Radio Frequency Identification with Compressed Sensing

Martin Mayer

Geboren am 19.09.1986
Matrikelnummer: 0725729

Fakultät für Elektrotechnik und Informationstechnik
Technische Universität Wien

Ausgeführt zum Zwecke der Erlangung des akademischen Grades eines *Doktors der technischen Wissenschaften*.

August 2016

Advisor

Univ.-Prof. Dr.-Ing. Norbert Görtz
Institute of Telecommunications
Technische Universität Wien
Österreich

Examiner

Prof. Ph.D. Michael E. Davies
Institute of Digital Communications
The University of Edinburgh
Scotland, United Kingdom

Abstract

Radio Frequency IDentification (RFID) is a ubiquitous technology to wirelessly identify transponders (tags) with a reader device. The contemporary Internet of Things (IoT) paradigm has spawned a tremendous amount of new applications in the realm of RFID. This thesis improves RFID for the IoT and other applications where a *multitude of tags* is identified by a single reader device.

Compressed Sensing (CS) is a signal processing technique to acquire and recover *sparse* signal vectors from under-sampling by solving *under-determined* linear systems of equations. Conventional RFID tag acquisition schemes like Frame Slotted ALOHA (FSA) are hampered by colliding tag responses, while the proposed CS-RFID approach *exploits collisions* during tag acquisition. Triggered by the reader, all tags *respond simultaneously* with their signature sequence. The superposition of the signature sequences at the reader is cast as a CS measurement, i.e., an under-determined linear system of equations, and the *tag acquisition is formulated as a CS recovery problem*. Versatile Approximate Message Passing (AMP) recovery algorithms are vetted and employed to solve the problem efficiently.

The information from the tag acquisition is utilized by identification protocols to reliably identify the tags. Two identification protocols are proposed: A very quick one for fixed inventories (set of objects that feature a tag for identification), and a more general one for arbitrary inventories.

The AMP recovery algorithms leverage a variable amount of prior knowledge. A novel algorithm that exploits *joint sparsity* and the *signal distribution* is introduced. The origins of joint sparsity in CS-RFID are highlighted — in particular, a reader with *multiple receive antennas* profits from this case.

A flexible *measurement setup* is proposed. It allows to control clock and data of several UHF RFID tags. The practical feasibility of CS-RFID is demonstrated, and the impact of detrimental effects is investigated.

Various analytical and numerical evaluations show that CS-RFID renders the identification of multiple tags *quicker,* more *noise robust* and more *energy efficient* than the state of the art.

Kurzfassung

Radio Frequency IDentification (RFID) ist eine Technologie, bei der Transponder (Tags) drahtlos von einem Lesegerät (Reader) identifiziert werden. Das aufstrebende Konzept des Internet of Things (IoT), zu Deutsch Internet der Dinge, bringt eine Vielzahl von neuen RFID Anwendungen mit sich. Diese Dissertation verbessert RFID für das IoT und andere Anwendungen, in welchen eine *Vielzahl von Tags* von einem einzigen Reader identifiziert werden sollen.

Compressed Sensing (CS) ist eine Signalverarbeitungstechnik, mit welcher spärlich besetzte Signalvektoren mittels Unterabtastung rekonstruiert werden können, indem *unterbestimmte* lineare Gleichungssysteme gelöst werden. Konventionelle Tag-Erfassungsmethoden wie Frame Slotted ALOHA (FSA) sind durch Kollisionen beeinträchtigt, während die vorgeschlagene CS-RFID Methode *Kollisionen ausnutzt*. Der Reader initiiert die Erfassung, und alle Tags antworten gleichzeitig mit einer Signatursequenz. Die Überlagerung dieser Sequenzen am Reader wird als CS Messung formuliert, und die *Erfassung der Tags ist ein CS Rekonstruktionsproblem*. Um dieses Problem zu Lösen werden vielseitige Approximate Message Passing (AMP) Rekonstruktionsalgorithmen verglichen und eingesetzt.

Die Information aus der Tag-Erfassungsphase wird anschließend von Identifikationsprotokollen benutzt, um alle Tags zuverlässig zu identifizieren. Zu diesem Zweck werden zwei Protokolle vorgestellt: Ein sehr schnelles für fixe Inventare (Objekte mit Tag), und ein allgemeineres für beliebige Inventare.

