Studienarbeit

TECHNISCHE UNIVERSITÄT DARMSTADT

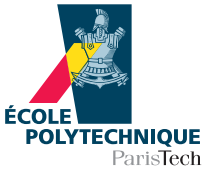
Ein modultheoretischer Beitrag zur sicheren Kommunikation

cand. ing. Christoph Hentschke

01. Juni 2009

REGELUNGSTHEORIE UND ROBOTIK rtr

Dipl.-Ing. K. Listmann



Prof. Dr. M. Fliess

Für meine Großmutter
Marianne Hentschke

Selbständigkeitserklärung

Hiermit erkläre ich, dass ich die von mir am heutigen Tage, dem Prüfungsausschuß des Fachbereichs Elektrotechnik und Informationstechnik der Technischen Universität Darmstadt, eingereichte Studienarbeit zum Thema

„Ein modultheoretischer Beitrag zur sicheren Kommunikation“

selbst verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie Zitate kenntlich gemacht habe.

Darmstadt, den 01. Juni 2009

Christoph Hentschke

Abstract

The consideration of a dynamical system as a modul enables a simple access to systemtheoretical properties. In this thesis a practical procedure for the verification of invertibility and controllability of piecewise linear discrete-time SISO systems is provided. Systems of this class are of interest for application in cryptography, because they include complex dynamics and an easy implementation at the same time. If a system complies with the mentioned properties, it provides the structural requirements for the application as a self-synchronizing stream cipher.

Zusammenfassung

Die Auffassung eines dynamischen Systems als Modul ermöglicht einen einfachen Zugang zu systemtheoretischen Eigenschaften. In dieser Arbeit wird ein beispielorientiertes Vorgehen zum Nachweis der Invertierbarkeit und Steuerbarkeit stückweise linearer diskreter SISO Systeme mit Hilfe der Modultheorie gezeigt. Systeme dieser Klasse sind interessant für Anwendungen im Bereich der Kryptographie, da sie eine komplexe Dynamik aufweisen, aber dennoch einfach zu realisieren sind. Erfüllt ein System die erwähnten Eigenschaften, besitzt es die strukturellen Voraussetzungen für die Verwendung zur selbstsynchronisierenden Stromverschlüsselung.

Résumé

Considérer un système dynamique comme un module permet un accès simple aux propriétés des systèmes. Dans ce manuscrit une procédure pour la vérification d'inversibilité et commandabilité des systèmes SISO linéaires par passages est présentée. Des systèmes de cette classe sont intéressants pour les applications en chryptographie car ils incluent une dynamique complexe ainsi qu'une implémentation simple. Si un système vérifie les propriétés mentionnées, ce système dispose de la structure requise pour une application de chiffrement par flot synchronisé autonome.

Danksagung

Angefangen mit der Organisation im Vorfeld, über die Betreuung während meines Aufenthaltes in Paris, bis hin zum Abschluss meiner Arbeit habe ich von vielen Seiten Unterstützung erhalten. Hierfür möchte ich mich herzlich bedanken!
Insbesondere bedanken möchte ich mich bei Herrn Listmann für dieses tolle Angebot, die Kontaktaufnahme zu Herr Prof. Fliess und die Betreuung seitens TU Darmstadt – ohne seinen Einsatz wäre dieser Aufenthalt nicht möglich gewesen. In gleicher Weise gilt mein Dank Herrn Prof. Fliess, der meine Betreuung und die Organisation an der École Polytechnique in Paris übernommen hat. Mit viel Geduld hat er mir die Grundlagen der Modultheorie nahegebracht und war für alle meine Fragen immer erreichbar.
Für den moralischen und auch finanziellen Beistand möchte ich mich bei meiner Mutter Gerda Hentschke und bei meiner Freundin Tanja Mayer, die mich in den Höhen und Tiefen meiner Arbeit begleitet haben, herzlich bedanken, ebenso bei meinen Freunden in Darmstadt.
Nicht zuletzt möchte ich mich bei Familie Joël und Martine Divier bedanken. Sie haben mir nicht nur eine Unterkunft zur Verfügung gestellt, sondern mich darüber hinaus freundschaftlich aufgenommen und mir zusammen mit meinem Kollegen Vincent Garcia den Start in Paris sehr erleichtert.

Remerciements

Depuis l'organisation préable en passant par le tutorat pendant mon séjour à Paris jusqu'à la fin de ma thèse, j'ai reçu des encouragements de toutes parts. Je remercie sincérement tous ceux qui m'ont soutenu!
Notamment je remercie Monsieur Listmann pour cette offre formidable, la prise de contact avec le Professeur Fliess et pour le tutorat du TU Darmstadt – Sans son effort mon séjour n'aurait pas été possible. Avec la même intensité, je remercie le Professeur Fliess, qui a accepté mon tutorat et l'organisation à l'École Polytechnique à Paris. Avec beaucoup de patience il m'a permis de me familiariser avec la théorie des modules et il était constamment disponible pour répondre à mes questions.
Pour l'assistance morale et financière aussi je remercie ma mère Gerda Hentschke et ma copine Tanja Mayer, qui m'ont aidé à traverser les hauts et les bas durant ma thèse, ainsi que mes amis de Darmstadt.
Le dernier remerciement mais pas le moindre, va à Joël et Martine Divier qui ont fait plus que mettre un logement à ma disposition en m'hébergeant très aimablement. Avec mon collègue Vincent Garcia ils m'ont aidé à m'acclimater à Paris.

Christoph Hentschke

Inhaltsverzeichnis

1 Einführung **1**

2 Selbstsynchronisierende Stromverschlüsselung **4**
2.1 Verschlüsselung 5
2.2 Übertragung und Entschlüsselung 8
2.3 Auswirkungen von Übertragungsfehlern 10
2.3.1 Bitslip I 10
2.3.2 Bitslip II 12
2.3.3 Bitfehler 13
2.4 Beispiel 13

3 Grundbegriffe der Algebra und der Modultheorie **19**

4 Moduln über einem nichtkommutativen Hauptidealring **23**
4.1 Systemmodul 25
4.2 Eingangs-Ausgangssystem 27

5 Invertierbarkeit und Steuerbarkeit **29**
5.1 Invertierbarkeit 30
5.2 Steuerbarkeit 32
5.3 Beispiel 33
5.3.1 Zeitabhängige Umschaltfunktion 36
5.3.2 Geheimtextabhängige Umschaltfunktion 40

6 Kryptoanalyse **43**
6.1 Brute-Force-Angriff 43
6.2 Chosen-Plaintext-Angriff 44

7 Zusammenfassung und Ausblick **46**

A Quellcode zu den Beispielen **48**

Kapitel 1

Einführung

Für eine sichere Übertragung von Informationen über eine unsichere Übertragungsstrecke ist eine Verschlüsselung der Informationen notwendig. Nach Beschaffenheit der Übertragungsstrecke und der Information sowie den verfügbaren Ressourcen bieten sich unterschiedliche Methoden zur Verschlüsselung an.
Im Mittelpunkt der Betrachtungen dieser Arbeit steht die selbstsynchronisierende Stromverschlüsselung – im weiteren kurz SSSC [1]: Der Geheimtext ist die symbolweise Überlagerung von Klar- und Schlüsseltext. Dieser Vorgang ist die eigentliche Verschlüsselung. Die Funktion, die aus Klar- und Schlüsseltext den Geheimtext generiert, wird als Verschlüsselungsfunktion bezeichnet. Die Länge von Klar-, Schlüssel- und Geheimtext ist identisch. Zu einem festen Zeitpunkt ergibt sich ein Symbol des Schlüsseltextes aus einer parametrierten Funktion einer endlichen Anzahl zurückliegender Symbole des Geheimtextes. Die Parameter der Funktion werden als Schlüssel bezeichnet, die Funktion als Schlüsselgenerator. Diese Struktur ermöglicht es auf der Seite des Empfängers, aus einer korrekten endlichen Folge von Symbolen des Geheimtextes und der Kenntnis des Schlüssels ein Symbol aus dem Schlüsseltext des Senders zu rekonstruieren. Wird der Geheimtext vollständig und korrekt übertragen, ist es dem Empfänger auf diese Weise möglich den vollständige Schlüsseltext des Senders zu erhalten. Aus der Kenntnis von Schlüssel- und Geheimtext kann der Empfänger den Klartext zurückgewinnen, indem die inverse Verschlüsselungsfunktion auf Schlüssel- und Geheimtext angewendet wird.
Die zur Synchronisation zwischen Sender und Empfänger nötigen Informationen sind bereits im Geheimtext enthalten, d.h. es müssen keine redundanten Symbole eingefügt werden, die allein der Synchronisation dienen. Dadurch bleibt die Übertragungsrate unbeeinflusst und der Aufwand für die Erweiterung einer bestehenden Übertragungsstrecke durch eine SSSC ist gering. Die SSSC ist robust gegenüber Synchronisationsverlust bedingt durch Fehler während der Übertragung und ermöglicht das Zuschalten des Empfängers zu einem beliebigen Zeitpunkt ohne Kenntnis der aktuellen Position im Geheimtext.
Die Widerstandskraft der Verschlüsselung gegen Angriffe nimmt mit der Vorhersagbarkeit des Schlüsseltextes ab. Der nicht zu verwirklichende Idealfall ist eine zufällige Folge von Symbolen als Schlüsseltext. Eine mögliche Annäherung stellt die Klasse der stückweise linearen diskreten Systeme dar. Diese verbinden Realisierbarkeit mit einer

[1] **S**elf-**S**ynchronizing **S**tream **C**ipher – Selbstynchronisierende Stromverschlüsselung

anspruchsvollen Dynamik. Wird ein solches System zur Verschlüsselung eingesetzt, bezeichnet man es als Message-embedded System. Auf Senderseite wird der Eingang des Systems mit der unverschlüsselten Nachricht gespeist. Am Ausgang liegt die verschlüsselte Nachricht an. Der Verlauf der inneren Zustände des Systems kann als Schlüsseltext interpretiert werden. Der Empfänger entschlüsselt die Nachricht mit Hilfe des inversen Systems. Die prinzipielle Darstellung der Verwendung eines Message-embedded Systems als Verschlüsselungssystem zeigt Abbildung 1.1.

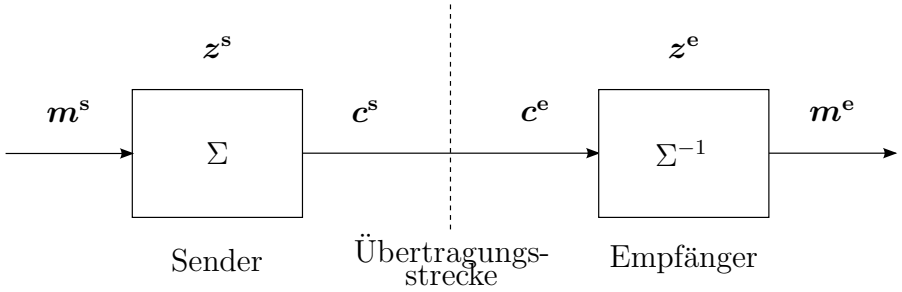


Abbildung 1.1: Message-embedded System ($\boldsymbol{m}$ – unverschlüsselte Nachricht bzw. Klartext, $\boldsymbol{z}$ – Schlüsseltext, $\boldsymbol{c}$ – verschlüsselte Nachricht bzw. Geheimtext

Allerdings erfüllt nicht jedes System dieser Klasse die strukturellen Voraussetzungen.
In [1] werden Methoden bereitgestellt, mit Hilfe derer geprüft werden kann, ob sich ein stückweise lineares diskretes SISO-System mit zeitabhängiger Umschaltfunktion zur selbstsynchronisierenden Stromverschlüsselung eignet. Dabei wird durch mühsame Matrizenrechnungen der relative Grad ermittelt und damit das System auf Invertierbarkeit geprüft – welche offensichtlich für die untersuchte Klasse von Systemen immer vorliegt. Der relative Grad wird bestimmt, indem die Bestimmungsgleichung für die Ausgangsgröße iteriert wird, bis eine direkte Abhängigkeit von der Ausgangsgröße vorliegt. Der relative Grad steht für die Anzahl benötigter Iterationen. Die Gültigkeit dieser Bedingung muss für jede mögliche Folge von Werten der Umschaltfunktion überprüft werden, was mit hohem Aufwand verbunden sein kann. Mit einer weiteren Kennzahl wird die Flachheit des Systemausgangs nachgewiesen. Diese Kennzahl ergibt sich ähnlich dem relativen Grad durch Herantasten an eine Bedingung für eine zuvor definierte inverse Transitionsmatrix. Die inverse Transitionsmatrix wiederum ergibt sich aus einer Matrizengleichung in Abhängigkeit von Eingangs-, System-, Ausgangs- und Durchgangsmatrix. Ist der relative Grad und die Flachheit des Systemausgangs gegeben, lässt sich das System in einer geeigneten Form darstellen und erfüllt die strukturellen Voraussetzungen für die Verwendung zur SSSC.
In der vorliegenden Arbeit wird diese Fragestellung nochmals aufgegriffen, aber durch einen anderen Zugang behandelt: Das stückweise lineare diskrete SISO-System wird zunächst als Speziallfall eines diskreten zeitvarianten Systems betrachtet, und dieses wiederum als Modul über einem nichtkommutativen Ring aufgefasst. Die algebraischen

Gleichungen, die die Beziehungen der Modulelemente beschreiben, spiegeln das Systemverhalten wieder. Sie werden durch einfache Umformungen in eine geeignete Darstellung überführt, so dass sich erkennen lässt, ob eine Basis für das Modul existiert. Dies ist gleichbedeutend mit der Steuerbarkeit des Systems. Ist der Systemausgang eine Basis, handelt es sich um einen flachen Ausgang. Die Frage der Invertierbarkeit lässt sich auf einen Vergleich des Ranges von den Modulelementen, die den Eingang repräsentieren und den Modulelementen, die den Ausgang repräsentieren, zurückführen. Das System ist demnach Invertierbar, wenn Ein- und Ausgang die selbe Anzahl linear unabhängiger Elemente besitzt. In diesem Fall stellt das System eine bijektive Abbildung zwischen Ein- und Ausgang dar. In beiden Fällen, Flachheit des Ausgangs und Invertierbarkeit, kann auf die Bestimmung künstlicher Kenngrößen verzichtet werden.
In Kapitel 2 werden die Grundlagen der Verschlüsselung, Übertragung und Entschlüsselung sowie die Einordnung der SSSC innerhalb der verschiedenen Verschlüsselungsverfahren beschrieben. Ebenso die Ursachen und Auswirkungen von Fehlern auf der Übertragungsstrecke. An einem primitiven Verschlüsselungssystem wird ausführlich auf die Robustheit gegenüber solchen Fehlern eingegangen und an Simulationsergebnissen veranschaulicht
Mit den nötigen Grundlagen der Algebra und Modultheorie beschäftigen sich Kapitel 3 und Kapitel 4. Anschließend wird die Anwendung der Modultheorie auf die Problemstellung in Kapitel 5 ausgeführt. D.h. anhand des Systemmoduls werden Rückschlüsse auf die systemtheoretischen Eigenschaften des Systems gezogen. An einem ausgewählten Beispiel wird das Vorgehen demonstriert und eine verschlüsselte, fehlerbehaftete Übertragung simuliert. Es folgt ein Vergleich des Verhaltens bei Synchronisationsverlust zwischen einem System mit einer zeitabhängig gewählten Umschaltfunktion und dem Fall, dass die Umschaltfunktion vom Geheimtext abhängt. Es zeigt sich, dass eine zeitabhängige Umschaltfunktion zu einem Verschlüsselungssystem führt, welches nicht die erwähnten Vorteile einer SSSC besitzt.
Ein entscheidendes Merkmal eines Verschlüsselungssystems ist die Frage der Sicherheit. Eine kurze Einführung mit Blick auf die vorgestellten Systeme bringt Kapitel 6. In Kapitel 7 werden die Ergebnisse der Arbeit zusammengefasst und noch offene Fragen diskutiert.

Kapitel 2

Selbstsynchronisierende Stromverschlüsselung

Die lange Geschichte der Kryptographie hat eine Vielzahl unterschiedlicher Methoden zur Verschlüsselung von Nachrichten hervorgebracht. Den Anfang machten die symmetrischen (oder auch klassischen) Verfahren: Sender und Empfänger verwenden den selben Schlüssel.
Erst in viel jüngerer Zeit fand dagegen die Entwicklung und Verwendung asynchroner (oder auch Public-Key) Verfahren statt. Sie gaben erstmals eine Antwort auf die bis dahin ungelöste Frage des sicheren Schlüsselaustausches bei synchronen Verfahren. Asynchrone Verfahren basieren auf einer Einwegfunktion. Mit Hilfe des sogenannten öffentlichen Schlüssels (Public-Key) des Empfängers wird eine Einwegfunktion parametriert und zur Verschlüsselung einer Nachricht auf Seite des Senders verwendet. Der öffentliche Schlüssel ist unbeschränkt zugänglich. Der umgekehrte Weg, also die Entschlüsselung der Nachricht, ist nur mit Kenntnis des geheimen Schlüssels möglich, der allein im Besitz des Empfängers der Nachricht ist. Eine sichere Übertragung von Information, die eingesetzt werden kann solange die Datenmenge begrenzt ist oder Geschwindigkeit keine Rolle spielt.
Die Kombination aus asynchronem Verfahren zum sicheren Austausch des Schlüssels und synchronem zum sicheren Austausch der Nachricht wird als hybrides Verfahren bezeichnet.
Die Selbstsynchronisierende Stromverschlüsselung gehört neben der synchronen Stromverschlüsselung und der Blockverschlüsselung zur Klasse der symmetrischen Verfahren (Abbildung 2.1). Die weitere Unterscheidung in Block- und Stromverschlüsselung richtet sich nach dem Umfang des Klartextes der in einem Durchgang verschlüsselt wird. Bei der Blockverschlüsselung sind dies (in der Regel) mehrere Symbole, ein Block, bei der Stromverschlüsselung ein einzelnes Symbol. Beide Verfahren lassen sich weiter aufteilen. Da sich die Betrachtungen dieser Arbeit nur auf die SSSC beziehen, wird hier aber nicht darauf eingegangen. Eine detailierte Einführung in die Kryptographie lässt sich in [2] (unter der angegebenen URL kann in sämtliche Kapitel des Buches Einsicht genommen werden) oder [3] finden.
In den folgenden Abschnitten wird der prinzipielle Ablauf des Ver- und Entschlüsse-

lungsvorgangs der SSSC beschrieben. Die Betrachtungen münden in einer allgemeinen Beschreibung der Struktur eines Verschlüsselungssystems mit der Eigenschaft der Selbstsynchronisation. In aktuellen Anwendungen wird diese Struktur durch eine Blockverschlüsselung erreicht, die in einem bestimmten selbstsynchronisierenden Modus, dem CFB (cipher feedback), mit einer (ineffizienten) Blockgröße von einem Symbol betrieben wird. Hinsichtlich einer effizienten Hardware-Implementierung einer SSSC wurde MOSQUITO und kurze Zeit später eine verbesserte Version, MOUSTIQUE, vorgeschlagen. Details können [4] und [5] entnommen werden. Eine Einführung speziell in die SSSC und einen Vorschlag zu einem automatenbasierten Entwurf gibt [6].

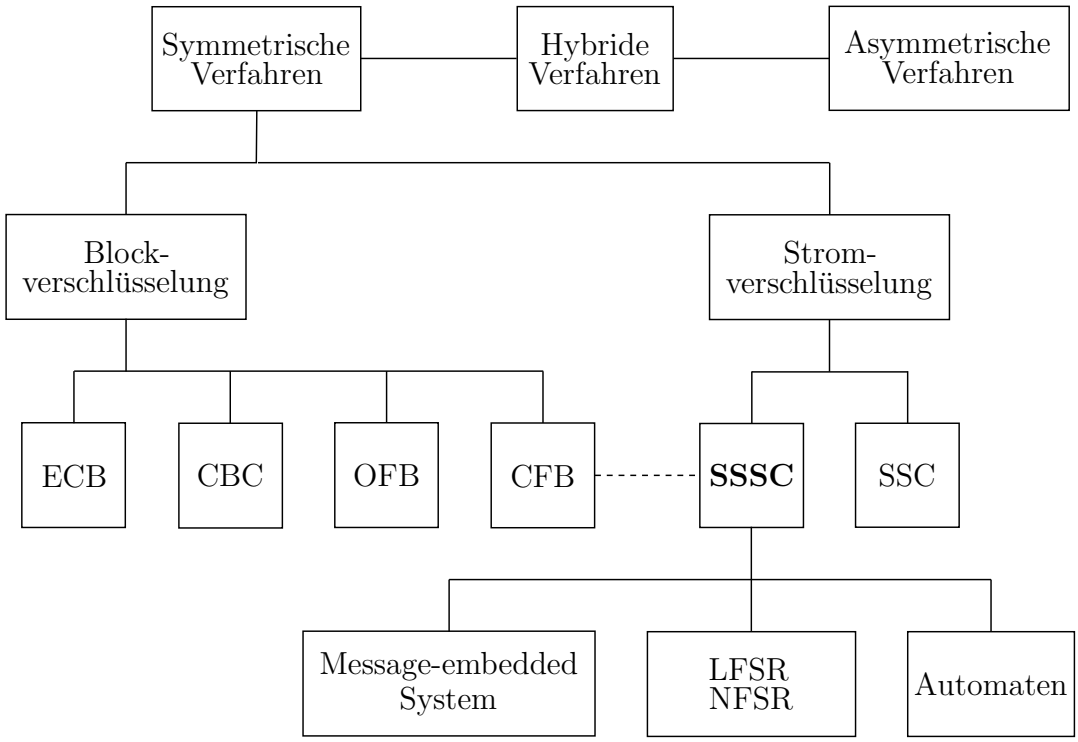


Abbildung 2.1: Klassifizierung der Verschlüsselungsverfahren (Die Abkürzungen: ECB – Electronic Codebook, CBC – Cipher-block Chaining, OFB – Output Feedback, CFB – Cipher feedback, SSC – Synchronous Stream Cipher, LFSR – Linear Feedback Shift Register, NFSR – Nonlinear Feedback Shift Register).

