}_{2,4}(\mathbb{F}_2)$ that represents a basis of the 2-dimensional subspace $A \subset \mathbb{F}_2^4$. We denote the matrix of unknowns by

$$G = \begin{pmatrix} g_{1,1} & g_{1,2} & g_{1,3} & g_{1,4}\\ g_{2,1} & g_{2,2} & g_{2,3} & g_{2,4} \end{pmatrix}$$

and we set (y_1, y_2) to be formal variables over $\mathbb{F}_2$. This way, a generic element of A can be expressed as $(y_1, y_2)\,G$ and so we can force the evaluation of p_1 on $(y_1, y_2)\,G$ to be zero, this is,

$$\begin{aligned} p_1\left((y_1,y_2)\,G\right) =& \left(g_{1,1}g_{2,2} + g_{1,2}g_{2,1} + g_{1,2}g_{2,4} + g_{1,4}g_{2,2}\right) y_1y_2\\ &+ \left(g_{1,1}g_{1,2} + g_{1,2}g_{1,4} + g_{1,2}\right) y_1\\ &+ \left(g_{2,1}g_{2,2} + g_{2,2}g_{2,4} + g_{2,2}\right) y_2\\ =&0 \end{aligned}$$

Now, since a degree-2 polynomial over $\mathbb{F}_2[y_1, y_2]$ is identically zero if and only if each of its coefficients is zero, we finally obtain the following equations in the unknown elements of the basis matrix:

$$\begin{aligned} g_{1,1}g_{2,2} + g_{1,2}g_{2,1} + g_{1,2}g_{2,4} + g_{1,4}g_{2,2} &= 0,\\ g_{1,1}g_{1,2} + g_{1,2}g_{1,4} + g_{1,2} &= 0,\\ g_{2,1}g_{2,2} + g_{2,2}g_{2,4} + g_{2,2} &= 0. \end{aligned}$$

Similarly, we define a matrix of unknowns in $\mathcal{M}_{2,4}(\mathbb{F}_2)$ as the basis of the 2-dimensional subspace $A^\perp \subset \mathbb{F}_2^4$, which we denote by

$$G^\perp = \begin{pmatrix} g^\perp_{1,1} & g^\perp_{1,2} & g^\perp_{1,3} & g^\perp_{1,4}\\ g^\perp_{2,1} & g^\perp_{2,2} & g^\perp_{2,3} & g^\perp_{2,4} \end{pmatrix}$$

and proceed as before. Only for completeness, the equations obtained from the polynomial q_1 in the unknowns $g^\perp_{1,1},\ldots,g^\perp_{2,4}$ are

$$\begin{aligned} g^\perp_{1,1}g^\perp_{1,2} + g^\perp_{1,1} + g^\perp_{1,2}g^\perp_{1,3} + g^\perp_{1,2}g^\perp_{1,4} + g^\perp_{1,3} + g^\perp_{1,4} &= 0,\\ g^\perp_{1,1}g^\perp_{2,2} + g^\perp_{1,2}g^\perp_{2,1} + g^\perp_{1,2}g^\perp_{2,3} + g^\perp_{1,2}g^\perp_{1,4} + g^\perp_{1,3}g^\perp_{2,2} + g^\perp_{1,4}g^\perp_{2,2} &= 0,\\ g^\perp_{2,1}g^\perp_{2,2} + g^\perp_{2,1} + g^\perp_{2,2}g^\perp_{2,3} + g^\perp_{2,2}g^\perp_{2,4} + g^\perp_{2,3} + g^\perp_{2,4} &= 0. \end{aligned}$$

Remark 3.2 *To obtain all the equations of the naive model, we would need to include the equations derived analogously from the polynomials $p_2, p_3, p_4, q_2, q_3, q_4$, whichever they are.*

After exemplifying how to construct the naive model we are ready to formally define it, which we do in the following proposition.

Proposition 3.3 (The Naive Model) *Let $(\mathbf{p}, \mathbf{q}) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ be a degree-d instance of the $\mathrm{HSP}_{|\mathbb{F}|}$, where $\mathbf{x} = (x_1, \ldots, x_n)$. Let the following matrices*

$$G = \left(g_{i,j}\right)_{\substack{1\leq i\leq n/2\\ 1\leq j\leq n}} \in \mathcal{M}_{n/2,n}\left(\mathbb{F}\right), \quad G^{\perp} = \left(g^{\perp}_{i,j}\right)_{\substack{1\leq i\leq n/2\\ 1\leq j\leq n}} \in \mathcal{M}_{n/2,n}\left(\mathbb{F}\right)$$

be matrices of unknowns. The naive model, denoted by $\mathrm{SysNaive}_{\mathrm{HSP}_{|\mathbb{F}|}}$, is defined as the following system of equations:

$$\mathrm{SysNaive}_{\mathrm{HSP}_{|\mathbb{F}|}} = \{\mathrm{Coeff}(p_i, t) = 0, \mathrm{Coeff}(q_j, t) = 0 : \ 1 \leq i \leq m, 1 \leq j \leq m, \\ t \in \mathrm{M}\left(\mathbb{F}[y_1, \ldots, y_{n/2}]\right)\},$$

where $\mathrm{Coeff}(p_i, t)$ denotes the coefficient of the monomial $t \in \mathrm{M}\left(\mathbb{F}[y_1, \ldots, y_{n/2}]\right)$ in the expression

$$p_i\left((y_1, \ldots, y_{n/2})G\right),$$

and $\mathrm{Coeff}(q_j, t)$ denotes the coefficient of the monomial $t \in \mathrm{M}\left(\mathbb{F}[y_1, \ldots, y_{n/2}]\right)$ in the expression

$$q_j\left((y_1, \ldots, y_{n/2})G^{\perp}\right).$$

$\mathrm{SysNaive}_{\mathrm{HSP}_{|\mathbb{F}|}}$ is a model for the $\mathrm{HSP}_{|\mathbb{F}|}$ with

$$\mathcal{O}\left(n^{d+1}\right)$$

equations over $\mathbb{F}$ in n^2 variables. Indeed, for every $n/2$-dimensional subspace $A \subset \mathbb{F}^n$ such that A is a solution of the $\mathrm{HSP}_{|\mathbb{F}|}$ on $(\mathbf{p}, \mathbf{q})$, it occurs that the entries of the matrices

$$G = \left(g_{i,j}\right) \in \mathcal{M}_{n/2,n}(\mathbb{F}), \quad G^{\perp} = \left(g^{\perp}_{i,j}\right) \in \mathcal{M}_{n/2,n}(\mathbb{F})$$

are solutions of the system $\mathrm{SysNaive}_{\mathrm{HSP}_{|\mathbb{F}|}}$, where G denotes a basis matrix of A and $G^{\perp}$ denotes a basis matrix of $A^{\perp}$.

Proof. Let $(\mathbf{p}, \mathbf{q}) \in \mathbb{F}[\boldsymbol{x}]^m \times \mathbb{F}[\boldsymbol{x}]^m$ be a degree-d instance of the $\mathrm{HSP}_{|\mathbb{F}|}$, where $\mathbf{x} = (x_1, \ldots, x_n)$, and let the matrices

$$G = \left(g_{i,j}\right)_{\substack{1\leq i\leq n/2\\ 1\leq j\leq n}} \in \mathcal{M}_{n/2,n}\left(\mathbb{F}\right), \quad G^{\perp} = \left(g^{\perp}_{i,j}\right)_{\substack{1\leq i\leq n/2\\ 1\leq j\leq n}} \in \mathcal{M}_{n/2,n}\left(\mathbb{F}\right)$$

denote a basis matrix of A and a basis matrix of $A^{\perp}$, respectively. Observe that for every $i, j \in \{1, \ldots, m\}$ it holds that

$$p_i\left(\left(y_1, \ldots, y_{n/2}\right) G\right) = \sum_{t\in\mathrm{M}\left(\mathbb{F}[y_1,\ldots,y_{n/2}]\right)} \mathrm{Coeff}(p_i, t)t,$$

$$q_j\left(\left(y_1, \ldots, y_{n/2}\right) G^{T}\right) = \sum_{t\in\mathrm{M}\left(\mathbb{F}[y_1,\ldots,y_{n/2}]\right)} \mathrm{Coeff}(q_j, t)t.$$

It follows that A is a solution of the HSP$_{|\mathbb{F}|}$ if and only if

$$
\begin{aligned}
&\begin{cases} p_i(A) & = 0, \quad \forall i \in \{1,\dots,m\}, \\ q_j(A^{\perp}) & = 0, \quad \forall j \in \{1,\dots,m\}, \end{cases}\\
\iff &\begin{cases} p_i\big((y_1,\dots,y_{n/2})\, G\big) & = 0, \quad \forall i \in \{1,\dots,m\}, \\ q_j\left(\big(y_1,\dots,y_{n/2}\big)\, G^T\right) & = 0, \quad \forall j \in \{1,\dots,m\}, \end{cases}\\
\iff &\begin{cases} \sum\limits_{t\in \mathrm{M}\left(\mathbb{F}[y_1,\dots,y_{n/2}]\right)} \mathrm{Coeff}(p_i,t)t & = 0, \quad \forall i \in \{1,\dots,m\}, \\ \sum\limits_{t\in \mathrm{M}\left(\mathbb{F}[y_1,\dots,y_{n/2}]\right)} \mathrm{Coeff}(q_j,t)t & = 0, \quad \forall j \in \{1,\dots,m\}, \end{cases}\\
\overset{*}{\iff} &\begin{cases} \mathrm{Coeff}(p_i,t) & = 0, \quad \forall i \in \{1,\dots,m\}, \\ \mathrm{Coeff}(q_j,t) & = 0, \quad \forall j \in \{1,\dots,m\}, \end{cases}
\end{aligned}
$$

this is, if and only if the entries of G, $G^{\perp}$ are solutions of the system SysNaive$_{\mathrm{HSP}_{|\mathbb{F}|}}$. Note that the last equivalence, $\overset{*}{\iff}$, derives from the fact that a polynomial over a finite field is identically zero if and only if its coefficients are.

Besides, note that for each $i \in \{1,\dots,m\}$, the polynomial p_i (respectively q_i) gives rise to at most as many equations of the form

$$\mathrm{Coeff}(p_i,t) = 0, \quad t \in \mathrm{M}\left(\mathbb{F}[y_1,\dots,y_{n/2}]\right),$$

as the total number of existing monomials in $\mathbb{F}\left[y_1,\dots,y_{n/2}\right]$ of degree less than or equal to d. The number of monomials in $\mathbb{F}[y_1,\dots,y_{n/2}]$ of a given degree $1 \leq deg \leq d$ equals the number of combination with repetitions of $n/2$ choose deg, which we denote by

$$\left(\!\!\binom{n/2}{deg}\!\!\right) := \binom{n/2+deg-1}{deg}.$$

Then, the total number of equations derived from p_i is at most

$$\left(\!\!\binom{n/2}{d}\!\!\right) + \left(\!\!\binom{n/2}{d-1}\!\!\right) + \ldots + \left(\!\!\binom{n/2}{1}\!\!\right) = \mathcal{O}\left(n^d\right).$$

Now, considering that $(\mathbf{p},\mathbf{q})$ is a pair of m-tuples of polynomials, with each of the polynomials giving rise to equations, and bearing in mind that $n \leq m \leq 2n$, the total of equations of SysNaive$_{\mathrm{HSP}_{|\mathbb{F}|}}$ is $\mathcal{O}(n^{d+1})$, as required. ■

The naive model is a first approach. However, it has characteristics that may slow down the computation of a Gröbner basis of the ideal associated to the system. First, the system has as many equivalent solutions as pairs of the form (G, G^T), where G, G^T are full rank matrices representing a basis of the subspace A and $A^{\perp}$ respectively. This equals

$$\left((|\mathbb{F}|^n-1)(|\mathbb{F}|^n-|\mathbb{F}|),\dots,(|\mathbb{F}|^n-|\mathbb{F}|^{n/2-1})\right)^2,$$

which grows rapidly. Second, we are not actively exploiting the orthogonality between A and $A^{\perp}$ since we have defined two uncorrelated basis matrices G and $G^{\perp}$, underusing information and possibly losing structure on $\mathrm{SysNaive}_{\mathrm{HSP}_{|\mathbb{F}|}}$. This suggests that the model can be refined.

In what follows we optimise the model in such a way that we use the orthogonality relation between the subspaces and we achieve uniqueness of the solution, at the expense of having a model that is only valid with a certain probability.

3.1.2 Our optimised model

The lesson we extract from the naive model is that if we want to construct a model for the $\mathrm{HSP}_{|\mathbb{F}|}$ that has a unique solution, then we need to impose some extra conditions on the subspace A. The following result is the key for the design of our model:

Lemma 3.4 *Let $(\mathbf{p},\mathbf{q}) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ be a degree-d instance of the* $\mathrm{HSP}_{|\mathbb{F}|}$ *and let $A \subset \mathbb{F}^n$ be a subspace of dimension $n/2$. If A is a solution of the instance $(\mathbf{p},\mathbf{q})$ of the* $\mathrm{HSP}_{|\mathbb{F}|}$*, then SA is also a solution of the instance $(\mathbf{p},\mathbf{q})$ of the* $\mathrm{HSP}_{|\mathbb{F}|}$ *for any $S \in \mathrm{GL}_{n/2}(\mathbb{F})$.*

Proof. Note that since $S \in \mathrm{GL}_{n/2}(\mathbb{F})$, the function

$$\begin{aligned}\mathbb{F}^{n/2} &\to \mathbb{F}^{n/2}\\ (y_1,\dots,y_{n/2}) &\mapsto (y_1,\dots,y_{n/2})S\end{aligned}$$

is a bijection. Indeed, surjectivity holds since given an element $(z_1,\dots,z_{n/2}) \in \mathbb{F}^{n/2}$,

$$\exists (y_1,\dots,y_{n/2}) \in \mathbb{F}^{n/2}, \quad (y_1,\dots,y_{n/2}) = (z_1,\dots,z_{n/2})S^{-1},$$

such that $(y_1,\dots,y_{n/2}) \mapsto (z_1,\dots,z_{n/2})$. Injectivity holds since

$$(y_1,\dots,y_{n/2})S = (y_1',\dots,y_{n/2}')S \Longrightarrow (y_1,\dots,y_{n/2}) = (y_1',\dots,y_{n/2}')$$

as a consequence of multiplying the equality by S^{-1}.

The bijectivity of the mapping guarantees that if $\left(y_1,\dots,y_{n/2}\right) \in \mathbb{F}^{n/2}$ is a vector of formal variables, then $(y_1,\dots,y_{n/2})S = (\tilde{y}_1,\dots,\tilde{y}_{n/2})$ is also a vector of formal variables, and so

$$p_i\left(\left(\tilde{y}_1,\dots,\tilde{y}_{n/2}\right)A\right) = 0 \quad \forall i \in \{1,\dots,m\},$$

as a consequence of $(\mathbf{p},\mathbf{q})$ being a solution of the $\mathrm{HSP}_{|\mathbb{F}|}$. Furthermore, the equation

$$q_j\left(\left(y_1,\dots,y_{n/2}\right)(SA)^{\perp}\right) = 0, \quad \forall j \in \{1,\dots,m\},$$

holds because $(SA)^{\perp} = A^{\perp}$. By definition of the orthogonality relation,

$$A^{\perp} = \{\mathbf{a}^{\perp} \in \mathbb{F}^n : \mathbf{a}\cdot\mathbf{a}^{\perp} = 0, \quad \forall \mathbf{a} \in A\}$$

where $\cdot$ is the standard scalar product over $\mathbb{F}^n$. Translating this condition into matrix form, $G^{\perp}$ is a basis matrix of $A^{\perp}$ if and only if $G^{\perp}G^T = 0$, where G is a basis matrix of A. Now it occurs that

$$G^{\perp}\left(SG\right)^T = G^{\perp}G^T S^T = 0$$

and so $(SA)^{\perp} = A^{\perp}$ as required. ∎

Lemma 3.4 allows us to construct a model for the $\mathrm{HSP}_{|\mathbb{F}|}$ with a unique solution provided that some condition on the subspace A is imposed. Let us explain how. Suppose that we have a degree-d instance $(\mathbf{p},\mathbf{q}) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ of the $\mathrm{HSP}_{|\mathbb{F}|}$ and that the entries of the bases G and $G^{\perp}$ are solutions of the system $\mathrm{SysNaive}_{\mathrm{HSP}_{|\mathbb{F}|}}$. Suppose that G has the form

$$G = \left(G_1|G_2\right), \quad G_1 \in \mathrm{GL}_{n/2}\left(\mathbb{F}\right). \tag{3.1}$$

If we set $S = {G_1}^{-1}$, then Lemma 3.4 guarantees that

$$SG = G_1^{-1}(G_1|G_2) = (I|G_1^{-1}G_2),$$

where $I \in \mathrm{GL}_{n/2}(\mathbb{F})$ is the identity matrix, is a basis of A. In this case the subspace A admits a basis of the form $(I|G_1^{-1}G_2)$, which is unique since it coincides with the reduced row echelon form of the matrix G. We say that the basis G is in systematic form.

Clearly, though, not every subspace admits a basis in systematic form. In fact, it only admits one if condition (3.1) is verified. The probability that a subspace admits a basis in systematic form coincides then with the probability that the square matrix of order $n/2$ denoted by G_1 is invertible, which is known to be

$$\gamma_{|\mathbb{F}|}(n/2) = \prod_{i=1}^{n/2}\left(1 - \frac{1}{|\mathbb{F}|^i}\right).$$

This probability can be approximated very roughly by

$$1 - \frac{1}{|\mathbb{F}|}$$

when n takes large values, which is reasonably high even when $|\mathbb{F}| = 2$. Condition 3.1 seems then a sensible restriction to impose over subspaces.

Note that a consequence of considering only solutions that admit a basis in systematic form is first the reduction of the number of unknowns and second the explicit use of the orthogonality relation between the subspaces since

$$\left(I|G_1^{-1}G_2\right)^{\perp} = \left((-G_1^{-1}G_2)^T|I\right).$$

By considering only subspaces that admit a basis in systematic form we can construct a refined model that fulfils the characteristics of a good model that we mentioned at the beginning of this chapter. The optimised model constructed by restricting to subspaces that admit a basis in systematic form is formalised in the following proposition.

Proposition 3.5 (Our Model) *Let $(\mathbf{p}, \mathbf{q}) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ be a degree-d instance of the* $\mathrm{HSP}_{|\mathbb{F}|}$. *Set $N = n^2/4$ and let $\left(y_1, \ldots, y_{n/2}\right) \in \mathbb{F}^{n/2}$ be formal variables. Let*

$$G = \left(g_{i,j}\right)_{\substack{1 \le i \le n/2 \\ 1 \le j \le n/2}} \in \mathcal{M}_{n/2,n/2}\left(\mathbb{F}\right)$$

be a matrix of unknowns. Our model, denoted by $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$, *is defined as the following system:*

$$\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}} = \{\mathrm{Coeff}(p_i, t) = 0, \mathrm{Coeff}(q_j, t) = 0 : \; 1 \le i \le m, 1 \le j \le m \\ t \in \mathrm{M}\left(\mathbb{F}[y_1, \ldots, y_{n/2}]\right)\},$$

where $\mathrm{Coeff}(p_i, t)$ *denotes the coefficient of the monomial $t \in \mathrm{M}\left(\mathbb{F}[y_1, \ldots, y_{n/2}]\right)$ in the expression*

$$p_i\left(\left(y_1, \ldots, y_{n/2}\right) \cdot (I|G)\right)$$

and $\mathrm{Coeff}(q_j, t)$ *denotes the coefficient of the monomial $t \in \mathrm{M}\left(\mathbb{F}[y_1, \ldots, y_{n/2}]\right)$ in the expression*

$$q_j\left(\left(y_1, \ldots, y_{n/2}\right) \cdot \left(-G^T|I\right)\right).$$

$\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$ *is a probabilistic model for the* $\mathrm{HSP}_{|\mathbb{F}|}$ *with*

$$\mathcal{O}\left(n^{d+1}\right)$$

equations over $\mathbb{F}$ in N variables. Indeed, if the $n/2$-dimensional subspace $A \subset \mathbb{F}^n$ is a solution of the $\mathrm{HSP}_{|\mathbb{F}|}$ *on $(\mathbf{p}, \mathbf{q})$, then A admits with probability $\gamma_{|\mathbb{F}|}(n/2)$ a basis $(I|G)$ in systematic form and the entries of $G \in \mathcal{M}_{n/2,n/2}(\mathbb{F})$ are solutions of the system* $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$.

We have constructed a refined model $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$ for the $\mathrm{HSP}_{|\mathbb{F}|}$. Before proceeding further, it may prove useful to spend some time exploring if the system presents different characteristics depending on the choice of the parameters of the $\mathrm{HSP}_{|\mathbb{F}|}$ instance and, if so, study the systems separately.

It turns out that the size of the ground field $\mathbb{F}$ impacts the structure of our model. We explain how in the following example.

Example 3.6 *Set $n = 4, d = 3$ and suppose that the degree-3 polynomial*

$$p_1(\mathbf{x}) = x_1x_2x_3 + x_2x_3x_4 + x_1x_2 + x_3x_4 + x_4 \in \mathbb{F}_2[\mathbf{x}], \quad \mathbf{x} = (x_1, x_2, x_3, x_4),$$

vanishes over a 2-dimensional subspace $A \subset \mathbb{F}_2^4$ that admits a basis in systematic form. According to our model $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$ *of* Proposition 3.5, *if we denote a basis in systematic form of A by*

$$G = \left(g_{i,j}\right)_{\substack{1 \le i \le 2 \\ 1 \le j \le 2}} \in \mathcal{M}_{2,2}\left(\mathbb{F}_2\right),$$

the equations of $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$ *derived from* p_1 *are the coefficients of all monomials of* $\mathbb{F}_2[y_1, y_2]$ *appearing in the expression*

$$\begin{aligned} p_1\left((y_1,\ldots,y_2)\,(I|G)\right) =& p_1(y_1, y_2, g_{1,1}y_1 + g_{2,1}y_2, g_{1,2}y_1 + g_{2,2}y_2)\\ =& {y_2}^3(g_{2,1}g_{2,2}) + {y_1}^2y_2(g_{1,1} + g_{1,1}g_{1,2})\\ &+ y_1{y_2}^2(g_{2,1} + g_{1,1}g_{2,2} + g_{2,1}g_{1,2})\\ &+ y_1y_2(1 + g_{1,1}g_{2,2} + g_{1,2}g_{2,1})\\ &+ {y_1}^2(g_{1,1}g_{1,2}) + {y_2}^2(g_{2,1}g_{2,2}) + g_{1,2}y_1 + g_{2,2}y_2. \end{aligned}$$

However, since $\mathbb{F}_2$ *is cyclic of order two, the following equalities hold*

$$\begin{aligned} g_{1,1}^2 = g_{1,1}, g_{1,2}^2 = g_{1,2}, g_{2,1}^2 = g_{2,1}, g_{2,2}^2 = g_{2,2},\\ {y_1}^2y_2 = y_1{y_2}^2 = y_1y_2,\\ y_1^2 = y_1,\\ y_2^2 = y_2, \end{aligned}$$

and produce simplifications on the expression above. More in particular, the expression above gets finally reduced to

$$\begin{aligned} p_1\left((y_1,\ldots,y_2)\,(I|G)\right) =&(g_{1,1}g_{1,2} + g_{1,1} + g_{2,1} + 1)y_1y_2\\ &+ (g_{1,1}g_{1,2} + g_{1,2})y_1 + g_{2,2}y_2 \end{aligned}$$

and the equations of $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$ *derived from* p_1 *after making the appropriate reductions are:*

$$\begin{aligned} g_{1,1}g_{1,2} + g_{1,1} + g_{2,1} + 1 = \mathrm{Coeff}(p_1, y_1y_2) + \mathrm{Coeff}(p_1, y_1{y_2}^2) + \mathrm{Coeff}(p_1, {y_1}^2y_2) = 0,\\ g_{1,1}g_{1,2} + g_{1,2} = \mathrm{Coeff}(p_1, y_1) + \mathrm{Coeff}(p_1, {y_1}^2) = 0,\\ g_{2,1}g_{2,2} + g_{2,2} = \mathrm{Coeff}(p_1, y_2) + \mathrm{Coeff}(p_1, {y_2}^2) = 0. \end{aligned}$$

This shows that our model for the HSP_2 *must take into account the field equations*

$$\begin{aligned} y_1^2 - y_1 = 0, y_2^2 - y_2 = 0,\\ g_{1,1}^2 - g_{1,1} = 0, g_{1,2}^2 - g_{1,2} = 0,\\ g_{2,1}^2 - g_{2,1} = 0, g_{2,2}^2 - g_{2,2} = 0. \end{aligned}$$

Abstracting a bit further, we can see that reductions do not only occur over $\mathbb{F}_2$*, but any time that the condition* $|\mathbb{F}| \leq d$ *is satisfied, since the field equations*

$$y_i^{|\mathbb{F}|} - y_i = 0, \quad \forall i \in \{1,\ldots,m\},$$

cause in this case that reductions take place. This is not the case if $|\mathbb{F}| > d$.

