# 1. Einleitung

## 1.1. Motivation des McEliece-Kryptosystems

Computertechnologie ist ein bedeutender Teil des täglichen Lebens. Umso wichtiger ist es, die mittels Computer verarbeiteten Daten vor dem Zugriff unbefugter Dritter zu schützen. Dies betrifft sensible Bankdaten, Patientendaten und persönliche Fotos gleichermaßen. Aktuell werden klassische kryptografische Verfahren wie RSA als asymmetrische kryptografische Verfahren genutzt. Mittels dieser Verfahren können geheime Schlüssel über einen unsicheren Kanal ausgetauscht und für schnellere symmetrische Verschlüsselungsverfahren wie AES genutzt werden.

Leistungsfähige Quantencomputer könnten in Zukunft in der Lage sein, klassische asymmetrische Verfahren, die auf dem Faktorisierungsproblem oder dem *diskreten Logarithmus-Problem (DLP)* basieren, zu brechen [19]. Damit ist die Vertraulichkeit jeder digital übertragenen Information gefährdet. *Post-Quanten-Kryptografie (PQC)* beschäftigt sich mit der Entwicklung quantensicherer Verschlüsselung auf nicht spezialisierter Hardware.

Das in dieser Arbeit vorgestellte *Classic McEliece-Kryptosystem* ist ein als besonders sicher eingestuftes Post-Quanten-Kryptosystem, das vom *Bundesamt für Sicherheit in der Informationstechnik (BSI)* empfohlen wird und im NIST Standardisierungsprozess für Post-Quanten-Kryptografie in der Runde der Finalisten war. Bei dem Verfahren handelt es sich um „ein codebasiertes Schlüsseltransportverfahren, basierend auf Niederreiters Variante, [...] instanziiert mit binären Goppa Codes“[11]. „Dessen Sicherheit basiert auf zwei Annahmen. Die erste Annahme ist, dass die verwendeten binären Goppa Codes nicht unterscheidbar von zufälligen linearen Codes sind. Die zweite Annahme ist, dass zufällige lineare Codes aufgrund des General-Decoding-

Problems sowohl auf Digitalrechnern als auch mithilfe von Quantencomputern nicht effizient decodiert werden können“[11]. Das Verfahren gehört mit seiner Erfindung im Jahre 1978 zu den ältesten ungebrochenen quantensicheren Verschlüsselungsverfahren. Obwohl es sich im NIST Standardisierungsprozess nicht durchsetzen konnte, bietet es eine Alternative zu dem von der NIST zugelassenen gitterbasierten Algorithmus Crystal-Kyber [5] und wird vom BSI weiterhin empfohlen.

## 1.2. Struktur der Arbeit

Die Arbeit ist in 5 Kapitel gegliedert.

In Kapitel 1 werden Voraussetzungen und Notationen eingeführt.

In Kapitel 2 wird ein kurzer Überblick über Quantencomputer und die Relevanz von Post-Quanten-Kryptografie gegeben. Es werden die Folgen der Algorithmen nach Shor und Grover für die Kryptografie aufgezeigt und das McEliece-Kryptosystem im Feld der quantensicheren Kryptografie eingeordnet.

In Kapitel 3 werden Goppa Codes definiert und grundlegende Eigenschaften dieser Codeklasse bewiesen. Neben der Kontrollmatrix wird die Dimension und der Minimalabstand beliebiger Goppa Codes hergeleitet. Anschließend wird eine bessere untere Schranke für die Minimaldistanz quadratfreier binärer Goppa Codes bewiesen.

In Abschnitt 3.3 erfolgt die Herleitung des Decodieralgorithmus nach Sugiyama für allgemeine Goppa Codes und des Decodieralgorithmus nach Patterson für irreduzible binäre Goppa Codes.

In Kapitel 4 wird die Schulbuchversion des McEliece-Kryptosystems und der Variante nach Niederreiter, inklusive Optimierung vorgestellt.