2.1 Verschlüsselung

Der Klartext. Die unverschlüsselte Nachricht, der Klartext $\boldsymbol{m}$, besteht aus einer beliebigen Folge von N Symbolen, die Elemente einer endlichen Menge $\mathcal{A}$, dem Alphabet

des Eingangs, sind. Die Gesamtheit aller möglichen Nachrichten ist der Nachrichtenraum $\mathcal{M}$. Das Symbol zum diskreten Zeitpunkt k wird mit m_k bezeichnet.

$$\boldsymbol{m} = (m_1, m_2, \ldots, m_N), \qquad \text{mit} \quad \boldsymbol{m} \in \mathcal{M}, \quad m_1, m_2, \ldots \in \mathcal{A}.$$

Eine geordnete Menge von Elementen oder eine Folge von Symbolen wird in Vektorschreibweise fett gedruckt dargestellt.

Der Schlüsseltext. Die Symbole z_n des Schlüsseltextes $\boldsymbol{z}$, die auch als zeitvarianter Schlüssel bezeichnet werden, werden während des Verschlüsselungsvorgangs aus einer Funktion $f_{\mathrm{z}}^{\mathrm{K}}$ in Abhängigkeit einer endlichen, festen Anzahl P zurückliegender Symbole des Geheimtextes $\boldsymbol{c}$ berechnet. Der Schlüsseltext besteht ebenfalls aus N Symbolen.
Die Funktion $f_{\mathrm{z}}^{\mathrm{K}}$ wird durch den (geheimen) Schlüssel $\boldsymbol{K}$,

$$\boldsymbol{K} = (K_1, \ldots, K_{N_{\mathrm{K}}}), \qquad \text{mit} \quad \boldsymbol{K} \in \mathcal{K},$$

parametriert und als Schlüsselgenerator bezeichnet. Die Menge aller möglichen Schlüssel ist der Schlüsselraum $\mathcal{K}$. Der Schlüssel besteht aus N_{K} Symbolen.
Im klassischen Entwurf von SSSC wird die notwendige Komplexität des Schlüsseltextes dadurch erreicht, dass der Schlüsselgenerator aus mehreren hintereinander geschalteten Schieberegistern aufgebaut wird. Durch diese mehrstufige Architektur des Schlüsselgenerators tritt eine Verzögerung auf. D.h. obwohl die benötigten vergangenen Werte des Geheimtextes für die Berechnung eines Symbols des Schlüsseltextes dem Schlüsselgenerator zur Verfügung stehen, wird das Symbol des Schlüsseltextes erst nach einer Verzögerung ausgegeben. Diese Verzögerung wird mit b_{s} abgekürzt und im englischen als „cipher function delay“ bezeichnet. Sie kann auch als eine Art Totzeit des Schlüsselgenerators interpretiert werden. Der Schlüsseltext wird über die Beziehungen

$$\begin{aligned} z_k &= f_{\mathrm{z}}^{\mathrm{K}}\left(c_{k-b_{\mathrm{s}}-P}, \ldots, c_{k-b_{\mathrm{s}}-1}\right), \qquad \text{mit} \quad K \in \mathcal{K} \\ \boldsymbol{z} &= (z_1, z_2, \ldots, z_N), \qquad \text{mit} \quad z_1, z_2, \ldots, z_N \in \mathcal{Z} \end{aligned} \tag{2.1}$$

bestimmt.

Der Geheimtext. Zum Zeitpunkt k ergibt sich das Symbol c_k des Geheimtextes als eine Funktion f_{c} in Abhängigkeit von m_k und z_k. Die Funktion f_{c} wird als Verschlüsselungsfunktion bezeichnet. Da es sich bei der Stromverschlüsselung um eine symbolweise Verschlüsselung handelt, muss auch der Geheimtext aus N Symbolen bestehen.

$$\begin{aligned} c_k &= f_{\mathrm{c}}\left(z_k, m_k\right) \\ \boldsymbol{c} &= (c_1, c_2, \ldots, c_N), \qquad \text{mit} \quad c_1, c_2, \ldots, c_N \in \mathcal{C}. \end{aligned} \tag{2.2}$$

Im Falle einer bitweisen Verschlüsselung, d.h. ein Symbol ist ein Bit und für $\mathcal{A}$, $\mathcal{Z}$ und $\mathcal{C}$ gilt

$$\mathcal{A} = \mathcal{Z} = \mathcal{C} = \{0, 1\},$$

handelt es sich bei der Verschlüsselungsfunktion f_c um eine einfache Addition modulo 2 oder auch XOR-Verknüpfung $\oplus$,

$$z_k = f_\mathrm{z}^\mathrm{K}\left(c_{k-b_\mathrm{s}-P}, \ldots, c_{k-b_\mathrm{s}-1}\right), \qquad k \in \mathcal{K}$$
$$c_k = z_k \oplus m_k.$$

Die Beziehungen (2.1) und (2.2) beschreiben allgemein die Struktur einer SSSC. Sie werden als kanonische Darstellung bezeichnet.
Bei der Initialisierung muss beachtet werden, dass für die Verschlüsselung der ersten $P+b_\mathrm{s}$ Symbole des Geheimtextes, $c_1 \ldots c_{P+b_\mathrm{s}}$ die zurückliegenden Symbole $c_{-b_\mathrm{s}-P}, \ldots, c_0$ des Geheimtextes noch nicht bekannt sind. Sie müssen durch einen sogenannten Initialisierungsvektor ($\boldsymbol{IV}$),

$$\boldsymbol{IV} = \left(c_{-b_\mathrm{s}-P}, \ldots, c_0\right),$$

aufgefüllt werden.
Eine anschauliche Darstellung des beschriebenen Verschlüsselungsvorgangs für ein Symbol m_i^s des Klartextes zeigt Abbildung 2.2.

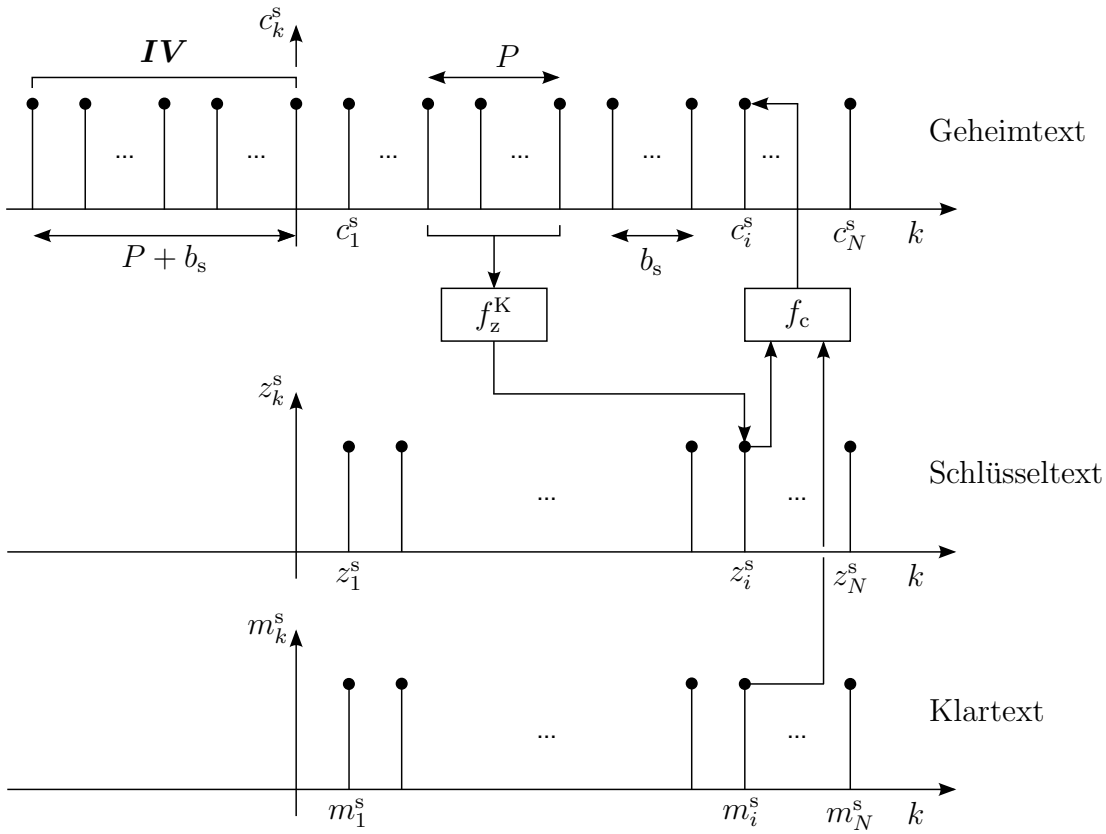


Abbildung 2.2: Verschlüsselung

2.2 Übertragung und Entschlüsselung

Auf Seite des Senders besteht die Übertragungsstrecke aus der im vorhergehenden Abschnitt beschriebenen Verschlüsselung. Der Aufbau des Empfängers besteht aus dem gleichen Schlüsselgenerator wie auf Senderseite aber der Umkehrfunktion der Verschlüsselungsfunktion als Entschlüsselungsfunktion.
Die besondere Struktur der SSSC ermöglicht es dem Empfänger aus einer Folge von P korrekt empfangenen zurückliegenden Symbolen des Geheimtextes das korrekte Schlüsseltextsymbol zum aktuellen Geheimtextsymbol zu bestimmen und es damit korrekt zu entschlüsseln. Daraus folgt, dass bei einem einmaligen Verlust der Synchronisation im ungünstigsten Fall $P + 1$ Symbole des Klartextes falsch entschlüsselt werden.
Ursachen für einen solchen Fehler können unterschiedliche Taktraten von Sender und Empfänger sein, die auf Seite des Empfängers periodisch zur wiederholten Wertung (Bitslip I) oder zum Übergehen (Bitslip II) von Symbolen führen. Wird ein einzelnes Symbol falsch übertragen (Bitfehler), d.h. der Wert des Symbols wird verfälscht, werden die nachfolgenden $P+1$ Symbole des Klartextes falsch entschlüsselt. Die Bezeichnungen Bitslip und Bitfehler werden hier unabhängig von der tatsächlichen Wertigkeit des Symbols verwendet. In den Abschnitten 2.3.1, 2.3.2 und 2.3.3 werden die Auswirkungen der unterschiedlichen Fehler während der Übertragung für eine SSSC, die sich durch eine allgemeine kanonische Darstellung der Form 2.1 und 2.2 beschreiben lässt, ausführlich erläutert. In Abschnitt 2.4 werden die Erkenntnisse an einem Beispiel überprüft und veranschaulicht.
Um Sender und Empfänger vor der Übertragung der eigentlichen Nachricht zu synchronisieren, muss lediglich eine beliebige Folge von P Symbolen gesendet werden. Dadurch ergibt sich für den Empfänger ebenfalls die Möglichkeit sich zu einem beliebigen Zeitpunkt auch ohne Kenntnis der exakten Position zuzuschalten und nach P Symbolen den Geheimtext korrekt zu entschlüsseln. Zusammengefasst sind die Vorteile der SSSC

+ die Möglichkeit einer einfachen Erweiterung einer bestehenden Übertragungsstrecke,
+ keine Reduktion der Übertragungsrate,
+ Robustheit gegenüber Synchronisationsverlust durch Fehler auf der Übertragungsstrecke und
+ die Möglichkeit des Empfängers, sich zu einem beliebigen Zeitpunkt der Übertragung zu zuschalten.

Der Nachteil der SSSC ergibt sich aus der Struktur des Schlüsselgenerators. Jedes Schlüsseltextsymbol wird nur dann korrekt erzeugt, wenn die benötigten P Geheimtextsymbole korrekt sind. Das bedeutet,

− jedes falsch übertragene Geheimtextsymbol resultiert in P falsch erzeugten Schlüsseltextsymbolen und damit auch falsch rekonstruierten Klartextsymbolen.

Die Anwendung einer SSSC setzt voraus, dass Sender und Empfänger im Besitz des selben Schlüssels und Initialisierungsvektors sind, wobei der Initialisierungsvektor nicht geheim gehalten werden muss. Offensichtlich kann durch Wahl des Initialisierungsvektors kein Einfluss auf die Sicherheit der Verschlüsselung genommen werden, da er lediglich eine Variation des Verlaufs der internen Zustände des Verschlüsselungssystems, aber keine Erweiterung des Schlüsselraumes bewirkt.
Das Problem des Schlüsselaustausches kann, wie in der Einführung zu diesem Kapitel erwähnt, durch ein asynchrones Verschlüsselungsverfahren gelöst werden.
Den prinzipiellen Aufbau einer Übertragungsstrecke für die SSSC zeigt Abbildung 2.3. Die Größen auf Seite des Senders sind mit „s“ gekennzeichnet, auf Seite des Empfängers mit „e“. Die Erkenntnis für die folgenden Betrachtungen: Ein Algorithmus, für den auf

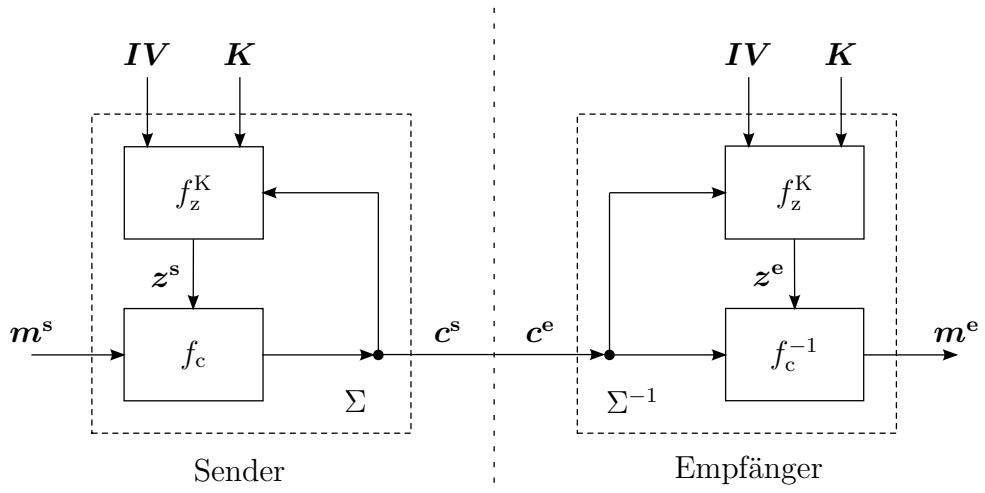


Abbildung 2.3: Übertragungsstrecke

Seite des Senders eine kanonische Darstellung entsprechend den Beziehungen (2.1) und (2.2),

$$z_k = f_z^K\left(c_{k-b_s-P}, \ldots, c_{k-b_s-1}\right)$$
$$c_k = f_c\left(z_k, m_k\right),$$

existiert, besitzt die Eigenschaften, die für eine selbständige Synchronisation nötig sind. Er erfüllt die strukturellen Voraussetzungen für die Verwendung zur SSSC. Die entsprechende Darstellung auf der Empfängerseite ist

$$z_k = f_z^K\left(c_{k-b_s-P}, \ldots, c_{k-b_s-1}\right)$$
$$m_k = f_c^{-1}\left(z_k, c_k\right).$$

Der Schlüsselgenerator auf Sender- und Empfängerseite sind identisch, die Entschlüsselungsfunktion ergibt sich als Umkehrfunktion der Verschlüsselungsfunktion. Die ge-

naue Realisierung als Zusammenschluss von Schieberegistern, als Automat oder als ein Message-embedded System ist dabei nicht von Bedeutung.

2.3 Auswirkungen von Übertragungsfehlern

In den folgenden Abschnitten werden die Auswirkungen der in 2.2 aufgeführten Übertragungsfehlern auf den Entschlüsselungsvorgang untersucht. Die Ergebnisse finden in Abschnitt 5.3 Verwendung, wenn die Anwendbarkeit von stückweise linearen diskreten SISO-Systemen geprüft wird.
Für die Notation gilt: Ein Symbol zum Zeitpunkt k in Klar-, Schlüssel- und Geheimtext trägt die Bezeichnung m_k^{s}, z_k^{s} und c_k^{s} auf Seite des Senders und entsprechend m_k^{e}, z_k^{e} und c_k^{e} auf Seite des Empfängers. Desweiteren wird in allen drei folgenden Betrachtungen angenommen, dass der Fehler im Geheimtext zum Zeitpunkt $k = i$ auftritt, das Symbol c_i^{e} also falsch ist. Die unverschlüsselte Nachricht hat eine Länge von N Symbolen.

2.3.1 Bitslip I

Es wird angenommen, dass der Empfänger mit einer höheren Taktrate als der Sender arbeitet, was zu einer sich periodisch wiederholenden doppelten Wertung von Symbolen auf Empfängerseite führt. Der Geheimtext $\boldsymbol{c}^{\mathbf{e}}$ ist länger als $\boldsymbol{c}^{\mathbf{s}}$.

Der Geheimtext. Der Fehler kann durch Auftrennen des Geheimtextes nach dem Zeitpunkt $k = i - 1$ und Einfügen des Symbols c_{i-1}^{s} an der Stelle $k = i$ modelliert werden. Es ergibt sich die Folge

$$\boldsymbol{c}^{\mathbf{e}} = \left(c_{-P+1}^{\mathrm{s}}, \ldots, c_0^{\mathrm{s}}, c_1^{\mathrm{s}}, \ldots, c_{i-1}^{\mathrm{s}}, c_{i-1}^{\mathrm{s}}, \ldots, c_{\mathrm{N}}^{\mathrm{s}}\right),$$

d.h. der Geheimtext auf Empfängerseite entspricht ab dem Zeitpunkt $k = i$ dem um einen Zeitschritt $\Delta = 1$ nach rechts verschobenen Geheimtext auf Senderseite $\boldsymbol{c}^{\mathbf{s}}$,

$$c_k^{\mathrm{e}} = c_{k-\Delta}^{\mathrm{s}}, \qquad \text{mit} \quad k \geq i. \tag{2.3}$$

Der Schlüsseltext. Als nächstes wird aus dem empfangenen Geheimtext der Schlüsseltext rekonstruiert. Zu einem Zeitpunkt k ergibt sich das Symbol z_k^{e} aus der Verschlüsselungsfunktion gemäß Beziehung (2.1) zu

$$z_k^{\mathrm{e}} = f_{\mathrm{z}}^{\mathrm{K}}\left(c_{k-\mathrm{b_s}-P}^{\mathrm{e}}, \ldots, c_{k-\mathrm{b_s}-1}^{\mathrm{e}}\right).$$

Der Schlüsseltext auf Empfängerseite hat einen Bezug zu dem auf Senderseite, wenn eine Darstellung der Form

$$z_k^{\mathrm{e}} = z_{k-\Delta}^{\mathrm{s}}, \qquad \text{mit} \quad \Delta = 1$$

existiert.
Wird die Verzögerung b_s berücksichtigt, beeinflusst der Fehler an der Stelle $k = i$ den Schlüsseltext zum ersten Mal an der Stelle $k = i + 1 + b_s$, denn mit (2.1) und (2.3) gilt

$$\begin{aligned} z^e_{i+1+b_s} &= f^K_z\left(c^e_{i+1-P}, \ldots, c^e_i\right) \\ &= f^K_z\left(c^s_{i+1-P}, \ldots, c^s_{i-\Delta}\right) \neq z^s_{i+1+b_s-\Delta} = f^K_z\left(c^s_{i+1-P-\Delta}, \ldots, c^s_{i-\Delta}\right). \end{aligned}$$

Die folgenden $P-2$ Symbole werden ebenfalls falsch rekonstruiert. Danach ist die Synchronisation abgeschlossen. Zum Zeitpunkt $k = i + 1 + b_s + P - 1 = i + b_s + P$ gilt für die Bestimmungsgleichung des entsprechenden Schlüsseltextsymbols

$$\begin{aligned} z^e_{i+b_s+P} &= f^K_z\left(c^e_i, \ldots, c^e_{i+P-1}\right) \\ &= f^K_z\left(c^s_{i-\Delta}, \ldots, c^s_{i+P-1-\Delta}\right) = z^s_{i+b_s+P-\Delta}, \end{aligned}$$

d.h. der Schlüsseltext auf Empfängerseite entspricht ab dem Zeitpunkt $k = i + b_s + P$ dem um einen Zeitschritt $\Delta = 1$ nach rechts verschobenen Schlüsseltext auf Senderseite,

$$z^e_k = z^s_{k-\Delta}, \qquad \text{mit} \quad k \geq i + b_s + P. \tag{2.4}$$

Die Schlüsseltextsymbole auf Empfängerseite, die keinen Bezug zum Schlüsseltext auf Seite des Senders besitzen, sind

$$\left\{z^e_{i+1+b_s}, \ldots, z^e_{i+1+b_s+P-2}\right\}, \qquad \text{mit} \quad P > 1. \tag{2.5}$$

Da $\boldsymbol{c}^{\mathbf{e}}$ nur Symbole enthält, die auch in $\boldsymbol{c}^{\mathbf{s}}$ enthalten sind, erzeugt der Schlüsselgenerator für $P = 1$ keine Schlüsseltextsymbole ohne Bezug zu $\boldsymbol{c}^{\mathbf{s}}$, da nur ein korrektes Symbol und nicht eine Folge von korrekten Symbolen benötigt wird. Allerdings ist das Schlüsseltextsymbol, welches sich aus dem doppelt gewerteten Geheimtextsymbol ergibt, redundant. Es hat den selben Wert wie das vorhergehende Symbol. Für eine praktische Anwendung als Verschlüsselungssystem dürfte der Fall $P = 1$ allerdings keine Rolle spielen.