By dedicating some time to analyse the model first, we have detected a difference between the cases $|\mathbb{F}| \leq d$ and $|\mathbb{F}| > d$, which is why we divide the study of the hardness of the $\mathrm{HSP}_{|\mathbb{F}|}$ into those two scenarios. We start studying the case $\mathrm{HSP}_{|\mathbb{F}|}$ when $|\mathbb{F}| > d$.

3.2 The $\mathrm{HSP}_{|\mathbb{F}|}$ for $|\mathbb{F}| > d$

We dedicate this section to analyse the system $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$ in the case $|\mathbb{F}| > d$, so this condition over the field $\mathbb{F}$ is assumed hereafter although not explicitly stated.

We first present an algorithm that solves the $\mathrm{HSP}_{|\mathbb{F}|}$ by detecting and exploiting the presence of linear equations in the system $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$. We show that our algorithm runs in randomised polynomial-time and that its probability of success is high, increasing with the size of $\mathbb{F}$. Our algorithm translates in a randomised polynomial-time cryptanalysis of the quantum money scheme of Aaronson-Christiano extended to a field $\mathbb{F}$ verifying that $|\mathbb{F}| > d$. Finally, we report experimental results supporting our theoretical findings and proving our algorithm to be very efficient in practice.

3.2.1 Our algorithm solving the $\mathrm{HSP}_{|\mathbb{F}|}$ for $|\mathbb{F}| > d$

As we saw in Example 3.6, in this scenario the equations in the system $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$ do not reduce modulo the field equations, which actually ends up causing the system to display a very favourable behaviour. In fact, the existence of linear equations in the system, along with the fact that we can characterise them and that there are sufficiently many of them, allows us to design an algorithm that solves the $\mathrm{HSP}_{|\mathbb{F}|}$.

In the following result we characterise the linear equations that are present in the system $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$.

Lemma 3.7 *Let* $(\mathbf{p}, \mathbf{q}) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ *be a degree-d instance of the* $\mathrm{HSP}_{|\mathbb{F}|}$. *For* $i \in \{1, \ldots, m\}$, *let* $p_i^{(1)}, q_i^{(1)}$ *denote the homogeneous components of degree* 1 *of* p_i *and* q_i, *respectively, that is:*

$$p_i{}^{(1)} = \sum_{j=1}^{n} \lambda^{\mathbf{p}}_{i,j} x_j, \quad \textit{where } \lambda^{\mathbf{p}}_{i,1}, \ldots, \lambda^{\mathbf{p}}_{i,n} \in \mathbb{F},$$

$$q_i{}^{(1)} = \sum_{j=1}^{n} \lambda^{\mathbf{q}}_{i,j} x_j, \quad \textit{where } \lambda^{\mathbf{q}}_{i,1}, \ldots, \lambda^{\mathbf{q}}_{i,n} \in \mathbb{F}.$$

For $i \in \{1, \ldots, m\}$ *and* $k \in \{1, \ldots, n/2\}$, *the linear equations*

$$\begin{cases} \sum\limits_{j=1}^{n/2} \lambda^{\mathbf{p}}_{i,j+n/2} g_{k,j} + \lambda^{\mathbf{p}}_{i,k}, \\ \sum\limits_{j=1}^{n/2} -\lambda^{\mathbf{q}}_{i,j} g_{j,k} + \lambda^{\mathbf{q}}_{i,k+n/2}, \end{cases}$$

are in $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$.

Proof. Recall that according to the definition of our model $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$ in Proposition 3.5, if the matrix of unknowns

$$G = (g_{i,j})_{1 \le i,j \le n/2} \in \mathcal{M}_{n/2}(\mathbb{F})$$

is such that $(I|G)$ represents a basis in systematic form of the subspace A that is solution of the degree-d instance $(\mathbf{p},\mathbf{q}) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ of the $\mathrm{HSP}_{|\mathbb{F}|}$, the equations in $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$ are

$$\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}} = \{\mathrm{Coeff}(p_i,t) = 0, \mathrm{Coeff}(q_j,t) = 0 : \ 1 \le i \le m, 1 \le j \le m, \\ t \in \mathrm{M}\left(\mathbb{F}[y_1,\ldots,y_{n/2}]\right)\},$$

where $\mathrm{Coeff}(p_i,t)$ is the coefficient of the monomial t in the expression

$$p_i\left(\left(y_1,\ldots,y_{n/2}\right)(I|G)\right),$$

which after expanding equals

$$p_i\left(y_1,\ldots,y_{n/2},\sum_{k=1}^{n/2} g_{k,1}y_k,\ldots,\sum_{k=1}^{n/2} g_{k,n/2}y_k\right), \tag{3.2}$$

and $\mathrm{Coeff}(q_i,t)$ is the coefficient of the monomial t in the expression

$$q_i\left(\left(y_1,\ldots,y_{n/2}\right)\left(-G^T|I\right)\right),$$

which after expanding it equals

$$q_i\left(\sum_{k=1}^{n/2} -g_{1,k}y_k,\ldots,\sum_{k=1}^{n/2} -g_{n/2,k}y_k, y_1,\ldots,y_{n/2}\right). \tag{3.3}$$

Recall from Example 3.6 that the condition $|\mathbb{F}| > d$ guarantees that the field equations

$$y_i^{|\mathbb{F}|} - y_i = 0, \quad \forall i \in \{1,\ldots,m\}$$

do not affect any of the equations of $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$ by causing reductions. This is, given a monomial

$$t \in \mathrm{M}_{d'}(\mathbb{F}[y_1,\ldots,y_{n/2}]),$$

no other monomial $t' \neq t$ in $\mathbb{F}[y_1,\ldots,y_{n/2}]$ reduces to t. Therefore, for any $i \in \{1,\ldots,m\}$, the expression $\mathrm{Coeff}(p_i,t)$ depends solely on the monomial t and more generally on the homogeneous component of degree d' of p_i. This is,

$$\mathrm{Coeff}(p_i,t) = \mathrm{Coeff}\left(p_i^{(d')},t\right).$$

Furthermore, the monomial t can be expressed as

$$t = y_{i_1}y_{i_2}\ldots y_{i_{d'}} \quad i_1,\ldots,i_{d'} \in \{1,\ldots,n/2\},$$

and observing the expression (3.2) we see that $\mathrm{Coeff}\left(p_i^{(d')},t\right)$ is of degree exactly d' if at least one of the coefficients of the monomials in the set

$$\{mon \in \mathrm{M}_{d'}(\mathbb{F}[\mathbf{x}]) : \ x_i \ \text{divides} \ mon \ \text{ for all } \ i \in \{n/2+1,\ldots,n\}\} \tag{3.4}$$

is non-zero. Indeed, the i-th component of the vector $(y_1, \dots, y_{n/2})(I|G)$ has dependency on the unknowns of G for all $i \in \{n/2+1, \dots, n\}$, whereas the j-th component of the vector does not have dependency on the unknowns of G for all $j \in \{1, \dots, n/2\}$. Thus the expression $\mathrm{Coeff}(p_i, t)$ is of degree exactly d' with probability

$$1 - \frac{1}{|\mathbb{F}|^{\binom{n/2}{d'}}},$$

which is overwhelming for large values of the parameters.

Remark 3.8 *An analogous argument can be used for q_i for every $q_i \in \{1, \dots, n\}$ to conclude the same, only considering that the set of expression* (3.4) *is now*

$$\{mon \in M_{d'}(\mathbb{F}[\mathbf{x}]) : \ x_i \ divides \ \ mon \ \ for \ all \ \ i \in \{1, \dots, n/2\}\}. \tag{3.5}$$

Therefore, the expressions

$$\mathrm{Coeff}\left(p_i^{(1)}, y_1\right), \dots, \mathrm{Coeff}\left(p_i^{(1)}, y_{n/2}\right), \mathrm{Coeff}\left(q_i^{(1)}, y_1\right), \dots, \mathrm{Coeff}\left(q_i^{(1)}, y_{n/2}\right) \tag{3.6}$$

are each of them linear with overwhelming probability. The expressions (3.6) above are the coefficients of the monomials $y_1, \dots, y_n$ in the expressions

$$\begin{cases} p_i^{(1)}\left(y_1, \dots, y_{n/2}, \sum\limits_{t=1}^{n/2} g_{t,1} y_t, \dots, \sum\limits_{t=1}^{n/2} g_{t,n/2} y_t\right), & 1 \leq i \leq m, \\ q_i^{(1)}\left(\sum\limits_{t=1}^{n/2} -g_{1,t} y_t, \dots, \sum\limits_{t=1}^{n/2} -g_{n/2,t} y_t, y_1, \dots, y_{n/2}\right), & 1 \leq i \leq m. \end{cases} \tag{3.7}$$

Substituting in (3.7) the expressions

$$\begin{aligned} {p_i}^{(1)} &= \sum_{j=1}^{n} \lambda^{\mathbf{p}}_{i,j} x_j, \quad \text{where } \lambda^{\mathbf{p}}_{i,1}, \dots, \lambda^{\mathbf{p}}_{i,n} \in \mathbb{F}, \\ {q_i}^{(1)} &= \sum_{j=1}^{n} \lambda^{\mathbf{q}}_{i,j} x_j, \quad \text{where } \lambda^{\mathbf{q}}_{i,1}, \dots, \lambda^{\mathbf{q}}_{i,n} \in \mathbb{F}, \end{aligned}$$

and expanding the expression of the evaluations, we see that for $k = 1, 2, \dots, n/2$,

$$\mathrm{Coeff}(p_i, y_k) = \mathrm{Coeff}\left(p_i^{(1)}, y_k\right) = \lambda^{\mathbf{p}}_{i,k} + \sum_{j=1}^{n/2} \lambda^{\mathbf{p}}_{i,j+n/2} g_{k,j}, \quad \forall i \in \{1, \dots, m\},$$

$$\mathrm{Coeff}(q_i, y_k) = \mathrm{Coeff}\left(q_i^{(1)}, y_k\right) = \lambda^{\mathbf{q}}_{i,k+n/2} - \sum_{j=1}^{n/2} \lambda^{\mathbf{q}}_{i,j} g_{j,k}, \quad \forall i \in \{1, \dots, m\},$$

as required. ■

After having found that there are linear equations in the system $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$, it is sensible to wonder how many of these linear equations happen to be linearly independent, just in case there are enough to find a solution.

Given $i \in \{1, \ldots, m\}$, Lemma 3.7 shows that there are $n/2$ linear equations derived from p_i (one derived from each linear monomial in $\mathbb{F}[y_1, \ldots, y_{n/2}]$) and analogously for q_i. Since $1 \leq i \leq m$ and $n \leq m \leq 2n$, the total number of linear equations in $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$ is

$$2m\frac{n}{2} = mn \geq 4N,$$

where N denotes the number of unknowns in $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$, this is, $n^2/4$. The system of linear equations present in $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$ is already overdetermined, so it is seems feasible that there are at least N of those linear equations that are linearly independent.

The following lemma shows that there are indeed N linearly independent linear equations in the system $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$ with overwhelming probability.

Lemma 3.9 *Let* $(\mathbf{p}, \mathbf{q}) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ *be a degree-d instance of the* $\mathrm{HSP}_{|\mathbb{F}|}$*. It occurs that at least* $N = n^2/4$ *linear equations from* $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$ *are linearly independent with probability*

$$\frac{\gamma_{|\mathbb{F}|}(m)}{\gamma_{|\mathbb{F}|}(m - n/2)}.$$

Proof. Recall that the linear equations of $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$ given by Lemma 3.7 are, for $i \in \{1, \ldots, m\}$ and $k \in \{1, \ldots, n/2\}$,

$$\begin{cases} \mathrm{Coeff}(p_i, y_k) = \sum\limits_{j=1}^{n/2} \lambda^{\mathbf{p}}_{i,j+n/2} g_{k,j} + \lambda^{\mathbf{p}}_{i,k} \\ \mathrm{Coeff}(q_i, y_k) = \sum\limits_{j=1}^{n/2} -\lambda^{\mathbf{q}}_{i,j} g_{j,k} + \lambda^{\mathbf{q}}_{i,k+n/2} \end{cases}$$

Let us construct the matrix of coefficients associated to the linear system of equations to examine its rank. The columns represent the unknowns

$$g_{1,1}, g_{1,2}, \ldots, g_{1,n/2}, g_{2,1}, g_{2,2}, \ldots, g_{2,n/2}, g_{n/2,1}, g_{n/2,2}, \ldots, g_{n/2,n}, 1,$$

whereas the entries of the matrix represent the coefficients of the linear equations

$$\mathrm{Coeff}(p_i, y_k) = 0, \quad \mathrm{Coeff}(q_i, y_k) = 0, \quad \forall i \in \{1, \ldots, m\}, \forall k \in \{1, \ldots, n/2\}.$$

in the above unknowns.

We obtain the matrix of coefficients given in (3.8).

Before continuing, recall from the proof of Lemma 3.7 that, on average, there are

$$mn - \lceil \frac{mn}{2^{n/2+1}} \rceil$$

rows of the matrix of coefficients that are non-zero, which tends to mn for large values of n.

$$\begin{pmatrix}
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
\lambda^{\mathbf{p}}_{i,n/2+1} & \lambda^{\mathbf{p}}_{i,n/2+2} & \cdots & \lambda^{\mathbf{p}}_{i,n} & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & \cdots & 0 & \lambda^{\mathbf{p}}_{i,1} \\
0 & 0 & \cdots & 0 & \lambda^{\mathbf{p}}_{i,n/2+1} & \lambda^{\mathbf{p}}_{i,n/2+2} & \cdots & \lambda^{\mathbf{p}}_{i,n} & \cdots & 0 & 0 & \cdots & 0 & \lambda^{\mathbf{p}}_{i,2} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \cdots & \lambda^{\mathbf{p}}_{i,n/2+1} & \lambda^{\mathbf{p}}_{i,n/2+2} & \cdots & \lambda^{\mathbf{p}}_{i,n} & \lambda^{\mathbf{p}}_{i,n/2} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
-\lambda^{\mathbf{q}}_{j,1} & 0 & \cdots & 0 & -\lambda^{\mathbf{q}}_{j,2} & 0 & \cdots & 0 & \cdots & -\lambda^{\mathbf{q}}_{j,n/2} & 0 & \cdots & 0 & \lambda^{\mathbf{q}}_{j,n/2+1} \\
0 & -\lambda^{\mathbf{q}}_{j,1} & \cdots & 0 & 0 & -\lambda^{\mathbf{q}}_{j,2} & \cdots & 0 & \cdots & 0 & -\lambda^{\mathbf{q}}_{j,n/2} & \cdots & 0 & \lambda^{\mathbf{q}}_{j,n/2+2} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
0 & 0 & \cdots & -\lambda^{\mathbf{q}}_{j,1} & 0 & 0 & \cdots & -\lambda^{\mathbf{q}}_{j,2} & \cdots & 0 & 0 & \cdots & -\lambda^{\mathbf{q}}_{j,n/2} & \lambda^{\mathbf{q}}_{j,n} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots
\end{pmatrix} \tag{3.8}$$

$$\begin{pmatrix}
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
\lambda^{\mathbf{p}}_{i_1,n/2+1} & \lambda^{\mathbf{p}}_{i_1,n/2+2} & \cdots & \lambda^{\mathbf{p}}_{i_1,n} & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 & \lambda^{\mathbf{p}}_{i_1,n/2+1} & \lambda^{\mathbf{p}}_{i_1,n/2+2} & \cdots & \lambda^{\mathbf{p}}_{i_1,n} & \cdots & 0 & 0 & \cdots & 0 \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \cdots & \lambda^{\mathbf{p}}_{i_1,n/2+1} & \lambda^{\mathbf{p}}_{i_1,n/2+2} & \cdots & \lambda^{\mathbf{p}}_{i_1,n} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
\lambda^{\mathbf{p}}_{i_{n/2},n/2+1} & \lambda^{\mathbf{p}}_{i_{n/2},n/2+2} & \cdots & \lambda^{\mathbf{p}}_{i_{n/2},n} & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 & \lambda^{\mathbf{p}}_{i_{n/2},n/2+1} & \lambda^{\mathbf{p}}_{i_{n/2},n/2+2} & \cdots & \lambda^{\mathbf{p}}_{i_{n/2},n} & \cdots & 0 & 0 & \cdots & 0 \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \cdots & \lambda^{\mathbf{p}}_{i_{n/2},n/2+1} & \lambda^{\mathbf{p}}_{i_{n/2},n/2+2} & \cdots & \lambda^{\mathbf{p}}_{i_{n/2},n}
\end{pmatrix} \tag{3.9}$$

We can see that the matrix of coefficients has a noticeable structure with symmetries. In particular we can see that, for given $i \in \{1, \ldots, m\}$ and $k \in \{1, \ldots, n/2\}$, if the row corresponding to the equation $\mathrm{Coeff}(p_i, y_k) = 0$ has non-trivial entries

$$\lambda^{\mathbf{p}}_{i,n/2+1}, \lambda^{\mathbf{p}}_{i,n/2+2}, \ldots, \lambda^{\mathbf{p}}_{i,n},$$

then the row corresponding to the equation $\mathrm{Coeff}(p_i, y_{k+1}) = 0$ has exactly the same non-trivial entries shifted $n/2$ positions to the right.

Furthermore, for given $i \in \{1, \ldots, m\}$ and $k \in \{1, \ldots, n/2\}$, if the row corresponding to the equation $\mathrm{Coeff}(q_j, y_k) = 0$ has non-trivial entries

$$\lambda^{\mathbf{q}}_{j,1}, \lambda^{\mathbf{q}}_{j,n/2+2}, \ldots, \lambda^{\mathbf{q}}_{j,n},$$

then the row $\mathrm{Coeff}(q_j, y_{k+1})$ has exactly the same non-trivial entries, each of them shifted one position to the right.

Let us now restrict our attention to the following $mn/2 \times N$ submatrix containing the equations $\mathrm{Coeff}(p_i, y_k)$, for $i \in \{1, \ldots, m\}$ and for all $k \in \{1, \ldots, n/2\}$,

$$\begin{pmatrix}
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
\lambda^{\mathbf{p}}_{i,n/2+1} & \cdots & \lambda^{\mathbf{p}}_{i,n} & 0 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\
0 & \cdots & 0 & \lambda^{\mathbf{p}}_{i,n/2+1} & \cdots & \lambda^{\mathbf{p}}_{i,n} & \cdots & 0 & \cdots & 0 \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
0 & \cdots & 0 & 0 & \cdots & 0 & \cdots & \lambda^{\mathbf{p}}_{i,n/2+1} & \cdots & \lambda^{\mathbf{p}}_{i,n} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots
\end{pmatrix} \tag{3.10}$$

Due to its very particular shape, the matrix (3.10) has rank N if the following $m \times n/2$ matrix

$$\begin{pmatrix}
\lambda^{\mathbf{p}}_{1,n/2+1} & \lambda^{\mathbf{p}}_{1,n/2+2} & \cdots & \lambda^{\mathbf{p}}_{1,n} \\
\lambda^{\mathbf{p}}_{2,n/2+1} & \lambda^{\mathbf{p}}_{2,n/2+2} & \cdots & \lambda^{\mathbf{p}}_{2,n} \\
\cdots & \cdots & \cdots & \cdots \\
\lambda^{\mathbf{p}}_{m,n/2+1} & \lambda^{\mathbf{p}}_{m,n/2+2} & \cdots & \lambda^{\mathbf{p}}_{m,n}
\end{pmatrix} \tag{3.11}$$

has rank N. Indeed, if the matrix (3.11) has rank N then there is a certain $n/2 \times n/2$ submatrix of the matrix (3.11) that has maximum rank. If we denote it by

$$\begin{pmatrix}
\lambda^{\mathbf{p}}_{i_1,n/2+1} & \lambda^{\mathbf{p}}_{i_1,n/2+2} & \cdots & \lambda^{\mathbf{p}}_{i_1,n} \\
\lambda^{\mathbf{p}}_{i_2,n/2+1} & \lambda^{\mathbf{p}}_{i_2,n/2+2} & \cdots & \lambda^{\mathbf{p}}_{i_2,n} \\
\cdots & \cdots & \cdots & \cdots \\
\lambda^{\mathbf{p}}_{i_{n/2},n/2+1} & \lambda^{\mathbf{p}}_{i_{n/2},n/2+2} & \cdots & \lambda^{\mathbf{p}}_{i_{n/2},n}
\end{pmatrix} \tag{3.12}$$

for $i_1, \ldots, i_{n/2} \in \{1, \ldots, m\}$, then the matrix (3.9) has full rank N, and since that matrix is a submatrix of the matrix (3.10), the matrix (3.10) has rank N as stated.