Nach der Vorstellung der Systeme findet in Abschnitt 4.3 ein Vergleich des McEliece-Verfahrens und der Variante nach Niederreiter statt.

In Abschnitt 4.4 wird anschließend jeweils ein konkretes Beispiel zur Verwendung der Systeme präsentiert. Das Kapitel endet mit einer Sicherheitsanalyse der Verfahren, welche in verfahrensunabhängige Angriffe und die Vorstellung eines Message-Resend-Angriffs auf das McEliece-Kryptosystem gegliedert ist.

In Abschnitt 4.6 werden kurz die Besonderheiten des bei der NIST eingereichten Classic McEliece Systems und dessen Parameterempfehlungen sowie Vor- und Nachteile diskutiert.

In Kapitel 5 wird die Arbeit mit einem kurzen Ausblick abgeschlossen.

In Anhang A findet sich eine Übersicht über die mathematischen Herleitungen als Unterstützung zum Verstehen der groben Beweisstruktur.

In Anhang B ist eine Referenzimplementierung des McEliece- und Niederreiter-Systems angegeben, mit der auch das Beispiel aus Abschnitt 4.4 erstellt wurde.

In Anhang C wird eine mögliche Konvertierung des McEliece-Kryptosystems hin zu einer CCA-2 sicheren Variante vorgestellt.

In Abbildung 1.1 ist der Zusammenhang der einzelnen Kapitel untereinander nochmals grafisch hervorgehoben.

## 1.3. Voraussetzungen

Dem Leser[1] sollten Grundlagen der Codierungstheorie, Algebra und Zahlentheorie bekannt sein. Insbesondere Aussagen über Lineare Codes und die Konstruktion und Darstellung der Elemente endlicher Körper (Galoiskörper) werden vorausgesetzt. Kenntnisse über zyklische Codes, BCH-Codes und RS-Codes sind zum Verständnis von Nutzen, jedoch nicht notwendig.

## 1.4. Notationen

Die hier vorgestellten Bezeichnungen werden in der gesamten Ausarbeitung genutzt, sofern sie an den entsprechenden Stellen nicht auf eine andere Art definiert worden sind.

Es bezeichnet $\mathbb{F}_q$ den endlichen Körper mit $q$ Elementen und $\mathbb{F}_{q^m}$ mit $m \in \mathbb{N}$ seinen Erweiterungskörper mit $q^m$ Elementen.[2] Beachte den Unterschied

[1] Aus Gründen der besseren Lesbarkeit wird das generische Maskulinum verwendet. Alle Personenbezeichnungen gelten damit gleichermaßen für alle Geschlechter.

[2] Die Anzahl der Elemente eines endlichen Körpers ist immer eine Primzahlpotenz und endliche Körper sind bis auf Isomorphie eindeutig. Das heißt für einen endlichen Körper mit $q$ Elementen existiert eine Primzahl $p$ und eine natürliche Zahl $m$, sodass $q = p^m$ gilt.

zwischen $\mathbb{F}_{q^m}$ und $\mathbb{F}_q^n$. Ersteres bezeichnet den Körper mit $q^m$ Elementen, wohingegen Letzteres einen Vektorraum der Dimension $n$ über dem Körper $\mathbb{F}_q$ mit $q$ Elementen bezeichnet.

Für einen Körper $K$ bezeichnet $K[x]$ die Menge aller Polynome beliebigen Grades in der Unbekannten $x$, mit Koeffizienten aus $K$ und $\deg(f)$ bezeichnet den Grad eines Polynoms $f(x)$ in $x$. Die Menge aller Polynome aus $K[x]$ vom Grad $t$ wird als $K[x]_t$ definiert. Im Verlauf der Ausarbeitung wird das Goppa Polynom $g(x)$ als Polynom in der Variablen $x$ vom Grad $t$ definiert.