Der Klartext. Ein Symbol des Klartextes kann aus dem entsprechenden Geheimtextsymbol korrekt entschlüsselt werden, wenn sowohl Geheimtextsymbol als auch das zugehörige Schlüsseltextsymbol korrekt sind. Aus (2.5) ergeben sich die falschen Schlüsselsymbole. Das falsche Geheimtextsymbol befindet sich an der Stelle $k = i$. Die Klartextsymbole auf Empfängerseite, die keinen Bezug zu Symbolen im ursprünglichen Klartext auf Senderseite besitzen, sind

$$\left\{m^e_i, m^e_{i+1+b_s}, \ldots, m^e_{i+1+b_s+P-2}\right\}. \tag{2.6}$$

Das Klartextsymbol m^e_i auf Empfängerseite hat keinen Bezug zum Klartext auf Senderseite, da das Geheimtextsymbol c^e_i falsch entschlüsselt wird. Allerdings ist die Information in c^e_i redundant, c^e_i und c^e_{i-1} sind identisch. Daraus folgt aus dem falschen Geheimtextsymbol c^e_i bzw. aus dem falsch rekonstruierte Klartextsymbol m^e_i kein Informationsverlust. Zusammengefasst ergibt sich, dass die Information von $P-1$ Klartextsymbolen nicht korrekt rekonstruiert werden kann.

2.3.2 Bitslip II

Der Empfänger arbeitet mit einer geringeren Taktrate als der Sender. Dies führt zu einem Übergehen von Symbolen. Der Geheimtext $\boldsymbol{c}^{\mathrm{e}}$ ist damit kürzer als $\boldsymbol{c}^{\mathrm{s}}$.

Der Geheimtext. Im Geheimtext wird das Symbol an der Stelle $k = i$ entfernt,

$$\boldsymbol{c}^{\mathrm{e}} = \left(c^{\mathrm{s}}_{-P+1}, \ldots, c^{\mathrm{s}}_{0}, c^{\mathrm{s}}_{1}, \ldots, c^{\mathrm{s}}_{i-1}, c^{\mathrm{s}}_{i+1}, \ldots, c^{\mathrm{s}}_{\mathrm{N}}\right),$$

d.h. der Geheimtext auf Empfängerseite entspricht ab dem Zeitpunkt $k = i$ dem um einen Zeitschritt $\Delta = 1$ nach links verschobenen Geheimtext auf Senderseite $\boldsymbol{c}^{\mathrm{s}}$,

$$c^{\mathrm{e}}_{k} = c^{\mathrm{s}}_{k+\Delta}, \qquad \text{mit} \quad k \geq i. \tag{2.7}$$

Der Schlüsseltext. Der Schlüsseltext $\boldsymbol{c}^{\mathrm{e}}$ entspricht $\boldsymbol{c}^{\mathrm{s}}$ bis einschließlich $k = i + \mathrm{b_s}$. Danach macht sich der Verlust des Geheimtextsymbols c^{s}_{i} bemerkbar, denn mit (2.1) und (2.7) gilt ab $k = i + \mathrm{b_s} + 1$

$$\begin{aligned} z^{\mathrm{e}}_{i+\mathrm{b_s}+1} &= f^{\mathrm{K}}_{\mathrm{z}}\left(c^{\mathrm{e}}_{i+1-P}, \ldots, c^{\mathrm{e}}_{i}\right) \\ &= f^{\mathrm{K}}_{\mathrm{z}}\left(c^{\mathrm{s}}_{i+1-P}, \ldots, c^{\mathrm{s}}_{i+\Delta}\right) \neq z^{\mathrm{s}}_{i+\mathrm{b_s}+1+\Delta} = f^{\mathrm{K}}_{\mathrm{z}}\left(c^{\mathrm{s}}_{i+1-P+\Delta}, \ldots, c^{\mathrm{s}}_{i+\Delta}\right). \end{aligned}$$

Der Ablauf der Synchronisation verhält sich ähnlich wie im vorhergehenden Fehlerfall. Es werden weitere $P - 2$ Symbole falsch rekonstruiert. Für $k = i + \mathrm{b_s} + P$ erhält man die Beziehung

$$\begin{aligned} z^{\mathrm{e}}_{i+\mathrm{b_s}+P} &= f^{\mathrm{K}}_{\mathrm{z}}\left(c^{\mathrm{e}}_{i}, \ldots, c^{\mathrm{e}}_{i+P-1}\right) \\ &= f^{\mathrm{K}}_{\mathrm{z}}\left(c^{\mathrm{s}}_{i-\Delta}, \ldots, c^{\mathrm{s}}_{i+P-1-\Delta}\right) = z^{\mathrm{s}}_{i+\mathrm{b_s}+P-\Delta}, \end{aligned}$$

d.h. der Schlüsseltext auf Empfängerseite entspricht ab dem Zeitpunkt $k = i + \mathrm{b_s} + P$ dem um einen Zeitschritt $\Delta = 1$ nach links verschobenen Schlüsseltext auf Senderseite,

$$z^{\mathrm{e}}_{k} = z^{\mathrm{s}}_{k+\Delta}, \qquad \text{mit} \quad k \geq i + \mathrm{b_s} + P.$$

Für die Schlüsseltextsymbole auf Empfängerseite, die keinen Bezug zum Schlüsseltext auf Seite des Senders besitzen, gilt ebenfalls die Erkenntnis aus (2.5).

Der Klartext. Die Klartextsymbole c^{e} ohne Bezug zu c^{s} sind

$$\left\{m^{\mathrm{e}}_{i}, m^{\mathrm{e}}_{i+1+\mathrm{b_s}}, \ldots, m^{\mathrm{e}}_{i+1+\mathrm{b_s}+P-2}\right\}. \tag{2.8}$$

Hinzu kommt der Verlust der Information über das Klartextsymbol c^{s}_{i} aufgrund des Übertragungsfehlers an sich. Insgesamt ergibt sich ein Verlust von $P + 1$ Symbolen pro Synchronisationsverlust durch einen Bitslip II.

2.3.3 Bitfehler

Der Geheimtext. Der Wert des Geheimtextsymbols zum Zeitpunkt $k = i$, c_i^{e}, wird um Δc_i verfälscht,

$$\boldsymbol{c}^{\mathbf{e}} = \left(c_{-P+1}^{\mathrm{s}}, \ldots, c_0^{\mathrm{s}}, c_1^{\mathrm{s}}, \ldots, c_i^{\mathrm{s}} + \Delta c_i, c_{i+1}^{\mathrm{s}}, \ldots, c_{\mathrm{N}}^{\mathrm{s}}\right).$$

Der Schlüsseltext. Die Symbole des Schlüsseltextes entsprechen ihrem Gegenüber auf Senderseite bis einschließlich $z_{i+\mathrm{b_s}}^{\mathrm{e}}$, denn für $k = i + \mathrm{b_s} + 1$ gilt

$$\begin{aligned} z_{i+\mathrm{b_s}+1}^{\mathrm{e}} &= f_{\mathrm{z}}^{\mathrm{K}}\left(c_{i+1-P}^{\mathrm{e}}, \ldots, c_i^{\mathrm{e}}\right) \\ &= f_{\mathrm{z}}^{\mathrm{K}}\left(c_{i+1-P}^{\mathrm{s}}, \ldots, c_i^{\mathrm{s}} + \Delta c_i\right) \neq z_{i+\mathrm{b_s}+1}^{\mathrm{s}} = f_{\mathrm{z}}^{\mathrm{K}}\left(c_{i+1-P}^{\mathrm{s}}, \ldots, c_i^{\mathrm{s}}\right). \end{aligned}$$

Die folgenden $P - 1$ Schlüsseltextsymbole sind offensichtlich ebenfalls falsch, da sie in Abhängigkeit des verfälschten Geheimtextsymbols bestimmt werden. Ab dem Zeitpunkt $k = i + \mathrm{b_s} + P$ stehen wieder korrekte Geheimtextsymbole zur Verfügung, so dass sich die falschen Schlüsseltextsymbole auf die Menge

$$\left\{z_{i+1+\mathrm{b_s}}^{\mathrm{e}}, \ldots, z_{i+\mathrm{b_s}+P}^{\mathrm{e}}\right\} \tag{2.9}$$

beschränken.

Der Klartext. Die Überlagerung von Geheimtext und Schlüsseltext führt in der Summe zu $P+1$ Klartextsymbolen, die nicht korrekt zurückgewonnen werden können. Dies sind die Symbole

$$\left\{\, m_i^{\mathrm{e}}, m_{i+1+\mathrm{b_s}}^{\mathrm{e}}, \ldots, m_{i+1+\mathrm{b_s}+P-1}^{\mathrm{e}} \,\right\}. \tag{2.10}$$

Dabei wird ein Fehler wird durch das falsche Geheimtextsymbol an sich, die restlichen P Symbole durch die Struktur des Verschlüsselungssystems verursacht.

2.4 Beispiel

Zur Verdeutlichung des Ablaufs von Ver- und Entschlüsselung und der selbständigen Synchronisation nach Auftreten der vorgestellten Übertragungsfehler wird ein einfaches Verschlüsselungssystem betrachtet. Dieses wird in Abschnitt 5.3 nochmals in abgewandelter Form, d.h. mit varianten Koeffizienten, aufgegriffen. Das System besitzt die kanonische Darstellung

$$\begin{aligned} z_k &= a_1\, c_{k-1} + a_0\, c_{k-2}, \\ c_k^{\mathrm{s}} &= z_k^{\mathrm{s}} + m_k^{\mathrm{s}}, \end{aligned} \qquad \text{mit} \quad a_0 = 1,\ a_1 = 170. \tag{2.11}$$

Wie aus (2.11) ersichtlich, ist der Schlüsselgenerator eine Funktion von zwei zum aktuellen Geheimtextsymbol unmittelbar zurückliegenden Geheimtextsymbolen. D.h. der Schlüsselgenerator arbeitet ohne Verzögerung und es gilt

$$P = 2 \quad \text{und} \quad b_{\mathrm{s}} = 0.$$

Aus der kanonischen Darstellung folgt für die Rekonstruktion des Klartextes auf Seite des Empfängers die Beziehung

$$m_k^{\mathrm{e}} = c_k^{\mathrm{e}} - z_k^{\mathrm{e}} \tag{2.12}$$

als Entschlüsselungsfunktion. Zur Rekonstruktion des Schlüsseltextes wird auf Empfängerseite der identische Algorithmus für den Schlüsselgenerator verwendet wie auf Senderseite.
Hinsichtlich anschaulicher Simulationswerte wird als Symbol kein Bit, sondern ein Byte definiert. Dies verringert die Wahrscheinlichkeit, dass sich in der Simulation durch eine ungünstige Wahl der Parameter Fehler überlagern und aufheben. Für das Alphabet des Eingangs ergibt sich

$$\mathcal{A} = \{0, 1, \ldots, 255\}.$$

Für den Wertebereich des Schlüsseltextes und des Geheimtextes wird dieselbe Annahme getroffen:

$$\mathcal{Z} = \mathcal{C} = \{0, 1, \ldots, 255\}.$$

Damit die Wertebereiche eingehalten werden, wird jede Rechenoperation Modulo 256 ausgeführt, d.h. die Ergebnisse der Bestimmungsgleichungen von Schlüssel- und Geheimtext (2.11) und Klartext (2.12) werden jeweils durch 256 dividiert und es wird mit dem Rest dieser Operation weitergerechnet. Die Modulo-Operation ist eine surjektive Abbildung auf den Ring der 256 Restklassen $0, \ldots, 255$. Unter der Annahme, dass der jeweilige Symbolwert den Wertebereich $\{0, \ldots, 255\}$ nicht überschreiten, ist die Modulo-Operation aber umkehrbar. Da dies hier der Fall ist, gilt:

$$c_k = (z_k + m_k) \mod 256 \quad \Longleftrightarrow \quad m_k = (c_k - z_k) \mod 256. \tag{2.13}$$

Der Initialisierungsvektor, der die beiden Geheimtextsymbole zur Bestimmung der ersten Schlüsseltextsymbole liefert, wird willkürlich zu

$$\boldsymbol{IV} = (50, 50)$$

gewählt.
Die Nachricht, der Klartext $\boldsymbol{m}^{\mathrm{s}}$, besteht aus einer Folge von $N = 10$ zufällig aus $\mathcal{A}$ generierten Symbolen. Für sämtliche betrachteten Übertragungsfehler wird jedoch eine identische Folge von Symbolen als Nachricht verwendet. Der Verlauf des Klartextes für den Fehlerfall Bitslip I kann Abbildung 2.4c entnommen werden. Abbildung 2.5c zeigt den Klartext bei einem Fehler vom Typ Bitslip II und Abbildung 2.6c im Fall eines Bitfehlers.
Der Schlüsseltext und Geheimtext auf Seite des Senders kann anhand der Beziehungen (2.11) berechnet werden. Der Verlauf der drei Schlüsseltexte ist in den Abbildungen 2.4b, 2.5b und 2.6b und der Verlauf der drei Geheimtexte in den Abbildungen 2.4a, 2.5a und 2.6a zu sehen. Die Zuordnung zwischen Abbildung und Fehler ist identisch mit den Abbildungen zum Verlauf der Klartexte: zuerst Bitslip I, dann Bitslip II und zuletzt

Bitfehler.
Die Generierung des fehlerbehafteten Geheimtextes auf Empfängerseite und daraus die Rekonstruktion von Schlüssel- und Klartext werden, getrennt nach den beiden Fehlerklassen, Bitslip und Bitfehler, in den folgenden Abschnitten erklärt. Der Fehler tritt an der Stelle $k = 5$ auf.

Bitslip I

Der Geheimtext. Das Geheimtextsymbol $k = 4$ wird doppelt gewertet. Die Folge der Geheimtextsymbole ist

$$\boldsymbol{c}^{\mathrm{e}} = \left(c^{\mathrm{s}}_{-1,\ldots,4}, c^{\mathrm{s}}_{4}, c^{\mathrm{s}}_{5,\ldots,10}\right).$$

Abbildung 2.4(a) zeigt die Überlagerung von $\boldsymbol{c}^{\mathrm{e}}$ mit $\boldsymbol{c}^{\mathrm{s}}$. D.h. es wird zunächst der Verlauf des Geheimtextes auf Empfängerseite abgebildet und anschließend der auf Senderseite. Sind beide Texte identisch, ist daher nur der letztere, der Geheimtext auf Senderseite, sichtbar. Die Zeitskala k richtet sich dagegen nach dem Geheimtext auf Empfängerseite $\boldsymbol{c}^{\mathrm{e}}$, da sich die weiteren Betrachtungen auch auf diesen beziehen. Der Geheimtext auf Senderseite wird daher an der Fehlerstelle aufgetrennt und gestreckt dargestellt. Für das Symbol an der Stelle $k = 5$ auf Empfängerseite gibt es kein entsprechendes Symbol auf Senderseite. An den Zeitpunkten $k = 0$ und $k = -1$ stehen die Symbole des Initialisierungsvektors. Bei den Verläufen von Schlüssel- und Klartext sind diese Stellen unbesetzt. Werden alle drei Verläufe untereinander angeordnet, lässt sich die Generierung des Schlüsseltextes aus dem Geheimtext und die Generierung des Geheimtextes aus Schlüssel und Klartext entsprechend Abbildung 2.2 nachvollziehen.

Der Schlüsseltext. Als nächstes wird der Schlüsseltext auf Senderseite gemäß Beziehung 2.11 bestimmt. Aus den Erkenntnissen von (2.5) erhält man die Menge der falschen Schlüsseltextsymbole zu

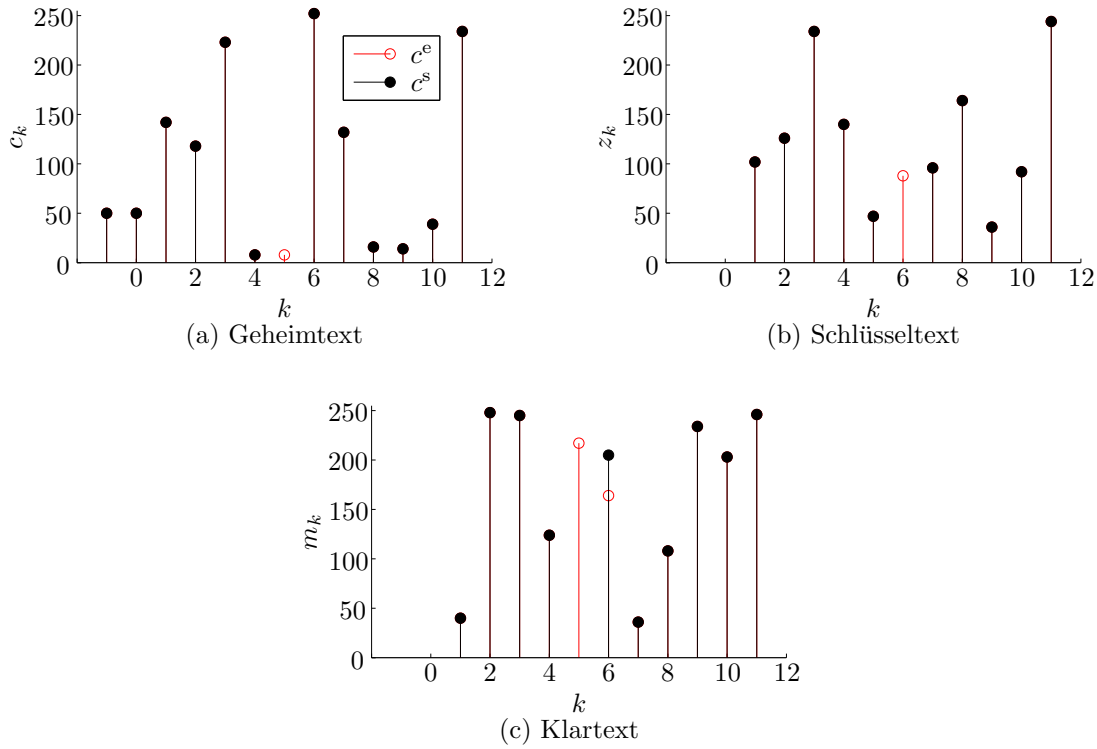
$$\{z^{\mathrm{e}}_{6}\}.$$

Die Überlagerung von $\boldsymbol{z}^{\mathrm{e}}$ mit $\boldsymbol{z}^{\mathrm{s}}$ zeigt Abbildung 2.4b.

Der Klartext. Der Klartext wird aus 2.12 berechnet. Mit (2.6) und den Zahlenwerten für dieses Beispiel ergeben sich die falschen Klartextsymbole zu

$$\{m^{\mathrm{e}}_{5}, m^{\mathrm{e}}_{6}\}.$$

Die entschlüsselte Folge von Klartextsymbolen am Ausgang des Empfängers überlagert mit der ursprünglichen Nachricht am Eingang des Senders zeigt Abbildung 2.4c.



(a) Geheimtext (b) Schlüsseltext

(c) Klartext

Abbildung 2.4: Bitslip I

Bitslip II

Der Geheimtext. Im Geheimtext $\boldsymbol{c}^{\mathrm{s}}$ wird das Symbol an der Stelle $k = 5$ entfernt. Der Empfänger erhält den Geheimtext

$$\boldsymbol{c}^{\mathrm{e}} = \left(c^{\mathrm{s}}_{-1,\ldots,4}, c^{\mathrm{s}}_{6,\ldots,10}\right).$$

Der resultierende Verlauf ist in Abbildung 2.5a zu sehen. Wie im vorherigen Fall orientiert sich die Zeitskala am Geheimtext auf Empfängerseite, $\boldsymbol{c}^{\mathrm{e}}$. Da dem Symbol c^{s}_5 kein Symbol auf Empfängerseite entspricht, wird die Zeitskala an dieser Stelle unterbrochen und um einen Zeitschritt versetzt fortgesetzt.