Therefore, we have proved that the matrix of coefficients has rank N if the $m \times n/2$ matrix (3.11) has full rank. Now, since the coefficients of the matrix (3.11) are uniformly random, the probability that the matrix (3.11) has full rank N coincides with the probability that a random $m \times n/2$ matrix has maximum rank N, which according to [Brent and McKay, 1987], is precisely

$$\frac{\left(1-\frac{1}{|\mathbb{F}|}\right)\dots\left(1-\frac{1}{|\mathbb{F}|^m}\right)}{\left(1-\frac{1}{|\mathbb{F}|}\right)\dots\left(1-\frac{1}{|\mathbb{F}|^{m-n/2}}\right)}=\frac{\gamma_{|\mathbb{F}|}(m)}{\gamma_{|\mathbb{F}|}(m-n/2)},$$

as required ■

At this point, the results from Lemma 3.7 and Lemma 3.9 ensure that there are enough linearly independent linear equations in $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$ to solve it with overwhelming probability, which gives rise to our algorithm to solve the $\mathrm{HSP}_{|\mathbb{F}|}$.

Theorem 3.10 *Let $|\mathbb{F}| > d$. There is a randomised polynomial-time algorithm solving degree-d instances of the $\mathrm{HSP}_{|\mathbb{F}|}$ in time:*

$$\mathcal{O}\left(n^{2\omega}\right),$$

where ω is the exponent of matrix multiplication, and with success probability

$$\frac{\gamma_{|\mathbb{F}|}(n/2)\gamma_{|\mathbb{F}|}(m)}{\gamma_{|\mathbb{F}|}(m-n/2)}.$$

The success probability of our algorithm can be asymptotically approximated by

$$1-\frac{1}{|\mathbb{F}|}.$$

Proof. The algorithm that solves the $\mathrm{HSP}_{|\mathbb{F}|}$ is a direct derivation from Lemma 3.7 and (3.9), namely:

```
Input: $(\mathbf{p},\mathbf{q}) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ a degree-$d$ instance of the $\mathrm{HSP}_{|\mathbb{F}|}$, with $|\mathbb{F}| > d$.
Compute the solution $E_A$ of the system of linear equations
    $\left\{ \lambda^{\mathbf{p}}_{i,k} + \sum_{j=1}^{n/2} \lambda^{\mathbf{p}}_{i,j+n/2} g_{k,j}, \quad \forall i \in \{1,\dots,m\}, \forall k \in \{1,\dots,n/2\} \right\}$
$A \leftarrow \mathrm{span}(E_A)$
return $A$
```

As for the probability of success of our algorithm, the algorithm succeeds first if the subspace solution A of the instance $(\mathbf{p},\mathbf{q}) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ of the $\mathrm{HSP}_{|\mathbb{F}|}$ admits a basis in systematic form (see Proposition 3.5), which occurs with probability

$$\gamma_{|\mathbb{F}|}(n/2),$$

and second if the system of linear equations has at least N linearly independent ones, which has been proved in Lemma 3.9 to happen with probability

$$\frac{\gamma_{|\mathbb{F}|}(m)}{\gamma_{|\mathbb{F}|}(m-n/2)}.$$

The product of both probabilities gives the desired probability of success. Furthermore, taking into account that

$$\gamma_{|\mathbb{F}|}(n) = \prod_{i=1}^{n} \left(1 - \frac{1}{|\mathbb{F}|^i}\right)$$

and

$$\lim_{n\to\infty} \gamma_{|\mathbb{F}|}(n) = 1 - \frac{1}{|\mathbb{F}|} + \mathcal{O}\left(\frac{1}{|\mathbb{F}|^2}\right),$$

it follows that

$$\lim_{n\to\infty} \frac{\gamma_{|\mathbb{F}|}(n/2)\gamma_{|\mathbb{F}|}(m)}{\gamma_{|\mathbb{F}|}(m-n/2)} = 1 - \frac{1}{|\mathbb{F}|} + \mathcal{O}\left(\frac{1}{|\mathbb{F}|^2}\right),$$

and so the asymptotic success probability of our algorithm increases as we increase the cardinality of $\mathbb{F}$. ■

Before moving on to present the experimental results of our algorithm, it is important to remark that the $\mathrm{HSP}_{|\mathbb{F}|}$ is the problem on which the security of the public-key quantum money scheme of Aaronson-Christiano relies. Their scheme is defined in [Aaronson and Christiano, 2013] over the field $\mathbb{F}_2$, but in the same paper it is left as an open question whether or not an extension of the scheme to other fields is possible. Our algorithm of Theorem 3.10 gives a negative answer to that question: it solves the $\mathrm{HSP}_{|\mathbb{F}|}$ in randomised polynomial time for degree-d instances of the $\mathrm{HSP}_{|\mathbb{F}|}$ that satisfy the condition $|\mathbb{F}| > d$, breaking Conjecture 2.4 and hence yielding a cryptanalysis of Aaronson-Christiano's quantum money scheme extended to fields other than the binary one.

3.2.2 Experimental results

Here in we present experimental results (see Tables 3.1 and 3.2) that complement our theoretical findings summed up in Theorem 3.10. All experiments were run on a 2.93 GHz Intel PC with 128 Gb of RAM with the MAGMA software (V2.20-10) [Bosma et al., 1997] and our MAGMA source code is available online on GitHub (https://github.com/Marta-PhD/solving-HSP-NHSP) and can also be read in Appendix A.

Due to the fact that generating instances of the $\mathrm{HSP}_{|\mathbb{F}|}$ is costly, our source code generates degree-d instances such that its subspace solution A can be expressed in systematic form and so our optimised model is always valid. Our algorithm behaves as expected from Theorem 3.10: it is very efficient in practice and it succeeds with high probability.

We start by choosing the parameters $d = 3$ and $m = n$ due to reasons of both speed and memory consumption. We select $m = n$ as this is the most unfavourable scenario for our attack (recall that $n \leq m \leq 2n$): as m increases, the system $\mathrm{Sys}_{\mathrm{HSP}_{|\mathbb{F}|}}$ becomes more overdefined. Furthermore, we perform experiments over fields of very different sizes to observe any potential variation of speed. Note that in Table 3.1, $\mathrm{Time}_{\mathrm{gen}}$ denotes the time needed to generate the degree-3 instances of the $\mathrm{HSP}_{|\mathbb{F}|}$ to which we applied our algorithm, and that Time is the time that our algorithm takes to output a solution (which essentially coincides with the time spent solving the linear system of Theorem 3.10).

	$d=3,\quad m=n$					
	$\lvert\mathbb{F}\rvert=5$			$\lvert\mathbb{F}\rvert=2^{16}+1$		
n	10	12	20	10	12	20
Time (in sec.)	0.00	0.00	0.02	0.00	0.00	0.03
Time_{gen} (in sec.)	1	2	135.1	1	4	244.7

Table 3.1: Performance of the cryptanalysis (Theorem 3.10) of the $\text{HSP}_{\lvert\mathbb{F}\rvert}$

We can see that the algorithm of Theorem 3.10 is very fast and that an increase of the size of the field does not imply an increase of the time that the algorithm takes to output a solution. However, we can see that the generation of the instances is significantly slower for $\lvert\mathbb{F}\rvert = 2^{16}+1$. As we mentioned in Remark 2.14, this occurs due to the fact that the speed of multiplications over a field depends on its size. More in particular, when applying Proposition 2.13 to generate m polynomials in $I_{d,A}$ and m polynomials in $I_{d,A^{\perp}}$, the first step takes $\mathcal{O}(mn^d)$ and the second step takes $\mathcal{O}(mdn^d)$, so the total cost is always dominated by the cost of the second step, i.e, the total cost is

$$\mathcal{O}\left(mnd^d\right).$$

However, the cost of the multiplications in $\mathbb{F}$ when $\lvert\mathbb{F}\rvert = 2^{16}+1$ is higher than when $\lvert\mathbb{F}\rvert = 5$.

We maintain the notation of Table 3.1 and the same choice of parameters $m = n$ for degree $d = 4$ for the reasons explained above. We report the following experiments.

	$d=4,\quad m=n$				
	$\lvert\mathbb{F}\rvert=5$		$\lvert\mathbb{F}\rvert=2^{16}+1$		
n	10	12	10	12	20
Time (in sec.)	0.00	0.00	0.00	0.00	0.03
Time_{gen} (in sec.)	8	40	18	107	5154.050

Table 3.2: Performance of our algorithm (Theorem 3.10) that solves the $\text{HSP}_{\lvert\mathbb{F}\rvert}$

The behaviour displayed by our algorithm for $d = 4$ is, in terms of speed, similar to the case $d = 3$: it is very fast and the generation of degree-4 instances of the $\text{HSP}_{\lvert\mathbb{F}\rvert}$ is slower as expected.

Regarding the probability of success of our algorithm assuming that our optimised model can be applied, bear in mind that the quotient

$$\frac{\gamma_{\lvert\mathbb{F}\rvert}\,(m)}{\gamma_{\lvert\mathbb{F}\rvert}\,(m-n/2)}$$

tends to 1 very quickly irrespective of the degree of the instances of the $\text{HSP}_{\lvert\mathbb{F}\rvert}$ and almost irrespective of the size of $\mathbb{F}$, which barely affects this tendency. Our algorithm succeeded for all the experiments that we performed on degree-3 and degree-4 instances of the HSP$_{|\mathbb{F}|}$.

Remark 3.11 *Note that the generation of instances of the* HSP$_{|\mathbb{F}|}$ *is rather slow in practice, probably due to a non-optimal implementation of the* `Evaluate` *function in* Magma *for symbolic polynomials.*

3.3 The HSP$_2$

We dedicate this section to study the hardness of the hidden subspaces problem over $\mathbb{F}_2$, a particular case satisfying the condition $|\mathbb{F}| \leq d$ and hence not covered in the previous analysis. The hidden subspaces problem was originally defined over $\mathbb{F}_2$ in [Aaronson and Christiano, 2013], so the results we obtain herein have a direct impact on the security of the original quantum money scheme of Aaronson-Christiano.

Let us anticipate that the strategy used in Section 3.2 cannot be replicated since linear equations no longer exist. However, our system modelling the hidden subspaces problem over the field $\mathbb{F}_2$ is still largely overdefined, which leaves open the possibility that the computation of a Gröbner basis is somehow efficient. Throughout this section we particularise our model for $\mathbb{F}_2$ and then we present a heuristic randomised polynomial-time algorithm that solves the HSP$_2$ assuming a conjecture about the degree of regularity of our system. After that, we present theoretical results that support the polynomial-time nature of our algorithm.

3.3.1 Structure of our model over the field $\mathbb{F}_2$

As we saw in Example 3.6, the field equations

$$g_{i,j}^2 - g_{i,j} = 0, \quad \forall i, j \in \{1, \ldots, m\},$$

produce reductions in the equations of our model from Proposition 3.5, so we have to include them in the system. Our model particularised to the case of $\mathbb{F}_2$ is as follows.

Proposition 3.12 *Let $(\boldsymbol{p}, \boldsymbol{q}) \in \mathbb{F}_2[\mathbf{x}]^m \times \mathbb{F}_2[\mathbf{x}]^m$ be a degree-d instance of the* HSP$_2$. *Set $N = n^2/4$, let $\left(y_1, \ldots, y_{n/2}\right) \in \mathbb{F}_2^{n/2}$ be formal variables and let*

$$G = \left(g_{i,j}\right)_{\substack{1 \leq i \leq n/2 \\ 1 \leq j \leq n/2}} \in \mathcal{M}_{n/2,n/2}\left(\mathbb{F}_2\right)$$

be a matrix of unknowns. Our model, denoted by Sys$_{\mathrm{HSP}_2}$, *is defined as the following system:*

$$\begin{aligned} \mathrm{Sys}_{\mathrm{HSP}_2} = \{ & \mathrm{Coeff}(p_i, t) = 0, \mathrm{Coeff}(q_j, t) = 0 \colon 1 \leq i \leq m, 1 \leq j \leq m, \\ & t \in M\left(\mathbb{F}_2[y_1, \ldots, y_{n/2}]\right)\} \bigcup \{g_{ij}^2 - g_{ij} \colon 1 \leq i \leq n/2, 1 \leq j \leq n/2\}, \end{aligned}$$

where $\mathrm{Coeff}(p_i, t)$ *denotes the coefficient of the monomial $t \in M\left(\mathbb{F}_2[y_1, \ldots, y_{n/2}]\right)$ in the expression*

$$p_i\left(\left(y_1, \ldots, y_{n/2}\right)(I|G)\right)$$

and $\mathrm{Coeff}(q_j, t)$ *denotes the coefficient of the monomial* $t \in M\left(\mathbb{F}_2[y_1, \ldots, y_{n/2}]\right)$ *in the expression*

$$q_j\left(\left(y_1, \ldots, y_{n/2}\right)\left(G^T|I\right)\right).$$

$\mathrm{Sys}_{\mathrm{HSP}_2}$ *is a probabilistic model of the* HSP_2 *of*

$$\mathcal{O}\left(n^{d+1}\right)$$

equations over $\mathbb{F}_2$ *in* N *variables. Indeed, if* $A \subset \mathbb{F}_2^n$ *is a solution of the* HSP_2 *on* $(\mathbf{p}, \mathbf{q})$*, then* A *admits with probability* $\gamma_2(n/2)$ *a basis* $(I|G)$ *in systematic form such that the entries of* $G \in \mathcal{M}_{n/2,n/2}(\mathbb{F})$ *are solutions of the system* $\mathrm{Sys}_{\mathrm{HSP}_2}$.

Proof. Our model $\mathrm{Sys}_{\mathrm{HSP}_2}$ is a particularisation of our model from Proposition 3.5 to the case HSP_2. Please note that $-G^T = G^T$ over $\mathbb{F}_2$, and that due to the field equations

$$y_i^2 - y_i = 0, \quad \forall i \in \{1, \ldots, n/2\},$$

the monomials in $\mathbb{F}_2[y_1, \ldots, y_{n/2}]$ are square-free. Therefore, the total number of monomials in $\mathbb{F}[y_1, \ldots, y_{n/2}]$ is now

$$\binom{n/2}{d} + \binom{n/2}{d-1} + \ldots + \binom{n/2}{1} = \mathcal{O}\left(n^d\right).$$

Since $(\mathbf{p}, \mathbf{q})$ is a pair of m-tuples of polynomials and considering that $n \leq m \leq 2m$, we obtain that the total number of equations of $\mathrm{Sys}_{\mathrm{HSP}_2}$ is $\mathcal{O}(n^{d+1})$ as desired. ∎

The fact that there are modular reductions taking place over $\mathbb{F}_2$ is less innocent than it seems and it is the ultimate reason why there are no linear equations in the system, contrary to what occurred in Section 3.2. Recall that in the case $|\mathbb{F}| > d$, equations of the form

$$\mathrm{Coeff}(p_i, y_j) = 0, \quad \mathrm{Coeff}(q_i, y_j) = 0, \quad 1 \leq i \leq m, 1 \leq j \leq n/2, \tag{3.13}$$

were linear. However, over $\mathbb{F}_2$ it occurs that

$$y_j = {y_j}^2 = {y_j}^3 = \ldots\ldots = {y_j}^d, \quad j \in \{1, \ldots, n/2\},$$

and so the analogous equations to the equations (3.13) present in our system after reductions are

$$\mathrm{Coeff}\left(p_i, y_j\right) + \mathrm{Coeff}\left(p_i, {y_j}^2\right) + \ldots + \mathrm{Coeff}\left(p_i, {y_j}^d\right) = 0, \quad 1 \leq i \leq m, 1 \leq j \leq n/2,$$

$$\mathrm{Coeff}\left(q_i, y_j\right) + \mathrm{Coeff}\left(q_i, {y_j}^2\right) + \ldots + \mathrm{Coeff}\left(q_i, {y_j}^d\right) = 0, \quad 1 \leq i \leq m, 1 \leq j \leq n/2,$$

which are not linear. In fact, they have degree d with high probability. Indeed, recall from the proof of Lemma 3.7 that the expression $\mathrm{Coeff}(p_i, {y_j}^d)$ has degree d if at least one of the coefficients of the monomials in the set $\mathrm{M}_d\left(\mathbb{F}[x_{n/2+1}, \ldots, x_n]\right)$ of p_i is non-zero, which occurs with probability

$$1 - \frac{1}{2^{\binom{n/2}{d}}},$$

which is very high for appropriately large values of the parameters.

Remark 3.13 *The same argument applies for the expressions*

$$\mathrm{Coeff}\left(q_i, {y_j}^d\right) = 0, \quad i \in \{1, \ldots, m\}, j \in \{1, \ldots, n/2\},$$

which are of degree d if at least one of the coefficients of the monomials in the set $\mathrm{M}_d\left(\mathbb{F}[x_1, \ldots, x_{n/2}]\right)$ of q_i is non-zero, which occurs as well with probability

$$1 - \frac{1}{2^{\binom{n/2}{d}}}.$$

Remark 3.14 *We wrote expressions of the form*

$$\mathrm{Coeff}\left(p_i, y_j^d\right) = 0, \quad \mathrm{Coeff}\left(q_i, y_j^d\right) = 0, \quad 1 \leq i \leq m, 1 \leq j \leq n/2,$$

only for the sake of clarity in order to explain that reductions take place and that an expression of the form

$$\mathrm{Coeff}(p_i, t), \quad t \in \mathrm{M}\left(\mathbb{F}[y_1, \ldots, y_{n/2}]\right),$$

gets contributions from all monomials $t' \in \mathrm{M}\left(\mathbb{F}[y_1, \ldots, y_{n/2}]\right)$ that reduce to t. In what follows, we write Coeff(p_i, t) *simply to refer to the final coefficient once the contributions coming from reductions over $\mathbb{F}_2[y_1, \ldots, y_{n/2}]$ have been added up.*

Therefore, our system $\mathrm{Sys}_{\mathrm{HSP}_2}$ consists —with very high probability— of equations of degree d over $\mathbb{F}_2$. However, the system is greatly overdetermined with $\mathcal{O}(n^{d+1})$ equations versus $n^2/4$ unknowns, so we can hope that computing a Gröbner basis of $\mathrm{Sys}_{\mathrm{HSP}_2}$ can be done efficiently. In what follows we conjecture that indeed it can.

3.3.2 Our heuristic algorithm solving the HSP$_2$

A priori we intended to run some tests only to get an impression of whether computing a Gröbner basis seemed to be efficient or not. A posteriori, a deeper study of our experimental results — which ended up pointing to the fact that $\mathrm{Sys}_{\mathrm{HSP}_2}$ is indeed much easier to solve than a semi-regular system with the same parameters — allowed us to conjecture the existence of a randomised algorithm that heuristically solves the HSP$_2$ in polynomial time.

We report experiments run on a 2.93 GHz Intel PC with 128 Gb. of RAM with the Magma software (V2.20-10) [Bosma et al., 1997] for the most disadvantageous choice of parameters in terms of overdefinition of the system, this is, $m = n$. The notation used in Tables 3.3 and 3.4 is the following: n is the number of variables of the polynomials of the pair $(\mathbf{p}, \mathbf{q})$, $N = n^2/4$ is the number of unknowns of the equations in the system $\mathrm{Sys}_{\mathrm{HSP}_2}$, $\mathrm{U}_{\mathrm{eqs}}$ is the upper bound on the number of equations in the system excluding the field equations (see Proposition 3.12), this is,

$$\binom{n/2}{d} + \binom{n/2}{d-1} + \ldots + \binom{n/2}{1},$$

and finally, d_{reg} is the degree of regularity observed in practice and $d_{\mathrm{reg}}^{\mathrm{sg}}$ is the theoretical degree of regularity treating the system as if it was semi-regular.

$d = 3$					
n	N	$\mathrm{U}_{\mathrm{eqs}}$	$d_{\mathrm{reg}}^{\mathrm{sg}}$	d_{reg}	Time (in sec)
8	16	224	4	3	1
10	25	500	5	3	1
12	36	984	5	3	2
14	49	1764	5	4	136
16	64	2944	6	4	150
18	81	4725	7	4	8000

Table 3.3: Behaviour of a Gröbner basis computation of the ideal associated to the system $\mathrm{Sys}_{\mathrm{HSP}_2}$ for degree-3 instances of the HSP_2

$d = 4$					
n	N	$\mathrm{U}_{\mathrm{eqs}}$	$d_{\mathrm{reg}}^{\mathrm{sg}}$	d_{reg}	Time (in sec)
8	16	240	6	4	1
10	25	600	6	4	1
12	36	1344	7	5	38
14	49	2744	8	5	3960

Table 3.4: Behaviour of a Gröbner basis computation of the ideal associated to the system $\mathrm{Sys}_{\mathrm{HSP}_2}$ for degree-4 instances of the HSP_2

Remark 3.15 *Recall that* Proposition 2.44 *establishes that the degree of regularity of a semi-regular system with degree, size and number of variables following the notation in* Table 3.3 *is computed as the index of the first non-positive coefficient of the series*

$$\frac{\prod_{i=1}^{U_{eqs}}(1-z^d)\prod_{i=1}^{N}(1-z^2)}{(1-z)^N} = \sum_{i\geq 0} c_i z^i. \tag{3.14}$$

If the system corresponding to the first row of Table 3.3 *was semi-regular, its degree of regularity would be computed by substituting the values* $n = 8, d = 3$, $U_{eqs} = 224, N = 16$ *in the expression* (3.14)*, which equals*

$$\frac{(1-z^3)^{224}(1-z^2)^{16}}{(1-z)^{16}} = 1 + 16z + 120z^2 + 336z^3 - 1764z^4 - \dots$$

and so the degree of regularity of such a semi-regular system would be 4*, as written in the table. Analogously computed for the rest of the cases.*

The first thing we observe in Tables 3.3 and 3.4 is that the number of equations of our system coincides with the upper bound $\mathrm{U}_{\mathrm{eqs}}$, this is, that the system is as overdetermined as it can be. This occurs because as we said in Remark 3.14 every equation of our model gets many contributions (from coefficients of monomials that get reduced by the field equations) and the probability that all these coefficients are zero tends to zero as the parameter n increases.

The next thing we observe is that solving the systems derived from our instances of the HSP_2 is easier than if they were actually semi-regular: the degree of regularity observed in practice is lower than the expected one, which suggests that there is an underlying structure in $\mathrm{Sys}_{\mathrm{HSP}_2}$. Not only the observed degree of regularity is lower than the expected one, but it also seems to increase less rapidly and stay bounded. This is precisely our conjecture.

Conjecture 3.16 *Given a degree d-instance $(\mathbf{p}, \mathbf{q}) \in \mathbb{F}_2[\mathbf{x}]^m \times \mathbb{F}_2[\mathbf{x}]^m$ of the* HSP_2*, the degree of regularity of our model* $\mathrm{Sys}_{\mathrm{HSP}_2}$ *is bounded above by $d + 1$.*

If our conjecture is true, it yields a heuristic randomised polynomial-time algorithm that solves the HSP_2.

Theorem 3.17 *Assuming* Conjecture 3.16*, there is a randomised polynomial-time algorithm, consisting in the computation of a Gröbner basis, solving the* HSP_2 *with complexity*

$$\mathcal{O}\left(n^{2\omega(d+1)}\right),$$

where ω is the exponent of matrix multiplication, and success probability

$$\gamma_2(n/2).$$

So far our experiments indicate that the system $\mathrm{Sys}_{\mathrm{HSP}_2}$ is indeed not semi-regular. In what follows we give theoretical results that further support our conjecture.