Die Parameter eines linearen Codes werden in dieser Arbeit mit $[n, k, d]_q$ bezeichnet. Dabei ist $n$ die Länge der Codewörter, $k$ die Anzahl der informationstragenden Symbole bzw. die Dimension des Codes, $d$ die Minimaldistanz des Codes und $q$ die Anzahl der Elemente des zugrundeliegenden endlichen Galoiskörpers ($q = |K|$).

Mit $\mathrm{wt}(C)$ wird das Minimalgewicht des Codes $C$ und mit $\mathrm{wt}(c)$ das Hamminggewicht eines Codewortes $c$ bezeichnet. Für lineare Codes sind die Minimaldistanz und das Minimalgewicht identisch, d.h. es gilt $\mathrm{wt}(C) = d(C)$ für alle linearen Codes $C$. Es sei darauf hingewiesen, dass $C$ den gesamten Code und $c$ ein Codewort aus dem Code $C$ bezeichnet. In der Nomenklatur des Codes wird $H$ für die Kontrollmatrix und $G$ für die Generatormatrix verwendet.

Bei der Übertragung von Nachrichten wird die uncodierte Nachricht (message) als $m$, die fehlerfrei codierte Nachricht (code word) als $c$ und ein Fehlervektor (error) mit Hammingdistanz höchstens $t$ als $e$ bezeichnet. Ein Empfänger erhält den Vektor $r$ (received), der oftmals die Summe der codierten Nachricht $c$ und des Fehlervektors $e$ ist. Als *Decodierung* wird der Prozess der Fehlerkorrektur (Berechnung von $c = r - e$) inklusive der Decodierung des fehlerfreien Codewortes $c$ hin zu einer uncodierten Nachricht $m$ bezeichnet.

Als *klassischer Computer* oder *Computer* wird ein auf Bits basierender Computer bezeichnet; als *Quantencomputer* dementsprechend ein Computer, der auf Qubits basiert. Algorithmen, die auf klassischen Computern eine polynomielle Laufzeit haben, werden *effiziente Algorithmen* genannt. Alle anderen Algorithmen werden dementsprechend als *ineffizient* bezeichnet und Probleme, für die ein polynomieller Algorithmus existiert (egal ob auf Quan-

tencomputern oder klassischen Computern), werden als *gebrochen* bezeichnet. Die vorgestellten Versionen des McEliece- und Niederreiter-Kryptosystems werden als *Schulbuchversionen* bezeichnet, da sie nur die strukturellen Ideen der Verfahren enthalten und nicht in der Praxis einsetzbar sind. Das *Classic McEliece-Verfahren* ist dagegen keine Schulbuchversion.

Für unbekannte Akronyme oder Abkürzungen sei auf die Liste der Akronyme am Ende der Ausarbeitung verwiesen.

Nachdem hiermit alle Notationen eingeführt wurden, beginnt im Folgenden der inhaltliche Teil der Arbeit.

# 3. Goppa Codes

## 3.1. Einleitung

Goppa Codes sind für Kryptosysteme von besonderer Bedeutung, da für sie im Gegensatz zu vielen anderen Codeklassen weiterhin die Annahme der Ununterscheidbarkeit von zufälligen linearen Codes gilt. Kombiniert mit der zweiten Annahme, dass zufällige lineare Codes aufgrund des General-Decoding-Problems (sowohl auf Digitalrechnern als auch mithilfe von Quantencomputern) nicht effizient decodiert werden können,[1] ist es möglich, mit einem effizienten Decodierverfahren für Goppa Codes eine quantensichere Einwegfunktion und damit ein quantensicheres Kryptosystem zu konstruieren. Der Kern dieses Kapitels ist es ein effizientes Decodierverfahren für Goppa Codes herzuleiten. Dazu werden Goppa Codes definiert und grundlegende Eigenschaften bewiesen. Das Kapitel ist angelehnt an die Arbeiten von Goppa [12], Huffman et al. [13], Baldoni et al. [1] und MacWilliams et al. [16].