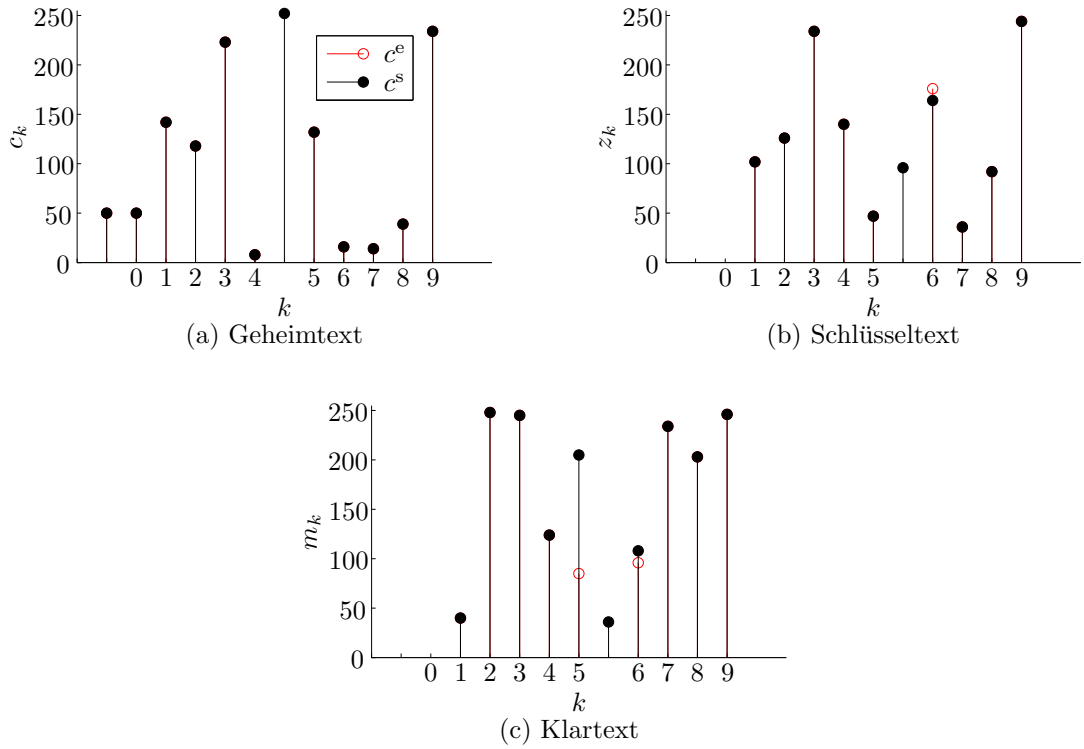
Der Schlüsseltext. Auch hier lässt sich aus (2.5) die Menge der falschen Schlüsseltextsymbole bestimmen:

$$\{z^{\mathrm{e}}_6\}.$$

Der Klartext. Nach (2.8) sind die falschen Klartextsymbole

$$\{m^{\mathrm{e}}_5, m^{\mathrm{e}}_6, \ldots\}.$$

Der Vergleich wird in Abbildung 2.5c gezogen. Insgesamt entsteht ein Verlust von $P+1 = 3$ Symbolen. Die P Symbole gehen zu Lasten des Verschlüsselungsverfahrens, der Verlust des weiteren Symbols entsteht durch den Fehler selbst.



(a) Geheimtext (b) Schlüsseltext

(c) Klartext

Abbildung 2.5: Bitslip II

Bitfehler

Der Geheimtext. Der Wert des Geheimtextsymbols zum Zeitpunkt $k = 5$, c_5^e, wird um Δc_5 verfälscht. Ein möglicher Unterschied in der Taktrate von Sender und Empfänger wird nicht berücksichtigt. Der Empfänger erhält den Geheimtext

$$\boldsymbol{c}^e = \left(c^s_{-1,\ldots,4}, c^s_5 + \Delta c_5, c^s_{6,\ldots,10}\right), \qquad \text{mit} \quad \Delta c_5 = 10.$$

In Abbildung 2.6a ist der Verlauf der Geheimtexte zu sehen. Die zeitskala richtet sich auch hier nach dem Geheimtext auf Empfängerseite.

Der Schlüsseltext. Die Symbole des Schlüsseltextes entsprechen ihrem Gegenüber auf Senderseite bis einschließlich z_5^e. Die folgenden $P = 2$ Schlüsseltextsymbole,

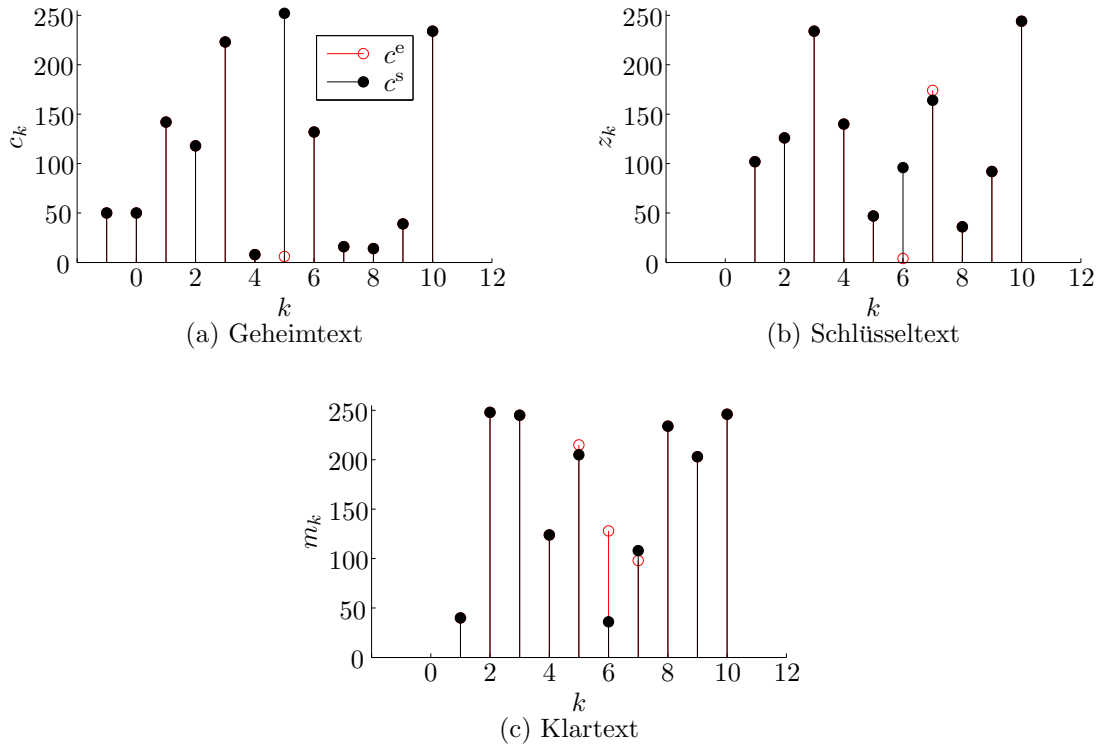
$$z_6^e \neq z_6^s \quad \text{und} \quad z_7^e \neq z_7^s,$$

sind falsch, da sie in Abhängigkeit des verfälschten Geheimtextsymbols bestimmt werden. Für die Bestimmung der Menge der falschen Schlüsseltextsymbole wird (2.9) benutzt. Für die Rekonstruktion weiterer Schlüsseltextsymbole stehen wieder die $P = 2$ benötigten Geheimtextsymbole in Übereinstimmung mit $\boldsymbol{c}^{\mathrm{s}}$ zur Verfügung. Es treten keine weiteren Fehler auf. In Abbildung 2.6b lässt sich dies nachvollziehen.

Der Klartext. In der Summe können $P+1 = 3$ Klartextsymbole nicht korrekt zurückgewonnen werden. Die Klartextsymbole, die falsch entschlüsselt werden, sind nach (2.10) die Symbole

$$\{\, m_5^{\mathrm{e}}, m_6^{\mathrm{e}}, m_7^{\mathrm{e}} \,\}\,.$$

Die Darstellung des Klartextes ist in Abbildung 2.6c zu finden. Gleich den vorherigen Abbildungen ist eine Überlagerung der Symbolfolge des Klartextes auf Empfängerseite mit der auf Senderseite zu sehen.



(a) Geheimtext

(b) Schlüsseltext

(c) Klartext

Abbildung 2.6: Bitfehler

Kapitel 3

Grundbegriffe der Algebra und der Modultheorie

Für das Verständnis des komplexen Aufbaus der in Kapitel 4 verwendeten Moduln, sind Kenntnisse über die zugrunde liegenden mathematischen Strukturen notwenig. Diese sind einerseits ein Polynomring in einer Veränderlichen, der die Koeffizienten der Modulelemente bereitstellt und andererseits ein Körper, aus dem wiederum die Koeffizienten der Veränderlichen im Polynomring entstammen. In den ersten Abschnitten werden diese Begriffe im hier benötigten Umfang erklärt ohne dabei Rücksicht auf eine mathemathisch exakte Darstellung zu nehmen. Eine vollständige Defintion kann in jeder Standardliteratur zur Algebra gefunden werden, z.B. in [7] oder [8].
Im Anschluss werden die Eigenschaften des Moduls sowie der Modulelemente und ihrer Beziehungen untereinander beschrieben und eine systemtheoretische Interpretation gegeben. Eine tiefer gehende Einführung in die Modultheorie, aber ohne dirketen Bezug zur Systemtheorie, enthält [9].

Abelsche Gruppe Eine abelsche (oder auch: kommutative) Gruppe $\mathcal{G}$ ist eine Menge von Elementen, für die eine Verknüpfung $\circ$ existiert. Die Verknüpfung weist jedem Paar von Elementen $a, b \in \mathcal{G}$ ein Element $a \circ b \in \mathcal{G}$ zu.
Die Verknüpfung ist assoziativ und kommutativ. Des Weiteren existiert ein neutrales Element e und zu einem beliebigen Element $a \in \mathcal{G}$ ein inverses Element $a^{-1} \in \mathcal{G}$, so dass gilt $a \circ a^{-1} = e$.

Ring Ein Ring $\mathcal{R}$ ist eine Menge von Elementen, für die zwei Verknüpfungen, die Addition $(+)$ und die Multiplikation $(\cdot)$ existieren. Bezüglich der Addition bilden die Elemente von $\mathcal{R}$ eine abelsche Gruppe mit dem neutralen Element 0.
Die Multiplikation ist assoziativ und es existiert ein neutrales Element 1. Die Existenz einens inversen Elements wird nicht gefordert. Falls die Multiplikation kommutativ ist, heißt $\mathcal{R}$ kommutativ.

Körper Ein Körper $\mathcal{K}$ ist ein kommutativer Ring, in welchem für jedes Element (außer Null) ein Inverses Elemente der Multiplikation existiert. Für das schwächere Kriterium

eines Schiefkörpers wird die Kommutativität der Multpilikation nicht gefordert.

Polynomring Die Elemente eines Polynomrings $k[\delta]$, Polynome in der Veränderlichen δ, erfüllen die strukturellen Anforderungen eines Ringes. Ein Element $\alpha \in k[\delta]$ besitzt allgemein die Form

$$\alpha = a_0 + a_1\delta + a_2\delta^2 + \ldots, \qquad \text{mit} \quad a_0, a_1, a_2, \ldots \in \mathcal{K},$$

wobei die Koeffizienten $a_0, a_1, a_2, \ldots$ die Elemente eines Körpers sind. Eine gewöhnliche lineare Differenzengleichung in der unbekannten Zeitfunktion y_k,

$$a_0 y_k + a_1 y_{k+1} + a_2 y_{k+2} + \ldots = 0,$$

lässt sich so durch

$$\alpha y_k = 0, \qquad \text{mit} \quad \alpha = a_0 + a_1\delta + a_2\delta^2 + \ldots \in k[\delta]$$

darstellen. Dabei ist δ der Verschiebeoperator. Sind die Koeffizienten aus $\mathcal{K}$ konstant oder zeitinvariant, ist $k[\delta]$ kommutativ, sind sie zeitvariant, ist $k[\delta]$ nichtkommutativ.

Hauptidealring In einem Hauptidealring ist jedes Ideal ein Hauptideal (oder auch zyklisches Ideal), d.h. jedes Ideal kann durch ein einzelnes Ringelement erzeugt werden. Ein Ideal $\mathcal{A}$ eines Ringes $\mathcal{R}$ ist ein Untermenge von $\mathcal{R}$, so dass die Multiplikation von einem Element r aus $\mathcal{R}$ und einem Element a aus $\mathcal{A}$ in $\mathcal{A}$ geschlossen ist,

$$r\, a \in \mathcal{A}, \qquad \text{mit} \quad r \in \mathcal{R},\ a \in \mathcal{A}.$$

Ein Polynomringe in einer Veränderlichen über einem Körper ist ein Hauptidealring.

Torsionsmodul Die Elemente eines Moduls $\mathcal{M}$ lassen sich aufteilen in Torsionselemente und freie Elemente. Für ein Torsionselement z^{t} gibt es einen Koeffizienten a, so dass gilt

$$a\, z_k^{\mathrm{t}} = 0, \qquad \text{mit} \quad a \in k[\delta], z \in \mathcal{M}.$$

Werden die Elemente von $\mathcal{M}$ als Systemgrößen aufgesfasst, so existiert für die Systemgröße z^{t} eine autonome Differenzengelichung, d.h. eine Differenzengelichung auf die durch andere Systemgrößen kein Einfluss genommen werden kann,

$$a_0\, z_k^{\mathrm{t}} + a_1\, z_{k+1}^{\mathrm{t}} + a_2\, z_{k+2}^{\mathrm{t}} + \ldots = 0, \qquad \text{mit} \quad a_0, a_1, a_2, \ldots \in \mathbb{R}.$$

Die Systemgröße z^{t} gehört zusammen mit den restlichen Torsionselementen zu einem nicht steuerbaren Teilsystem von $\mathcal{M}$. Dieses Teilsystem ist ein Untermodul von $\mathcal{M}$ und wird als Torsionsmodul bezeichnet. Generell gilt: Enthält ein Modul nur Torsionselement, wird es als Torsionsmodul bezeichnet.

Freies Modul Lässt sich ein Element z aus dem Modul $\mathcal{M}$ als Linearkombination anderer Elemente darstellen, gibt es also eine Beziehung der Form

$$z = \beta_1\, b_1 + \beta_2\, b_2 + \ldots, \qquad \text{mit} \quad z, b_1, b_2, \ldots \in \mathcal{M},\ \beta_1, \beta_2, \ldots \in k[\delta],$$

ist das Element z frei. Ein Modul dessen Elemente alle frei sind, wird als freier Modul bezeichnet. Sidn die Elemente $b_1, b_2, \ldots$ linear unabhängig sind sie eine Basis dieses freien Moduls. Ein freies Modul besitzt immer eine Basis.

Quotientenmodul Die Struktur die sich durch Bilden des Quotientenmoduls ergibt, spielt eine zentrale Rolle in der modultheoretischen Betrachtung linearer Systeme.
Sei $\mathcal{M}$ ein $k[\delta]$-Modul und $\mathcal{U}$ ein Untermodul von $\mathcal{M}$. Das Element m ist ein Elemente aus $\mathcal{M}$, das Element u ist ein Element aus $\mathcal{U}$, $m \in \mathcal{M}$ und $u \in \mathcal{U}$. Die Bildung des Quotionenmoduls $\mathcal{M}/\mathcal{U}$ ordnet einem beliebigen Element m seine sogennante Restklasse $m + \mathcal{U}$ zu. Für m existiert auch die Bezichnung Repräsentant der Restklasse $m + \mathcal{U}$. Lässt sich ein beliebiges Element $m_1 \in \mathcal{M}$ durch eine Linearkombination von Elementen aus $\mathcal{U}$ darstellen, ist es also in U enthalten, bewirkt dies ein Abbildung von m_1 nach $\mathcal{U}$. Ein weiteres beliebiges Element $m_2 \in \mathcal{M}$ wird ebenfalls der selben Restklasse zugeordnet. Ein Unterschied zweier beliebiger kanonischer Bilder $\bar{m}_1$ und $\bar{m}_2$ der Elemente m_1 und m_2 aus $\mathcal{M}$ in $\mathcal{M}/\mathcal{U}$ existiert also nur, wenn der Unterschied nicht durch eine Linearkombination von Elementen aus $\mathcal{U}$ darstellbar ist. Der Kern dieser Abbildung sind demnach die Elemente des Moduls $\mathcal{U}$, denn eine Operation mit Elementen aus $\mathcal{U}$ in $\mathcal{M}/\mathcal{U}$ bewirkt keine Änderung. Eine beliebige Linearkombination kanonischer Bilder der Element von $\mathcal{U}$ sind demnach in $\mathcal{M}/\mathcal{U}$ Null, d.h. stellen das neutrale Element dar. Diese Abbildung wird als Homomorphismus bezeichent. Der Zusammenhang zwischen dem Rang von $\mathcal{M}$, Untermodul $\mathcal{U}$ und Quotientenmodul $\mathcal{M}/\mathcal{U}$ ist

$$\operatorname{rg} \mathcal{M}/\mathcal{U} = \operatorname{rg} \mathcal{M} - \operatorname{rg} \mathcal{U}. \tag{3.1}$$

Die Bildung des Quotientenmoduls bei der Erzeugung des Systemmoduls zeigt 4.1.

Basis Ein endlich erzeugter Modul enthält eine endliche Anzahl von Elementen, die nicht zwingend linear unabhängig sein müssen. Ein Erzeugendensystem dieses Moduls ist eine Menge von Elementen des Moduls, so dass alle anderen, verbleibenden Elemente durch Linarkombination der Elemente des Erzeugendensystems dargestellt werden können. Für einen Modul $\mathcal{M}$, für den die Elemente a, b und c ein Erzeugendensystem darstellen, schreibt man

$$\mathcal{M} = [a, b, c]\,.$$

Sind die Elemente des Erzeugendensystems linear unabhängig, stellt das Erzeugendensystem eine Basis des Moduls dar. Im Gegensatz zu Vektorräumen besitzen Moduln im

Allgemeinen nicht zwingend eine Basis.
Existiert eine Basis b, exisitiert auch für jedes Element z des Moduls eine Beziehung

$$z = \beta\, b, \qquad \text{mit} \quad z, b \in \mathcal{M},\ \beta \in k[\delta].$$

Rang Der Rang eines Moduls steht für die Anzahl linear unabhängiger Elemente des Moduls. Der Rang des Moduls entspricht dem Rang seiner Basis und damit der Anzahl der Elemente, die nötig sind, um den Modul aufzuspannen.

Kapitel 4

Moduln über einem nichtkommutativen Hauptidealring

Ein Modul stellt eine Verallgemeinerung eines Vektorraumes dar. Im Gegensatz zu einem Vektorraum, der über einem Schiefkörper definiert ist, ist ein Modul über einem Ring, und hier im Speziellen über einem Hauptidealring, definiert. Umgekehrt kann ein Vektorraum als Spezialfall eines Moduls betrachtet werden. Die Koeffizienten der Elemente eines Moduls sind die Elemente eines Hauptidealrings: eines Polynomrings $k[\delta]$ in der Veränderlichen δ über dem Körper der reellen Zahlen. Der Polynomring kann kommutativ oder nichtkommutativ sein.
In der Konsequenz kann nicht zu jedem beliebigen Koeffizienten eines Modulelements ein Inverses gefunden werden. D.h. zu einem beliebigen Element $\alpha \in k[\delta]$ existiert im Allgemeinen kein $\beta \in k[\delta]$, so dass die Multiplikation beider Elemente die 1, das neutrale Element der Multiplikation, ergibt,

$$\alpha\beta \neq 1, \quad \beta\alpha \neq 1, \qquad \text{mit} \quad \alpha, \beta \in k[\delta].$$

Sei M ein Modul über einem Polynomring $k[\delta]$, kurz $[\delta]$-Modul, und m ein beliebiges Element aus M. Existiert für m eine Beziehung der Form

$$\alpha m = 0, \qquad \text{mit} \quad \alpha \in k[\delta],\ m \in M \tag{4.1}$$

kann daraus nicht abgeleitet werden, dass einer der beteiligten Faktoren Null sein muss. Das Element m kann im Gegenteil sogar eine sehr komplexe Struktur besitzen. Wäre m jedoch ein Element eines Vektorraums, würde aus (4.1) mit $\alpha \neq 0$ folgen, dass m der Nullvektor ist.
Durch die oben beschriebenen Eigenschaften eines Moduls und seiner Elemente lässt sich ein beliebiges lineares zeitdiskretes System Σ, das durch ein System von linearen Differenzengleichungen beschrieben wird, als ein endlich erzeugtes $k[\delta]$-Modul auffassen. Eine allgemeine Form eines Systems von Differenzengleichungen für eine System dieser Klasse ist (4.3). Das zu einem linearen System gehörende $k[\delta]$-Modul wird im Weiteren als Systemmodul oder gleich dem System als Σ bezeichnet. Das Systemmodul ist keine weitere Möglichkeit der Darstellung eines Systems, wie z.B. die Zustandsraumdarstellung, sondern es ist das System. D.h. ein lineares zeitdiskretes System ist ein $k[\delta]$-Modul. Daher

wird die Bezeichnung Σ gleichermaßen für das System und das Systemmodul verwendet. Wie ein solches Systemmodul erzeugt werden kann, wird in Abschnitt 4.1 gezeigt.
Die Unbestimmte δ im Polynomring $k[\delta]$ ist ein Verschiebeoperator um eine Zeiteinheit nach rechts. Für eine beliebige Systemgröße w_k von Σ gilt der Zusammenhang

$$\delta w_k = w_{k+1}.$$

Wird der Schiebeoperator auf das Produkt aus einem zeitvarianten Koeffizienten α_k und einer Systemgröße w_k angewendet, gilt

$$\delta a_k \, w_k = a_{k+1} \, w_{k+1}, \qquad \text{mit} \quad a \in \mathbb{R}. \tag{4.2}$$

Die Frage der Zeitvarianz eines Systems beeinflusst dabei nur die Kommutativität des dem Systemmodul zu Grunde liegenden Polynomringes $k[\delta]$: Ist das System zeitinvariant, dann ist $k[\delta]$ kommutativ, ist das System zeitvariant, ist $k[\delta]$ nichtkommutativ. Die Elemente des Polynomringes $k[\delta]$ sind von der Form

$$a_0 + a_1 \, \delta + a_2 \, \delta^2 + \ldots, \qquad \text{mit} \quad a_0, a_1, a_2, \ldots \in k.$$

Im vorliegenden Fall, also zur Nachbildung von linearen Differenzengleichungen, ist k der Körper der reellen Zahlen $\mathbb{R}$; k wird als Grundkörper bezeichnet. Handelt es sich um zeitvariante Differenzengleichungen, sind die Elemente von k ebenfalls zeitvariant. Sind die Differenzengleichungen zeitinvariant, ist k ein Konstantenkörper. Betrachtet man zwei Elemente α und β aus dem Polynomring $k[\delta]$,

$$\begin{aligned} \alpha &= a_0 + a_1 \, \delta + \ldots, \\ \beta &= b_0 + b_1 \, \delta + \ldots, \end{aligned} \qquad \text{mit} \quad a_0, a_1, \ldots, b_0, b_1, \ldots \in k,$$

dann ist die Gleichung

$$\begin{aligned} \alpha\beta &= (a_0 + a_1 \, \delta + \ldots)\,(b_0 + b_1 \, \delta + \ldots) = a_0 \, b_0 + a_0 \, b_1 \, \delta + a_1 \, \delta \, b_0 + a_1 \, \delta \, b_1 \, \delta \\ &= b_0 \, a_0 + b_0 \, a_1 \, \delta + b_1 \, \delta \, b_0 + b_1 \, \delta \, b_1 \, \delta = \beta\alpha \end{aligned}$$

mit (4.2) im Allgemeinen nur dann richtig, wenn k ein Konstantenkörper ist und somit

$$a_i = a_j \quad \text{und} \quad b_i = b_j \qquad \forall \, i, j \in \mathbb{N}$$

gilt. Der Polynomring $k[\delta]$ ist demnach nur dann kommutativ, wenn der Grundkörper k ein Konstantenkörper ist und damit gleichbedeutend die Differenzengleichungen des Systems zeitinvariant sind.
Den ersten Vorschlag der Auffasung eines linearen Systemes als Modul machte Michel Fliess in [10]. Vom selben Autor stammen die Artikel [11], [12] und [13], die ebenfalls eine gute Einführung in die Theorie der Moduln geben. Der Nachweis verschiedener systemtheoretischer Eigenschaften mit Hilfe der Modultheorie und die Anwendung an Beispielen kann in [14] gefunden werden. In [15] wird eine algebraische Heranführung an grundsätzliche Fragen bei der Untersuchung linearer Systeme gegeben. Vom gleichen Autor, aber mehr fokussiert auf die Fragestellung in dieser Arbeit, ist [16].