3.3.3 Theoretic results supporting our heuristic algorithm: degree falls

The goal of this section is to provide theoretical arguments that support the fact that the system of equations $\mathrm{Sys}_{\mathrm{HSP}_2}$ is not semi-regular and thus our Conjecture 3.16. Our main achievement is proving that linear combinations of certain equations of degree d in $\mathrm{Sys}_{\mathrm{HSP}_2}$ lead to equations of degree lower than d.

During a Gröbner basis computation, when a combination of equations yields an equation of a lower degree than that of the equations combined we say that there is a degree fall. This is typically a behaviour which does not occur in a semi-regular system of equations and so it is a first step towards proving Conjecture 3.16. Even more, combining linearly the equations of $\mathrm{Sys}_{\mathrm{HSP}_2}$ is actually the first computation carried out by an algorithm aiming at computing a Gröbner basis, and as a consequence of our finding, solving the system of degree d-equations $\mathrm{Sys}_{\mathrm{HSP}_2}$ presents a degree fall in a single step.

Let $(\mathbf{p}, \mathbf{q}) \in \mathbb{F}_2[\mathbf{x}]^m \times \mathbb{F}_2[\mathbf{x}]^m$ be a degree-d instance of the HSP_2 and let us consider a pair formed by one polynomial of $\mathbf{p}$ and one polynomial of $\mathbf{q}$, which we denote simply by (p, q), ignoring subindices to avoid an overcomplication of the notation. It turns out that from a pair (p, q) it is possible to obtain an equation of degree lower than d. Let us set

$$N_d = \binom{n/2}{d}$$

and let us order lexicographically the monomials of degree d in the sets

$$\mathrm{M}_d\left(\mathbb{F}_2\left[y_1,\ldots,y_{n/2}\right]\right),\mathrm{M}_d\left(\mathbb{F}_2\left[x_{n/2+1},\ldots,x_n\right]\right),\mathrm{M}_d\left(\mathbb{F}_2\left[x_1,\ldots,x_{n/2}\right]\right),$$

obtaining, respectively, the ordered sequences

$$\begin{aligned} t_1 &\prec_{\text{lex}} \ldots \prec_{\text{lex}} t_{N_d}, \\ m_1 &\prec_{\text{lex}} \ldots \prec_{\text{lex}} m_{N_d}, \\ m_1^{\perp} &\prec_{\text{lex}} \ldots \prec_{\text{lex}} m_{N_d}^{\perp}. \end{aligned}$$

We can express the pair (p,q) in the following manner.

$$\begin{cases} p &= \alpha_1 m_1 + \ldots + \alpha_{N_d} m_{N_d} + p', \quad \text{with } \alpha_1,\ldots,\alpha_{N_d} \in \mathbb{F}_2, \\ q &= \beta_1 m_1^{\perp} + \ldots + \beta_{N_d} m_{N_d}^{\perp} + q', \quad \text{with } \beta_1,\ldots,\beta_{N_d} \in \mathbb{F}_2, \end{cases} \tag{3.15}$$

where p' is thus a polynomial in $\mathbb{F}_2\left[\mathbf{x}\right]$ that does not have any term with a monomial in $\mathrm{M}_d\left(\mathbb{F}_2[x_{n/2+1},\ldots,x_n]\right)$ and q' is a polynomial in $\mathbb{F}_2\left[\mathbf{x}\right]$ that does not have any term with a monomial in $\mathrm{M}_d\left(\mathbb{F}_2[x_1,\ldots,x_{n/2}]\right)$.

As a consequence, the only monomials of degree d that can be present in p' are divisible by x_i for some $i \in \{1,\ldots,n/2\}$. Analogously, the only monomials of degree d that can be present in q' are divisible by x_j for some $j \in \{n/2+1,\ldots,n\}$, which implies that

$$\mathrm{Coeff}(p',t)^{(d)} = 0 = \mathrm{Coeff}(q',t)^{(d)}, \quad \forall t \in \mathrm{M}\left(\mathbb{F}_2[y_1,\ldots,y_{n/2}]\right).$$

To see why, recall that $\mathrm{Coeff}(p',t)$ and $\mathrm{Coeff}(q',t)$ are the coefficients of the monomial $t \in \mathrm{M}(\mathbb{F}[y_1,\ldots,y_{n/2}])$ in the expressions

$$\begin{cases} p'\left(\left(y_1,\ldots,y_{n/2}\right)(I|G)\right) = p'\left(y_1,\ldots,y_{n/2},\sum\limits_{k=1}^{n/2} g_{k,1}y_k,\ldots,\sum\limits_{k=1}^{n/2} g_{k,n/2}y_k\right), \\ q'\left(\left(y_1,\ldots,y_{n/2}\right)\left(G^T|I\right)\right) = q'\left(\sum\limits_{k=1}^{n/2} g_{1,k}y_k,\ldots,\sum\limits_{k=1}^{n/2} g_{n/2,k}y_k,y_1,\ldots,y_{n/2}\right), \end{cases}$$

for $1 \le i \le m$, and since p' has only monomials in $\mathbb{F}[\mathbf{x}]$ of degree d that are divisible by

$$x_i, \quad \text{for some } i \in \{1,\ldots,n/2\},$$

the equation $\mathrm{Coeff}(p',t)$ is of degree at most $d-1$ due to the i-th component of the vector $(y_1,\ldots,y_{n/2})(I|G)$ having no dependency on the unknowns of G. Therefore, the homogeneous component of degree d of $\mathrm{Coeff}(p',t)$ is zero as stated, and analogously for the homogeneous component of degree d of $\mathrm{Coeff}(q',t)$.

Therefore, we can write

$$\begin{aligned} \mathrm{Coeff}(p,t)^{(d)} &= \alpha_1 \mathrm{Coeff}(m_1,t)^{(d)} + \ldots + \alpha_{N_d}\mathrm{Coeff}(m_{N_d},t)^{(d)}, \\ \mathrm{Coeff}(q,t)^{(d)} &= \beta_1 \mathrm{Coeff}(m_1^{\perp},t)^{(d)} + \ldots + \beta_{N_d}\mathrm{Coeff}(m_{N_d}^{\perp},t)^{(d)}. \end{aligned}$$

We will see that the fact that G and G^T have the same entries in different positions will produce relations between the homogeneous components of degree d of the equations in $\mathrm{Sys}_{\mathrm{HSP}_2}$, which will ultimately allow us to combine expressions of our system in a certain manner such that the homogeneous component of degree d of such a combination gets cancelled out and so the combination is of degree at most $d-1$. The appropriate way to combine equations of $\mathrm{Sys}_{\mathrm{HSP}_2}$ to obtain equations of degree lower than d is inferred from the following result, which shows that the structure of $\mathrm{Sys}_{\mathrm{HSP}_2}$ is indeed very particular.

Proposition 3.18 *Let $(\mathbf{p},\mathbf{q}) \in \mathbb{F}_2[\mathbf{x}]^m \times \mathbb{F}_2[\mathbf{x}]$ be a degree-d instance of the* HSP_2*. Let $(p,q) \in \mathbb{F}_2[\mathbf{x}] \times \mathbb{F}_2[\mathbf{x}]$ be a pair in the set*

$$\{(p_i, q_j)\colon 1 \le i \le m, 1 \le j \le m\}$$

and set $N_d = \binom{n/2}{d}$. Let us order lexicographically the monomials of degree d in the sets

$$\mathrm{M}_d\left(\mathbb{F}_2\left[y_1,\ldots,y_{n/2}\right]\right), \mathrm{M}_d\left(\mathbb{F}_2\left[x_{n/2+1},\ldots,x_n\right]\right), \mathrm{M}_d\left(\mathbb{F}_2\left[x_1,\ldots,x_{n/2}\right]\right),$$

obtaining, respectively, the ordered sequences

$$\begin{aligned} t_1 &\prec_{lex} \ldots \prec_{lex} t_{N_d}, \\ m_1 &\prec_{lex} \ldots \prec_{lex} m_{N_d}, \\ m_1^{\perp} &\prec_{lex} \ldots \prec_{lex} m_{N_d}^{\perp}. \end{aligned}$$

For all $i,j \in \{1,\ldots,N_d\}$, it holds that:

$$\mathrm{Coeff}\left(m_i, t_j\right)^{(d)} = \mathrm{Coeff}\left(m_j^{\perp}, t_i\right)^{(d)}. \tag{3.16}$$

Proof. Let m_i be a degree-d monomial in $\mathbb{F}_2[x_{n/2+1},\ldots,x_n]$ and let t_j be a degree-d monomial in $\mathbb{F}_2[y_1,\ldots,y_{n/2}]$ for some $i,j \in \{1,\ldots,N_d\}$. The monomials m_i and t_j can be expressed as

$$\begin{aligned} m_i &= x_{i_1+n/2}x_{i_2+n/2}\cdots x_{i_d+n/2}, \quad \text{for some } i_1,\ldots,i_d \in \{1,\ldots,n/2\}, \\ t_j &= y_{j_1}y_{j_2}\cdots y_{j_d}, \quad \text{for some } j_1,\ldots,j_d \in \{1,\ldots,n/2\}. \end{aligned}$$

Let us focus on the left-hand side of the equation (3.16). It holds that:

$$\mathrm{Coeff}(m_i,t_j)^{(d)} = \mathrm{Coeff}\left(\prod_{k=1}^{d} x_{i_k+n/2}, y_{j_1}y_{j_2}\cdots y_{j_d}\right)^{(d)}.$$

Observing the system

$$\begin{cases} p\left(\left(y_1,\ldots,y_{n/2}\right)(I|G)\right) = p\left(y_1,\ldots,y_{n/2},\sum\limits_{t=1}^{n/2} g_{t,1}y_t,\ldots,\sum\limits_{t=1}^{n/2} g_{t,n/2}y_t\right), \\ q\left(\left(y_1,\ldots,y_{n/2}\right)\left(G^T|I\right)\right) = q\left(\sum\limits_{t=1}^{n/2} g_{1,t}y_t,\ldots,\sum\limits_{t=1}^{n/2} g_{n/2,t}y_t, y_1,\ldots,y_{n/2}\right), \end{cases} \tag{3.17}$$

for $1 \leq i \leq m$, it holds that the expression

$$\mathrm{Coeff}\left(\prod_{k=1}^{d} x_{i_k+n/2}, y_{j_1}y_{j_2}\dots y_{j_d}\right)^{(d)}$$

coincides with the degree-d homogeneous component of the coefficient of $y_{j_1}y_{j_2}\cdots y_{j_d}$ in the expression

$$\prod_{k=1}^{d}\sum_{\ell=1}^{n/2} g_{\ell,i_k}y_\ell,$$

which in turn coincides with the homogeneous component of degree d of the coefficient of $y_{j_2}\dots y_{j_d}$ in the expression

$$\prod_{k=1}^{d}\sum_{\ell=1}^{d} g_{j_\ell,i_k}y_{j_\ell}.$$

After expanding the latter expression, we finally obtain that

$$\mathrm{Coeff}(m_i, t_j)^{(d)} = \sum_{\pi\in S_d}\prod_{k=1}^{d} g_{j_{\pi(k)},i_k}. \tag{3.18}$$

Let us focus now on the right-hand side of the equation (3.16). Note that the rings $\mathbb{F}_2[y_1,\dots,y_{n/2}]$ and $\mathbb{F}_2[x_1,\dots,x_{n/2}]$ are related through a bijection $\phi : y_i \mapsto x_i$, so the j-th term in the ordered sequence

$$m_1^\perp \prec_{\text{lex}} \dots \prec_{\text{lex}} m_{N_d}^\perp$$

is the image through ϕ of the j-th element of the ordered sequence

$$t_1 \prec_{\text{lex}} \dots \prec_{\text{lex}} t_{N_d},$$

this is,

$$m_j^\perp = x_{j_1}x_{j_2}\dots x_{j_d}.$$

Analogously, the rings and $\mathbb{F}_2[x_{n/2+1},\dots,x_n]$ and $\mathbb{F}_2[y_1,\dots,y_{n/2}]$ are related through a bijection $\phi : x_i \mapsto x_{i-n/2}$, and so the i-th monomial in the ordered sequence

$$t_1 \prec_{\text{lex}} \dots \prec_{\text{lex}} t_{N_d}$$

coincides with the image through ϕ of the i-th monomial in the ordered sequence

$$m_1 \prec_{\text{lex}} \dots \prec_{\text{lex}} m_{N_d},$$

and so

$$t_i = y_{i_1}y_{i_2}\dots y_{i_d}.$$

Now it occurs that

$$\mathrm{Coeff}(m_j^\perp, t_i)^{(d)} = \mathrm{Coeff}\left(\prod_{k=1}^{d} x_{j_k}, y_{i_1}y_{i_2}\dots y_{i_d}\right)^{(d)}.$$

Observing the system (3.17), it holds that

$$\mathrm{Coeff}\left(\prod_{k=1}^{d} x_{j_k}, y_{i_1}y_{i_2}\ldots y_{i_d}\right)^{(d)}$$

coincides with the degree-d homogeneous component of the coefficient of $y_{i_1}y_{i_2}\cdots y_{i_d}$ in the expression

$$\prod_{k=1}^{d}\sum_{\ell=1}^{n/2} g_{j_k,\ell}y_\ell,$$

which in turn coincides with the homogeneous component of degree d of the coefficient of $y_{i_1}y_{i_2}\ldots y_{i_d}$ in the expression

$$\prod_{k=1}^{d}\sum_{\ell=1}^{d} g_{j_k,i_\ell}y_{i_\ell}.$$

After expanding the latter expression, we finally obtain that

$$\sum_{\pi\in S_d}\prod_{k=1}^{d} g_{j_k,i_{\pi(k)}}. \tag{3.19}$$

Finally, expressions (3.18) and (3.19) coincide since

$$\sum_{\pi\in S_d}\prod_{k=1}^{d} g_{j_k,i_{\pi(k)}} = \sum_{\pi\in S_d}\prod_{k=1}^{d} g_{j_{\pi^{-1}(\pi(k))}}, i_{\pi(k)} = \sum_{\pi^{-1}\in S_d}\prod_{k'=1}^{d} g_{j_{\pi^{-1}(k')}}, i_{k'}.$$

■

The following theorem allows us to exploit the structural symmetries that the system Sys$_{\mathrm{HSP}_2}$ presents and thus generate equations of degree lower than d from appropriately selected linear combinations of equations in the system.

Theorem 3.19 *Taking into account the notation of* Proposition 3.18, *for a given* $t \in \mathrm{M}\left(\mathbb{F}[y_1,\ldots,y_{n/2}]\right)$, *bear in mind that*

$$\begin{aligned}\mathrm{Coeff}(p,t)^{(d)} &= \alpha_1\mathrm{Coeff}(m_1,t)^{(d)} + \ldots + \alpha_{N_d}\mathrm{Coeff}(m_{N_d},t)^{(d)},\\ \mathrm{Coeff}(q,t)^{(d)} &= \beta_1\mathrm{Coeff}(m_1^\perp,t)^{(d)} + \ldots + \beta_{N_d}\mathrm{Coeff}(m_{N_d}^\perp,t)^{(d)}.\end{aligned}$$

Under these conditions, there exist indices $i,j\in\{1,\ldots,N_d\}$ *such that the equation*

$$\mathrm{Coeff}(p,t_j)+\mathrm{Coeff}(q,t_i)+\sum_{k\neq i}\mathrm{Coeff}(q,t_k)+\sum_{\ell\neq j}\mathrm{Coeff}(p,t_\ell)=0$$

is of degree lower than d.

Proof. Denote by $i, j \in \{1, \ldots, N_d\}$ the smallest indices such that $\alpha_i \neq 0$, $\beta_j \neq 0$. Using Proposition 3.18, it holds that

$$\operatorname{Coeff}(m_i, t_j)^{(d)} = \operatorname{Coeff}(m_j^{\perp}, t_i)^{(d)}. \tag{3.20}$$

For every $k \neq i$ and for every $\ell \neq j$, we can use Proposition 3.18 to establish some more equalities:

$$\operatorname{Coeff}(m_k, t_j)^{(d)} = \operatorname{Coeff}(m_j^{\perp}, t_k)^{(d)}, \tag{3.21}$$

$$\operatorname{Coeff}(m_\ell^{\perp}, t_i)^{(d)} = \operatorname{Coeff}(m_i, t_\ell)^{(d)}, \tag{3.22}$$

$$\operatorname{Coeff}(m_\ell^{\perp}, t_k)^{(d)} = \operatorname{Coeff}(m_k, t_\ell)^{(d)}. \tag{3.23}$$

The following relations hold:

$$\begin{aligned}
\operatorname{Coeff}(m_i, t_j)^{(d)} + \sum_{k \neq i} \operatorname{Coeff}(m_k, t_j)^{(d)} &= \operatorname{Coeff}(p, t_j)^{(d)}, \\
\operatorname{Coeff}(m_j^{\perp}, t_i)^{(d)} + \sum_{\ell \neq j} \operatorname{Coeff}(m_\ell^{\perp}, t_i)^{(d)} &= \operatorname{Coeff}(q, t_i)^{(d)}, \\
\sum_{k \neq i} \operatorname{Coeff}(m_j^{\perp}, t_k)^{(d)} + \sum_{\substack{\ell \neq j, \\ k \neq i}} \operatorname{Coeff}(m_\ell^{\perp}, t_k)^{(d)} &= \sum_{k \neq i} \operatorname{Coeff}(q, t_k)^{(d)}, \\
\sum_{\ell \neq j} \operatorname{Coeff}(m_i, t_\ell)^{(d)} + \sum_{\substack{k \neq i \\ \ell \neq j}} \operatorname{Coeff}(m_k, t_\ell)^{(d)} &= \sum_{\ell \neq j} \operatorname{Coeff}(p, t_\ell)^{(d)}.
\end{aligned}$$

Now taking into account the equalities (3.20), (3.21), (3.22), (3.23), the expression obtained by adding up the left-hand side of all the equalities above equals zero, which means that

$$\operatorname{Coeff}(p, t_j) + \operatorname{Coeff}(q, t_i) + \sum_{k \neq i} \operatorname{Coeff}(q, t_k) + \sum_{\ell \neq j} \operatorname{Coeff}(p, t_\ell) = 0,$$

as required. ∎

This result can be immediately used to generate low-degree equations from the equations in the system $\mathrm{Sys}_{\mathrm{HSP}_2}$.

Corollary 3.20 *Let* $(\mathbf{p}, \mathbf{q}) \in \mathbb{F}_2[\mathbf{x}]^m \times \mathbb{F}_2[\mathbf{x}]^m$ *be a degree-d instance of the* HSP_2. *We can generate* $\mathcal{O}(m^2)$ *equations of degree lower than d, which are linear combinations of the degree-d equations of* $\mathrm{Sys}_{\mathrm{HSP}_2}$.

Proof. We apply simply Theorem 3.19 to each pair of polynomials $(p_i, q_j) \in \mathbb{F}_2[\mathbf{x}] \times \mathbb{F}_2[\mathbf{x}]$. From the proof of Theorem 3.19, it is clear that these equations are linear combinations of the equations from $\mathrm{Sys}_{\mathrm{HSP}_2}$. ∎

Before moving on to present some experimental results, it is important to remark that our algorithm of Theorem 3.17 solves the HSP_2 in heuristic randomised polynomial time for degree-d instances, thus yielding in practice a cryptanalysis of Aaronson-Christiano's quantum money scheme.

3.3.4 Experiments

Since there are equations of degree lower than d in $\mathrm{Sys}_{\mathrm{HSP}_2}$, it is natural to wonder whether it is algebraically meaningful or not, which translates into studying whether the equations of degree lower than d are linearly independent. To conclude this chapter, in Tables 3.5 and 3.6 we report experimental results on the number of equations of degree lower than d generated from the equations of $\mathrm{Sys}_{\mathrm{HSP}_2}$ (as in Corollary 3.20) which happen to be linearly independent. In the tables, we denote by $\#\mathrm{eqs}_{\mathrm{pr}}$ the number of linearly independent equations obtained in practice and by $\#\mathrm{eqs}_{\mathrm{th}}$ the maximum number of linearly independent equations that can be obtained, which is at most the total number of equations of degree lower than d generated, i.e., m^2.

	$d = 3$	
	$\#\mathrm{eqs}_{\mathrm{pr}}$	$\#\mathrm{eqs}_{\mathrm{th}}$
$m = n = 10$	99	100
$m = n = 12$	144	144
$m = n = 14$	196	196
$m = n = 16$	256	256

Table 3.5: Degree-2 linearly independent equations obtained from degree-3 instances of the HSP$_2$

	$d = 4$	
	$\#\mathrm{eqs}_{\mathrm{pr}}$	$\#\mathrm{eqs}_{\mathrm{th}}$
$m = n = 10$	71	100
$m = n = 12$	144	144
$m = n = 14$	196	196
$m = n = 16$	256	256

Table 3.6: Degree-3 linearly independent equations of obtained from degree-4 instances of the HSP$_2$

We observe that the behaviour is unstable for small values of the parameters. This is partially due to the fact that if a polynomial p_i does not have terms of degree d in $\mathbb{F}_2[x_{n/2+1}, \ldots, x_n]$, then we do not get equations of degree lower than d applying Theorem 3.19 to the pair (p_i, q_k) for all $k \in \{1, \ldots, m\}$ (and analogously for q_i). This happens with probability

$$\frac{1}{2^{N_d}},$$

which is not too small for low parameters, i.e, if N_d is low. So, for small parameters it is possible that we obtain a number of equations of degree lower than d which is less than m^2. However, if this is the case there are equations of degree lower than d anyway, derived from the terms of degree $d-1$ or lower. We see that the behaviour stabilises for big enough values of the parameters m, n obtaining as many equations of degree lower than d as possible.

Chapter 4

Cryptanalysis of Aaronson and Christiano's scheme: The noisy case

This chapter is dedicated to analyse the hardness of the hidden subspaces problem with noise, proposed by Scott Aaronson and Paul Christiano at STOC$'$12 aiming at enhancing the hardness of the hidden subspaces problem.

We design a randomised polynomial-time algorithm that solves degree-d instances of the noisy hidden subspaces problem over fields $\mathbb{F}$ of prime size that satisfy the condition $|\mathbb{F}| > d$, and we report experimental results on its performance that confirm that it is efficient in practice. Over the field $\mathbb{F}_2$ we present a probabilistic algorithm that combines exhaustive search and our algorithm from Chapter 3, yielding a probability of success that exceeds the one conjectured by its authors.

Since the security of the noisy version of the public-key quantum money scheme of Aaronson-Christiano relies on the hardness of the noisy hidden subspaces problem, our results yield a cryptanalysis of any extension of the scheme to a field $\mathbb{F}$ for degree-d instances satisfying the condition $|\mathbb{F}| > d$. Over the field $\mathbb{F}_2$ we break a conjecture made by the authors about the hardness of the noisy hidden subspaces problem.

4.1 The $\mathrm{NHSP}_{|\mathbb{F}|}$ for $|\mathbb{F}| > d$

In Chapter 3 we divided the study of the hardness of the hidden subspaces problem into two different scenarios, so it seems only sensible to divide the study of the noisy version of the former problem into the same cases. We dedicate this section to analyse the hardness of degree-d instances of the $\mathrm{NHSP}_{|\mathbb{F}|}$ whenever $|\mathbb{F}| > d$, so this condition over the field $\mathbb{F}$ is assumed hereafter although not explicitly stated.