Goppa Codes wurden 1970 vom russischen Mathematiker Valery Goppa in seinem Paper „A new class of linear correcting codes“ eingeführt [12]. Nach Goppa hat diese Codeklasse die besonderen Vorteile, wie zyklische Codes durch ein Generatorpolynom spezifiziert zu sein. Doch im Gegensatz zu zyklischen Codes erlaube der Grad des Generatorpolynoms eine Abschätzung der Parameter eines Goppa Codes. Die einzigen zyklischen Codes, die nach Goppa ebenfalls diese Eigenschaft aufweisen, seien BCH-Codes, die durch Goppa Codes verallgemeinert würden (vgl. [12]).

[1] Da es sich hierbei um Annahmen handelt, sind die beiden Aussagen bisher nicht bewiesen oder widerlegt worden. Sie bilden die Grundlage für die Sicherheit des McEliece Kryptosystems.

## 3.2. Definition und Parameter von Goppa Codes

Goppa Codes der Länge $n$ über $\mathbb{F}_q$ sind wie folgt definiert:[2]

**Definition 3.1** (Goppa Codes)**.** Sei $\mathbb{F}_q$ ein endlicher Körper und $m \in \mathbb{N}$ beliebig.

Es seien $L = (\alpha_1, \ldots, \alpha_n)$ ein $n$-Tupel paarweise verschiedener Elemente aus $\mathbb{F}_{q^m}$ und $g(x) \in \mathbb{F}_{q^m}[x]$ ein normiertes Polynom mit $g(\alpha_i) \neq 0$ für $i = 1, \ldots, n$. Dann heißt der lineare Code

$$\Gamma(L, g) := \left\{ c = (c_1, \ldots, c_n) \in \mathbb{F}_q^n : R_c(x) = \sum_{i=1}^{n} \frac{c_i}{x - \alpha_i} \equiv 0 \bmod g(x) \right\}$$

über $\mathbb{F}_q$ (klassischer) *Goppa Code* der Länge $n$ zum *Goppa-Polynom* $g(x)$. $L$ wird auch *Support* des Goppa Codes genannt.

Ist $g(x)$ zudem irreduzibel, so wird $\Gamma(L, g)$ als *irreduzibler Goppa Code* bezeichnet.

Es ist zu bemerken, dass das Inverse von $x - \alpha_i \bmod g(x)$ existiert, da aus $g(\alpha_i) \neq 0$ folgt, dass der ggT von $(x - \alpha_i)$ und $g(x)$ gleich 1 ist. Aus dem Lemma von Bézout folgt dann die Existenz.

**Satz 3.2** (Goppa Codes sind lineare Codes)**.** Goppa Codes sind lineare Codes, da

$$R_{ac}(x) = \sum_{i=1}^{n} \frac{ac_i}{x - \alpha_i} = a \sum_{i=1}^{n} \frac{c_i}{x - \alpha_i} \equiv a \cdot 0 \equiv 0 \bmod g(x)$$

für alle $a \in \mathbb{F}_q, c \in \Gamma(L, g)$ gilt und

$$R_{c_1+c_2}(x) = \sum_{i=1}^{n} \frac{c_1 + c_2}{x - \alpha_i} = \sum_{i=1}^{n} \frac{c_1}{x - \alpha_i} + \sum_{i=1}^{n} \frac{c_2}{x - \alpha_i} \equiv 0 + 0 \equiv 0 \bmod g(x)$$

für alle $c_1, c_2 \in \Gamma(L, g)$ gilt.

[2] Beachte, dass im Originalpaper nur binäre Goppa Codes betrachtet werden, die erst in späteren Papern generalisiert wurden. Hier wird eine Definition über beliebigen Galoiskörpern gegeben.