Die AMP Rekonstruktionsalgorithmen können eine variable Menge an Vorwissen ausnutzten. Ein neuer Algorithmus wird präsentiert, der in der Lage ist, "Joint Sparsity" und die statistische Signalverteilung auszunutzen. Dies ist insbesondere bei einem Reader mit *mehreren Empfangsantennen* relevant und nützlich.

Ein *flexibler Messaufbau*, der es ermöglicht, Daten und Takt von mehreren UHF RFID Tags zu kontrollieren, wird eingeführt. Die praktische Realisierbarkeit von CS-RFID wird demonstriert, und der Einfluss von störenden Effekten wird untersucht.

Verschiedene analytische und numerische Auswertungen belegen, dass CS-RFID bei der Identifikation von mehreren Tags *schneller*, *robuster gegen Rauschen* und *energieeffizienter* als der aktuelle Stand der Technik ist.

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgments.

<div align="right">

Martin Mayer
August 2016

</div>

Acknowledgments

Table of Contents

Notation and Definitions

Scalars, vectors and matrices:

Deterministic and random scalar:	a and a
Deterministic and random vector:	$\mathbf{a}$ and $\mathbf{a}$
Deterministic and random matrix:	$\mathbf{A}$ and $\mathbf{A}$
Columns of an $M \times N$ matrix:	$\mathbf{A} = [\mathbf{a}_1, ..., \mathbf{a}_N]$
Rows of an $M \times N$ matrix:	$\mathbf{A} = [\mathbf{A}_{1,:}; ...; \mathbf{A}_{M,:}]$
Entry of matrix, m-th row and n-th column:	$(\mathbf{A})_{m,n} \equiv a_{m,n}$
Complex conjugate:	$(\cdot)^*$
Transposition of vector or matrix:	$(\cdot)^{\mathrm{T}}$
Hermitian adjoint of vector or matrix:	$(\cdot)^{\mathrm{H}}$
Inverse of matrix:	$(\cdot)^{-1}$
Determinant of a matrix:	$\lvert \cdot \rvert$
$N \times N$ identity matrix:	$\mathbf{I}_N$
$M \times N$ all-zero matrix:	$\mathbf{0}_{M \times N}$
$M \times N$ all-one matrix:	$\mathbf{1}_{M \times N}$
Vector $\mathbf{a}$ to diagonal matrix $\mathbf{A}$ operation:	$\mathbf{A} = \mathrm{diag}\,(\mathbf{a})$
Sum of diagonal entries of $M \times M$ matrix:	$\mathrm{trace}(\mathbf{A}) = \sum_{m=1}^{M} a_{m,m}$

Sets:

Set (denoted in calligraphic font):	$\mathcal{S}$
Cardinality (number of elements):	$\lvert \mathcal{S} \rvert$
Vector with entries described by set (set contains indices):	$\mathbf{a}_{\mathcal{S}}$

Distributions:

x is Gaussian distributed with mean μ and variance σ^2:	$\mathsf{x} \sim \mathcal{N}(\mu, \sigma^2)$
x is Laplace distributed with mean μ and scale parameter κ:	$\mathsf{x} \sim \mathcal{L}(\mu, \kappa)$
x is Gamma distr. with shape param. k and inv. scale param. θ:	$\mathsf{x} \sim \Gamma(k, \theta)$

Gaussian distribution with mean μ and variance σ^2 evaluated at x:

$$\mathcal{N}(x; \mu, \sigma^2) \equiv \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(x - \mu)^2\right).$$

Laplace distribution with mean μ and variance $2\kappa^2$ evaluated at x:

$$\mathcal{L}(x; \mu, \kappa) \equiv \frac{1}{2\kappa} \exp\left(-\frac{|x - \mu|}{\kappa}\right).$$

Gamma distribution with shape parameter k and inverse scale parameter θ evaluated at x ($\Gamma_{\mathrm{f}}(\cdot)$ denotes the gamma function):

$$\Gamma(x; k, \theta) \equiv \frac{x^{k-1} e^{-x/\theta}}{\theta^k \Gamma_{\mathrm{f}}(k)}.$$

Binomial distribution with n trials and success probability p evaluated at k:

$$p_B(k; n, p) \equiv \binom{n}{k} p^k (1-p)^{n-k}.$$

The abbreviation "i.i.d." stands for independent and identically distributed.