First we present an algorithm that solves degree-d instances of the $\mathrm{NHSP}_{|\mathbb{F}|}$, which we derived from an algorithm that solves only instances of the NHSP_2 that are linear. Our algorithm runs in polynomial time and it is randomised, succeeding with an overwhelming probability. Second we report experimental results that support our theoretic findings, this is, they experimentally verify that our algorithm is very efficient in practice and that its probability of success is overwhelming.

4.1.1 Our algorithm solving the $\mathrm{NHSP}_{|\mathbb{F}|}$ for $|\mathbb{F}| > d$

Our algorithm solving degree-d instances of the $\mathrm{NHSP}_{|\mathbb{F}|}$ is based upon a simple algorithm that solves just linear instances of the NHSP_2. The latter algorithm was mentioned by the authors themselves in [Aaronson and Christiano, 2013] and stated over the field $\mathbb{F}_2$, but it can be extended to work over any other field. This result was supposed to be a mere remark on why they were choosing the instances underlying their scheme to be non-linear. However, it turns out that it can be extended to build an algorithm that solves instances of the $\mathrm{NHSP}_{|\mathbb{F}|}$ of degree higher than one, and more in particular of degree less than the cardinality of the field $\mathbb{F}$.

Let us first describe the algorithm that solves linear instances of the NHSP_2 and point out that the size of the field does not affect its validity. The result concerning the algorithm that solves linear instances of the NHSP_2 is stated in the paper of the authors as follows.

Lemma 4.1 [Aaronson and Christiano, 2013, Claim 6.10] *Let us consider a degree-*1 *instance of the* NHSP_2*,* $(\mathbf{p},\mathbf{q}) \in \mathbb{F}_2[\mathbf{x}]^m \times \mathbb{F}_2[\mathbf{x}]^m$*, such that the subspace* $A \subset \mathbb{F}^n$ *is a solution of it. Then, there exists an algorithm that recovers* A *in randomised polynomial time.*

To see why there exists an algorithm such as the one in Lemma 4.1, the key observation is the following. If $(\mathbf{p},\mathbf{q}) \in \mathbb{F}_2[\mathbf{x}]^m \times \mathbb{F}_2[\mathbf{x}]^m$ is a degree-1 instance of the NHSP_2, then it occurs that for any $i \in \{1,\dots,m\}$,

$$q_i \text{ vanishes on } A^\perp \iff q_i(\mathbf{x}) = \lambda_i^{\mathbf{q}}\mathbf{x} \text{ for some } \lambda_i^{\mathbf{q}} \in A. \tag{4.1}$$

Note that this characterisation is inferred from the orthogonality relation between A and $A^\perp$. Indeed, if for a given $i \in \{1,\dots,m\}$ we express the linear polynomial q_i as

$$q_i(\mathbf{x}) = \sum_{j=1}^{n} \lambda_{i,j}^{\mathbf{q}} x_i = \lambda_i^{\mathbf{q}}\mathbf{x}, \quad \text{where } \lambda_i^{\mathbf{q}} = (\lambda_{i,1}^{\mathbf{q}},\dots,\lambda_{i,n}^{\mathbf{q}}) \in \mathbb{F}^n,$$

then by definition of orthogonality it occurs that

$$\left(A^{\perp}\right)^{\perp} = \{\mathbf{a} \in \mathbb{F}^n \colon \mathbf{a} \cdot \mathbf{a}^{\perp} = 0, \quad \forall \mathbf{a}^{\perp} \in A^{\perp}\} = A,$$

which translates precisely into the condition (4.1).

Remark 4.2 *The analogous property*

$$p_i \text{ vanishes on } A \iff p_i\left(\mathbf{x}\right) = \lambda_i^{\mathbf{p}}\mathbf{x} \text{ for some } \lambda_i^{\mathbf{p}} \in A^{\perp} \tag{4.2}$$

holds, but for simplicity of exposition we focus on the characterisation (4.1)*.*

Also note that the property (4.1) *holds over any field* $\mathbb{F}$*, as it is just a consequence of the orthogonality relation between* A *and* $A^{\perp}$*.*

Now the condition (4.1) guarantees that for each polynomial q_i that truly vanishes on $A^{\perp}$, the element $\lambda_i^{\mathbf{q}}$ belongs to A. Therefore, among the set

$$\{\lambda_i^{\mathbf{q}} \in \mathbb{F}^n \colon q_i\left(\mathbf{x}\right) = \lambda_i^{\mathbf{q}}\mathbf{x}, \quad i \in \{1, \ldots, m\}\} \tag{4.3}$$

there must be $\lceil(1-\varepsilon)m\rceil$ elements, as many as the number of non-noisy polynomials of $\mathbf{q}$, that belong to A. The remaining εm elements, as many as the number of noisy polynomials of $\mathbf{q}$, do not belong to A except with a negligible probability.

Remark 4.3 *Recall that every noisy polynomial is chosen such that it vanishes on a* $n/2$*-dimensional subspace that is different from* $A^{\perp}$*. The probability that it still accidentally vanishes on* A *coincides with the probability that it vanishes on every element of one of its basis, this is,*

$$\frac{1}{|\mathbb{F}|^{n/2}}.$$

Therefore, if we could distinguish whether a given scalar in the set (4.3) belonged to A or not, then that would yield an algorithm to learn at most $\lceil(1-\varepsilon)m\rceil$ elements that belong to A.

Remark 4.4 *Note that the set* (4.3) *contains as many non-zero elements as the number of polynomials of* $\mathbf{q}$ *whose linear homogeneous components are not zero, so there are at most* $\lceil(1-\varepsilon)m\rceil$ *elements of* A *in the set* (4.3)*.*

It turns out that deciding whether a given element of $\mathbb{F}^n$ belongs to A or not can be done efficiently. To explain how, let us introduce a couple of definitions that will avoid overcomplicated notation later on.

Definition 4.5 *Let* $(\mathbf{p}, \mathbf{q}) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ *be a degree-*d *instance of the* NHSP$_{|\mathbb{F}|}$*. We define the weight of a vector* $v \in \mathbb{F}^n$ *with respect to* $\mathbf{p}$ *(resp.* $\mathbf{q}$*), denoted by* $w_{|\mathbb{F}|}^{\mathbf{p}}\left(v\right)$ *(resp.* $w_{|\mathbb{F}|}^{\mathbf{q}}\left(v\right)$*), as the cardinal of the set*

$$W_v^{\mathbf{p}} = \{p_i : p_i\left(v\right) \neq 0,\, i = 1, \ldots, m\} \quad (\text{resp.}\, W_v^{\mathbf{q}} = \{q_i : q_i\left(v\right) \neq 0,\, i = 1, \ldots, m\})$$

this is, $w_{|\mathbb{F}|}^{\mathbf{p}}\left(v\right) = |W_v^{\mathbf{p}}|$ *(resp.* $w_{|\mathbb{F}|}^{\mathbf{q}}\left(v\right) = |W_v^{\mathbf{q}}|$*).*

Definition 4.6 *Let* $(\mathbf{p}, \mathbf{q}) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ *be a degree-d instance of the* $\mathrm{NHSP}_{|\mathbb{F}|}$*. We define the set* $Z^{\mathbf{p}}_{|\mathbb{F}|} \subset \mathbb{F}^n$ *(resp.* $Z^{\mathbf{q}}_{|\mathbb{F}|} \subset \mathbb{F}^n$*) as*

$$Z^{\mathbf{p}}_{|\mathbb{F}|} = \{v \in \mathbb{F}^n : w^{\mathbf{p}}_{|\mathbb{F}|}(v) < \varepsilon\beta n\} \quad (\textit{resp.}\, Z^{\mathbf{q}}_{|\mathbb{F}|} = \{v \in \mathbb{F}^n : w^{\mathbf{q}}_{|\mathbb{F}|}(v) < \varepsilon\beta n\}).$$

Bearing these definitions in mind, the following result essentially states that the polynomials of $\mathbf{p}$ can be used as an oracle to test membership in A.

Lemma 4.7 [Aaronson and Christiano, 2013, Lemma 6.4] *Let us consider a degree-d instance of the* NHSP_2*,* $(\mathbf{p}, \mathbf{q}) \in \mathbb{F}_2[\mathbf{x}]^m \times \mathbb{F}_2[\mathbf{x}]^m$*. Then,*

$$A \subseteq Z^{\mathbf{p}}_2 \textit{ and } Pr\left[A = Z^{\mathbf{p}}_2\right] = 1 - 2^{-\Omega(n)},$$

where $Pr[\cdot]$ *denotes the probability of an event.*

Remark 4.8 Lemma 4.7 *applies over a generic field* $\mathbb{F}$*, occurring in that case that*

$$A \subseteq Z^{\mathbf{p}}_{|\mathbb{F}|} \textit{ and } Pr\left[A = Z^{\mathbf{p}}_{|\mathbb{F}|}\right] = 1 - |\mathbb{F}|^{-\Omega(n)}$$

Indeed, Lemma 4.7 allows to test if a given element z belongs to A with overwhelming probability by verifying that the only polynomials of $\mathbf{p}$ that do not vanish on z are at most the noisy ones, which is equivalent to checking whether $z \in Z^{\mathbf{p}}_{|\mathbb{F}|}$.

Therefore, we are now in a position to recover all the elements of the set (4.3) that belong to A, this is, to recover at most $\lceil (1-\varepsilon)m \rceil$ elements of A. A lower bound of the probability that we can extract a basis of A from a set of $\lceil (1-\varepsilon)m \rceil$ elements of A can be computed using the following theorem.

Theorem 4.9 [Brent and McKay, 1987, Theorem 1.1] *Recall, to begin with, that the* $|\mathbb{F}|$*-rank of a matrix* $G \in \mathcal{M}_{j,k}(\mathbb{F})$ *is the greatest integer* ℓ *such that* G *has a* $\ell \times \ell$ *submatrix (not necessarily contiguous) whose determinant is non-zero. For integers* $n \geq 1$*,* $\Delta \geq 0$ *and* $0 \leq \delta \leq n$*, define*

$$P_{\Delta,\delta}(n, p)$$

to be the probability that a random $(n + \Delta) \times n$ *matrix over* $\mathbb{F}$ *has* $|\mathbb{F}|$*-rank* $n - \delta$*, and set* $P_{\Delta,0}(0, p) = 1$*. For an integer* k *and indeterminate* t*, set*

$$\Pi_k(t) = (1 - t)(1 - t^2) \ldots (1 - t^k).$$

It occurs that:

$$P_{\Delta,\delta}(n, |\mathbb{F}|) = \frac{1}{|\mathbb{F}|^{\delta(\delta+\Delta)}} \frac{\Pi_n(1/|\mathbb{F}|)}{\Pi_\delta(1/|\mathbb{F}|)\Pi_{n-\delta}(1/|\mathbb{F}|)} \frac{\Pi_{n+\Delta}(1/|\mathbb{F}|)}{\Pi_{\delta+\Delta}(1/|\mathbb{F}|)}.$$

As a consequence of Theorem 4.9, the probability that we can extract $n/2$ linearly independent elements among a collection of $\lceil (1-\varepsilon)m \rceil$ elements coincides with the probability that a certain $\lceil (1-\varepsilon)m \rceil \times n$ matrix over $\mathbb{F}$ has $|\mathbb{F}|$-rank $n/2$, which is greater than the probability that a certain $\lceil (1-\varepsilon)m \rceil \times n/2$ matrix has $|\mathbb{F}|$-rank $n/2$. Replacing in Theorem 4.9 (n by $n/2$, Δ by $\lceil (1-\varepsilon)m \rceil - n/2$ and δ by 0), the probability that a certain $\lceil (1-\varepsilon)m \rceil \times n/2$ matrix has $|\mathbb{F}|$-rank $n/2$ is

$$(4.4) \qquad \prod_{i=\lceil (1-\varepsilon)m \rceil - n/2+1}^{\lceil (1-\varepsilon)m \rceil} \left(1 - \frac{1}{|\mathbb{F}|^i}\right) = \frac{\gamma_{|\mathbb{F}|}\left(\lceil (1-\varepsilon)m \rceil\right)}{\gamma_{|\mathbb{F}|}\left(\lceil (1-\varepsilon)m \rceil - n/2\right)},$$

which is overwhelming for large values of n.

Therefore, the algorithm that solves a degree-1 instance $(\mathbf{p}, \mathbf{q}) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ of the NHSP$_{|\mathbb{F}|}$ consists in checking how many of the elements in the set (4.3) are in A (see Lemma 4.7), as we can then extract a basis of A with probability at least (4.4) provided that the set (4.3) has $\lceil (1-\varepsilon)m \rceil$ elements. Note that the algorithm runs in polynomial time, essentially requiring m computations of the weight of each element in the set (4.3) with respect to $\mathbf{p}$ and the computation of the $|\mathbb{F}|$-rank of a matrix.

Now, as we said in the beginning, it is possible to extend the algorithm solving linear instances of the NHSP$_{|\mathbb{F}|}$ to an algorithm that solves degree-d instances of the NHSP$_{|\mathbb{F}|}$ whenever $|\mathbb{F}| > d$. As it happened in Chapter 3, the condition $|\mathbb{F}| > d$ is very favourable for exactly the same reasons of no reductions modulo the field equations taking place. The following lemma shows that solving a degree-d instance of the NHSP$_{|\mathbb{F}|}$ can be reduced to solving a degree-1 instance of the same problem.

Lemma 4.10 *Whenever $|\mathbb{F}| > d$, any degree-d instance $(\mathbf{p}, \mathbf{q}) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ of the* NHSP$_{|\mathbb{F}|}$ *can be reduced to a degree-*1 *instance of the* NHSP$_{|\mathbb{F}|}$.

Proof. Let $(\mathbf{p}, \mathbf{q}) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ be a degree-d instance of the NHSP$_{|\mathbb{F}|}$. Recall that A is a solution of the instance $(\mathbf{p}, \mathbf{q})$ if

$$\begin{aligned} p_i(A) &= 0, \quad \forall i \in I_{\mathbf{p}}, \\ q_j(A^{\perp}) &= 0, \quad \forall j \in I_{\mathbf{q}}. \end{aligned}$$

Recall from the proof of Lemma 3.7 that for any degree-d polynomial $r \in \mathbb{F}[\mathbf{x}]$ vanishing on a $n/2$-dimensional subspace B we can write

$$r = r^{(d)} + r^{(d-1)} + \ldots + r^{(1)},$$

so if we denote an unknown basis matrix of B by $G \in \mathcal{M}_{n/2,n}(\mathbb{F})$ and we set $\mathbf{y} = (y_1, \ldots, y_{n/2}) \in \mathbb{F}^{n/2}$ to be a vector of formal variables as usual, it holds that

$$(4.5) \qquad r(\mathbf{y}G) = r^{(d)}(\mathbf{y}G) + r^{(d-1)}(\mathbf{y}G) + \ldots + r^{(1)}(\mathbf{y}G).$$

Since $|\mathbb{F}| > d$, no monomial of $\mathbb{F}[y_1, \ldots, y_{n/2}]$ reduces modulo the field equations

$$y_i^{|\mathbb{F}|} - y_i = 0, \quad i \in \{1, \ldots, n/2\}.$$

Therefore, given $1 \leq \deg \leq d$, the set of monomials of $\mathbb{F}[y_1, \ldots, y_{n/2}]$ that are present in the homogeneous component $r^{(\deg)}$ is disjoint to the set of monomials that are present in every other homogeneous component of r of a different degree. And now, since a polynomial over $\mathbb{F}[y_1, \dots, y_{n/2}]$ vanishes if and only if all its coefficients are zero, it must occur that

$$r^{(\deg)} = 0, \quad \forall \deg \in \{1, \dots, d\},$$

and in particular, that

$$r^{(1)} = 0.$$

As a consequence, if A is a solution of the degree-d instance $(\mathbf{p}, \mathbf{q})$ of the $\mathrm{NHSP}_{|\mathbb{F}|}$, then A is also a solution of the degree-1 instance

$$(\mathbf{p}^{(1)}, \mathbf{q}^{(1)}),$$

as required. ■

First reducing a degree-d instance $(\mathbf{p}, \mathbf{q})$ of the $\mathrm{NHSP}_{|\mathbb{F}|}$ to the degree-1 instance of the $\mathrm{NHSP}_{|\mathbb{F}|}$ as in Lemma 4.10 and then applying the algorithm of Lemma 4.1 that solves linear instances yields an algorithm that solves degree-d instances of the $\mathrm{NHSP}_{|\mathbb{F}|}$, as detailed in the following theorem.

Theorem 4.11 *Let $|\mathbb{F}| > d$. There is a randomised polynomial-time algorithm that solves the* $\mathrm{NHSP}_{|\mathbb{F}|}$ *with complexity*

$$\mathcal{O}\left(m^{\omega} d n^{d}\right)$$

and probability of success at least

$$\left(1 - \frac{1}{|\mathbb{F}|^{n}}\right)^{\lceil (1-\varepsilon)m \rceil} \frac{\gamma_{|\mathbb{F}|}\left(\lceil (1-\varepsilon)m \rceil\right)}{\gamma_{|\mathbb{F}|}\left(\lceil (1-\varepsilon)m \rceil - n/2\right)}$$

Proof. Let $(\mathbf{p}, \mathbf{q}) \in \mathbb{F}[\mathbf{x}]^{m} \times \mathbb{F}[\mathbf{x}]^{m}$ be a degree-d instance of the $\mathrm{NHSP}_{|\mathbb{F}|}$. Denoting the homogeneous components of degree 1 of $\mathbf{p}$ and $\mathbf{q}$ by

$$p_i^{(1)}(\mathbf{x}) = \lambda_i^{\mathbf{p}} \mathbf{x}, \quad q_i^{(1)}(\mathbf{x}) = \lambda_i^{\mathbf{q}} \mathbf{x}, \quad \text{where } \lambda_i^{\mathbf{p}}, \lambda_i^{\mathbf{q}} \in \mathbb{F}^{n},\ i \in \{1, \dots, m\},$$

the following algorithm solves the $\mathrm{NHSP}_{|\mathbb{F}|}$ as an application of Lemma 4.1 and Lemma 4.10:

```
Input: $(\mathbf{p}, \mathbf{q}) \in \mathbb{F}[\mathbf{x}]^{m} \times \mathbb{F}[\mathbf{x}]^{m}$ a degree-$d$ instance of the $\mathrm{NHSP}_{|\mathbb{F}|}$, with $|\mathbb{F}| > d$.
$E_A \leftarrow \emptyset$
for j=1 to m do
//We identify which scalars $\lambda_i^{\mathbf{q}}$ of the set (4.3) belong to $A$
   weight $\leftarrow w_{|\mathbb{F}|}^{\mathbf{p}}\left(\lambda_j^{\mathbf{q}}\right)$
   if $m -$ weight $\geq \lceil (1-\varepsilon) \rceil m$ then
      $E_A \leftarrow E_A \cup \{\lambda_j^{\mathbf{q}}\}$
   end if
end for
```

```
if $\dim(\mathrm{span}(E_A)) = n/2$ then
//We check if there are $n/2$ elements of $E_A$ that are linearly independent
   $A \leftarrow \mathrm{span}(E_A)$.
   return $A$
else
   print “The algorithm fails”
end if
```

The complexity of the algorithm is that of computing m weights with respect to each of the m polynomials of $\mathbf{p}$, with each weight involving m evaluations of a polynomial in $\mathbb{F}[\mathbf{x}]$, and that of computing a rank, which is

$$\mathcal{O}\left(m^2 dn^d\right) + \mathcal{O}\left(\lceil (1-\varepsilon)m \rceil^{\omega}\right) = \mathcal{O}\left(m^{\omega} dn^d\right)$$

Regarding the success probability, the algorithm above succeeds if among the $\lceil (1-\varepsilon)m \rceil$ elements from E_A there are $n/2$ linearly independent ones, this is, if the corresponding $\lceil (1-\varepsilon)m \rceil \times n$ matrix has rank $n/2$. The probability that a $\lceil (1-\varepsilon)m \rceil \times n$ matrix has rank $n/2$ is at least the probability that a certain $\lceil (1-\varepsilon)m \rceil \times n/2$ submatrix has rank $n/2$, which equals the expression obtained in (4.4), namely

$$\frac{\gamma_{|\mathbb{F}|}\left(\lceil (1-\varepsilon)m \rceil\right)}{\gamma_{|\mathbb{F}|}\left(\lceil (1-\varepsilon)m \rceil - n/2\right)}.$$

Furthermore, in order to succeed the algorithm also needs that in the instance $(\mathbf{p}, \mathbf{q})$, all the $\lceil (1-\varepsilon)m \rceil$ non-noisy polynomials of $\mathbf{q}$ have linear terms. Given a certain $i \in \{1, \ldots, m\}$, the probability that a non-noisy polynomial q_i has a linear term is $1 - 1/|\mathbb{F}|^n$ and it is independent from any other q_j (with $j \neq i$) having linear terms. Therefore, the event that the $\lceil (1-\varepsilon)m \rceil$ non-noisy polynomials of $\mathbf{q}$ have linear terms equals:

$$\left(1 - \frac{1}{|\mathbb{F}|^n}\right)^{\lceil (1-\varepsilon)m \rceil}$$

Therefore, the overall probability that the algorithm succeeds given any degree-d instance of the NHSP$_{|\mathbb{F}|}$ is at least:

$$\left(1 - \frac{1}{|\mathbb{F}|^n}\right)^{\lceil (1-\varepsilon)m \rceil} \frac{\gamma_{|\mathbb{F}|}\left(\lceil (1-\varepsilon)m \rceil\right)}{\gamma_{|\mathbb{F}|}\left(\lceil (1-\varepsilon)m \rceil - n/2\right)}$$

■

Remark 4.12 *Note that in the unlikely case that the algorithm above fails to recover a basis of A it is still possible to try to recover a basis of $A^{\perp}$ instead using the characterisation of* Remark 4.2 *involving the scalars $\lambda_i^{\mathbf{p}}$.*

This theorem gives a randomised polynomial-time algorithm solving degree-d instances of the NHSP$_{|\mathbb{F}|}$ whenever $|\mathbb{F}| > d$, which means that the noisy scheme of Aaronson and Christiano is not secure extended to fields $\mathbb{F}$ such that $|\mathbb{F}| > d$. Besides, the probability of success of our algorithm is overwhelming, so the break can be considered total. However, note that the attack depends on the polynomials having linear terms, so if the authors change the parameters of their scheme then our attack could be avoided. A possible way to do this would be by considering homogeneous instances of degree at least three.

4.1.2 Experimental results

In this section we report experimental results (see Tables 4.1 and 4.2) to complement our theoretical results of Theorem 4.11. All experiments were run on a 2.93 GHz Intel PC with 128 Gb of RAM with the MAGMA software (V2.20-10) [Bosma et al., 1997] and the MAGMA source code is written in Appendix A and available online on GitHub (`https://github.com/Marta-PhD/solving-HSP-NHSP`).