### 3.2.1. Kontrollmatrix (und Generatormatrix)

Im vorherigen Abschnitt wurde gezeigt, dass Goppa Codes lineare Codes sind. Da für jeden linearen Code eine Generator- und Kontrollmatrix existiert, wird im Folgenden die Kontrollmatrix von Goppa Codes hergeleitet.[3]

Die Existenz von $(x - \alpha_i)^{-1}$ aus der Definition von Goppa Codes wurde bereits gezeigt. Das Inverse Element zu $(x - \alpha_i)^{-1}$ lässt sich mittels des erweiterten euklidischen Algorithmus herleiten und ist nach Huffman et al. [13] wie folgt angeben

$$\frac{1}{x-\alpha_i} = 1 \cdot \frac{1}{x-\alpha_i} \equiv \left(\frac{g(\alpha_i)}{g(\alpha_i)} - \frac{g(x)}{g(\alpha_i)}\right) \frac{1}{x-\alpha_i} \equiv -\frac{1}{g(\alpha_i)} \frac{g(x) - g(\alpha_i)}{x-\alpha_i} \bmod g(x). \tag{3.1}$$

Ersetzt man in Definition 3.1 den Bruch $\frac{1}{x-\alpha_i}$ gemäß Gleichung 3.1, so ist $c$ genau dann ein Codewort, wenn[4]

$$R_c(x) = \sum_{i=1}^{n} \frac{c_i}{x-\alpha_i} \equiv \sum_{i=1}^{n} c_i \frac{g(x) - g(\alpha_i)}{x-\alpha_i} g(\alpha_i)^{-1} \equiv 0 \bmod g(x) \tag{3.2}$$

gilt. In Gleichung 3.2 kann die modulo Operation weggelassen werden, da der Grad von

$$\sum_{i=1}^{n} c_i \frac{g(x) - g(\alpha_i)}{x-\alpha_i} g(\alpha_i)^{-1}$$

durch die Division von $g(x)$ durch $x - \alpha_i$ stets kleiner ist als der von $g(x)$. Ein Codewort $c$ ist also genau dann in $\Gamma(L, g)$ enthalten, wenn

$$\sum_{i=1}^{n} c_i \frac{g(x) - g(\alpha_i)}{x-\alpha_i} g(\alpha_i)^{-1} = 0 \tag{3.3}$$

gilt. Nun wird der Bruch

$$\frac{g(x) - g(\alpha_i)}{x-\alpha_i}$$

genauer analysiert, indem ein Goppa-Polynom eines spezifischen Grades ange-

[3] Die Kontrollmatrix statt der Generatormatrix herzuleiten ist durch die Definition von Goppa Codes motiviert.

[4] Man kann das Minus weglassen, da $0 \equiv -0$ gilt.

nommen wird. Sei $g \in \mathbb{F}_{q^m}[x]$ ein Goppa-Polynom vom Grad $t$. Dann hat g die Form

$$g = \sum_{j=0}^{t} g_j x^j, \text{ mit } g_t = 1.$$

Die Ersetzung von $g(x)$ und $g(\alpha_i)$ im betrachteten Bruch führt zu

$$\begin{aligned}
&\frac{g(x) - g(\alpha_i)}{x - \alpha_i} = \sum_{j=0}^{t} g_j \frac{(x^j - \alpha_i^j)}{x - \alpha_i} \\
&\overset{5}{=} \sum_{j=0}^{t} g_j \sum_{k=0}^{j-1} x^k \alpha_i^{j-1-k} \\
&= g_0 \cdot 0 + g_1 + g_2(\alpha_i^1 + x^1) + \cdots + g_t(\alpha_i^{t-1} + \alpha_i^{t-2}x + \cdots + x^{t-1}) \\
&= \sum_{k=0}^{t-1} x^k \sum_{j=k+1}^{t} g_j \alpha_i^{j-1-k}. \qquad (3.4)
\end{aligned}$$