4.1 Systemmodul

Die Ausgangssituation ist ein System von q unabhängigen linearen Differenzengleichungen in s Variablen, den Systemgleichungen. Diese beschreiben das dynamische Verhalten eines Systems Σ. Die s Variablen $w_1 \dots w_s$ werden als Systemgrößen bezeichnet. Ihre Wahl ist für ein gegebenes System nicht eindeutig, jede Linearkombination aus den Systemgleichungen ist ebenfalls eine Beschreibung des Systemverhaltens. Es wird dabei zunächst nicht zwischen Eingangs- und Ausgangsvariablen und Variablen, welche die inneren Zustände des Systems beschreiben, unterschieden. Allgemein sind die Systemgleichungen von der Form

$$\begin{aligned} \left(a_{11}^0 + a_{11}^1\,\delta + a_{11}^2\,\delta^2 + \dots\right)\, w_1 + \dots + \left(a_{1s}^0 + a_{1s}^1\,\delta + a_{1s}^2\,\delta^2 + \dots\right)\, w_s &= 0 \\ &\vdots \\ \left(a_{q1}^0 + a_{q1}^1\,\delta + a_{q1}^2\,\delta^2 + \dots\right)\, w_1 + \dots + \left(a_{qs}^0 + a_{qs}^1\,\delta + a_{qs}^2\,\delta^2 + \dots\right)\, w_s &= 0. \end{aligned} \tag{4.3}$$

Die Koeffizienten $a_{11}^0, a_{11}^1, \dots$ sind Elemente aus dem Körper der reellen Zahlen $\mathbb{R}$,

$$a_{11}^0, a_{11}^1, \dots \in \mathbb{R}.$$

Wählt man die Koeffizienten der Systemgrößen aus dem nichtkommutativen Polynomring $k[\delta]$, lässt sich (4.3) kompakter schreiben:

$$\begin{aligned} \alpha_{11}\, w_1 + \dots + \alpha_{1s}\, w_s &= 0 \\ &\vdots \qquad \text{mit} \quad \alpha_{11}, \dots, \alpha_{qs} \in k[\delta] \\ \alpha_{q1}\, w_1 + \dots + \alpha_{qs}\, w_s &= 0. \end{aligned} \tag{4.4}$$

Die Matrix der Koeffizienten wird als Präsentationsmatrix $\boldsymbol{P}$ bezeichnet. Die Matrix $\boldsymbol{P}$ ist eine Polynommatrix, ihre Einträge sind die Polynome $\alpha_{11}, \dots, \alpha_{qs}$ aus dem Polynomring $k[\delta]$,

$$\boldsymbol{P} = \begin{pmatrix} \alpha_{11} & \dots & \alpha_{1s} \\ \vdots & \ddots & \vdots \\ \alpha_{q1} & \dots & \alpha_{qs}. \end{pmatrix}, \qquad \text{mit} \quad \alpha_{11}, \dots, \alpha_{qs} \in k[\delta]$$

Mit $\boldsymbol{P}$ lässt sich (4.4) in der Matrixform

$$\boldsymbol{P} \begin{pmatrix} w_1 \\ \vdots \\ w_s \end{pmatrix} = 0, \qquad \text{mit} \quad \boldsymbol{P} \in k[\delta]^{qxs} \tag{4.5}$$

darstellen.
Das Systemmoduls Σ zu (4.5) lässt sich mit Hilfe Präsentationsmatrix $\boldsymbol{P}$ folgendermaßen

erzeugen: Es existiert immer ein freies Modul F mit einer Basis b_F. Die Elemente der Basis sind

$$b_\mathrm{F} = \{W_1, \ldots, W_s\},$$

der Rang von F entspricht der Anzahl der Basiselemente und damit der Anzahl der Systemgrößen von Σ:

$$\mathrm{rg}\ F = s.$$

Die Basiselemente werden mit $W_1, \ldots, W_s$ bezeichnet, da sie die Repräsentanten der kanonischen Bilder $w_1, \ldots, w_s$ sind. Die kanonischen Bilder sind die Elemente des Systemmoduls und können als Systemgrößen des entsprechenden Systems interpretiert werden. Zwischen $W_1, \ldots W_s$ und den Systemgrößen besteht jedoch kein direkter Zusammenhang. Es existiert ebenso ein Modul E mit dem Erzeugendensystem $\{E_1, \ldots, E_q\}$. Die Elemente des Erzeugendensystems ergeben sich aus Linearkombinationen der Basiselemente von F, dadurch ist E ein (freies) Untermodul von F. Die Art der Linearkombinationen wird durch die Systemgleichungen (4.4) bzw. von der Präsentationsmatrix $\boldsymbol{P}$ vorgegeben

$$\begin{pmatrix} E_1 \\ \vdots \\ E_q \end{pmatrix} = \boldsymbol{P} \begin{pmatrix} W_1 \\ \vdots \\ W_s \end{pmatrix}.$$

Da die Differenzengleichungen (4.3) linear unabhängig sind, gilt dies auch für die Zeilen von $\boldsymbol{P}$ und damit auch für die Elemente des Erzeugendensystems von E. D.h. $\{E_1, \ldots, E_q\}$ ist eine Basis von E,

$$b_\mathrm{E} = \{E_1, \ldots, E_q\},$$

der Rang von E ist

$$\mathrm{rg}\ E = q. \tag{4.6}$$

Die Elemente von F müssen keinerlei Beziehungen erfüllen und können daher allen erdenklichen Differenzengleichungen in den Variablen $W_1, \ldots, W_s$ genügen. Sie besitzen noch keinen Bezug zu einem konkreten System, dessen Verhalten durch die Systemgleichungen (4.3) beschrieben werden kann.
Lässt sich das Verhalten eines Systems in den Systemvariablen $W_1, \ldots, W_s$ mit Hilfe eines Erzeugendensystems $\{E_1, \ldots, E_q\}$ von E durch

$$\begin{aligned} E_1 &= 0 \\ &\vdots \\ E_q &= 0 \end{aligned}$$

beschreiben, so ist dies ebenfalls durch jedes weitere Erzeugendensystem von E möglich. Das Modul E lässt sich als die Gesamtheit aller möglichen nichttrivialen Systemgleichungen zu einem konkreten System interpretieren.

Die Beziehungen der Elemente des Systemmodul Σ sollen durch die Systemgleichungen des entsprechenden Systems gegeben sein. Diese Struktur ergibt sich aus den oben erzeugten Moduln F und E durch Bildung des Quotientenmoduls,

$$\Sigma = F/E.$$

Die kanonische Abbildung $\varphi : F \to F/E$ ordnet jedem Element in F sein kanonisches Bild in F/E zu. Innerhalb der Erläuterungen zur Bildung des Systemmoduls werden die kanonischen Bilder mit kleinen Buchstaben bezeichnet, ihre Repräsentanten, also die Elemente in F, mit großen. Der Kern der Abbildung φ ist

$$\ker \varphi = E,$$

d.h. sämtliche kanonischen Bilder $e_1, \ldots, e_p$ der Elemente $E_1, \ldots, E_p$ im Untermodul E von F, erfüllen in Σ die Bedingung $e = 0$ (Abbildung 4.1). Damit wird deutlich, dass der Quotientenmodul F/E als das System Σ aufgefasst werden kann. Der Rang von Σ ist nach (3.1)

$$\operatorname{rg} \Sigma = \operatorname{rg} F - \operatorname{rg} E = s - q.$$

Aus der Untersuchung des Quotientenmoduls F/E können nun Rückschlüsse auf Eigenschaften des Systems Σ geschlossen werden.

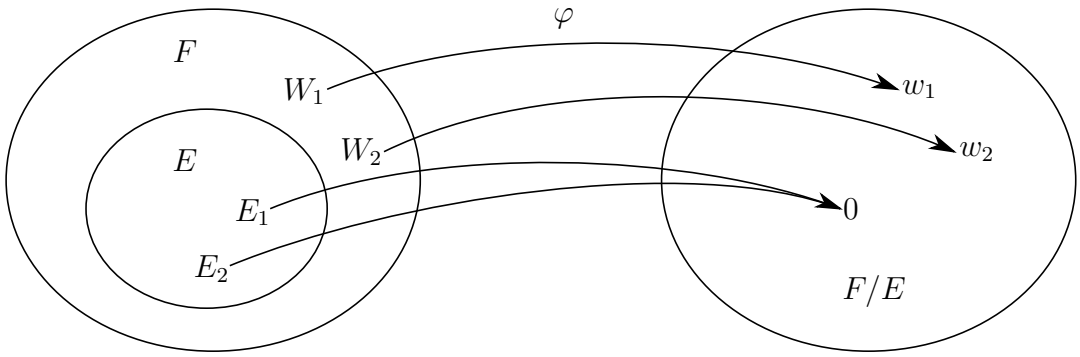


Abbildung 4.1: Erzeugung des Systemmoduls

4.2 Eingangs-Ausgangssystem

In der bisherigen Betrachtung wurde auf eine Interpretation der Systemgrößen bzw. der Elemente des Systemmoduls verzichtet. Unabhängig von der konkreten Realisierung des Systems kann anhand des Systemmoduls ein Eingang u_k und ein Ausgang y_k definiert werden. Die Eingangsgrößen sind $u_k = u_k^1, \ldots, u_k^m$, die Ausgangsgrößen $y_k = y_k^1, \ldots, y_k^p$. Da u_k und y_k Elemente von Σ sind, erzeugen sie jeweils ein Untermodul von Σ. Das

Untermodul $[u_k]$ mit dem Eingang als Erzeugendensystem ist das Eingangsmodul, das Untermodul $[y_k]$, erzeugt durch den Ausgang, ist das Ausgangsmodul.
Die Eingangsgrößen werden so gewählt, dass durch sie eine Beeinflussung einer beliebigen Systemgröße z möglich ist. Im Systemmodul entspricht dies einer Beziehung der Elemente der Form

$$\alpha\, z_k = \beta_1\, u_{1,k} + \beta_2\, u_{2,k} + \ldots, \qquad \text{mit} \quad \alpha, \beta_1, \beta_2, \ldots \in k[\delta],\ z, u_k \in \Sigma.$$

Damit ein solcher Zusammenhang im Systemmodul besteht, muss das Quotientenmodul aus Systemmodul und Eingangsmodul ein Torsionsmodul sein, d.h.

$$\operatorname{rg} \Sigma/[u_k] = 0.$$

Eine Systemgröße ist eine Ausgangsgröße, wenn die Elemente des Eingangs und die des Ausgangs ein Erzeugendensystem des Systemmoduls sind,

$$[u_k,\ y_k] = \Sigma.$$

Ist dies der Fall, wird $[u_k,\ y_k]$ als Eingangs-Ausgangs-System bezeichnet. Wenn nicht, bezeichnet man $[u_k,\ y_k]$ als Eingangs-Ausgangs-Teilsystem und es gilt

$$\Sigma = [u_k,\ y_k] \oplus \Sigma/[u_k,\ y_k].$$

Kapitel 5

Invertierbarkeit und Steuerbarkeit

Die Idee chaotische dynamische Systeme zur Verschlüsselung von Nachrichten zu benutzen, verzweigt sich in unterschiedliche Varianten der Umsetzung. Eine erfolgsversprechende Möglichkeit ist, das System mit der zu verschlüsselnden Nachricht $\boldsymbol{m}$ als Eingangssignal zu beaufschlagen. Ein solches System wird als Message-embedded System bezeichnet. Der Geheimtext $\boldsymbol{c}$ ist der Ausgang des Systems, der Schlüsseltext $\boldsymbol{z}$ ergibt sich aus einer Kombination interner Zustände. Unter bestimmten Voraussetzungen lässt sich der Schlüsseltext als eine Folge von zurückliegenden Symbolen des Geheimtextes ausdrücken. Hinsichtlich des Synchronisationsproblems zwischen Sender und Empfänger bringt dies für ein Verschlüsselungssystem wünschenswerte Eigenschaften mit sich. Die Systeme, die im Folgenden auf diese Voraussetzungen geprüft werden, entstammen der Klasse der stückweise linearen diskreten SISO-Systeme.
Als Ausgangssituation wird, wie auch in [1], von einer Systembeschreibung in Zustandsraumdarstellung ausgegangen. Ein stückweise lineares diskretes SISO-System der Ordnung n besitzt die Zustandsraumdarstellung

$$\begin{aligned} x_{k+1} &= A_j\, x_k + B_j\, u_k \\ y_k &= C_j\, x_k + D_j\, u_k. \end{aligned} \tag{5.1}$$

Der Index j der Matrizen A_j, B_j, C_j und D_j ergibt sich zu einem Zeitpunkt k aus einer Umschaltfunktion $\sigma\,(k, x_k)$ in Abhängigkeit des Zeitpunkts selbst und der Zustände des Systems x_k zu diesem Zeitpunkt,

$$j = \sigma\,(k, x_k) \in \{1, \ldots, J\}.$$

Die möglichen Matrizen, die angenommen werden können, sind die Elemente der endlichen Mengen

$$\begin{aligned} A_j &\in \{A_1, \ldots, A_J\} \in \mathbb{R}^{n\times n}, \\ B_j &\in \{B_1, \ldots, B_J\} \in \mathbb{R}^{n\times 1}, \\ C_j &\in \{C_1, \ldots, C_J\} \in \mathbb{R}^{1\times n} \quad \text{und} \\ D_j &\in \{D_1, \ldots, D_J\} \in \mathbb{R}. \end{aligned}$$

Es wird zunächst angenommen, die Umschaltfunktion $\sigma\,(k, x_k)$ sei eine Funktion allein vom Zeitpunkt k, der aktuelle Zustand x_k hat keinen Einfluss: $\sigma\,(k, x_k) = \sigma(k)$. Das

System (5.1) lässt sich dann offensichtlich als ein Spezialfall eines linearen diskreten zeitvarianten SISO-System der Form

$$\begin{aligned} x_{k+1} &= A_k\, x_k + B_k\, u_k \\ y_k &= C_k\, x_k + D_k\, u_k. \end{aligned} \tag{5.2}$$

interpretieren. Denn zu jedem Zeitpunkt ist eine Variation der Koeffizienten der Matrizen möglich. Aber auch im Fall einer allgemeinen Umschaltfunktion, $\sigma\,(k, x_k) = \sigma\,(k, x_k)$, kann das Verhalten durch 5.2 beschrieben werden. Denn obwohl die Auswahl der Matrizen A_j, B_j, C_j und D_j von einer eventuell nichtlinearen Funktion der Zustände abhängt, erhält man zu jedem Zeitpunkt ein lineares System, denn der Wert der Koeffizienten ist nicht mit dem Wert der Systemzustände gekoppelt. Die Matrizen des Systems mit einer einer Umschaltfunktion $\sigma\,(k, x_k)$ sind Elemente der gleichen Menge, wie auch im Fall der allein zeitabhängigen Umschaltfunktion. Die Abweichung beschränkt sich auf die Art der Umschaltsequenz, also die Folge von Werten, die die Umschaltfunktion mit der Zeit annimmt. Dies hat jedoch keinen Einfluss auf die Klassifizierung des Systemverhaltens. Damit lässt sich zusammenfassen, dass ein stückweise lineares diskretes SISO-System als ein Spezialfall der Klasse linearer diskreter zeitvarianter SISO-System aufgefasst werden kann – unbeachtet der Umschaltfunktion.
Mit den Erkenntnissen aus [1] ergibt sich die Frage nach der Invertierbarkeit und der Flachheit des Systemausgangs der Klasse von Systemen der Form (5.1). An diese Eigenschaften ist die Verwendbarkeit zur SSSC gekoppelt. Die Vorgehensweise in [1] wird in 1 kurz erläutert. In den folgenden Abschnitten wird die Notwendigkeit dieser Eigenschaften erläutert und der Nachweis dafür aus Sicht der in Kapitel 4 vorgestellten Modultheorie untersucht. Anschließend wird die Anwendung an einem konkreten Beispiel gezeigt. Die Ausführung erfolgt anhand eines linearen diskreten zeitvarianten SISO-Systems. Wie soeben begründet stellt dies eine verallgemeinerte Betrachtung dar.