The experiments show that the algorithm that solves degree-d instances of the $\mathrm{NHSP}_{|\mathbb{F}|}$ behaves as expected: it is very efficient in practice and it succeeds with overwhelming probability.

In Table 4.1 we start by choosing the smallest degree possible ($d = 3$), the average value for the noise in order not to have a biased impression of speed ($\varepsilon = 0.25$) and as few public polynomials as possible to get the lowest possible success probability ($\beta = 3/(1-2\varepsilon)^2$). Furthermore, we perform experiments over fields $\mathbb{F}$ of very different sizes to observe any potential variation of speed.

	$d = 3,\ \ \varepsilon = 0.25,\ \ m = \lceil \beta n \rceil$							
	$\lvert\mathbb{F}\rvert = 5$				$\lvert\mathbb{F}\rvert = 2^{16}+1$			
n	10	12	14	20	10	12	14	20
Time (in sec.)	0.25	0.6	1.59	13.13	0.43	1.03	3.09	20.14

Table 4.1: Performance of our algorithm (Theorem 4.11) solving the $\mathrm{NHSP}_{|\mathbb{F}|}$ for $d = 3$

We can see that the algorithm is very fast and that increasing the size of the base field does not entail a significant decrease of the speed. However, note that a decrease of speed is occurring during the generation of degree-3 instances of the $\mathrm{NHSP}_{|\mathbb{F}|}$, which is due to the fact that the cost of multiplications over $\mathbb{F}$ increases as the cardinality of the field increases.

Next, we maintain the same choice of parameters only increasing the degree of the instance ($d = 4$):

	$d = 4,\ \ \varepsilon = 0.25,\ \ m = \lceil \beta n \rceil$					
	$\lvert\mathbb{F}\rvert = 5$			$\lvert\mathbb{F}\rvert = 2^{16}+1$		
n	10	12	14	10	12	14
Time (in sec.)	1.28	4.41	11.32	2.54	7.85	17.96

Table 4.2: Performance of our algorithm (Theorem 4.11) solving the $\mathrm{NHSP}_{|\mathbb{F}|}$ for $d = 4$

We can see that our algorithm experienced a slight decrease of speed with respect to the case $d = 3$ (note that the cost of evaluating p_i over a certain element increases with the degree) but it is still very fast. The decrease of speed when generating an instance over a field with $|\mathbb{F}| = 5$ and $|\mathbb{F}| = 2^{16} + 1$ is again noticeable.

Finally, our attack succeeded for all the instances we performed experiments on. This is what is expected from Theorem 4.11, since particularising the expression of the probability of success for the most disadvantageous parameters we performed experiments on ($d = 3$, $n = 10$ and $\varepsilon = 0.25$) we obtain

$$\frac{\gamma_{|\mathbb{F}|}\left(\lceil (1-0.25)120 \rceil\right)}{\gamma_{|\mathbb{F}|}\left(\lceil (1-0.25)120 \rceil - 5\right)} \approx 0.9\overset{(60}{\cdots}9033,$$

for the first factor of the expression, whereas the second factor amounts to

$$\left(1 - \frac{1}{5^{10}}\right)^{\lceil (1-0.25)120 \rceil} \approx 0.9\overset{(5}{\cdots}90784.$$

Undoubtedly, the probability of success —which is the product of both factors above— for the most disadvantageous parameters we chose is already overwhelming. Furthermore, both factors of the probability in Theorem 4.11 increase with n and $|\mathbb{F}|$, and so asymptotically the situation only improves.

4.2 The NHSP$_2$

This section is dedicated to study the hardness of the hidden subspaces problem with noise over $\mathbb{F}_2$, a particular case verifying the condition $|\mathbb{F}| \leq d$ and so not affected by the results we obtained in Section 4.1.1. The noisy hidden subspaces problem is originally defined over $\mathbb{F}_2$ in [Aaronson and Christiano, 2013], and thus the results we obtain here have a direct impact on the noisy quantum money scheme proposed by the authors.

As it happened in Chapter 3, the field $\mathbb{F}_2$ is very particular and the algorithm that works under the condition $|\mathbb{F}| > d$ does not apply (as in this case Lemma 4.10 is no longer true due to reductions modulo the field equations).

The algorithm we present runs in polynomial time and it is randomised, with its success probability being $\Omega(2^{-n/2})$ provided that the proportion of noise $\varepsilon \in (0, \varepsilon)$ lies within a certain range. Achieving this probability demonstrates that Conjecture 2.4, made by the authors of the scheme precisely claiming the contrary, is false. Even more, the conjecture states that no quantum algorithm exists succeeding with such probability either and ours is purely classical.

Our algorithm simply combines both an exhaustive search and the algorithm that solves HSP$_2$ (see Theorem 3.17) from Chapter 3. Given a degree-d instance $(\mathbf{p}, \mathbf{q}) \in \mathbb{F}_2[\mathbf{x}]^m \times \mathbb{F}_2[\mathbf{x}]^m$ of the NHSP$_2$, the idea is to choose n polynomials at random from $\mathbf{p}$, say $\{p_i\}_{\{i \in \mathcal{F}\}}$ with $|\mathcal{F}| = n$, and hoping that they happen to be non-noisy. If they are, the algorithm for the HSP$_2$ of Theorem 3.17 succeeds and if they are not, the algorithm for the HSP$_2$ fails and we repeat the process of choosing n polynomials from $\mathbf{p}$ at random.

The algorithm above succeeds if the randomly-chosen polynomials in the set $\{p_i\}_{\{i\in\mathcal{F}\}}$ chosen at random are non-noisy and if the algorithm for the HSP$_2$ succeeds for the instance $\{p_i\}_{\{i\in\mathcal{F}\}}$. The probability that n polynomials chosen at random from $\mathbf{p}$ are non-noisy is approximated by the following result.

Lemma 4.13 *Given a degree-d instance $(\mathbf{p},\mathbf{q}) \in \mathbb{F}_2[\mathbf{x}]^m \times \mathbb{F}_2[\mathbf{x}]^m$ of the* NHSP$_2$*, the probability of choosing n polynomials from $\mathbf{p}$ (analogously from $\mathbf{q}$) that are non-noisy is given by the quotient*

$$\frac{\binom{\lceil(1-\varepsilon)\beta n\rceil}{n}}{\binom{\lceil\beta n\rceil}{n}}.$$

Furthermore, the asymptotic expression of the above probability, which we denote by $P_n^{\varepsilon,\beta}$, is as follows:

$$P_n^{\varepsilon,\beta} = \left[\left(\frac{\beta-1}{\beta}\right)^{\beta-1}\left(1+\frac{1}{(\varepsilon-1)\,\beta}\right)^{(\varepsilon-1)\beta}\left(1-\varepsilon-\frac{1}{\beta}\right)\right]^{n}.$$

Proof. We rely on the following asymptotic approximation of a binomial coefficient,

$$\log_2\binom{b}{a} \approx bH_2\left(\frac{a}{b}\right), \quad a,b\in\mathbb{N},$$

which derives from Stirling's approximation

$$\log n! \approx n\log,$$

where

$$H_2(x) = -x\log_2 x - (1-x)\log_2(1-x)$$

is the binary entropy. By means of these approximations, we get the expression above. ■

As the algorithm solving HSP$_2$ is randomised, we also need to take into consideration its probability of success to determine the overall probability of success of the algorithm solving the NHSP$_2$. The following result sums it up.

Theorem 4.14 *Given a degree-d instance $(\mathbf{p},\mathbf{q}) \in \mathbb{F}_2[\mathbf{x}]^m \times \mathbb{F}_2[\mathbf{x}]^m$ of the* NHSP$_2$*, the algorithm*

```
Input: $(\mathbf{p},\mathbf{q}) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ a degree-$d$ instance of the NHSP$_{|\mathbb{F}|}$, with $|\mathbb{F}| \leq d$.
$p_{i_1},\dots,p_{i_n} \leftarrow n$ polynomials chosen at random from the set $\{p_1,\dots,p_m\}$
Apply the algorithm from Theorem 3.17 on $p_{i_1},\dots,p_{i_n}$.
if the algorithm from Theorem 3.17 yields a solution then
    return $A$
else
    print "The algorithm fails"
end if
```

runs in polynomial time and succeeds with probability

$$\gamma_2\,(n/2)\,P_n^{\varepsilon,\beta}.$$

Proof. Extracting from $\mathbf{p}$ at random a family of polynomials $\{p_i\}_{\{i\in\mathcal{F}\}}$, with $|\mathcal{F}| = n$, takes $\mathcal{O}\,(n)$ time, and so the running time of the algorithm is $\mathcal{O}(n)$ times the running time of the algorithm solving the HSP$_2$, which is polynomial (Theorem 3.17). Note that, heuristically, applying the algorithm of Theorem 3.17 to a degree-d instance $(\mathbf{p},\mathbf{q})$ works without changes if only the polynomials of $\mathbf{p}$ are being considered.

Concerning the probability, on the one hand the probability that the n randomly chosen polynomials are non-noisy is $P_n^{\varepsilon,\beta}$, and on the other hand the algorithm for HSP$_2$ works with probability $\gamma_2\,(n/2)$. ■

Finally, we prove that the algorithm succeeds with probability $\Omega(2^{-n/2})$ if the proportion of noise lies within a certain range. We set $\beta = 3/(1-2\varepsilon)^2$ since this is the least advantageous choice.

Theorem 4.15 *Let $\beta = 3/(1-2\varepsilon)^2$. Given a degree-$d$ instance $(\mathbf{p},\mathbf{q}) \in \mathbb{F}_2[\mathbf{x}]^m \times \mathbb{F}_2[\mathbf{x}]^m$ of the* NHSP$_2$*, the algorithm has an asymptotic probability of success*

$$c_\varepsilon{}^{-n/2}, \quad \textit{with } c_\varepsilon < 2,$$

for $\varepsilon \in (0,\varepsilon_\beta]$, $0.2836336067907370 < \varepsilon_\beta < 0.2836336067907371$.

Proof. Once a proportion of noise ε is fixed, the expression $P_n^{\varepsilon,\beta}$ depends solely on n. Since the success probability of the algorithm is $\gamma_2\,(n/2)\cdot P_n^{\varepsilon,\beta}$ and considering that $\lim_{n\to\infty}\gamma_2\,(n/2) \approx 0.288788$, the success probability of the algorithm is $\Omega(2^{-n/2})$ whenever the following holds

$$P_n^{\varepsilon,\beta} = \Omega(2^{-n/2}) \iff P_n^{\varepsilon,\beta} > 2^{-n/2}.$$

Solving numerically the latter inequality, we obtain that this happens when

$$0 \le \varepsilon \le \varepsilon_\beta, \quad \text{with}\, 0.2836336067907370 < \varepsilon_\beta < 0.2836336067907371.$$

■

Remark 4.16 *The success probability $\Omega(2^{-n/2})$ of our algorithm can be made any higher (e.g., $\Omega(2^{-n/3})$) at the expense of suitably reducing the width of the interval $(0,\varepsilon_\beta]$ within which such success probability is reached.*

Remark 4.17 *It is also easy to check that increasing (but fixed) values of β give increasing values of the ε_β boundary; this fact justifies the choice of β in the statement of* Theorem 4.15.

This theorem implies that, for $\varepsilon \in (0, \varepsilon_\beta]$, there is a non-quantum polynomial-time algorithm solving the NHSP_2 with a success probability that is $\Omega(2^{-n/2})$. This contradicts the conjecture of Aaronson and Christiano about the noisy version of their scheme (see Conjecture 2.4) and it is, to the best of our knowledge, the first non-quantum but classical algorithm that does so. Finally, we want to remark that we do not include experimental results in this section because the running time of this algorithm is $\mathcal{O}(n)$ times the running time of the algorithm that solves the HSP_2, so we consider that it does not provide any meaningful information aside from the one already displayed in Section 4.1.2.

Chapter 5

Conclusions, Contributions and future work

In this chapter we address our conclusions on the hardness of the hidden subspaces problem and its noisy counterpart, revisiting our contributions, and we briefly comment what are the implications over the noise-free and the noisy version of Aaronson-Christiano's scheme.

We also discuss particularly relevant concurrent work done by other authors and sketch several lines of research that could be explored in the future.

5.1 Conclusions

At the start of this thesis, in the context of post-quantum cryptography gaining relevance due to the threat that efficient quantum computing would pose to modern cryptography, two problems claimed to be quantum-resistant struck our attention. These problems were the $\mathrm{HSP}_{|\mathbb{F}|}$ and the $\mathrm{NHSP}_{|\mathbb{F}|}$, which underlay the security of two versions of the first public-key quantum money scheme with a security proof, Aaronson-Christiano's. We noticed that no one had neither approached yet the analysis of the hardness of the hidden subspaces problem and its noisy counterpart nor studied the security of Aaronson-Christiano's scheme. This thesis aimed at filling this gap and, once finished, the global picture is as follows.

- The HSP_2 can be heuristically solved in randomised polynomial time as long as it is instantiated as the authors specify (see Theorem 3.17). As a consequence, the noise-free version of the scheme of Aaronson and Christiano is not secure.

- The $\mathrm{HSP}_{|\mathbb{F}|}$ can be solved in randomised polynomial time for degree-d instances of the problem that satisfy the condition $|\mathbb{F}| > d$ (see Theorem 3.10). This algorithm yields a cryptanalysis of any extension of the scheme of Aaronson-Christiano to a finite field satisfying the former condition.

- The NHSP_2 can be solved in polynomial time with a probability that exceeds that established in Conjecture 2.4 for roughly half of the choices of the noise parameter, thus disproving it (see Theorem 4.14). This conjecture was stated by the authors and it constitutes the assumption under which their noisy scheme achieves perfect completeness and negligible soundness error. Therefore, the scheme is no longer secure.

- The $\mathrm{NHSP}_{|\mathbb{F}|}$ can be solved in randomised polynomial time for degree-d instances of the problem that satisfy the condition $|\mathbb{F}| > d$ (see Theorem 4.11). The algorithm solving the $\mathrm{NHSP}_{|\mathbb{F}|}$ yields a cryptanalysis of the noisy version of Aaronson-Christiano's scheme extended to any field $\mathbb{F}$ that satisfies the former condition. In particular, the $\mathrm{NHSP}_{|\mathbb{F}|}$ is not harder than the $\mathrm{HSP}_{|\mathbb{F}|}$ in this setting, since we achieved a non-quantum reduction from the noisy version of the problem to the noiseless one.

Note that our findings on the noisy hidden subspaces problem, which were obtained from a non-quantum but classical perspective, already suggested that the noisy version of the problem may be no harder than the noise-free version of it irrespective of the field. Concurrently to this thesis, Paul Christiano and Or Sattath confirmed it by proving that there is in fact a quantum reduction from the noisy hidden subspaces problem to the noiseless hidden subspaces problem [Aaronson, 2016, Aaronson, 2018].

5.2 Contributions

For a given a degree-d instance of the $\mathrm{HSP}_{|\mathbb{F}|}$, the system that models the $\mathrm{HSP}_{|\mathbb{F}|}$ presents a different structure depending on whether $|\mathbb{F}| > d$ or $|\mathbb{F}| \leq d$. This is why we divided the study of the $\mathrm{HSP}_{|\mathbb{F}|}$ and the $\mathrm{NHSP}_{|\mathbb{F}|}$ into two different scenarios.

5.2.1 Contributions regarding the $\mathrm{HSP}_{|\mathbb{F}|}$

In the scenario $|\mathbb{F}| > d$ we found out that there were linear equations in the system, which turned out to be sufficiently many for it to be solved by considering the linear equations only. In this sense, we obtained a randomised polynomial-time algorithm that solves degree-d instances of the $\mathrm{HSP}_{\mathbb{F}}$ and performs well in terms of efficiency (see Tables 3.1, 3.2). In particular, our main result in this regard is the following.

Theorem 3.10 *Let $|\mathbb{F}| > d$. The algorithm*

```
Input: $(\mathbf{p}, \mathbf{q}) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ a degree-$d$ instance of the $\mathrm{HSP}_{|\mathbb{F}|}$, with $|\mathbb{F}| > d$.
Compute the solution $E_A$ of the system of linear equations
    $\left\{ \lambda^{\mathbf{p}}_{i,k} + \sum_{j=1}^{n/2} \lambda^{\mathbf{p}}_{i,j+n/2} g_{k,j}, \quad \forall i \in \{1, \ldots, m\}, \forall k \in \{1, \ldots, n/2\} \right\}$
$A \leftarrow \mathrm{span}(E_A)$
return $A$
```

runs in randomised polynomial-time and it solves the $\mathrm{HSP}_{|\mathbb{F}|}$ *in*

$$\mathcal{O}\left(n^{2\omega}\right)$$

with success probability

$$\frac{\gamma_{|\mathbb{F}|}(n/2)\gamma_{|\mathbb{F}|}(m)}{\gamma_{|\mathbb{F}|}(m-n/2)},$$

which can be asymptotically approximated by $1 - \frac{1}{|\mathbb{F}|}$.

Unfortunately, in the case $|\mathbb{F}| \leq d$ there were no linear equations in the system. However, we noticed that the degree of regularity of our system was lower than the expected degree of regularity if the system was semi-regular. In fact, it seemed to stay bounded (see Conjecture 3.16). This behaviour suggested that the system had likely some internal structure that could be exploited, possibly imposed by the orthogonality relation between the subspaces A and $A^{\perp}$. Indeed, we proved in Theorem 3.19 that there were a kind of symmetries in the equations that allowed to construct equations of degree strictly lower than d from linear combinations of equations in the system. These are the main findings that support our algorithm solving the HSP_2, which is efficient in practice (see Tables 3.5, 3.6). It is the following.

Theorem 3.17 *Assuming* Conjecture 3.16*, there is a randomised polynomial-time algorithm (consisting on the computation of a Gröbner basis) solving degree-d instances of* HSP_2 *with a complexity of*

$$\mathcal{O}\left(n^{2\omega(d+1)}\right),$$

and success probability

$$\gamma_2(n/2).$$

Our results yield a heuristic cryptanalysis of Aaronson-Christiano's noise-free scheme, as well as a cryptanalysis of any extension of Aaronson-Christiano's noise-free scheme to any field $\mathbb{F}$ other than $\mathbb{F}_2$.

5.2.2 Contributions regarding the $\mathrm{NHSP}_{|\mathbb{F}|}$

Given a degree-d instance of the $\mathrm{NHSP}_{|\mathbb{F}|}$, we found out that there is a reduction from the NHSP to the HSP whenever the condition $|\mathbb{F}| > d$ is satisfied, which allows to apply our algorithm that solves the $\mathrm{HSP}_{|\mathbb{F}|}$. The main result concerning our algorithm solving the $\mathrm{NHSP}_{|\mathbb{F}|}$, which achieves good results in terms of efficiency and success probability (see Tables 4.1, 4.2), is the following.

Theorem 4.11 *The algorithm*

```
Input: $(\mathbf{p},\mathbf{q}) \in \mathbb{F}[\mathbf{x}]^m \times \mathbb{F}[\mathbf{x}]^m$ a degree-$d$ instance of the $\mathrm{NHSP}_{|\mathbb{F}|}$, with $|\mathbb{F}| > d$.
$E_A \leftarrow \emptyset$
for j=1 to m do
//We identify which scalars $\lambda_i^{\mathbf{q}}$ of the set (4.3) belong to $A$
    weight $\leftarrow w_{|\mathbb{F}|}^{\mathbf{p}}\left(\lambda_j^{\mathbf{q}}\right)$
    if $m -$ weight $\geq \lceil(1-\varepsilon)\rceil m$ then
        $E_A \leftarrow E_A \cup \{\lambda_j^{\mathbf{q}}\}$
    end if
end for
if $\dim(\mathrm{span}(E_A)) = n/2$ then
//We check if there are $n/2$ elements of $E_A$ that are linearly independent
    $A \leftarrow \mathrm{span}(E_A)$.
    return $A$
else
    print "The algorithm fails"
end if
```

solves the $\mathrm{NHSP}_{|\mathbb{F}|}$*, with* $|\mathbb{F}| > d$*, with complexity*

$$\mathcal{O}\left(m^{\omega} d n^{d}\right)$$

and probability of success at least

$$\left(1-\frac{1}{|\mathbb{F}|^n}\right)^{\lceil(1-\varepsilon)m\rceil} \frac{\gamma_{|\mathbb{F}|}\left(\lceil(1-\varepsilon)m\rceil\right)}{\gamma_{|\mathbb{F}|}\left(\lceil(1-\varepsilon)m\rceil - n/2\right)}.$$

In the case of the NHSP_2 we could achieve no reduction, but exhaustive search alone combined with our algorithm for the hidden subspaces problem performs better than $\mathcal{O}\left(2^{-n/2}\right)$ for slightly more than half of the values that the noise can take.

Theorem 4.15 *Let* $\beta = 3/(1-2\varepsilon)^2$*. Given a degree-*$d$ *instance* $(\mathbf{p},\mathbf{q}) \in \mathbb{F}_2[\mathbf{x}]^m \times \mathbb{F}_2[\mathbf{x}]^m$ *of the* NHSP_2*, the algorithm has an asymptotic probability of success*

$$c_\varepsilon{}^{-n/2}, \quad \text{with}\, c_\varepsilon < 2,$$

for $\varepsilon \in (0, \varepsilon_\beta]$, $0.2836336067907370 < \varepsilon_\beta < 0.2836336067907371$.

This disproves Conjecture 3.16, which was assumed by Aaronson-Christiano and it is the main assumption on which they base their noisy quantum money scheme.

5.3 Future work

The problem of designing a public-key quantum money scheme with a proof of security remains open. In this regard, perhaps the more natural research area that arises from this thesis is exploring if another instantiation of the $\mathrm{HSP}_{|\mathbb{F}|}$ can make the $\mathrm{HSP}_{|\mathbb{F}|}$ hard and thus the scheme of Aaronson-Christiano secure. Studying whether or not choosing the instances to be homogeneous makes a difference on the hardness of the $\mathrm{HSP}_{|\mathbb{F}|}$ might be a good point to start. However, let us say that our uninformed guess is that A and $A^\perp$ are too structured not to impose an exploitable structure on the system of equations. Note that throughout this thesis we have not contemplated the case $|\mathbb{F}| \leq d$ for fields other than $\mathbb{F}_2$, so that is also worth a look, although we believe that the behaviour might be somehow analogous. These studies would be from a purely classical (as in non-quantum) perspective.

From a quantum perspective there is some other research could be carried out. Zhandry claims in [Zhandry, 2017] that instantiating the quantum money scheme of Aaronson-Christiano with indistinguishability obfuscation that is secure against quantum computers yields a secure quantum money scheme. Another proposal of a quantum money scheme, based on the scheme of Aaronson-Christiano, is given in [Ben-David and Sattath, 2017], so it should be further investigated.

As an anecdote, in the process of trying to find a distinguisher to detect noisy polynomials over $\mathbb{F}_2$, we noticed that the number of zeros of the sum of two noisy polynomials was greater that the number of zeros of the sum of either two non-noisy polynomials or one noisy polynomial and one non-noisy one. We dedicated some time to apply statistical techniques to test how good of a distinguisher it was, but it did not seem to perform better than exhaustive search. However, we have not fully exhausted the options of study derived from this observation, so it might be something interesting to look at more in depth.