Das Einsetzen von Gleichung 3.4 in Gleichung 3.3 liefert

$$\begin{aligned}
0 &= \sum_{i=1}^{n} c_i \frac{g(x) - g(\alpha_i)}{x - \alpha_i} g(\alpha_i)^{-1} \\
&= \sum_{i=1}^{n} c_i g(\alpha_i)^{-1} \sum_{k=0}^{t-1} x^k \sum_{j=k+1}^{t} g_j \alpha_i^{j-1-k} \\
&\overset{6}{=} \sum_{k=0}^{t-1} \left( \sum_{i=1}^{n} c_i g(\alpha_i)^{-1} \sum_{j=k+1}^{t} g_j \alpha_i^{j-1-k} \right) x^k.
\end{aligned}$$

Nach einem Koeffizientenvergleich gilt genau dann, dass $c \in \Gamma(L, g)$, wenn alle Koeffizienten der $x^k$ gleich Null sind, d.h. in Formeln muss gelten

[5] Bemerke, dass $x^j - a^j = (x - a) \cdot \left(a^{j-1} + xa^{j-2} + x^2a^{j-3} + \ldots + x^{j-2}a + x^{j-1}\right)$ gilt.

[6] Bemerke, dass hier wieder nur eine Vertauschung der Summationsreihenfolge vorgenommen wurde, damit die Summe als Linearkombination der $x^k$ dargestellt ist.

$$c \in \Gamma(L,g) \Leftrightarrow \forall\, 0 \leq k \leq t-1 : \sum_{i=1}^{n} \left( c_i g(\alpha_i)^{-1} \sum_{j=k+1}^{t} \left( g_j \alpha_i^{j-1-k} \right) \right) = 0 \quad (3.5)$$

$$\Leftrightarrow Hc^T = 0, \text{ mit}$$

$$H = \begin{bmatrix} g(\alpha_1)^{-1} g_t & g(\alpha_2)^{-1} g_t & \cdots & g(\alpha_n)^{-1} g_t \\ g(\alpha_1)^{-1} \left( g_{t-1} + g_t \alpha_1 \right) & g(\alpha_2)^{-1} \left( g_{t-1} + g_t \alpha_2 \right) & \cdots & g(\alpha_n)^{-1} \left( g_{t-1} + g_t \alpha_n \right) \\ \vdots & \vdots & \cdots & \vdots \\ g(\alpha_1)^{-1} \sum_{j=1}^{t} g_j \alpha_1^{j-1} & g(\alpha_2)^{-1} \sum_{j=1}^{t} g_j \alpha_2^{j-1} & \cdots & g(\alpha_n)^{-1} \sum_{j=1}^{t} g_j \alpha_n^{j-1} \end{bmatrix}.$$

Die Struktur der Matrix ergibt sich direkt aus Gleichung 3.5, wobei der Parameter $k$ beginnend bei $(t-1)$ bis 0 iteriert wird. Dabei entsprechen die Zeilen der Matrix der jeweiligen Iteration der $t-1 \geq k \geq 0$ und innerhalb einer Zeile ist die Summe

$$\sum_{i=1}^{n} \left( c_i g(\alpha_i)^{-1} \sum_{j=k+1}^{t} \left( g_j \alpha_i^{j-1-k} \right) \right)$$

als Iteration über die $1 \leq i \leq n$ dargestellt. Die Matrix $H$ kann nach MacWilliams et al. [16] wiederum als Produkt der Matrizen

$$H = \begin{bmatrix} g_t & 0 & \cdots & 0 \\ g_{t-1} & g_t & \cdots & 0 \\ g_{t-2} & g_{t-1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_1 & g_2 & \cdots & g_t \end{bmatrix} \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \alpha_1 & \alpha_2 & \cdots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \cdots & \alpha_n^2 \\ \vdots & \vdots & \cdots & \vdots \\ \alpha_1^{t-1} & \alpha_2^{t-1} & \cdots & \alpha_n^{t-1} \end{bmatrix} \begin{bmatrix} g(\alpha_1)^{-1} & & 0 \\ & \ddots & \\ 0 & & g(\alpha_n)^{-1} \end{bmatrix}$$
$$:= CXY$$

geschrieben werden. Diese Produktschreibweise lässt sich durch Ausmultiplizieren verifizieren. Dabei entspricht die Multiplikation mit $Y$ von rechts