5.1 Invertierbarkeit

Bei einem Verschlüsselungssystem muss es offensichtlich möglich sein aus dem Geheimtext eindeutig den Klartext zu rekonstruieren; die empfangene Nachricht wäre sonst unbrauchbar. Ist das System (5.2) (links-)invertierbar, existiert ein inverses System mit dem Eingang y_k, dem Geheimtext, und dem Ausgang u_k, dem Klartext. Die inneren Zustände sind erst bei der Untersuchung des Ausgangs auf Flachheit von Interesse.
Der Systemmodul Σ zu einem System der Form (5.2) lässt sich, wie in Abschnitt 4.1 beschrieben, konstruieren. Der Rang von Σ muss entsprechend (3.1)

$$\operatorname{rg} \Sigma = 1$$

sein. Unter der Annahme, dass der Eingang u_k linear unabhängig ist und nur ein Element enthält, ergibt sich für das Eingangsmodul $[u_k]$ der Rang ebenfalls

$$\operatorname{rg} [u_k] = 1.$$

Die Annahme der linearen Unabhängigkeit eines Eingangs mit nur einem Elemente ist für die vorliegende Anwendung als Verschlüsselungssystem immer erfüllt. Denn der Eingang, die Nachricht, muss keinerlei Bedingungen gehorchen. Die lineare Unabhängigkeit ist nicht gegeben, wenn der Eingäng eine Systemgröße eines autonomen Systems ist.
Der Rang des durch den Ausgang y_k erzeugten Ausgangsmoduls $[y_k]$ kann maximal denselben Rang erreichen; ein System mit nur einem unabhängigen Eingang kann nicht mehrere unabhängige Ausgänge besitzen. Ist der Rang kleiner, $[y_k]$ also ein Torsionsmodul, besteht kein Zusammenhang zwischen dem Ausgang und dem System Σ. Der Ausgang genügt in diesem Fall einer autonomen Differenzengleichung, einer Differenzengleichung auf die durch andere Systemgrößen kein Einfluss genommen werden kann. Dieser Fall kann hinischtlich der Anwendung ausgeschlossen werden. Deswegen gilt für den Rang von $[y_k]$

$$\operatorname{rg}\,[y_k] = 1.$$

Wird der Quotientenmodul aus Systemmodul und Ausgangsmodul gebildet, gilt für den Rang des Quotientenmoduls mit (3.1)

$$\operatorname{rg}\,\Sigma/[y_k] = \operatorname{rg}\,\Sigma - \operatorname{rg}\,[y_k] = 0.$$

Im Weiteren werden die kanonischen Bilder im Quotientenmodul mit den gestrichenen Größen der Repräsentanten bezeichnet.
Das Quotientenmodul $\Sigma/[y_k]$ enthält keine unabhängigen Elemente, es ist ein Torsionsmodul, jedes Element ist ein Torsionselement. Das bedeutet, für ein beliebiges Element $\bar{z}$ in $\Sigma/[y_k]$ existiert ein Element α in $k[\delta]$, so dass die Gleichung

$$\gamma_k \bar{z} = 0, \qquad \text{mit} \quad \gamma_k \in k[\delta],\ \bar{z} \in \Sigma/[y_k]$$

erfüllt ist. Für den Repräsentanten z_k im Systemmodul ergibt sich die Bedingung

$$\gamma z_k \in [y_k] \qquad \text{bzw.} \quad \gamma_k z_k = \alpha_k y_k, \qquad \text{mit} \quad \gamma_k, \alpha_k \in k[\delta],\ z_k, y_k \in \Sigma.$$

Eine endliche Folge von z_k und zurückliegenden Werten lässt sich also aus einer Linerakombination von y_k und zurückliegenden Werten darstellen. Neben dem eigentlichen Systemeingang u_k hat y_k ebenfalls Einfluss auf eine beliebige Systemgröße z in Σ. Das bedeutet, y_k ist neben u_k ein unabhängiger Eingang des Systems. Daraus folgt, dass neben dem System mit Eingang u_k und Ausgang y_k auch das (inverse) System mit Eingang y_k und Ausgang u_k existiert, unabhängig von der genauen Realisierung eines Systems der Form (5.2). Dies bedeutet, das System (5.2) ist folglich immer invertierbar!
Es gibt immer einen Zusammenhang zwischen einer Linerakombination des Eingangs und einer endlichen Anzahl zurückliegender Werte und einer des Ausgangs und seinen zurückliegenden Werten. Es existiert also immer ein (beobachtbares) Eingangs-Ausgangs-Teilsystem, welches durch $[u_k,\ y_k]$ erzeugt wird, und dessen Elemente einer Beziehung der Form

$$a_k\, y_k + a_{k+1}\, y_{k+1} + \ldots = b_k\, u_k + b_{k+1}\, u_{k+1} + \ldots \tag{5.3}$$

genügen.
Die Eigenschaft der Invertierbarkeit bedeutet: Aus der Kenntnis einer endlichen Folge des Eingangssignals und der Umschaltfunktion kann die Folge des Ausgangssignals eindeutig bestimmt werden. Und für den umgekehrten Fall, die Bestimmung des Eingangssignals aus dem Ausgangssignals unter Kenntnis der Umschaltfunktion, gilt diese offensichtlich auch.
Wäre dies nicht der Fall, würden die Größen u_k und y_k zu unterschiedlichen Systemen gehören, die keinen Einfluss aufeinander ausüben.
Das System Σ ist die direkte Summe von Eingangs-Ausgangs-Teilsystem $[u_k,\ y_k]$ und dem Quotientenmodul von Σ und $[u_k,\ y_k]$,

$$\Sigma = [u_k, y_k] \oplus \Sigma/[u_k, y_k].$$

5.2 Steuerbarkeit

Mit der Eigenschaft der Invertierbarkeit besteht die prinzipielle Möglichkeit auf Seite des Empfängers aus dem Geheimtext den Klartext eindeutig zurückzugewinnen.
Für ein Verschlüsselungssystem ist es wünschenswert, ein Symbol des Klartextes nach einer endlichen Zeit bzw. aus einer endlichen Folge von Symbolen aus dem Geheimtext zu erhalten. Daher muss eine weitere Bedingung an das System gestellt werden: Der Eingang u_k muss als eine endliche Folge von Symbolen des Ausgangs y_k und zurückliegenden Werten dargestellt werden können. Für die Systemgrößen muss

$$u_k = a_k\, y_k + a_{k+1}\, y_{k+1} + a_{k+2}\, y_{k+2} + ... + a_{k+P-1}\, y_{k+P-1} + a_{k+P+b_s}\, y_{k+P+b_s} \tag{5.4}$$

gelten. Es werden die Bezeichnungen aus Kapitel 2 benutzt. Im Systemmodul entspricht dies

$$u_k = a_k\, y_k + \delta a_k\, y_k + \delta^2 a_k\, y_k + ... + \delta^{P-1} a_k\, y_k + \delta^{P+b_s} a_k\, y_k,$$

wobei es nicht nötig ist, eine Beziehung für ein Klartextsymbol zum Zeitpunkt k (u_k) zu finden. Eine Verschiebung des Klartextes um Δ $(u_{k+\Delta})$ bzw. eine zum Geheimtext verzögerte Rekonstruktion der Symbole des Klartextes, ermöglicht weiterhin eine eindeutige Bestimmung des Klartextes. Im Gegensatz zum zeitkontinuierlichen Fall, wo neben dem Verlauf der Ableitung einer Größe auch deren Anfangsbedingung bekannt sein muss, reicht im diskreten Fall die Kenntnis der zeitverschobenen Größe aus.
Damit eine Beziehung der Form (5.4) existiert, muss das Eingangs-Ausgangs-Teilsystem (5.3) eine Basis y_k besitzen, das Teilsystem mit dem unabhängigen Eingang y_k also steuerbar sein.
Es exisitiert ein Zusammenhang zwischen der Steuerbarkeit eines Systems und der Existenz einer Basis des entsprechenden Systemmoduls. Existiert für ein Systemmodul Σ eine Basis b, so kann jedes Element von Σ als Linearkombination von b dargestellt werden. Im gegebenen Fall bedeutet dies, dass ein beliebiger Zustand des Elementes u_k durch eine Linearkombination von y_k dargestellt werden kann.

Wird das inverse System mit dem Eingang y_k und dem Ausgang u_k betrachtet, so existiert zu jedem beliebigen Zustand des Ausgangs u_k entsprechend (5.4) eine Folge von gewichteten Eingangssignalen $a_k\, y_k + \delta a_k\, y_k + \delta^2 a_k\, y_k + ...$, die den Ausgang des Systems in diesen Zustand überführen. Während bei der Invertierbarkeit ein eindeutiger Zusammenhang zwischen einer Signalfolge des Eingangs y_k und einer Signalfolge des Ausgangs u_k gefordert wird,

$$a_k\, y_k + a_{k+1}\, y_{k+1} + \ldots = b_k\, u_k + b_{k+1}\, u_{k+1} + \ldots$$

kann mit Hilfe eines flachen Ausgangs y_k zu jedem Zeitpunkt der Eingang als Folge von Ausgangssignalen dargestellt werden,

$$u_k = a_k\, y_k + a_{k+1}\, y_{k+1} + \ldots.$$

Es wird angenommen, dass eine Darstellung von (5.2) oder einem Eingangs-Ausgangs-Teilsystem von (5.2) in der Form (5.4) existiert. Wird diese Differenzengleichung um P Zeitschritte nach rechts verschoben und nach y_k umgestellt, erhält man

$$\begin{aligned} y_k &= a_{k-P}\, y_{k-P} + \ldots + a_{k-1}\, y_{k-1} + b_k\, u_k \\ &= \sum_{n=1}^{P} a_{k-n}\, y_{k-n} + b_k\, u_k. \end{aligned} \tag{5.5}$$

Wobei zu Gunsten der Anschaulichkeit $\Delta = P + b_\mathrm{s}$ und $b_\mathrm{s} = 0$ angenommen wurde. Die Koeffizienten a_i und b_j sind jeweils Elemente des Körpers der reellen Zahlen. Sie ergeben sich in Abhängigkeit der Werte der Umschaltfunktion $j = \sigma(k, y_k)$ zu den einzelnen Zeitpunkten innerhalb des Zeitintervalls $[k-P, k-1]$ aus den Einträgen der Matrizen A_j, B_j, C_j und D_j. Die Abfolge der Werte der Umschaltfunktion innerhalb dieses Intervalls wird im Weiteren als Umschaltsequenz bezeichnet und die Umschaltsequenz mit ϵ abgekürzt,

$$\epsilon = \Big(\sigma(k-P, y_{k-P}), \ldots, \sigma(k-P, y_{k-P})\Big).$$

Um die Abhängigkeit von Koeffizienten und Umschaltsequenz deutlich zu machen wird (5.5) im Weiteren als

$$y_k = \sum_{n=1}^{P} a_n^\epsilon\, y_{k-n} + b^\epsilon\, u_k \tag{5.6}$$

geschrieben. Die Koeffizienten sind nicht mehr direkt vom Zeitpunkt k abhängig. Sie werden aber indirekt über die vom Zeitintervall $[k-P, k-1]$ und damit von k abhängige Umschaltsequenz ϵ beeinflusst.

5.3 Beispiel

An einem Beispielsystem wird im Folgenden die Anwendung der in den vorhergehenden Abschnitten beschriebenen Werkzeuge gezeigt. Die Struktur des Beispiels ist inspiriert

durch das in [1] verwendete Beispiel. Hinsichtlich anschaulicher Simulationsergebnisse werden die Parameter hier jedoch abweichend gewählt.
Wir betrachten ein lineares diskretes zeitvariantes System in Zustandsraumdarstellung

$$\begin{aligned} \begin{pmatrix} x_{1,k+1} \\ x_{2,k+1} \end{pmatrix} &= \begin{pmatrix} a_{1,k} & 1 \\ a_{0,k} & 0 \end{pmatrix} \begin{pmatrix} x_{1,k} \\ x_{2,k} \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u_k \\ y_k &= \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} x_{1,k} \\ x_{2,k} \end{pmatrix}. \end{aligned} \tag{5.7}$$

Die zeitvarianten Koeffizienten $a_{0,k}$ und $a_{1,k}$ ergeben sich aus einer zunächst nicht näher bestimmten Umschaltfunktion σ als Elemente einer endlichen geordneten Menge möglicher Werte,

$$\begin{aligned} a_{0,k} &\in \{130, 140, -150, -160\}, \\ b_{1,k} &\in \{-10, 20, -30, 40\}, \end{aligned} \qquad \text{mit} \quad k = 1, \ldots, J = 4.$$

Für die Elemente des Systemmoduls Σ, deren Beziehungen durch die Differenzengleichungen des Systems beschrieben werden, gilt

$$\delta x_{1,k} - a_{1,k}\, x_{1,k} - x_{2,k} = 0 \tag{5.8}$$

$$\delta x_{2,k} - a_{0,k}\, x_{1,k} - u_k = 0. \tag{5.9}$$

Die Erzeugung des Systemmoduls orientiert sich an den in Kapitel 4 beschriebenen Schritten: Es wird ein freies Modul F erzeugt, dessen Rang der Anzahl der Systemgrößen von Σ (u_k, $x_{1,k}$, und $x_{2,k}$) entspricht,

$$F = [W_1, W_2, W_3], \quad \text{rg } F = 3.$$

Ein zweites freies Modul $E \subset F$ wird mit Hilfe der beiden Differenzengleichungen aus (5.7) und den Elementen in E erzeugt,

$$F = [E_1, E_2], \quad \text{rg } E = 2,$$

wobei

$$E_1 = X_{1,k} - a_{1,k}\, X_{1,k} - X_{2,k}, \quad \text{und} \quad E_2 = \delta X_{2,k} - a_{0,k}\, X_{1,k} - U_k.$$

Die Elemente $X_{i,j}$ sind wieder die Repräsentanten der kanonischen Bilder $x_{i,j}$ im Systemmodul Σ nach Bildung des Quotientenmoduls F/E.
Es wird das Quotientenmodul aus F und E gebildet. Innerhalb des Quotientenmoduls $\Sigma = F/E$ gelten die Beziehungen (5.8) und (5.9). Es ergibt sich aus (3.1) rg $\Sigma = 1$ für den Rang von Σ. Wie bereits in Abschnitt 5.1 erkannt, ist das Quotientenmodul aus Systemmodul und Ausgangsmodul $[y_k = x_{1,k}]$ ein Torsionsmodul und das System (5.7) invertierbar.

Aus (5.8) sieht man, dass sich das Element $x_{2,k}$ aus einer $k[\delta]$-Linearkombination von $x_{1,k}$ darstellen lässt:

$$x_{2,k} = \delta x_{1,k} - a_{1,k}\, x_{1,k}.$$

Mit diesem Zusammenhang kann aus (5.9) auch für das Element u_k eine Darstellung als $k[\delta]$-Linearkombination von $x_{1,k}$ angegeben werden:

$$\begin{aligned} u_k &= \delta\, x_{2,k} - a_{0,k}\, x_{1,k} \\ &= \delta\,(\delta x_{1,k} - a_{1,k}\, x_{1,k}) - a_{0,k}\, x_{1,k} \\ &= \delta^2\, x_{1,k} - \delta\, a_{1,k}\, x_{1,k} - a_{0,k}\, x_{1,k} \end{aligned} \tag{5.10}$$

Da offensichtlich alle Elemente von Σ als $k[\delta]$-Linearkombination von $x_{1,k}$ dargestellt werden können, ist $x_{1,k}$ eine Basis von Σ und der Ausgang des Systems $y_k = x_{1,k}$ ein flacher Ausgang. Die Differenzengleichung in u_k und y_k ergibt sich aus (5.10) zu

$$y_{k+2} = a_{1,k+1}\, y_{k+1} + a_{0,k}\, y_k + u_k.$$

Damit besitzt das System die strukturellen Voraussetzungen zur selbstsynchronisierenden Stromverschlüsselung. Die kanonische Darstellung in der Form (2.1) und (2.2) mit $c_k = y_k$ und $m_k = u_k$ ist

$$\begin{aligned} z_k &= a_{1,k+1}\, c_{k+1} + a_{0,k+1}\, c_k, \\ c_{k+2} &= z_k + m_k. \end{aligned} \tag{5.11}$$

Die ersten beiden Symbole des Geheimtextes sind dem Initialisierungsvektor entnommen. Verschiebt man den Geheimtext soweit nach links, bis das Geheimtextsymbol c_1^{s} das erste Symbol ist, das Informationen über den Klartext enthält, erhält man

$$\boldsymbol{c}^{\mathbf{s}} = (c_{-1}^{\mathrm{s}}, c_0^{\mathrm{s}}, c_1^{\mathrm{s}}, \ldots, c_{N_{\mathrm{m}}}^{\mathrm{s}}) \quad \text{statt} \quad \boldsymbol{c}^{\mathbf{s}} = (c_1^{\mathrm{s}}, \ldots, c_{N_{\mathrm{m}}+2}^{\mathrm{s}})$$

und (5.11) lässt sich anschaulicher durch

$$\begin{aligned} z_k &= f_{\mathrm{z}}^{\mathrm{K}}\,(c_{k-2}, c_{k-1}) = a_{1,k-1}\, c_{k-1} + a_{0,k-2}\, c_{k-2}, \\ c_k &= f_{\mathrm{c}}\,(z_k, m_k) = z_k + m_k \end{aligned} \tag{5.12}$$

darstellen.

Auf Seite des Senders enthält der aktuelle Wert des Geheimtextes die Information über den zwei Schritte zurückliegenden Wert des Klartextes verschlüsselt in Abhängigkeit der letzten zwei Werte des Geheimtextes. Umgekehrt kann auf Empfängerseite aus zwei aufeinander folgenden unmittelbar zurückliegenden Symbolen des Geheimtextes das aktuelle Symbol des Schlüsseltextes rekonstruiert und damit das entsprechende Symbol des Geheimtextes entschlüsselt werden.

Für eine konkrete Betrachtung nehmen wir an, dass die Nachricht $\boldsymbol{m}^{\mathbf{s}}$ eine Länge von $N = 60$ Symbolen hat und aus einer beliebigen Folge von Bytes besteht (Abbildung 5.3 oder 5.6). Das Alphabet des Eingangs ist somit

$$\mathcal{A} = \{0, \ldots, 255\}$$

und der Nachrichtenraum

$$\mathcal{M} = \mathcal{A}^{\mathrm{N}} = \{0, \ldots, 255\}^{60}.$$

Für den Wertebereich des Schlüsseltextes $\mathcal{Z}$ und den des Geheimtextes $\mathcal{C}$ wird der gleiche Wertebereich gewählt

$$\mathcal{Z} = \mathcal{C} = \{0, \ldots, 255\}^{60}.$$

Die Einhaltung des Wertebereichs wird durch eine Division modulo 256 nach jeder Rechenoperation gewährleistet. Abschnitt 2.4 enthält eine kurze Erläuterung zu den nötigen Annahmen und Einschränkungen.
Zur Bestimmung des ersten Symbols des Schlüsseltextes, z_1, müssen die $P = 2$ zurückliegenden Symbole y_0 und y_{-1} des Geheimtextes bekannt sein. Dies wird durch einen willkürlich gewählten Initialisierungsvektor erreicht, der die fehlenden Stellen auffüllt:

$$\boldsymbol{IV} = (50, 50)\,.$$

Im Folgendenden wird die Robustheit von zwei Verschlüsselungssystemen gegenüber dem Synchronisationsverlust zwischen Sender und Empfänger betrachtet. Da der Verlust eines Symbols eine ähnliche Herausforderung an das Synchronisationsverhalten stellt wie die doppelte Wertung eines Symbols, wird nur der letztere Fall untersucht, da er sich anschaulicher darstellen lässt. Dies entspricht dem in Abschnitt 2.3.1 vorgestellten Bitslip I. Die Reaktion des Verschlüsselungssystems auf eine Verfälschung eines Symbols (Bitfehler) während der Übertragung stimmt mit der aus Abschnitt 2.3.3 beobachteten überein.
Die beiden Verschlüsselungssysteme unterscheiden sich in der Umschaltfunktion, die die Kriterien für den Wechsel zwischen zwei linearen Modi, d.h. zwischen den möglichen Koeffizienten $a_{0,k}$ und $a_{1,k}$ festlegt. In Abschnitt 5.3.1 wird der Fall der zeitabhängigen Umschaltfunktion, in Abschnitt 5.3.2 der Fall der geheimtextabhängigen Umschaltfunktion behandelt.

5.3.1 Zeitabhängige Umschaltfunktion

Für das erste Verschlüsselungssystem wird die Umschaltfunktion $\sigma(k, x_k)$ als eine Funktion des Zeitpunktes k gewählt. Zu jedem Zeitpunkt k liefert $\sigma(k, x_k)$ einen Wert, anhand dessen die beiden varianten Koeffizienten $a_{0,k}$ und $a_{1,k}$ für den aktuellen Zeitpunkt festgelegt werden. Die Anzahl möglicher Werte bzw. Modi beträgt $J = 4$. Die Verwendung einer zeitabhängigen Umschaltfunktion entspricht dem Vorschlag in [1]. Es gilt

$$\sigma(k, x_k) = \sigma(k), \qquad \text{mit} \quad \sigma \in \{1, \ldots, J = 4\}.$$

Im gegebenen Fall wird $\sigma(k)$ so gewählt, dass die geordnete Menge der Koeffizienten zyklisch durchschritten wird. Alle 10 Zeitschritte erfolgt ein Wechsel. Nach den ersten

Umschaltvorgängen nehmen die Koeffizienten die Werte

$$a_{0,k} = \begin{cases} -10 & \text{wenn } k \leq 0 \\ 20 & \text{wenn } 1 \leq k \leq 10 \\ -30 & \text{wenn } 11 \leq k \leq 20 \\ 40 & \text{wenn } 21 \leq k \leq 30 \\ -10 & \text{wenn } 31 \leq k \leq 40 \\ \vdots & \end{cases} \quad \text{und} \quad a_{1,k} = \begin{cases} 130 & \text{wenn } k \leq 0 \\ 140 & \text{wenn } 1 \leq k \leq 10 \\ -150 & \text{wenn } 11 \leq k \leq 20 \\ -160 & \text{wenn } 21 \leq k \leq 30 \\ 130 & \text{wenn } 31 \leq k \leq 40 \\ \vdots & \end{cases}$$

an. Die Abgrenzung von Bereichen konstanter Koeffizienten sind in den folgenden Abbildungen gestrichelt eingezeichnet.
Aus Gründen der Anschaulichkeit ist die Umschaltfunktion in diesem Fall vorgegeben, d.h. Bestandteil des Algorithmus. In [1] wird begründet, warum dies aus Sicherheitsaspekten ungeeignet ist. Eine bessere Möglichkeit ist eine variable, schlüsselabhängie Wahl der Umschaltfunktion.

Der Geheimtext. Der Verlauf des Geheimtextes auf Empfängerseite ($\boldsymbol{c}^{\mathrm{e}}$) ergibt sich aus dem Geheimtext auf Seite des Senders ($\boldsymbol{c}^{\mathrm{s}}$) durch die doppelte Wertung der Symbole zum Zeitpunkt $k = 15$ und $k = 35$ des Geheimtextes $\boldsymbol{c}^{\mathrm{s}}$. Es gilt also

$$\boldsymbol{c}^{\mathrm{e}} = \left(c^{\mathrm{s}}_{-1,\ldots,15}, c^{\mathrm{s}}_{15}, c^{\mathrm{s}}_{16,\ldots,35}, c^{\mathrm{s}}_{35}, c^{\mathrm{s}}_{36,\ldots,60}\right).$$

Die falschen Geheimtextsymbole sind an den Stellen $i_1 = 16$ und $i_2 = 37$. Der Geheimtext $\boldsymbol{c}^{\mathrm{e}}$ ist damit um zwei Symbole länger als $\boldsymbol{c}^{\mathrm{s}}$. Ein Vergleich zwischen vor und nach der Übertragung ist in Abbildung 5.1 dargestellt.