Norms:

The l_p-*norm* of a vector $\mathbf{x} \in \mathbb{C}^N$ is defined as

$$\|\mathbf{x}\|_p := \left(\sum_{n=1}^{N} |x_n|^p\right)^{\frac{1}{p}}.$$

The *Frobenius norm* of a matrix $\mathbf{A} \in \mathbb{C}^{M \times N}$ is defined as

$$\|\mathbf{A}\|_{\mathrm{F}} := \sqrt{\mathrm{trace}\left(\mathbf{A}^{\mathrm{H}}\mathbf{A}\right)} = \sqrt{\sum_{m=1}^{M} \sum_{n=1}^{N} |a_{m,n}|^2}.$$

Miscellaneous:

The *Mean Squared Error (MSE)* between a vector $\mathbf{a} \in \mathbb{C}^N$ and a vector $\mathbf{b} \in \mathbb{C}^N$ is defined as

$$\mathrm{MSE}(\mathbf{a}, \mathbf{b}) := \frac{1}{N} \|\mathbf{a} - \mathbf{b}\|_2^2 = \frac{1}{N} \sum_{n=1}^{N} |a_n - b_n|^2.$$

The MSE between two matrices $\mathbf{A} \in \mathbb{C}^{M \times N}$ and $\mathbf{B} \in \mathbb{C}^{M \times N}$ is defined as

$$\mathrm{MSE}(\mathbf{A}, \mathbf{B}) := \frac{1}{MN} \|\mathbf{A} - \mathbf{B}\|_{\mathrm{F}}^2.$$

The *Normalized Mean Squared Error (NMSE)* between two vectors is defined as

$$\text{NMSE}(\mathbf{a}, \mathbf{b}) := \frac{\|\mathbf{a} - \mathbf{b}\|_2^2}{\|\mathbf{a}\|_2^2},$$

the NMSE between two matrices as

$$\text{NMSE}(\mathbf{A}, \mathbf{B}) := \frac{\|\mathbf{A} - \mathbf{B}\|_F^2}{\|\mathbf{A}\|_F^2}.$$

Chapter 1

Introduction

We are living at the dawn of the Second Machine Age [1], where inventions are infused by the rapid evolution of digital base technologies. The proliferation of the internet through advances in wireless communications has propelled us into the age of big data [2], where mobile devices occupy more network resources than the long-established desktops, while the amount of transferred data is increasing steeper than ever. We are witnessing a society with an ever-growing hunger for data, and inventing the means to satiate this hunger excites the scientific community with new paradigms.

The Internet of Things (IoT) [3–5] is such a contemporary paradigm that combines several base technologies to achieve the bigger picture: Enabling new technologies based on massive Machine-to-Machine (M2M) communication and interconnectivity. A rapidly growing number of physical objects are being connected to the internet, helping new applications to flourish, and spawning a tremendous flood of data that has to be processed and stored efficiently. Following two base technologies are seen as crucial IoT enablers [4–6], and relevant to this thesis:

- **Radio Frequency IDentification (RFID)** is a ubiquitous technology where a reader device wirelessly identifies tags [7, 8]. The tags can be built incredibly small, battery-less and low-cost, thereby facilitating high-volume productions. The information stored on a tag ranges from simple product codes used to, e.g., identify items in a store, to sensor data used to, e.g., monitor wireless sensor networks. The applications of RFID are manifold [9], and its deployment is expected to grow tremendously under the IoT paradigm [10–12].

- **Compressed Sensing (CS)** is a signal processing technique to acquire sparse signal vectors in a compressive manner and to fully recover sparse signal vec-

tors from under-determined linear systems of equations, i.e., from fewer samples than previously believed [13–15]. It is a key enabler to gather the huge amounts of data from sensors, social networks and many other applications in the age of big data. The IoT is governed by massive M2M communication that contributes to the big data phenomenon, calling for techniques such as CS that help to reduce the amount of stored and processed data.

The IoT applications involving RFID span all major sectors surrounding our daily lives [5]: Healthcare, transportation, industry, market, school, vehicles, smart homes, and agriculture. The IoT allows physical objects to communicate and share information, and to be located and identified. Each of the sectors named above contains objects that interact "intra-sector", i.e., on relatively short distances. Consider RFID tags equipped with sensors used to monitor a person's health, continuously read out by a central device. In another application, all items in a warehouse are equipped with RFID tags, continuously read out by a central device to monitor the stock. Information is also spread "inter-sector", i.e., over relatively long distances via the internet. The personal health information could be uploaded to a server for logging purposes, while the warehouse monitor information may be forwarded to the company headquarters.