Appendix A

Magma codes

In this appendix we include the source codes (in Magma software [Bosma et al., 1997]) of the algorithms for the $\text{HSP}_{|\mathbb{F}|}$ and the $\text{NHSP}_{|\mathbb{F}|}$ of Chapter 3 and Chapter 4, so the experiments can be reproduced or different ones can be run if desired. The codes are also available online on GitHub (`https://github.com/Marta-PhD/solving-HSP-NHSP`).

A.1 Generating and solving the $\text{HSP}_{\mathbb{F}}$

The main routine regarding the $\text{HSP}_{|\mathbb{F}|}$ is `Solve`, which generates a degree-d instance of the $\text{HSP}_{|\mathbb{F}}$ as described by Aaronson-Christiano and then finds a solution for it. This routine has several dependencies, so we include a tree of code dependencies in Figure A.1 to make it more visual.

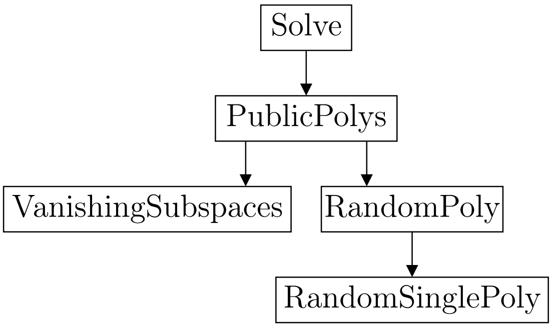


Figure A.1: Dependencies of the main routine `Solve` for the $\text{HSP}_{|\mathbb{F}|}$ case

`Solve` takes as parameters the cardinality of the finite field $\mathbb{F}$, the number m of polynomials to vanish on each A and $A^{\perp}$, the number of variables n of each polynomial and the degree d of all the polynomials. Note that by including the cardinality of the base field as a parameter in the code we cover both the case $|\mathbb{F}| > d$ and the case $|\mathbb{F}| = 2$ in the same piece of code.

We include here all the subroutines. `VanishingSubspaces` generates uniformly at random an $n/2$-dimensional subspace A over $\mathbb{F}^n$ and computes its orthogonal, $A^\perp$.

```
function VanishingSubspaces(q, n)
// Generation of the subspace E=(0|I) in
// n number of variables
E:=ZeroMatrix(GF(q), n div 2, n);
for i:=1 to n div 2 do
   E[i,i]:=1;
end for;

// Generation of A=(A1|A2) where A1 is invertible of size n/2
// and A2 random of size n/2
repeat
   A1:=Random(KMatrixSpace(GF(q), n div 2, n div 2));
until Rank(A1) eq n div 2;
A:=Random(KMatrixSpace(GF(q), n div 2, n));
for i:=1 to n div 2 do
   for j:=1 to n div 2 do
      A[i,j]:=A1[i,j];
   end for;
end for;

// Computation of the change of basis matrix L (s.t. E=AL)
LT,N:=Solution(Transpose(A),Transpose(E));
W:=ZeroMatrix(GF(q),n,n);
repeat
   for i:=1 to n do
      W[i]:=W[i]+LT[i]+Random(N);
   end for;
until Rank(W) eq n;
L:=Transpose(W);

AO:=KernelMatrix(Transpose(A));
LOT,NO:=Solution(Transpose(AO),Transpose(E));
WO:=ZeroMatrix(GF(q),n,n);

// Computation of the change of basis matrix LO (s.t. E=AO LO)
repeat
   for i:=1 to n do
      WO[i]:=WO[i]+LOT[i]+Random(NO);
   end for;
until Rank(WO) eq n;
LO:=Transpose(WO);
```

```
return A, AO, L, LO;
end function;
```

GenerateMonomials is an auxiliary function that returns the list of monomials of degree $[1, \ldots, d]$, for a given d, in the given ring of polynomials R, such that are divisible by an element in the set $\{x_{n/2+1}, \ldots, x_n\}$.

```
function GenerateMonomials(n, d, EqR, R)
Mon:=[];

for degree:=1 to d do
   for a in MonomialsOfDegree(R, degree) do
      for k:=n div 2+1 to n do
         if IsDivisibleBy(a, R.k) then
            if NormalForm(a, EqR) notin Mon then
               Mon:=Mon cat [a];
               break k;
            end if;
         end if;
      end for;
   end for;
end for;
return Mon;
end function;
```

RandomSinglePoly, RandomPoly and PublicPolys are concerned with generating random sets of polynomials that vanish on A and $A^{\perp}$. RandomSinglePoly generates uniformly at random a polynomial of degree d with coefficients in $\mathbb{F}$ that vanishes on the subspace E generated by $\{x_{n/2+1}, \ldots, x_n\}$. Observe that both take as input the list of monomials generated by GenerateMonomials.

```
function RandomSinglePoly(q, d, Mon)
// Generation of a polynomial of degree d uniformly at random
// vanishing over E

repeat
   P := &+ [Random(GF(q))*Mon[j] : j in [1..#Mon]];
until TotalDegree(P) eq d;
return P;
end function;
```

RandomPoly generates two sets of m uniformly random polynomials of degree d with coefficients in $\mathbb{F}$ vanishing on the subspace E.

```
function RandomPoly(q, m, d, Mon)
```

```
P:=[]; Q:=[];
P[1]:=RandomSinglePoly(q, d, Mon);
Q[1]:=RandomSinglePoly(q, d, Mon);

Y1:={P[1]}; Y2:={Q[1]};
for i:=2 to m do
   repeat
      P[i]:=RandomSinglePoly(q, d, Mon);
// Generation of m polynomials of degree d uniformly
// at random vanishing over E...
   until {P[i]} meet Y1 eq {};
// assuring that the m generated polynomials are distinct
   Y1:=Y1 join {P[i]};
end for;

for j:=2 to m do
   repeat
      Q[j]:=RandomSinglePoly(q, d, Mon);
// Generation of m polynomials of degree d uniformly
// at random vanishing over E...
   until {Q[j]} meet Y2 eq {};
// assuring that the m generated polynomials are distinct
   Y2:=Y2 join {Q[j]};
end for;

return P, Q;
end function;
```

The routine `PublicPolys` transforms (according to Proposition 2.13) the two sets of m uniformly random polynomials of degree d vanishing on E into two sets of m polynomials vanishing on A and $A^{\perp}$, respectively.

```
function PublicPolys(q, m, n, d, R)

EqR:=[R.i^q-R.i : i in [1..Rank(R)]];

Mon:=GenerateMonomials(n, d, EqR, R);
Pb,Qb:=RandomPoly(q, m, d, Mon);

A,AO,L,LO:=VanishingSubspaces(q, n);

g:=[&+ [L[j,i]*R.j  : j in [1..n]] : i in [1..n]];
h:=[&+ [LO[j,i]*R.j : j in [1..n]] : i in [1..n]];

P:=[]; Q:=[];
```

```
for k:=1 to m do
   P[k]:=NormalForm(Evaluate(Pb[k],g),EqR);
   Q[k]:=NormalForm(Evaluate(Qb[k],h),EqR);
end for;

return A, L, AO, LO, P, Q;
end function;
```

Finally, the main routine `Solve` calls `PublicPolys` to generate a degree-d random instance of the HSP$_{|\mathbb{F}|}$ and then goes on to solve Sys$_{\text{HSP}_{|\mathbb{F}|}}$ of Proposition 3.5 distinguishing between the cases $|\mathbb{F}| > d$ and $|\mathbb{F}| = 2$ as in Chapter 3.

```
function Solve(q, m, n, d)

R:=PolynomialRing(GF(q), n);

// Generation of a degree-d instance of HSP
A,AO,L,LO,P,Q:=PublicPolys(q, m, n, d, R);

G<[g]>:=PolynomialRing(GF(q), n^2 div 4, "grevlex");
EqG:=[G.i^q-G.i : i in [1..n^2 div 4]];
F:=PolynomialRing(G,n div 2);
EqF:=[F.i^q-F.i : i in [1..Rank(F)]];

// Algorithm for HSP when q > d
if q gt d then
   h:=[];
   for i:=1 to n div 2 do
      h[i]:=F.i;
      h[i+n div 2]:=&+[G.(i+n div 2*t)*F.(t+1) : t in [0..n div 2-1]];
   end for;
   Ho:=[];
   for k:=1 to m do
      Ho[k]:=HomogeneousComponent(P[k],1);
   end for;
   z:=[];
   for k:=1 to m do
      z[k]:=Evaluate(Ho[k],h);
   end for;
   peqs:=[];
   for k:=1 to m do
      peqs[k]:=Coefficients(z[k]);
   end for;
   newlisteqs:=[];
   for k:=1 to m do
      newlisteqs:=newlisteqs cat [a: a in peqs[k]];
   end for;
   I:=ideal<G|newlisteqs>;
   J:=Variety(I);
end if;

// Algorithm for HSP when q = 2
if q eq 2 then
   h1:=[];
   h2:=[];
   for i:=1 to n div 2 do
      h1[i]        := F.i;
      h1[i+n div 2]:= &+[G.(i+(t-1)*(n div 2))*F.t:t in [1..n div 2]];
      h2[i]        := &+[G.(t+(i-1)*(n div 2))*F.t:t in [1..n div 2]];
      h2[i+n div 2]:= F.i;
   end for;

   z:=[];
   w:=[];
   for k:=1 to m do
      z[k]:=NormalForm(Evaluate(P[k],h1),EqF);
      w[k]:=NormalForm(Evaluate(Q[k],h2),EqF);
   end for;

   peqs:=[];qeqs:=[];
   for k:=1 to m do
      peqs[k]:=Coefficients(z[k]);
      qeqs[k]:=Coefficients(w[k]);
   end for;

   Listp:=[];Listq:=[];
   for k:=1 to m do
      for i:=1 to #peqs[k] do
         Listp:=Listp cat [peqs[k][i]];
      end for;
      for i:=1 to #qeqs[k] do
         Listq:=Listq cat [qeqs[k][i]];
      end for;
   end for;
   eqs:=Listp cat Listq;
   I:=ideal<G | eqs,EqG>;
   SetVerbose("Faugere",1);
   J:=GroebnerBasis(I);
end if;
return P,Q,J;
end function;
```

A.2 Generating and solving the $\mathrm{NHSP}_{|\mathbb{F}|}$ for $|\mathbb{F}| > d$

As before, the main routine regarding the $\mathrm{NHSP}_{|\mathbb{F}|}$ is `Solve`, which generates a degree-d instance of the $\mathrm{NHSP}_{|\mathbb{F}|}$ and then finds a solution for it, only that now we only implemented the case $|\mathbb{F}| > d$. The scenario $|\mathbb{F}| = 2$ covered in Chapter 4 is not implemented. `Solve` has several dependencies, so again we include a tree of code dependencies in Figure A.2 to make it more visual.

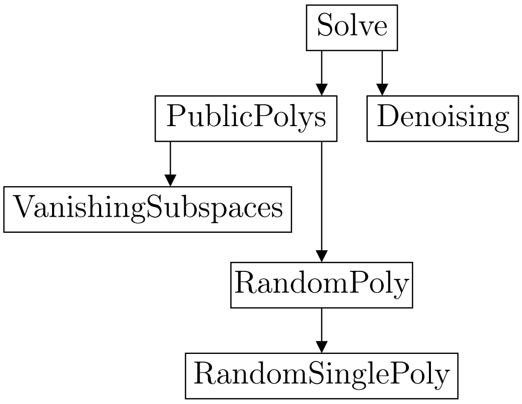


Figure A.2: Dependencies of the main routine `Solve` for the $\mathrm{NHSP}_{|\mathbb{F}|}$ case

In the noisy case, `Solve` takes as parameters the cardinality of the finite field, the number n of variables of each polynomial, the proportion $\varepsilon \in (0, 1/2)$ of noise and the degree d of the polynomials.

We include here all the subroutines. `VanishingSubspaces` generates uniformly at random an $n/2$-dimensional subspace A over $\mathbb{F}^n$ and computes its orthogonal $A^{\perp}$, as well as other $\lceil 2\epsilon m \rceil$ uniformly random subspaces of dimension $n/2$ over which the polynomials added as noise will vanish.

```
function VanishingSubspaces(pr, n, delta)
   // Generation of the subspace E=(0|I)
   E:=ZeroMatrix(GF(pr), n div 2, n);
   for i:=1 to n div 2 do
      E[i,i] := 1;
   end for;

   // Generation of A=(A1|A2) where A1 is invertible of size n/2
   // and A2 random of size n/2
   repeat
      A1:=Random(KMatrixSpace(GF(pr), n div 2, n div 2));
   until Rank(A1) eq n div 2;
   A:=Random(KMatrixSpace(GF(pr), n div 2, n));
   for i:=1 to n div 2 do
      for j:=1 to n div 2 do
         A[i,j]:=A1[i,j];
      end for;
   end for;

   // Computation of the change of basis matrix L (s.t. E=AL)
   LT,N := Solution(Transpose(A), Transpose(E));
   W    := ZeroMatrix(GF(pr), n, n);
   repeat
      for i:=1 to n do
         W[i] := W[i] + LT[i] + Random(N);
      end for;
      L := Transpose(W);
   until Rank(L) eq n and A*L eq E;

   // Computation of the change of basis matrix LO (s.t. E=AO LO)
   AO     := KernelMatrix(Transpose(A));
   LOT,NO := Solution(Transpose(AO), Transpose(E));
   WO     := ZeroMatrix(GF(pr), n, n);
   repeat
      for i:=1 to n do
         WO[i] := WO[i] + LOT[i] + Random(NO);
      end for;
      LO := Transpose(WO);
   until Rank(LO) eq n and AO*LO eq E;

   B := [];
   for i:=1 to delta do
      repeat
         B[i] := Random(KMatrixSpace(GF(pr), n div 2, n));
      until Rank(B[i]) eq n div 2;
   end for;

   LT:=[]; N:=[]; BL:=[];
   for i:=1 to delta do
      LT[i], N[i] := Solution(Transpose(B[i]), Transpose(E));
      W := ZeroMatrix(GF(pr), n, n);
      repeat
         for j:=1 to n do
            W[j] := W[j] + LT[i][j] + Random(N[i]);
         end for;
         BL[i] := Transpose(W);
      until Rank(BL[i]) eq n and B[i]*BL[i] eq E;
   end for;

   BO:=[];
   for i:=1 to delta do
      repeat
```

```
      BO[i] := Random(KMatrixSpace(GF(pr), n div 2, n));
    until Rank(BO[i]) eq n div 2;
  end for;

  LOT:=[]; NO:=[]; BOL:=[];
  for i:=1 to delta do
    LOT[i], NO[i] := Solution(Transpose(BO[i]), Transpose(E));
    WO := ZeroMatrix(GF(pr), n, n);
    repeat
      for j:=1 to n do
        WO[j] := WO[j] + LOT[i][j] + Random(NO[i]);
      end for;
      BOL[i] := Transpose(WO);
    until Rank(BOL[i]) eq n and BO[i]*BOL[i] eq E;
  end for;

  return A, AO, B, BO, L, LO, BL, BOL;
end function;
```

GetMonomials is an auxiliary function that returns the list of monomials of degree $[1, \ldots, d]$, for a given d, in the given ring of polynomials R, such that are divisible by an element in the set $\{x_{n/2+1}, \ldots, x_n\}$.

```
function GetMonomials(R, d)
  Mon := [];
  n   := Rank(R);
  for degree:=1 to d do
    for a in MonomialsOfDegree(R, degree) do
      for k:=n div 2 + 1 to n do
        if IsDivisibleBy(a, R.k) then
          Mon := Mon cat [a];
          break k;
        end if;
      end for;
    end for;
  end for;
  return Mon;
end function;
```

As it happened before, RandomSinglePoly, RandomPoly and PublicPolys are concerned with the generation of an instance of the NHSP$_{|\mathbb{F}|}$. RandomSinglePoly generates uniformly at random a polynomial of degree d with coefficients in $\mathbb{F}^n$ that vanishes on the subspace E generated by $\{x_{n/2+1}, \ldots, x_n\}$.

```
function RandomSinglePoly(R, Mon, d)
  pr := Characteristic(R);
```

```
   n  := Rank(R);
   repeat
      p := &+[Random(Set([0..pr-1]))*Mon[j] : j in [1..#Mon]];
   until TotalDegree(p) eq d;
   return p;
end function;
```

The routine RandomPoly generates two sets of m polynomials of degree d chosen uniformly at random with coefficients in $\mathbb{F}$ vanishing on the subspace E. As an auxiliary, we use the routine GetMonomials.

```
function RandomPoly(R, m, d)
   p := [];
   q := [];
   Mon  := GetMonomials(R, d);

   p[1] := RandomSinglePoly(R, Mon, d);
   q[1] := RandomSinglePoly(R, Mon, d);
   Y1   := {p[1]};
   Y2   := {q[1]};

   for i:=2 to m do
      repeat
         p[i] := RandomSinglePoly(R, Mon, d);
      until {p[i]} meet Y1 eq {};

      Y1 := Y1 join {p[i]};
   end for;

   for i:=2 to m do
      repeat
         q[i] := RandomSinglePoly(R, Mon, d);
      until {q[i]} meet Y2 eq {};

      Y2:=Y2 join {q[i]};
   end for;

   return p, q;
end function;
```

The routine PublicPolys generates the following:

- A list of indices K_1 (resp. K_2), with $|K_1| = |K_2| = \lceil (1-\varepsilon)\, m \rceil$, and $\lceil (1-\varepsilon)\, m \rceil$ polynomials with indices in K_1 (resp. K_2) that vanish on A (resp. $A^{\perp}$).
- A list of indices J_1 (resp. J_2), with $|J_1| = |J_2| = \lceil \varepsilon m \rceil$, and $\lceil \varepsilon m \rceil$ polynomials

with indices in J_1 (resp. J_2) that vanish on the $2\lceil \varepsilon m \rceil$ random $n/2$-dimensional subspaces generated with `VanishingSubspaces`.

```
function PublicPolys(R, m, d, delta)
   pr := Characteristic(R);
   n   := Rank(R);

   J1  := [];
   J2  := [];

   for i:=1 to delta do
      repeat
         tmp := Random([1..m]);
      until tmp notin J1;
      J1 := J1 cat [tmp];
   end for;

   for i:=1 to delta do
      repeat
         tmp := Random([1..m]);
      until tmp notin J2;
      J2 := J2 cat [tmp];
   end for;

   Sort(~J1);
   Sort(~J2);

   K1 := [ x : x in [1..m] | x notin J1];
   K2 := [ x : x in [1..m] | x notin J2];

   pb, qb := RandomPoly(R, m, d);

   T:=[]; BT:=[]; C:=[]; BC:=[];
   for i:=1 to m do
      T[i]  := Terms(pb[i]);
      C[i]  := Coefficients(pb[i]);
      BT[i] := Terms(qb[i]);
      BC[i] := Coefficients(qb[i]);
   end for;

   S:=[];
   for i:=1 to m do
      S[i] := [];
      for j:=1 to #T[i] do
         S[i][j] := [];
         for k:=1 to n do
            S[i][j][k] := 0;
            repeat
               if IsDivisibleBy(T[i][j], R.k) then
                  S[i][j][k] := S[i][j][k] + 1;
                  T[i][j]    := T[i][j]/R.k;
               end if;
            until IsDivisibleBy(T[i][j], R.k) eq false;
         end for;
      end for;
   end for;

   BS:=[];
   for i:=1 to m do
      BS[i] := [];
      for j:=1 to #BT[i] do
         BS[i][j] := [];
         for k:=1 to n do
            BS[i][j][k] := 0;
            repeat
               if IsDivisibleBy(BT[i][j],R.k) then
                  BS[i][j][k] := BS[i][j][k] + 1;
                  BT[i][j]    := BT[i][j]/R.k;
               end if;
            until IsDivisibleBy(BT[i][j], R.k) eq false;
         end for;
      end for;
   end for;

   A, AO, B, BO, L, LO, BL, BOL := VanishingSubspaces(pr, n, delta);
   Unk := Matrix(R, 1, n, [R.i   : i in [1..n]]);
   NL  := Matrix(R, n, n, [L[i]  : i in [1..n]]);
   NLO := Matrix(R, n, n, [LO[i] : i in [1..n]]);

   NBL:=[]; NBOL:=[];
   for i:=1 to delta do
      NBL[i]  := Matrix(R, n, n, [BL[i][j]  : j in [1..n]]);
      NBOL[i] := Matrix(R, n, n, [BOL[i][j] : j in [1..n]]);
   end for;

   xL  := Unk*NL;
   xLO := Unk*NLO;
   xBL :=[]; xBLO:=[];
   for i:=1 to delta do
      xBL[i]  := Unk*NBL[i];
      xBLO[i] := Unk*NBOL[i];
   end for;
```

```
   p:=[]; q:=[]; b:=0;
   for i:=1 to m do
      if i in K1 then
         p[i] := &+[C[i,j]*&*[xL[1][k]^S[i][j][k] : k in [1..n]
                 | S[i][j][k] ne 0] : j in [1..#T[i]]];
      else
         b:=b+1;
         p[i] := &+[C[i,j]*&*[xBL[b][1][k]^S[i][j][k] : k in [1..n]
                 | S[i][j][k] ne 0] : j in [1..#T[i]]];
      end if;
   end for;

   c:=0;
   for i:=1 to m do
      if i in K2 then
         q[i] := &+[BC[i,j]*&*[xLO[1][k]^BS[i][j][k] : k in [1..n]
                 | BS[i][j][k] ne 0] : j in [1..#BT[i]]];
      else
         c:=c+1;
         q[i] := &+[BC[i,j]*&*[xBLO[c][1][k]^BS[i][j][k] : k in [1..n]
                 | BS[i][j][k] ne 0] : j in [1..#BT[i]]];
      end if;
   end for;

   return p, q;

end function;
```

Now, the routine Denoising implements the algorithm of Theorem 4.11 solving the NHSP$_{|\mathbb{F}|}$.

```
function Denoising(R, m, p, q, gamma)
   pr          := Characteristic(R);
   n           := Rank(R);
   Lambdaq     := [];
   ListEval    := [];
   LambdaOK    := [];

   for i:=1 to #q do
      Lambdaq[i]   := [];
      ListEval[i]  := [];
      for j:=1 to n do
         Lambdaq[i][j] := MonomialCoefficient(q[i], R.j);
      end for;
      for k:=1 to #p do
         ListEval[i][k] := Evaluate(p[k],Lambdaq[i]);
      end for;

      numberzeros := #[ x : x in [1..#ListEval[i]]
                                 | ListEval[i][x] eq 0];
      if numberzeros ge gamma then
         LambdaOK := LambdaOK cat [Lambdaq[i]];
      end if;
   end for;

   EltsA := ZeroMatrix(GF(pr), #LambdaOK, n);
   for i:=1 to #LambdaOK do
      for j:=1 to #LambdaOK[i] do
         EltsA[i][j] := LambdaOK[i][j];
      end for;
   end for;

   BasisA := [];
   if Rank(EchelonForm(EltsA)) ge n div 2 then
      for i:=1 to n div 2 do
         BasisA := BasisA cat [EchelonForm(EltsA)[i]];
      end for;
   else
      "Algorithm failed.";
   end if;

   return BasisA;

end function;
```