# 4. Das McEliece-Kryptosystem und seine Varianten

Aufbauend auf den Erkenntnissen über Goppa Codes ist es nun möglich das McEliece-Kryptosystem und seine Varianten einzuführen. Bei dem Verfahren handelt es sich um ein codebasiertes asymmetrisches Verschlüsselungsverfahren, das auf den folgenden zwei Annahmen beruht.

1. Binäre Goppa Codes sind nicht unterscheidbar von zufälligen linearen Codes.

2. Zufällige lineare Codes sind aufgrund des General-Decoding-Problems sowohl auf Digitalrechnern als auch Quantencomputern nicht effizient decodierbar.

Das General-Decoding-Problem, auf dem die kryptografische Einwegfunktion des McEliece-Verfahrens basiert, ist wie folgt definiert.

**Definition 4.1** (General Decoding)**.** Gegeben sei ein empfangenes Wort $x$, decodiere $x$ zu dem am dichtesten liegenden Codewort $c$ eines beliebigen Codes (bzgl. der Hamming-Distanz).

**Bemerkung 4.2.** Sei $x = c + e$, wobei $e$ den Übertragungsfehler bezeichnet, so ist das Finden von $e$ äquivalent zum General-Decoding-Problem.

Es existieren viele Codefamilien mit effizienten Decodierverfahren (zum Beispiel Goppa Codes), aber das allgemeine General-Decoding-Problem ist NP-schwer [2]. Mittels eines effizienten Decodieralgorithmus für Goppa Codes und der Schwere des General-Decoding-Problems ist die Grundlage für die kryptografische Einwegfunktion des McEliece-Kryptosystems gegeben.

# 5. Zusammenfassung und Ausblick

In dieser Ausarbeitung wurde die Codeklasse der Goppa Codes mit ihren grundlegenden Eigenschaften und möglichen Decodieralgorithmen vorgestellt. Aufbauend darauf wurden sowohl das McEliece-, das Niederreiter- und das Classic McEliece-Kryptosystem eingeführt und deren Sicherheitseigenschaften analysiert. Codebasierte Verfahren gelten als besonders sicher, haben sehr kurze Chiffrate und ermöglichen eine schnellere Ver- und Entschlüsselung als RSA, was sie zukünftig besonders attraktiv für kryptografische Anwendungen auf mobilen Endgeräten macht.

Trotz des schnell steigenden, verfügbaren Speicherplatzes stellen sich die großen öffentlichen Schlüssel jedoch als Herausforderung für eine Verbreitung codebasierter Kryptosysteme heraus. Aktuell wird, um diesem Problem zu begegnen, die Verwendung von quasi-zyklischen (QC) moderate Density Parity-Check (MDPC)-Codes anstelle von Goppa Codes vorgeschlagen. In der Vergangenheit stellte sich jedoch für viele vorgeschlagene Codeklassen die Ununterscheidbarkeitsannahme von zufälligen linearen Codes als falsch heraus, sodass das Vertrauen in neue Codes begrenzt ist. Eine Analyse neuer Codeklassen mit kürzeren Schlüssellängen sollte damit Hauptgegenstand zukünftiger Forschung sein. Denn bei der Verwendung effizienterer Codes ergäbe sich eine höhere Praktikabilität codebasierter Kryptosysteme. Ein weiterer Vorteil wäre, dass sie als Backup dienen könnten, sollte sich die Ununterscheidbarkeitsannahme für Goppa Codes in der Zukunft als falsch erweisen.