While the intra-sector interaction hinges on technologies like RFID, the inter-sector communication relies on the internet and technologies to tap into it, like the upcoming 5th generation of mobile networks [16, 17]. The focus of this thesis lies on the intra-sector wireless communication with RFID tags. A myriad of RFID applications are envisioned under the IoT paradigm, many involving the communication with a multitude of RFID tags at once. By equipping everyday objects with RFID tags, the internet is extended to the IoT.

In this thesis, I propose prescriptions of how to apply big data signal processing techniques to RFID in order to strongly improve the communication with a multitude of RFID tags, rendering their identification quick, robust and energy efficient. I answer the following question: *"How to improve RFID for the IoT?"*

1.1 The Relevance of RFID

The roots of RFID can presumably be traced back to the application of radar in World War II [18] — it was discovered that a plane that performs a roll changes the reflected radio signal that was originally emitted by the radar station. This facilitated the identification of friendly planes, without necessitating the planes to actively transmit information. Since then, RFID technology has come a long way. The radar station has become the RFID reader that transmits a Continuous Wave (CW) carrier signal, and the planes have been replaced by incredibly small and cheap tags that change the reflection coefficient of their antenna in order to evoke amplitude changes in the reflected signal and convey information to the reader — this principle is called *backscatter communication*.
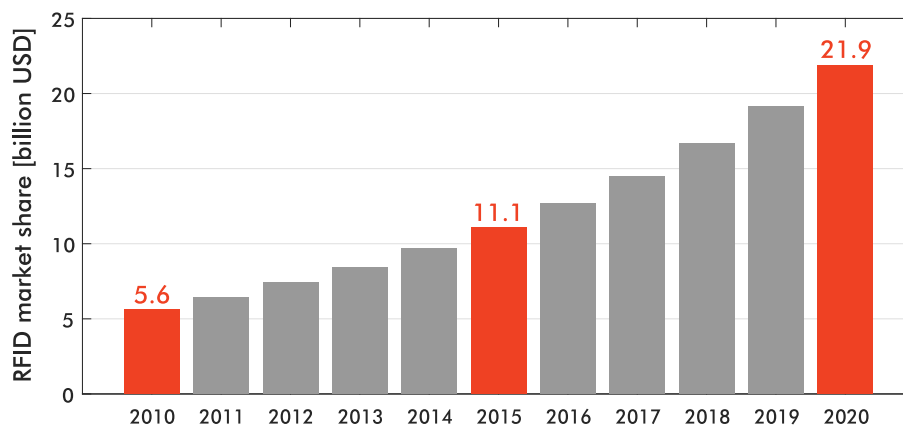
Figure 1.1 Projected size of the global market for RFID tags.

Nowadays, RFID is a ubiquitous technology deployed in retail, supply chain management, aviation, healthcare, public transport, and many other areas [8, 10, 19]. Figure 1.1 [20] illustrates the size of the global market for RFID tags over recent years, and its prospective size by 2020. The utilization of RFID tags has doubled from 2010 to 2015, and is projected to continue this trend in the years to come. The expected growth in RFID is attributed to the strongly increasing deployment in transport, logistics, healthcare and other new applications emerging under the IoT paradigm. The recently founded RAIN RFID industry alliance [12] will further promote the adoption of Ultra High Frequency (UHF) RFID to identify, locate, authenticate and engage everyday items. Furthermore, the widespread concept of Industry 4.0 foresees a heightened RFID usage [21].
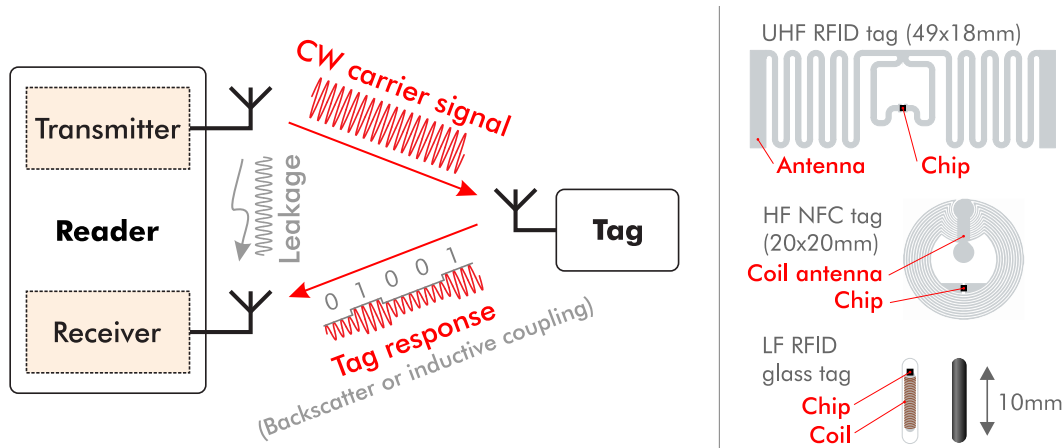
Figure 1.2 Generic RFID setup.