Der Schlüsseltext. Innerhalb von Bereichen mit konstanten Koeffizienten sind die Erkenntnisse aus 2.3.1 auch hier gültig. Das erste Symbol von $\boldsymbol{c}^{\mathrm{e}}$, das von $\boldsymbol{c}^{\mathrm{s}}$ abweicht, ist c^{e}_{16}. Nach (2.5) stimmt der Schlüsseltext $\boldsymbol{c}^{\mathrm{e}}$ daher bis einschließlich Symbol z^{e}_{16} mit $\boldsymbol{z}^{\mathrm{s}}$ überein,

$$z^{\mathrm{e}}_k = z^{\mathrm{s}}_k, \qquad \text{mit} \quad 1 \leq k < 17.$$

Das Symbol z^{e}_{17} wird falsch rekonstruiert,

$$z^{\mathrm{e}}_k \neq z^{\mathrm{s}}_k, \qquad \text{mit} \quad k = 17.$$

Nach (2.4) entsprechen die darauf folgenden Symbole zunächst wieder dem um einen Zeitschritt nach rechts verschobenen Schlüsseltext $\boldsymbol{z}^{\mathrm{s}}$,

$$z^{\mathrm{e}}_k = z^{\mathrm{s}}_{k-\Delta}, \qquad \text{mit} \quad 17 < k < 22,\ \Delta = 1.$$

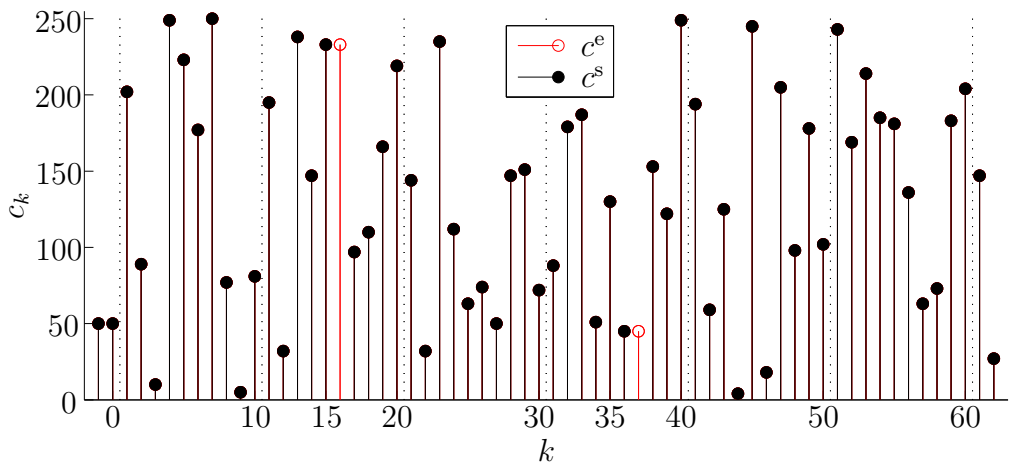


Abbildung 5.1: Zeitabhängige Umschaltfunktion, Geheimtext

Beim Übergang von $k = 20$ nach $k = 21$ bedingt die Umschaltfunktion eine Variation der Koeffizienten, d.h. $a_{20} \neq a_{21}$. An der Stelle $k = 22$ und $k = 23$ tritt erneut eine Abweichung auf, denn an der Stelle $k = 22$ gilt

$$\begin{aligned} z^{\mathrm{e}}_{22} &= a_{21}\, c^{\mathrm{e}}_{21} + a_{20}\, c^{\mathrm{e}}_{20} \\ &= a_{21}\, c^{\mathrm{s}}_{20} + a_{19}\, c^{\mathrm{s}}_{19} \neq z^{\mathrm{s}}_{22-\Delta} = a_{21-\Delta}\, c^{\mathrm{s}}_{21-\Delta} + a_{20-\Delta}\, c^{\mathrm{s}}_{20-\Delta} = a_{20}\, c^{\mathrm{s}}_{20} + a_{19}\, c^{\mathrm{s}}_{19}. \end{aligned}$$

und Entsprechendes an der Telle $k = 23$. Einen Zeitschritt später, $k = 24$, sind die Koeffizienten der Geheimtextsymbole $a_{0,k-1}$ und $a_{0,k-1-\Delta}$ bzw. $a_{1,k-1}$ und $a_{1,k-1-\Delta}$ wieder identisch und die Symbole der Schlüsseltexte ebenfalls:

$$z^{\mathrm{e}}_k = z^{\mathrm{s}}_{k-\Delta}, \qquad \text{mit} \quad 24 < k < 32$$

Aber: Ein erneuter Wechsel der Koeffizienten führt einen Zeitschritt später auch zu einem erneuten Auftreten des Problems, da die Schlüsselgeneratoren auf Sender- und Empfängerseite weiter mit einem Taktversatz arbeiten.
Der zweite Bitslip-Fehler an der Stelle $k = 37$ bedingt nach Abschluss der Synchronisation zwei falsche Symbole im Schlüsseltext im Anschluss an jede Variation der Koeffizienten, da es zwei Zeitschritte benötigt, bis sich die Koeffizienten der Schlüsselgeneratoren auf beiden Seiten angeglichen haben. Jeder Bitslip-Fehler verschlechtert kummulativ die Rekonstruktion des Schlüsseltextes (Abbildung 5.2).

Der Klartext. Der Klartext ist die gewichteten Überlagerung von Geheim- und Schlüsseltext. Die Bestimmungsgleichung ergibt sich durch Umformung der Beziehung (5.12) zu

$$m^{\mathrm{e}}_k = c^{\mathrm{e}}_k - z^{\mathrm{e}}_k. \tag{5.13}$$

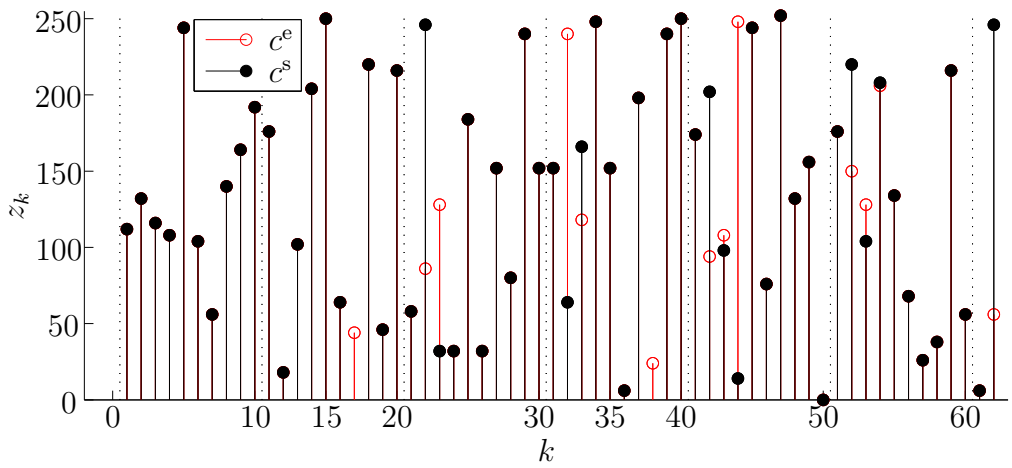


Abbildung 5.2: Zeitabhängige Umschaltfunktion, Schlüsseltext

Wie im Fall der zeitinvarianten Koeffizienten führt jedes falsche Symbol in Geheim- oder Schlüsseltext zu einem Verlust des zugeordneten Symbols im Klartext. Den Verlauf zeigt Abbildung 5.3. Unter der Annahme, dass in einem realen Übertragungssystem keine unterschiedlichen Bitslip-Fehler auftreten können, sich das Problem der Synchronisation also nicht durch Überlagerung unterschiedlicher Fehler von selbst lösen kann, bietet das vorgestellte Message-embedded System mit einer zeitabhängigen Umschaltfunktion nicht die gewünschten Vorteile einer SSSC, aber deren Nachteile.

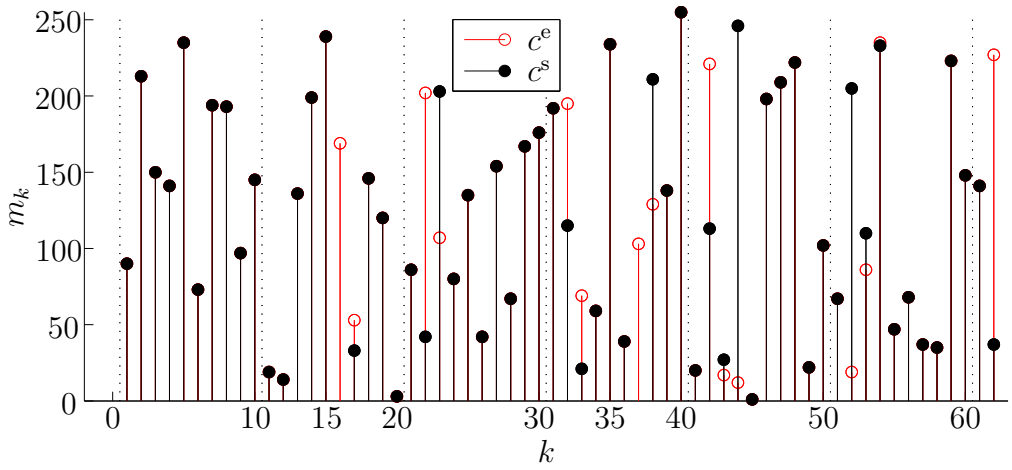


Abbildung 5.3: Zeitabhängige Umschaltfunktion, Klartext

5.3.2 Geheimtextabhängige Umschaltfunktion

Im zweiten Fall wird die Umschaltfunktion σ als eine Funktion des Geheimtextes c_k gewählt. Zu jedem Zeitpunkt k wird der Wert der Umschaltfunktion und damit die varianten Koeffizienten $a_{0,k}$ und $a_{1,k}$ anhand des Wertes des aktuellen Geheimtextsymbols ausgewählt. Der Wertemenge der Umschaltfunktion enthält $J = 4$ Elemente. Für die Umschaltfunktion gilt

$$\sigma = \sigma(c_k), \qquad \text{mit} \quad \sigma \in \{1, \ldots, J = 4\}.$$

Im gegebenen Fall wird der Wertebereich des Geheimtextes in vier Bereiche unterteilt, die den vier möglichen Werten von $\sigma(c_k)$ entsprechend den Beziehungen

$$a_{0,k} = \begin{cases} -10 & \text{wenn} \quad \bmod 4 = 0 \\ 20 & \text{wenn} \quad \bmod 4 = 1 \\ -30 & \text{wenn} \quad \bmod 4 = 2 \\ 40 & \text{wenn} \quad \bmod 4 = 3 \end{cases} \quad \text{und} \quad a_{1,k} = \begin{cases} 130 & \text{wenn} \quad \bmod 4 = 0 \\ 140 & \text{wenn} \quad \bmod 4 = 1 \\ -150 & \text{wenn} \quad \bmod 4 = 2 \\ -160 & \text{wenn} \quad \bmod 4 = 3 \end{cases}$$

über eine Operation Modulo 4 zugeordnet sind.

Der Geheimtext. Die Verfälschung des Geheimtextes stimmt mit der in Abschnitt 5.3.1 überein. Aufgrund der unterschiedlichen Umschaltfunktionen sind die Werte jedoch nicht zwingend identisch. Der Verlauf ist in Abbildung 5.4 zu sehen.

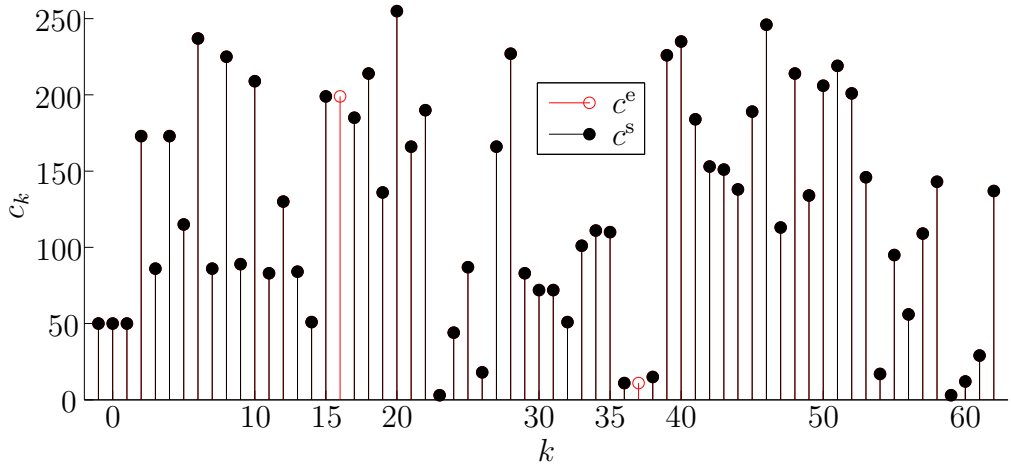


Abbildung 5.4: Geheimtextabhängige Umschaltfunktion, Geheimtext

Der Schlüsseltext. Der erste kritische Symbol im Schlüsseltext ist an der Stelle $k = 22$. Zu diesem Zeitpunkt ergibt sich das Schlüsseltextsymbol aus

$$\begin{aligned} z_{22}^{\mathrm{e}} &= a_{\sigma(c_{21}^{\mathrm{e}})}\, c_{21}^{\mathrm{e}} + a_{\sigma(c_{20}^{\mathrm{e}})}\, c_{20}^{\mathrm{s}} \\ &= a_{\sigma(c_{21}^{\mathrm{e}})}\, c_{20}^{\mathrm{s}} + a_{\sigma(c_{20}^{\mathrm{e}})}\, c_{19}^{\mathrm{s}} = a_{\sigma(c_{20}^{\mathrm{s}})}\, c_{20}^{\mathrm{s}} + a_{\sigma(c_{19}^{\mathrm{s}})}\, c_{19}^{\mathrm{s}} = z_{21}^{\mathrm{s}}. \end{aligned}$$

Das Schlüsseltextsymbol wird korrekt rekonstruiert. Einen Zeitschritt später kommt es zum selben Verhalten. Es lässt sich erkennen, dass der zeitverschobene Koeffizientenwechsel im Fall der geheimtextabhängigen Umschaltfunktion nicht zu einem Taktversatz der beiden Schlüsselgeneratoren führt. Denn diese werden durch die Geheimtextsymbole unabhängig der exakten Position im Geheimtext synchronisiert.
Das Verhalten des Schlüsseltextes im Fall eines Bitslip-Fehlers entspricht den allgemeinen Betrachtungen aus Abschnitt 2.3.1. Die Zeitvarianz hat daher keinen Einfluss auf die Robustheit gegenüber Synchronisationsverlust. Den Verlauf zeigt Abbildung 5.5.

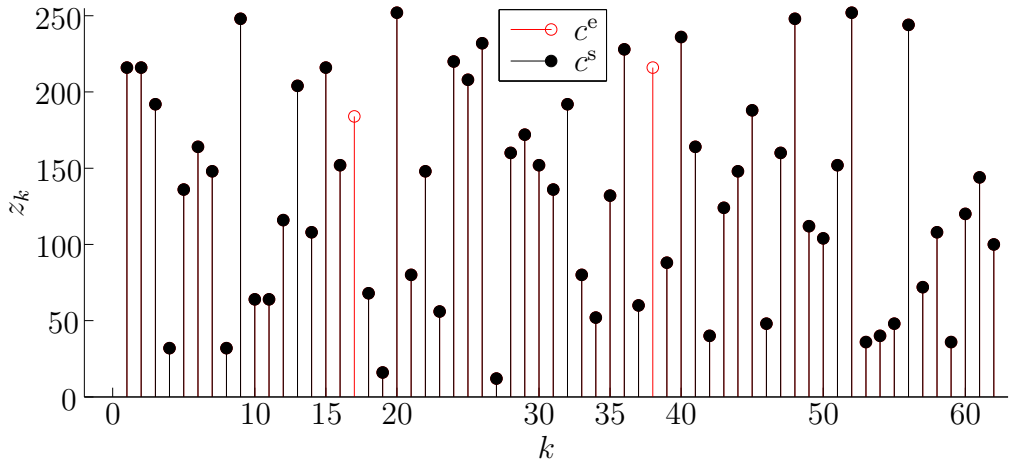


Abbildung 5.5: Geheimtextabhängige Umschaltfunktion, Schlüsseltext

Der Klartext. Der Verlauf des Klartextes kann vollständig aus den Betrachtungen aus Abschnitt 2.3.1 abgeleitet werden: Jeder Bitslip-Fehler resultiert in $P = 2$ falschen Klartextsymbolen, wobei ein Klartextsymbol redundant ist, da es aus der Entschlüsselung eines doppelt gewerteten Geheimtextsymbols stammt. Somit bleibt es beim Verlust von nur einem Symbol der ursprünglichen Nachricht. Der Verlauf des Klartextes wird in Abbildung 5.6 gezeigt.
Eine geheimtextabhängige Wahl der Umschaltfunktion verhindert offensichtlich das Auftreten eines Taktversatzes der beiden Schlüsselgeneratoren nach einem Bitslip-Fehler und das damit verbundene Auftreten von Entschlüsselungsfehlern nach jedem Koeffizientenwechsel. Mit einer geheimtextabhängigen Umschaltfunktion lässt sich ein stück-

weise lineares diskretes SISO-System in Form eines Message-embedde System zur SSSC verwenden.

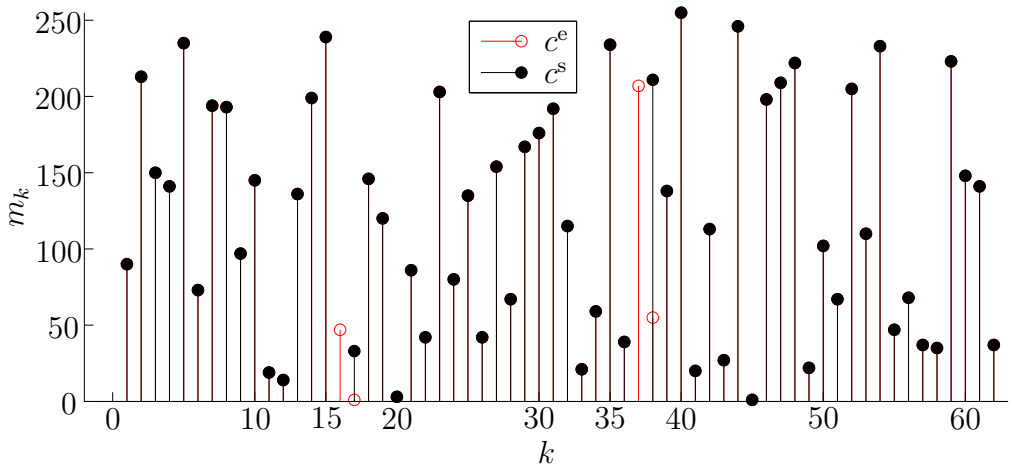


Abbildung 5.6: Geheimtextabhängige Umschaltfunktion, Klartext

Kapitel 6

Kryptoanalyse

Die Eigenschaft, die über die Verwendbarkeit eines Message-embedded Systems und allgemein jedes Algorithmus zur Verschlüsselung von Nachrichten entscheidet, ist die Widerstandskraft gegen Entschlüsselungsversuche des Geheimtextes durch Dritte. Die Untersuchung eines Verschlüsselungssystems nach Schwachstellen gegenüber Angreifern ist die Kryptoanalyse. Die Bewertung der Sicherheit erfolgt unter der Annahme, dass dem Angreifer sämtliche Information über das Verschlüsselungssystem zur Verfügung stehen; die Sicherheit muss allein vom geheimen Schlüssel abhängen. Diese Annahme wird als KERKHOFF-Prinzip bezeichnet.
Wir betrachten dabei zwei Fälle: Die einfachsten Vorgehensweisen für den Angreifer ist das Ausprobieren sämtlicher Schlüssel innerhalb des Schlüsselraumes. Diese Methode wird als Brute-Force-Angriff bezeichnet und führt im gegebenen Fall auf die Frage nach der geeigneten Parametrisierung des Systems bzw. dem Zusammenhang von Systemparametern und dem geheimen Schlüssel.
Im zweiten Fall wird davon ausgegangen, dass der Angreifer Zugriff auf das Verschlüsselungssystem hat. Es ist ihm möglich einen beliebigen Klartext vorzugeben und den resultieren Geheimtext zu erhalten. Aus systemtheoretischer Sicht entspricht dies einer Parameteridenifikation. Dieses Vorgehen wird nach den Rahmenbedingungen als Chosen-Plaintext-Angriff bezeichnet. Angriffe, die sich die Eigenheiten des Verschlüsselungsalgorithmus zunutze machen, um aus Bruchstücken des Klar- und Geheimtextes Informationen über den geheimen Schlüssel zu gewinnen, werden allgemein als Algebraischer Angriff bezeichnet. Die Chosen-Plaintext Attack stellt mit die größte Herausforderung an ein Verschlüsselungssystem dar.
In [1] und [17] können Verfahren zur Bewertung und Ermittlung der Sicherheit gefunden werden. Im Folgenden werden die zugrunde liegenden Ideen vorgestellt und die Ausgangssituation für die Kryptoanalyse erläutert.

6.1 Brute-Force-Angriff

Beim Brute-Force-Angriff wird der Algorithmus des Verschlüsselungssystems nicht betrachtet. Die Sicherheit ist in erster Linie eine Frage des Schlüsselraumes $\mathcal{K}$, d.h. der Schlüssellänge N_{K} und des Wertebereichs bzw. des Alphabets eines Schlüsselsymbols.