Finally, the main routine Solve integrates the process of generating and solving a degree-d instance of the NHSP$_{|\mathbb{F}|}$ with a proportion ε of noise.

```
function Solve(q, n, epsilon, d)
   R     := PolynomialRing(GF(q), n);
   beta  := 3/(1 - 2*epsilon)^2;
   m     := Ceiling(beta*n);
   gamma := Ceiling((1-epsilon)*m);
   delta := m - gamma;

   P, Q   := PublicPolys(R, m, d, delta);
   BasisA := Denoising(R, m, P, Q, gamma);

   return BasisA;
end function;
```

References

[Aaronson, 2009] Aaronson, S. (2009). Quantum copy-protection and quantum money. In *Proceedings of the 24th Annual IEEE Conference on Computational Complexity, CCC 2009, Paris, France, 15-18 July 2009*, pages 229–242. 1.1, 2.1

[Aaronson, 2016] Aaronson, S. (2016). Public-key quantum money. In *Lecture notes for the 28th McGill Invitational Workshop on Computational Complexity*, pages 81–88. arXiv:1607.05256. 5.1

[Aaronson, 2018] Aaronson, S. (Retrieved in 2018). More wrong things I said in papers. Shtetl-Optimized, The Blog of Scott Aaronson. https://www.scottaaronson.com/blog/?p=2854. 5.1

[Aaronson and Christiano, 2013] Aaronson, S. and Christiano, P. (2013). Quantum money from hidden subspaces. *Theory of Computing*, 9:349–401. 1, 1.1, 1.3.1, 2.1, 2.2, 2.3, 2.1, 2.3, 2.4, 2.12, 2.15, 2.16, 3.2.1, 3.3, 4.1.1, 4.1, 4.7, 4.2

[Alkim et al., 2015] Alkim, E., Ducas, L., Pöppelmann, T., and Schwabe, P. (2015). Post-quantum key exchange - a new hope. Cryptology ePrint Archive, Report 2015/1092. https://eprint.iacr.org/2015/1092. 2.3

[Alon et al., 1998] Alon, N., Krivelevich, M., and Sudakov, B. (1998). Finding a large hidden clique in a random graph. In *Proc. ACM-SIAM Symp. on Discrete Algorithms (SODA)*, pages 594—598. 2.1

[Bardet, 2004] Bardet, M. (2004). *Étude des systèmes algébriques surdéterminés. Applications aux codes correcteurs et à la cryptographie.* PhD thesis, Université de Paris VI. 2.5.3

[Bardet et al., 2015] Bardet, M., Faugère, J.-C., and Salvy, B. (2015). On the complexity of the F_5 Gröbner basis algorithm. *Journal of Symbolic Computation*, 70:49–70. 2.40

[Ben-David and Sattath, 2017] Ben-David, S. and Sattath, O. (2017). Quantum tokens for digital signatures. Cryptology ePrint Archive, Report 2017/094. https://eprint.iacr.org/2017/094. 5.3

[Bennett et al., 1982] Bennett, C. H., Brassard, G., Breidbard, S., and Wiesner, S. (1982). Quantum cryptography, or unforgeable subway tokens. In *Advances in Cryptology - CRYPTO 1982*, pages 267–275. 1.1, 2.1

[Bernstein et al., 2009] Bernstein, D. J., Buchmann, J. A., and Dahmen, E. (2009). *Post-Quantum Cryptography*. Springer Berlin Heidelberg. 1

[Bernstein et al., 2015] Bernstein, D. J., Hopwood, D., Hülsing, A., Lange, T., Niederhagen, R., Papachristodoulou, L., Schneider, M., Schwabe, P., and Wilcox-O'Hearn, Z. (2015). SPHINCS: Practical stateless hash-based signatures. *EUROCRYPT 2015*, pages 368–397. 1

[Bos et al., 2015] Bos, J. W., Costello, C., Naehrig, M., and Stebila, D. (2015). Post-quantum key exchange for the TLS protocol from the ring learning with errors problem. In *IEEE Symposium on Security and Privacy*, pages 553–570, San Jose, CA, USA. IEEE Computer Society Press. 2.3

[Bosma et al., 1997] Bosma, W., Cannon, J. J., and Playoust, C. (1997). The Magma algebra system i: The user language. *Journal of Symbolic Computation*, 24(3-4):235–265. 1.2, 3.2.2, 3.3.2, 4.1.2, A

[Braeken et al., 2005] Braeken, A., Wolf, C., and Preneel, B. (2005). A study of the security of unbalanced oil and vinegar signature schemes. In Menezes, A., editor, *Topics in Cryptology - CT-RSA 2005*, volume 3376 of *Lecture Notes Comput. Sci.*, pages 29–43. Springer Berlin Heidelberg. 1

[Brent and McKay, 1987] Brent, R. P. and McKay, B. D. (1987). Determinants and rank of random matrices over $\mathbb{Z}_m$. *Discrete Math.*, 66:35–50. 3.2.1, 4.9

[Buchberger, 1965] Buchberger, B. (1965). *Ein Algorithmus zum Auffinden der Basiselemente des Restklassenringes nach einem nulldimensionalen Polynomideal*. PhD thesis, Leopold-Franzens-Universität. 2.5.1, 2.5.2

[Buchberger, 1970] Buchberger, B. (1970). Ein algorithmisches Kriterium für die Lösbarkeit eines algebraischen Gleichungs-systems. *Aequationes mathematicae*, 4/3:374–383. 2.5.2

[Buchberger, 1979] Buchberger, B. (1979). A criterion for detecting unnecessary reductions in the construction of Gröbner bases. In Ng, E. W., editor, *Symbolic and Algebraic Computation*, volume 72 of *Lecture Notes Comput. Sci.*, pages 3–21. Springer Berlin Heidelberg. 2.5.2

[Buchberger, 1983] Buchberger, B. (1983). A note on the complexity of constructing Gröbner bases. In van Hulzen, J., editor, *Computer Algebra*, volume 162 of *Lecture Notes in Computer Science*, pages 137–145. Springer. 2.5.2

[Buchmann et al., 2011] Buchmann, J., Dahmen, E., and Hülsing, A. (2011). XMSS - a practical forward secure signature scheme based on minimal security assumptions. *PQCrypto 2011*, pages 117–129. 1

[Conde Pena et al., 2014] Conde Pena, M., Durán Díaz, R., Hernández Encinas, L., and Muñoz Masqué, J. (2014). The isomorphism of polynomials problem applied to multivariate quadratic cryptography. In Herrero, A., Baruque, B., Klett, F., Abraham, A., Snášel, V., Carvalho, A. C., Bringas, P. G., Zelinka, I., Quintián, H., and Corchado, E., editors, *International Joint Conference SOCO'13-CISIS'13-ICEUTE'13*, volume 239 of *Advances in Intelligent Systems and Computing*, pages 567–576. Springer International Publishing. 1

[Conde Pena et al., 2015] Conde Pena, M., Faugère, J.-C., and Perret, L. (2015). Algebraic cryptanalysis of a quantum money scheme the noise-free case. In Katz, J., editor, *Public-Key Cryptography – PKC 2015*, pages 194–213, Berlin, Heidelberg. Springer Berlin Heidelberg. (document), 1, 1.3.2

[Conde Pena et al., 2018] Conde Pena, M., Hernández Encinas, L., Durán Díaz, R., Faugère, J.-C., and Perret, L. (2018). Non-quantum cryptanalysis of the noisy version of Aaronson-Christiano's quantum money scheme. Accepted for publication in "IET Information Security". (document), 1, 1.3.3

[Conover, 2018] Conover, E. (2018). Google moves toward quantum supremacy with 72-qubit computer. Science News. https://www.sciencenews.org/article/google-moves-toward-quantum-supremacy-72-qubit-computer. 1

[Cook, 1971] Cook, S. (1971). The complexity of theorem proving procedures. In *Proceedings of the Third Annual ACM Symposium on Theory of Computing*, pages 151–158. 2.5

[Cook, 2000] Cook, S. (2000). *The P versus NP Problem*. Clay Mathematics Institute. 2.5

[Courtois et al., 2001] Courtois, N., Finiasz, M., and Sendrier, N. (2001). How to achieve a McEliece-based digital signature scheme. *ASIACRYPT 2001*, 2248:157–174. 1

[Cox et al., 2007] Cox, D. A., Little, J., and O'Shea, D. (2007). *Ideals, Varieties, and Algorithms*. Springer, 3rd edition edition. 2.5

[Dieks, 1982] Dieks, D. (1982). Communication by EPR devices. *Physics Letters A*, 92A(6):271–272. 1

[Diffie and Hellman, 1976] Diffie, W. and Hellman, M. E. (1976). New directions in Cryptography. *IEEE Transactions on Information Theory*, 22:644–654. 1

[Dubé, 1990] Dubé, T. W. (1990). The structure of polynomial ideals and Gröbner bases. *SIAM Journal on Computing*, 19(4):750. 2.5

[Ducas et al., 2013] Ducas, L., Durmus, A., Lepoint, T., and Lyubashevsky, V. (2013). Lattice signatures and bimodal Gaussians. *CRYPTO 2013*, pages 40–56. 1

[ElGamal, 1985] ElGamal, T. (1985). A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, 31(4):469–472. 1

[Farhi et al., 2012] Farhi, E., Gosset, D., Hassidim, A., Lutomirski, A., and Shor, P. W. (2012). Quantum money from knots. In *Innovations in Theoretical Computer Science 2012, Cambridge, MA, USA, January 8-10, 2012*, pages 276–289. 1.1, 2.1

[Faugère et al., 1993] Faugère, J., Gianni, P., Lazard, D., and Mora, T. (1993). Efficient computation of zero-dimensional Gröbner bases by change of ordering. *J. Symbolic Computation*, 16:329–344. 2.5.2

[Faugère, 1999] Faugère, J.-C. (1999). A new efficient algorithm for computing Gröbner bases (F_4). *Journal of Pure and Applied Algebra*, 139:61–88. 1.2, 2.5.2

[Faugère, 2002] Faugère, J.-C. (2002). A new efficient algorithm for computing Gröbner bases without reduction to zero (F_5). In Press, A., editor, *International Symposium on Symbolic and Algebraic Computation-ISAAC 2002*, pages 75–83. 1.2, 2.5.2, 2.42

[Faugère et al., 2014] Faugère, J.-C., Gaudry, P., Huot, L., and Renault, G. (2014). Sub-cubic change of ordering for Gröbner basis: a probabilistic approach. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation*. ACM. 2.5.2

[Faugère and Joux, 2003] Faugère, J.-C. and Joux, A. (2003). Algebraic cryptanalysis of hidden field equation (HFE) cryptosystems using Gröbner bases. *CRYPTO 2003*, 2729:44–60. 1

[Faugère et al., 2013] Faugère, J.-C., Safey El Din, M., and Verron, T. (2013). On the complexity of computing Gröbner bases for quasi-homogeneous systems. In *Proceedings of the 38th International Symposium on Symbolic and Algebraic Computation*, ISSAC '13, pages 189–196, New York, NY, USA. ACM. 2.5.3

[Gavinsky, 2012] Gavinsky, D. (2012). Quantum money with classical verification. In *Proceedings of the 27th Conference on Computational Complexity, CCC 2012, Porto, Portugal, June 26-29, 2012*, pages 42–52. IEEE. 1.1, 2.1

[Geiselmann et al., 2003] Geiselmann, W., Meier, W., and Steinwandt, R. (2003). An attack on the isomorphisms of polynomials problem with one secret. *Int. J. Inf. Sec.*, 2(1):59–64. 1

[Giusti, 1984] Giusti, M. (1984). Some effectivity problems in polynomial ideal theory. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation*, volume 174 of *Lecture Note in Computer Science*, pages 159–171, Cambridge,UK. Springer. 2.5.3

[Giusti and Lazard, 1983] Giusti, M. and Lazard, D. (1983). Complexity of standard basis computations, related algebraic problems and their common double exponential behaviour. In *Lecture Notes for the Conference "Computer and Commutative Algebra"*, Geneve, Italy. Publication du Centre de Mathématiques de l'École Polytechnique. 2.5

[Güneysu et al., 2012] Güneysu, T., Lyubashevsky, V., and Pöppelmann, T. (2012). Practical lattice-based cryptography: A signature scheme for embedded systems. *CHES 2012*, pages 530–547. 1, 2.3

[Goldreich et al., 1997] Goldreich, O., Goldwasser, S., and Halevi, S. (1997). Public-key cryptosystems from lattice reduction problems. In Kaliski, Burton S., J., editor, *Advances in Cryptology - CRYPTO 1997*, volume 1294 of *Lecture Notes Comput. Sci.*, pages 112–131. Springer Berlin Heidelberg. 2.3

[Google, 2018] Google (Retrieved in 2018). Quantum AI. https://ai.google/research/teams/applied-science/quantum-ai/. 1

[Grover, 1997] Grover, L. K. (1997). Quantum mechanics helps in searching for a needle in a haystack. *Physical Review Letters*, 79:325–328. 1

[IBM, 2018] IBM (Retrieved in 2018). The future is quantum. https://www.research.ibm.com/ibm-q/. 1

[Institute for Quantum Computing, 2018a] Institute for Quantum Computing (Retrieved in 2018). ETSI/IQC quantum safe workshop 2017. September 13-15, 2017,London, UK. http://www.etsi.org/news-events/events/1173-etsi-iqc-quantum-safe-workshop-2017. 1

[Institute for Quantum Computing, 2018b] Institute for Quantum Computing (Retrieved in 2018). ETSI/IQC quantum safe workshop 2018. November 6-8, 2018, Beijing, China. http://www.etsi.org/news-events/events/1296-etsi-iqc-quantum-safe-workshop-2018. 1

[Karp, 1972] Karp, R. M. (1972). Reducibility among combinatorial problems. In Miller, R. E. and Thatcher, J. W., editors, *Complexity of Computer Computations*, pages 85–103. Plenum, New York. 2.5

[Kawachi et al., 2007] Kawachi, A., Tanaka, K., and Xagawa, K. (2007). Multi-bit cryptosystems based on lattice problems. In *Public Key Cryptography - PKC 2007*, pages 315–329. 2.3

[Koblitz, 1987] Koblitz, N. (1987). Elliptic curve cryptosystems. *Mathematics of Computation*, 48:203–209. 1

[Kollreider and Buchberger, 1978] Kollreider, C. and Buchberger, B. (1978). An improved algorithmic construction of Gröbner-bases for polynomial ideals. *ACM SIGSAM Bulletin*, 12(2):27–36. 2.5.2

[Lamport, 1979] Lamport, L. (1979). Constructing digital signatures from a one way function. *Technical Report SRI-CSL-98, SRI International Computer Science Laboratory.* 1

[Lazard, 1983] Lazard, D. (1983). Gröbner bases, Gaussian elimination and resolution of systems of algebraic equations. In Hulzen, J., editor, *Computer Algebra*, volume 162 of *Lecture Notes in Computer Science*, pages 146–156. Springer Berlin Heidelberg. 2.5.2, 2.5.2, 2.36, 2.5.3

[Le Gall, 2014] Le Gall, F. (2014). Powers of tensors and fast matrix multiplication. In *Proceedings of the 39th International Symposium on Symbolic and Algebraic Computation*, ISSAC '14, pages 296–303, New York, NY, USA. ACM. 2.2

[Lutomirski, 2010] Lutomirski, A. (2010). An online attack against Wiesner's quantum money. Quantum physics, arXiv on-line archive. https://arxiv.org/abs/1010.0256. 2.1

[Lutomirski, 2011] Lutomirski, A. (2011). Component mixers and a hardness result for counterfeiting quantum money. Quantum physics, arXiv on-line archive. https://arxiv.org/abs/1107.0321. 1.1, 2.1

[Lutomirski et al., 2010] Lutomirski, A., Aaronson, S., Farhi, E., Gosset, D., Kelner, J. A., Hassidim, A., and Shor, P. W. (2010). Breaking and making quantum money: Toward a new quantum cryptographic protocol. In *Innovations in Computer Science - ICS 2010, Tsinghua University, Beijing, China, January 5-7, 2010. Proceedings*, pages 20–31. 2.1

[Macario-Rat et al., 2013] Macario-Rat, G., Plut, J., and Gilbert, H. (2013). New insight into the isomorphism of polynomial problem IP1S and its use in Cryptography. In Sako, K. and Sarkar, P., editors, *Advances in Cryptology - ASIACRYPT 2013*, volume 8269 of *Lecture Notes Comput. Sci.*, pages 117–133. Springer Berlin Heidelberg. 1

[Macaulay, 1994] Macaulay, F. S. (1994). *The algebraic theory of modular systems.* Cambridge University Press, Cambridge, UK. Revised reprint of the 1916 original. 2.5.2

[Matsumoto and Imai, 1988] Matsumoto, T. and Imai, H. (1988). Public quadratic polynomial-tuples for efficient signature-verification and message-encryption. In Günther, C. G., editor, *Advances in Cryptology - EUROCRYPT '88*, volume 330 of *Lecture Notes Comput. Sci.*, pages 419–453. Springer Berlin Heidelberg. 1

[Mayr and Meyer, 1982] Mayr, E. W. and Meyer, A. R. (1982). The complexity of the word problem for commutative semigroups and polynomial ideals. *Adv. in Maths.*, 46:305–329. 2.5

[McEliece, 1978] McEliece, R. J. (1978). A public-key cryptosystem based on algebraic coding theory. Technical Report 42-44, Jet Propulsion Laboratory. 1

[Merkle, 1979] Merkle, R. C. (1979). *Secrecy, authentication, and public key systems.* PhD thesis, Stanford University. 1

[Microsoft, 2018] Microsoft (Retrieved in 2018). Get the quantum edge with Azure. https://www.microsoft.com/en-us/quantum/. 1

[Miller, 1985] Miller, V. S. (1985). Use of elliptic curves in cryptography. In H.C., W., editor, *Conference on the Theory and Application of Cryptographic Techniques - CRYPTO 1985*, volume 218 of *Lecture Notes Comput. Sci.*, pages 417–426. Springer Berlin Heidelberg. 1

[Molina et al., 2012] Molina, A., Vidick, T., and Watrous, J. (2012). Optimal counterfeiting attacks and generalizations for Wiesner's quantum money. Quantum physics, arXiv on-line archive. https://arxiv.org/abs/1202.4010. 1.1, 2.1

[Mosca and Stebila, 2010] Mosca, M. and Stebila, D. (2010). Quantum coins. *Error-Correcting Codes, Finite Geometry and Cryptography*, 523:35–47. 1.1, 2.1

[Nakamoto, 2008] Nakamoto, S. (2008). Bitcoin P2P e-cash paper. 1

[Nakamoto, 2009] Nakamoto, S. (2009). Bitcoin: A peer-to-peer electronic cash system. 1

[Niederreiter, 1986] Niederreiter, H. (1986). A public-key cryptosystem based on algebraic coding theory. *Problems of Control and Information Theory*, 15:19–34. 1

[NIST, 2018] NIST (2018). Post-quantum cryptography project. https://csrc.nist.gov/Projects/Post-Quantum-Cryptography. 1

[Park, 1970] Park, J. L. (1970). The concept of transition in quantum mechanics. *Foundations of Physics*, 1(1):23–33. 1

[Pastawski et al., 2011] Pastawski, F., Yao, N. Y., Jiang, L., Lukin, M. D., and Cirac, J. I. (2011). Unforgeable noise-tolerant quantum tokens. Quantum physics, arXiv on-line archive. https://arxiv.org/abs/1112.5456. 1.1, 2.1

[Patarin, 1996] Patarin, J. (1996). Hidden fields equations (HFE) and isomorphisms of polynomials (IP): Two new families of asymmetric algorithms. In Maurer, U., editor, *Advances in Cryptology - EUROCRYPT 1996*, volume 1070 of *Lecture Notes Comput. Sci.*, pages 33–48. Springer Berlin Heidelberg. 1

[Patarin et al., 1998] Patarin, J., Goubin, L., and Courtois, N. (1998). Improved algorithms for isomorphisms of polynomials. In Nyberg, K., editor, *Advances in Cryptology - EUROCRYPT 1998*, volume 1403 of *Lecture Notes Comput. Sci.*, pages 184–200. Springer Berlin Heidelberg. 1

[Peikert, 2009] Peikert, C. (2009). Public-key cryptosystems from the worst-case shortest vector problem. In *Proceedings of the 41st annual ACM symposium on Theory of computing*, pages 333–342. ACM. 2.3

[Peikert, 2014] Peikert, C. (2014). Lattice cryptography for the Internet. *PQCrypto 2014*, pages 197–219. 1, 2.3

[Peikert et al., 2008] Peikert, C., Vaikuntanathan, V., and Waters, B. (2008). A framework for efficient and composable oblivious transfer. In *CRYPTO 2008*, pages 554–571. 2.3

[Perret, 2005] Perret, L. (2005). A fast cryptanalysis of the isomorphism of polynomials with one secret problem. In Cramer, R., editor, *Advances in Cryptology - EUROCRYPT 2005*, volume 3494 of *Lecture Notes Comput. Sci.*, pages 354–370. Springer Berlin Heidelberg. 1

[Regev, 2009] Regev, O. (2009). On lattices, learning with errors, random linear codes, and cryptography. *J. ACM*, 56(6):34:1–34:40. 2.3

[Rivest et al., 1978] Rivest, R., Shamir, A., and Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21:120–126. 1

[Shannon, 1948] Shannon, C. E. (1948). A mathematical theory of communications. *The Bell System Technical Journal*, 27:623–656. 1

[Shor, 1997] Shor, P. W. (1997). Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.*, 26(5):1484–1509. 1

[Singh, 2011] Singh, S. (2011). *The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography*. Knopf Doubleday Publishing Group. 1

[Strassen, 1969] Strassen, V. (1969). Gaussian elimination is not optimal. *Numerische Mathematik*, 13(4):354–356. 2.2

[Wiesner, 1983] Wiesner, S. (1983). Conjugate coding. *ACM SIGACT News*, 15(1):78–88. 1, 1.1

[Wootters and Zurek, 1982] Wootters, W. K. and Zurek, W. H. (1982). A single quantum cannot be cloned. *Nature*, 299:802–803. 1

[Wootters and Zurek, 2009] Wootters, W. K. and Zurek, W. H. (2009). The no-cloning theorem. *Physics Today*, pages 76–77. 1

[Zhandry, 2017] Zhandry, M. (2017). Quantum lightning never strikes the same state twice. Cryptology ePrint Archive, Report 2017/1080. `https://eprint.iacr.org/2017/1080`. 5.3

[Zhang et al., 2015] Zhang, J., Zhang, Z., Ding, J., Snook, M., and Özgür Dagdelen (2015). Authenticated key exchange from ideal lattices. *EUROCRYPT 2015*, pages 719–751. 1