# A. Anhang: Übersicht über die Herleitungen

Dieses Kapitel enthält Abbildungen, die eine Übersicht über die Beweisstruktur dieser Arbeit geben.

# B. Anhang: Implementierung in SageMath

In diesem Abschnitt werden die Kernelemente einer Implementierung des McEliece- und Niederreiter-Kryptosystems in SageMath vorgestellt. Die Implementierungen sind an das Whitepaper der TU Eindhoven [17] angelehnt und nutzen den Decodieralgorithmus nach Patterson (vgl. Algorithmus 3.2). Die Kontrollmatrix wird nur durch Multiplikation der Matrizen $X$ und $Y$ (ohne die invertierbare Matrix $C$) berechnet. Es sei auch darauf hingewiesen, dass durch die Implementierung der systematischen Form die Generatormatrix $G$ die Form $(G''|I_k)$ anstelle von $(I_k|G'')$ hat. Die Kontrollmatrix hat entsprechend die Form $(I_{n-k}|H'')$ anstelle von $(H''|I_{n-k})$.

## B.1. Implementierung des McEliece- und Niederreiter-Kryptosystems

Analog zu Abschnitt 4.1 besteht die Implementierung des McEliece-Kryptosystems aus einem Schlüsselerzeugungsalgorithmus (vgl. Implementierung B.1), einem Verschlüsselungsalgorithmus (vgl. Implementierung B.3) und einem Entschlüsselungsalgorithmus (vgl. Implementierung B.4).

Bei der Schlüsselerzeugung aus Implementierung B.1 ist es möglich die Parameter $m$ und $t$ des Goppa Codes und die Art, wie das Goppa-Polynom und der Support gewählt werden sollen, zu spezifizieren. In der Praxis wird das Goppa-Polynom immer zufällig gewählt. Zu Demonstrationszwecken ist es in dieser Implementierung auch möglich deterministische Wahlen des Po-

# C. Anhang: Konvertierung des McEliece-Kryptosystems

In diesem Kapitel wird die Konvertierung der Schulbuchversion des McEliece-Kryptosystems hin zu einer CCA-2 sicheren Variante nach Fujisaki und Okamoto präsentiert [10]. Die Ausführungen sind angelehnt an die Arbeit von Engelbert et al. [6]. Die Konvertierung nutzt Verschlüsselungsalgorithmus C.1 und Entschlüsselungsalgorithmus C.2 und folgt dabei den Notationen aus Tabelle C.1.

Tabelle C.1.: Bezeichnungen in den Konvertierungsalgorithmen

| | |
|---|---|
| $r$ | Zufallszahl |
| $R$ | Zufallszahlengenerator mit einem Seed fester Länge |
| Conv | Bijektive Abbildung zwischen den Zahlen $\mathbb{Z}\backslash\mathbb{Z}\binom{n}{t}$ und den entsprechenden Fehlervektoren der Länge $n$ vom Gewicht $t$. |
| $H$ | Hashfunktion mit Output der Länge $\log_2(\binom{n}{t})$ |
| $\mathcal{E}(m,z)=c$ | McEliece Verschlüsselung der Nachricht $m$ mit Fehlervektor $z$ |
| $D(c)=m$ | Entschlüsselungsfunktion $D$ des McEliece-Kryptosystems |
| $\mathrm{MSB}_n(m)$ | Die rechten $n$ Bits der Nachricht $m$ |
| $\mathrm{LSB}_n(m)$ | Die linken $n$ Bits der Nachricht $m$ |
| $\|$ | Konkatenation |

Das resultierende Chiffrat besteht aus

$$c = (c_1 \| c_2) = \left( \mathcal{E}\Big(r, \underbrace{\mathrm{Conv}(H(r\|m))}_{z}\Big) \| R(r) + m) \right).$$

Durch die Konvertierung werden Known-Plaintext, Related Message, Reaction