Zunächst wird nach einem beliebigen Verfahren ein Schlüssel $\boldsymbol{K}$ aus dem Schlüsselraum $\mathcal{K}$ ausgewählt. Mit diesem Schlüssel wird versucht den Geheimtext zu entschlüsseln. Gelingt dies nicht, wird der nächste Schlüssel ausgewählt, solange, bis die Entschlüsselung gelingt oder sämtliche mögliche Schlüssel ausprobiert wurden.
Wie erwähnt, ist der Brute-Force-Angriff unabhängig vom Verschlüsselungssystem anwendbar und die Widerstandskraft gegen einen Brute-Force-Angriff von der Beschaffenheit des geheimen Schlüssels abhängig. Wird ein Message-embedded System zur Verschlüsselung verwendet, kann durch eine ungünstig gewählter Schlüssel bzw. durch eine ungünstige Zuordnung von Schlüssel und Systemparamtern der Angriff erleichtert werden. Nämlich dann, wenn das System trotz der Paramtrierung mit unterschiedlichen Schlüsseln das gleiche Eingangs-Ausgangs-Verhalten zeigt. Umgekehrt bedeutet dies, dass sich die Symbole $K_1, \ldots, K_{N_\mathrm{K}}$ des Schlüssels $\boldsymbol{K}$ eindeutig als eine Funktion der Eingangs- und Ausgangsfolge

$$K_i = f_i\left(y_k, y_{k+1}, \ldots, u_k, u_{k+1}, \ldots\right), \qquad \text{mit} \quad i = 1, \ldots, N_\mathrm{K}$$

schreiben lassen müssen, um die maximale Sicherheit für einen gegebenen Schlüsselraum zu gewährleisten. Ist dies der Fall, werden die Schlüsselsymbole bzw. die Systemparameter als identifizierbar bezeichnet. Verfahren für die Untersuchung eines allgemeinen SISO-Systems auf Identifizierbarkeit werden in [17] vorgestellt, in [1] können Aussagen speziell zu den hier vorgestellten Systemklassen gefunden werden.

6.2 Chosen-Plaintext-Angriff

Der Chosen-Plaintext-Angriff ist mit der stärkste Angriff mit dem gegen ein Verschlüsselungssystem vorgegangen werden kann. Bei diesem Angriff ist es dem Angreifer möglich in gleicher Weise wie der Anwender auf das Verschlüsselungssystem zuzugreifen, d.h. der Angreifer kann dem Verschlüsselungssystem einen beliebig gewählten Klartext vorgeben und erhält den zugehöreigen Geheimtext. Damit stehen im sämtliche Größen mit der Ausnahme des geheimen Schlüssels zur Verfügung.
Wir betrachten zunächst die Bestimmungsgleichung für ein Geheimtextsymbol zum Zeitpunkt k gemäß (5.6) aus Sicht des Anwenders:

$$c_k^\mathrm{s} = \sum_{n=1}^{P} a_n^\epsilon \, c_{k-n} + b^\epsilon \, m_k.$$

Dem Anwender ist dabei der geheime Schlüssel $\boldsymbol{K}$ bekannt, damit auch der Wert der Umschaltfunktion σ zu jedem Zeitpunkt und die Umschaltsequenz ϵ. Für jedes Geheimtextsymbol existiert eine Bestimmungsgleichung.
Es wird davon ausgegangen, dass die Umschaltfunktion nicht Teil des Algorithmus ist, sondern in Abhängigkeit des geheimen Schlüssels generiert wird. Ohne Kenntnis des geheimen Schlüssels kann der Angreifer die genaue Umschaltsequenz ϵ nicht bestimmen,

sondern nur die Anzahl möglicher Umschaltsequenzen $\epsilon_1, \ldots, \epsilon_{N_\epsilon}$. Diese Anzahl ist endlich, da sowohl die Umschaltfunktion eine endliche Wertemenge aus J Elementen besitzt als auch das Zeitintervall $[k-P, k-1]$ eine endliche Menge von diskreten Zeitpunkten enthält. Damit ist die Grundlage des Angreifers die Kenntnis der N_σ möglichen Bestimmungsgleichungen für ein Geheimtextsymbol zum Zeitpunkt k:

$$\begin{aligned} c_k^{\mathrm{s}} &= \sum_{n=1}^{P} a_n^{\epsilon_1}\, c_{k-n} + b^{\epsilon_1}\, m_k \\ &\vdots \\ c_k^{\mathrm{s}} &= \sum_{n=1}^{P} a_n^{\epsilon_{N_\epsilon}}\, c_{k-n} + b^{\epsilon_{N_\epsilon}}\, m_k. \end{aligned}$$

Das Ziel des Angriffs ist die Identifikation der Koeffizienten und aus den Koeffizienten die Bestimmung des geheimen Schlüssels. Ist dem Angreifer der geheime Schlüssel bekannt, kann er offensichtlich das Verschlüsselungssystem nachbilden und somit zu jedem beliebigen Geheimtext den zugehörgen Klartext bestimmen.
Ein Vorschlag für die Behandlung dieses Identifikationsproblems kann in [1] und [18] gefunden werden. Das Ergebnis führt zu einer Komplexität des Angriffs, die für große M_{N_ϵ} durch

$$\mathcal{O}(M_{N_\epsilon}^3)$$

angenähert werden kann. Wobei sich M_{N_ϵ} aus

$$M_{N_\epsilon}^3 = \binom{N_\epsilon + P - 1}{N} = \frac{(N_\epsilon + P - 1)!}{N_\epsilon!(P-1)!}$$

ergibt. D.h. bei einer Zunahme der Paramteranzahl, wächst M_{N_ϵ} schneller als die Anzahl der Parameter, was die Voraussetzung eines guten Verschlüsselungsalgorithmus ist.

Kapitel 7

Zusammenfassung und Ausblick

In dieser Arbeit werden drei Themengebiete berührt: die Kryptographie, die Modultheorie und die Systemtheorie. Die Kryptographie stellt mit der selbstsynchronisierenden Stromverschlüsselung die Anwendung. Die Realisierung dieses Verschlüsselungssystems kann auf unterschiedliche Art und Weise geschehen. Von Interesse sind die Größen Klartext, Schlüsseltext und Geheimtext. Ihre Beziehungen lassen sich aber stets in einer besonderen Form darstellen, der sogennanten kanonischen Darstellung. Es wird der Vorschlag einer Realisierung durch ein lineares diskretes SISO-System untersucht. Am Eingang des Systems liegt die unverschlüsselte Nachricht an, am Ausgang die verschlüsselte. Die inneren Zustände des Systems lassen sich als Schlüsseltext interpretieren. Wird ein System als Verschlüsselungssystem eingesetzt, bezeichnet man es als Messege-embedded System. Unbeachtet der Umschaltbedingung für den Wechsel zwischen zwei linearen Modi, lässt sich ein solches System als Speziallfall eines linearen zeitvarianten diskreten SISO-Systems darstellen.
Aus systemtheoretischer Sicht existiert eine kanonische Darstellung für ein lineares zeitvariantes SISO-System dann, wenn es invertierbar ist und der Ausgang des Systems ein flacher Ausgang ist. Der Nachweis dieser Eigenschaften kann auf unterschiedlichen Wegen erbracht werden. In dieser Arbeit wird das System als Modul über einem nichtkommutativen Polynomring, dem Systemmodul, aufgefasst. Die Elemente des Systemmoduls entsprechen den Systemgrößen und ihre Beziehungen untereinander repräsentieren die Differentialgleichungen. Wird beachtet, dass im Polynomring die Kommutativität nicht gilt, kann mit den Modulelementen gerechnet und aus den algebraischen Eigenschaften des Systemmoduls auf systemtheoretische Eigenschaften des entsprechenden Systems geschlossen werden.
Es wird gezeigt, dass ein SISO-System immer invertierbar ist, d.h. es existiert immer eine eindeutige Beziehung zwischen Eingang und Ausgang, da beide aus jeweils einer unabhängigen Größe erzeugt werden.
Für den Nachweis der Flachheit des Systemausgangs wird die Basis des Systemmoduls bestimmt. Ist der Systemausgang die Basis, kann jedes beliebige Element des Systemmoduls und damit jede beliebige Größe im System durch den Ausgang und Iterationen des Ausgangs dargestellt werden. Ist dies der Fall, wird der Systemausgang als flacher Ausgang bezeichnet. Dieses Vorgehen wird an einem einfachen Beispiel demonstriert.

Das Verhalten der selbstsynchronisierenden Stromverschlüsselung bei Auftreten eines Bitslip und eines Bitfehlers wird allgemein untersucht und vergleichen. Anhand des Beispielsystems, für das die Invertierbarkeit und die Flachheit des Ausgangs nachgewiesen wurde, wird ebenfalls das Fehlerverhalten untersucht. Dabei wird zwischen zwei unterschiedlichen Umschaltfunktionen, zeitabhängig und geheimtextabhängig, verglichen. Es wird gezeigt, dass ein stückweise lineares diskretes SISO-System mit zeitabhängiger Umschaltfunktion, obwohl es die strukturellen Voraussetzungen erfüllt, zur Verwendung als selbstsynchronisierende Stromverschlüsselung nicht geeignet ist. Wird eine geheimtextabhängige Umschaltfunktion angenommen, ist das resultierende Verschlüsselungssystem dagegen robust gegenüber Synchronisationsverlust.
Die Frage der Sicherheit des Verschlüsselungssystems gegenüber einem Brute-Force-Angriff und einem Chosen-Plaintext-Angriff wird erläutert, auf quantitative Angaben wird jedoch verzichtet. Natürlich ist die Sicherheit eines Verschlüsselungssystem gegenüber Angriffen das ausschlaggebende Kriterium bei der Bewertung der tatsächlichen Verwendbarkeit. Daher ist eine saubere Darstellung der Sicherheit eines stückweise lineares diskretes SISO-System mit geheimtextabhängiger Umschaltfunktion eine interessante Frage, die es noch zu klären gilt.
Desweiteren wird die Anwendung der Modultheorie nur an einem primitiven Beispiel gezeigt, dass die geforderten Eigenschaften erfüllt. Es fehlt die Behandlung eines komplexeren Systems mit nicht steuerbarem und nicht beobachtbarem Teilsystem, anhand dessen die Extraktion des steuer- und beobachtbaren Eingangs-Ausgangs-Teilsystems demonstriert werden kann. Diese Arbeit stellt nur einen Einstieg in diese Betrachtungsweise dar und bietet was die Anwendbarkeit und Abgeschlossenheit betrifft noch viel Raum zur Erweiterung.
Bevor ein System auf die strukturellen Voraussetzungen für die Verwendung zur selbstsynchronisierenden Stromverschlüsselung untersucht werden kann, muss es zunächst gefunden werden. Es stellt sich die Frage, wie Kandidaten für die vorgestellte Testumgebung gefunden werden können.
Besteht in aktuellen Anwendungen Bedarf nach einer selbstsynchronisierenden Stromverschlüsselung wird eine adaptierte Blockverschlüsselung eingesetzt. Diese bietet nachweislich Sicherheit gegen Angreifer mit dem Nachteil einer ineffizienten Implementierung. Ein Vergleich zwischen einer selbstsynchronisierenden Stromverschlüsselung realisiert mit einer modifizierten Blockverschlüsselung und einer Realisierung durch ein Message-embedded System hinsichtlich Aufwand und Rechenbedarf der Implementierung dürfte ebenfalls eine Interessante Erweiterung darstellen.

Anhang A

Quellcode zu den Beispielen

Die Simulation einer verschlüsselten Übertragung zusammen mit den entsprechenden Abbildungen in den Abschnitt 2.4, 5.3.1 und 5.3.2wurden mit Hilfe von MATLAB durchgeführt und erstellt. Im Folgenden werden die relevanten Ausschnitte der zugrunde liegenden Quellcodes gezeigt.

Invariantes System

```
% Generation of a random massage (pleintext)
nM = 10;
ms = round(256*rand(1,nM));

% Arbitrarily chosen initialisation vector (IV)
iv = [50,50];

% Initialisation
zs = zeros(1,nM);
cs = zeros(1,nM+2);
cs(1:2) = iv;
ze1 = zeros(1,nM+1); ze2 = zeros(1,nM-1);
 ze3 = zeros(1,nM);
me1 = zeros(1,nM+1); me2 = zeros(1,nM-1);
 me3 = zeros(1,nM);


% ---------------------------------------------------------
% Transmitter
% ---------------------------------------------------------

for k = 1:nM
    % Key generator
    zs(k) = 170*cs(k+1) + cs(k); zs(k) = mod(zs(k),256);
```

```
    % Encription function
    cs(k+2) = zs(k) + ms(k); cs(k+2) = mod(cs(k+2),256);
end;


% ---------------------------------------------------------
% Link
% ---------------------------------------------------------

% Error type 1: Byte 4 is counted twice
ce1 = [cs(1:6),cs(6),cs(7:end)];

% Error type 2: Byte 5 is lost
ce2 = [cs(1:6),cs(8:end)];

% Error type 3: Byte 5 is distorted
ce3 = cs;
ce3(7) = mod(ce3(7) + 10,256);


% ---------------------------------------------------------
% Receiver
% ---------------------------------------------------------

% Error type 1
for k = 1:nM+1
    % Key generator
    ze1(k) = 170*ce1(k+1) + ce1(k);
    ze1(k) = mod(ze1(k),256);
    % Decription function
    me1(k) = ce1(k+2) - ze1(k);
    me1(k) = mod(me1(k),256);
end;

% Error type 2
for k = 1:nM-1
    ze2(k) = 170*ce2(k+1) + ce2(k);
    ze2(k) = mod(ze2(k),256);
    me2(k) = ce2(k+2) - ze2(k);
    me2(k) = mod(me2(k),256);
end;
```

```
% Error type 3
for k = 1:nM
    ze3(k) = 170*ce3(k+1) + ce3(k);
    ze3(k) = mod(ze3(k),256);
    me3(k) = ce3(k+2) - ze3(k);
    me3(k) = mod(me3(k),256);
end;
```

Zeitabhängige Umschaltfunktion

```
% Generation of a random massage (pleintext)
nM = 60;
ms = round(256*rand(1,nM));

% Arbitrarily chosen initialisation vector (IV)
iv = [50,50];

% Initialisation
zs = zeros(1,nM);
cs = zeros(1,nM+2);
cs(1:2) = iv;
ze = zeros(1,nM+2);
me = zeros(1,nM+2);

% Set of values for the time-varying coefficients
a0 = [-10,20,-30,40];
a1 = [130,140,-150,-160,130];

% ------------------------------------------------------------
% Transmitter
% ------------------------------------------------------------

for k = 1:nM
    % Key generator
    zs(k) = a1( sigmaK(k+1-2) )*cs(k+1) + ...
        a0( sigmaK(k-2) )*cs(k);
    zs(k) = mod(zs(k),256);
    % Encription function
    cs(k+2) = zs(k) + ms(k);
    cs(k+2) = mod(cs(k+2),256);
end;
```

```
% ------------------------------------------------------
% Link
% ------------------------------------------------------

% Byte 15(index 17) and byte 35(index 37) are counted twice
ce = [cs(1:17),cs(17:37),cs(37:end)];


% ------------------------------------------------------
% Receiver
% ------------------------------------------------------

for k = 1:nM+2
    % Key generator (keytext)
    ze(k) = a1( sigmaK(k+1-2) )*ce(k+1) + ...
        a0( sigmaK(k-2) )*ce(k);
    ze(k) = mod(ze(k),256);
    % Decription function (ciphertext)
    me(k) = ce(k+2) - ze(k);
    me(k) = mod(me(k),256);
end;
```

```
% ------------------------------------------------------
% Switching function -- depending on k
% ------------------------------------------------------

function j = sigmaK(k)

% The time-index is related to the plaintext and keytext
if k<1
    j = 1;
elseif k>=1 && k<=10
    j = 2;
elseif k>=11 && k<=20
    j = 3;
elseif k>=21 && k<=30
    j = 4;
elseif k>=31 && k<=40
```

```
    j = 1;
elseif k>=41 && k<=50
    j = 2;
elseif k>=51 && k<=60
    j = 3;
else
    j = 4;
end;
```

Geheimtextabhängige Umschaltfunktion

```
% Generation of a random massage (pleintext)
nM = 60;
ms = round(256*rand(1,nM));

% Arbitrarily chosen initialisation vector (IV)
iv = [50,50];

% Initialisation
zs = zeros(1,nM);
cs = zeros(1,nM+2);
cs(1:2) = iv;
ze = zeros(1,nM+2);
me = zeros(1,nM+2);

% Set of values for the time-varying coefficients
a0 = [-10,20,-30,40];
a1 = [130,140,-150,-160,130];


% ------------------------------------------------------------
% Transmitter
% ------------------------------------------------------------

for k = 1:nM
    % Key generator (keytext)
    zs(k) = a1( sigmaC(cs(k+1)) )*cs(k+1) + ...
        a0( sigmaC(cs(k)) )*cs(k);
    zs(k) = mod(zs(k),256);
    % Encription function (ciphertext)
    cs(k+2) = zs(k) + ms(k);
```

```
    cs(k+2) = mod(cs(k+2),256);
end;


% ------------------------------------------------------
% Link
% ------------------------------------------------------

% Byte 15(index 17) and byte 35(index 37) are counted twice
ce = [cs(1:17),cs(17:37),cs(37:end)];


% ------------------------------------------------------
% Receiver
% ------------------------------------------------------

for k = 1:nM+2
    % Key generator
    ze(k) = a1( sigmaC(ce(k+1)) )*ce(k+1) + ...
        a0( sigmaC(ce(k)) )*ce(k);
    ze(k) = mod(ze(k),256);
    % Decription function
    me(k) = ce(k+2) - ze(k);
    me(k) = mod(me(k),256);
end;
```

```
% ------------------------------------------------------
% Switching function -- depending on c
% ------------------------------------------------------

function j = sigmaC(c)

j = mod(c,4)+1;
```

Literaturverzeichnis

[1] P. V. Tan, G. Millerioux, and J. Daafouz, "Invertibility, flatness and identifiability of switched linear dynamical systems: An application to secure communications," in *Proc. 47th IEEE Conference on Decision and Control CDC 2008*, 9–11 Dec. 2008, pp. 959–964.

[2] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*. http://www.cacr.math.uwaterloo.ca/hac/: CRC, Aug. 2001, vol. 5.

[3] H. Delfs and H. Knebl, *Introduction to Cryptography*, U. Maurer, Ed. Springer, 2007, vol. 2.

[4] J. Daemen and P. Kitsos, "The self-synchronizing stream cipher mosquito," STMicroelectronics Belgium; Hellenic Open University, Tech. Rep., Dec 2005.

[5] ——, "The self-synchronizing stream cipher moustique," in *New Stream Cipher Designs*, ser. Lecture Notes in Computer Science. Springer, 2008, vol. 4986/2008, pp. 210–223.

[6] U. M. Maurer, "New approaches to the design of self-synchronizing stream ciphers," in *Advances in Cryptology – EUROCRYPT '91*, ser. Lecture Notes in Computer Science. Springer, 1991, vol. 547/1991, pp. 458–471.

[7] S. Bosch, *Algebra*. Springer, 2005, vol. 6.

[8] J. C. Jantzen and J. Schwermer, *Algebra*. Springer, 2005, vol. 1.

[9] A. Hölzle, "Einführung in die Modultheorie," Juli 2007. [Online]. Available: http://www.mathematik-netz.de/

[10] M. Fliess, "Some basic structural properties of generalized linear systems," in *Systems & control letters*. Elsevier Science, 1990, vol. 15, no. 5, pp. 391–396.

[11] ——, "Reversible linear and nonlinear discrete-time dynamics," *IEEE Trans. Autom. Control*, vol. 37, no. 8, pp. 1144–1153, Aug. 1992.

[12] ——, "Une interprétation algébrique de la transformation de laplace et des matrices de transfert," in *Linear algebra and its applications*. Elsevier Science, 1994, vol. 203-04, pp. 429–442.

[13] M. Fliess, C. Join, and H. Sira-Ramirez, "Robust residual generation for linear fault diagnosis: an algebraic setting with examples," in *International Journal of Control*. Taylor & Francis, 2004, vol. 77, no. 14, pp. 1223–1242.

[14] J. Rudolph, "Lineare systeme: Ein modultheoretischer zugang," Institut für Regelungs- und Steuerungstheorie, TU Dresden, Tech. Rep., Nov 2008.

[15] H. Bourlès, *Systèmes linéaires*. Lavoisier, 2006, vol. 1.

[16] ——, "Structural properties of discrete and continuous linear time-varying systems: A unified approach," in *Lecture notes in control and information sciences. Advanced topics in control systems theory. Conference, Paris , FRANCE (2004)*, vol. 311. Springer, 2005, pp. 225–280.

[17] F. Anstett, G. Millerioux, and G. Bloch, "Message-embedded cryptosystems: Cryptanalysis and identifiability," in *Proc. and 2005 European Control Conference Decision and Control CDC-ECC '05. 44th IEEE Conference on*, 2005, pp. 2548–2553.

[18] R. Vidal, S. Soatto, Y. Ma, and S. Sastry, "An algebraic geometric approach to the identification of a class of linear hybrid systems," in *Proc. 42nd IEEE Conference on Decision and Control*, vol. 1, 2003, pp. 167–172.