A major trend is to bring technology closer to our bodies. A key invention of the last decade was the smartphone that extends our fingertips to reach into the internet. In recent years, smartphones have been equipped with Near Field Communication (NFC) [22], a technology to read out High Frequency (HF) RFID tags at close proximity that can be used for, e.g., mobile payment applications. It is to be expected that future smartphones will be equipped with UHF RFID readers that are able to read out tags within the range of several meters — several companies already offer proper smartphone adapters. This would promote new applications such as indoor positioning [23]. Another strongly growing field that brings RFID closer to our bodies are wearables that incorporate RFID in everyday apparel, and sensors that are placed on our skin [24–27]. Further down the road, RFID technology will grow into our bodies in the form of implants [28–32]. Implants based on RFID technology bear the huge advantage of being almost maintenance free due to their battery-less operation. Sensor information can be read out in a noninvasive manner.

I will now describe the technological foundations of RFID, followed by a survey of identification protocols.

1.1.1 A Brief Introduction to RFID Technology

The basic RFID setup, illustrated by Figure 1.2, comprises a reader device and one or several tags that are to be identified by the reader. Identification refers to the exchange of data between tag and reader, which is achieved by means of a wireless radio link. The reader is the "sophisticated", active part of the communication that coordinates the identification process. The tags primarily consist of a basic in-

	Low Frequency (LF)	High Frequency (HF)	Ultra High Frequency (UHF)	Microwave
Frequency	$< 135\,\mathrm{kHz}$	$13.56\,\mathrm{MHz}$	$860 - 950\,\mathrm{MHz}$	$2.45\,\mathrm{GHz}$
Read range	$1\,\mathrm{m}$	$0.2\,\mathrm{m}$	$7\,\mathrm{m}$	$1\,\mathrm{m}$
Antenna type	Coil ($> 100\,\mathrm{turns}$)	Coil ($< 10\,\mathrm{turns}$)	Dipole antenna	Dipole antenna
Standards	ISO 18000-2, ISO 11784/85	ISO 18000-3, ISO 14443, NFC	ISO 18000-63, EPCglobal Gen2	ISO 18000-4
Coupling	Inductively-coupled systems		Backscatter systems	
Tag type	Passive		Passive, semi-passive, active	
Power consum.	Lowest ←——————————————————————————→ Highest			
Bandwidth	Lowest ←——————————————————————————→ Highest			
Complexity	Lowest ←——————————————————————————→ Highest			
Material influ.	Lowest ←——————————————————————————→ Highest			

Table 1.1 Classification of RFID.

tegrated circuit –the "chip"– that stores information and performs simple processing tasks, and an antenna or coil for the wireless data transfer.

The established frequency bands for RFID are depicted in Table 1.1 [33, 34], along with their implications on the read-out distance and the type of communication. The depicted read ranges are typical values for passive tags that are powered by the reader; semi-passive or active tags achieve read ranges of up to $100\,\mathrm{m}$ (a definition of the tag types follows below). The range-limiting factor is the allowed transmit power that is dictated by the individual standards. With increasing operating (carrier-) frequency, the bandwidth and data rate increase. However, wireless communication at higher frequencies is stronger affected by materials, in particular conductive liquids and metal. While Low Frequency (LF) and HF tags can be submerged in liquids or implanted into animals, UHF and microwave tags require free space propagation and a certain clearance from liquids and metal. Tags that operate in the LF and HF bands employ coil antennas and use inductive coupling for energy- and data-transfer. Tags that operate in the UHF or microwave bands employ dipole antennas and use electromagnetic wave propagation for energy- and data-transfer.

Considering the latter, the prevalent technique to convey data from tag to reader is called *backscatter modulation* [8, 35]. In such, the reader emits a CW carrier signal that is reflected at the tag. The tag changes the load impedance of its antenna according to its binary transmit data. This evokes changes in the reflection coefficient that ultimately result in amplitude changes in the reflected signal. Consequently, the backscattered CW exhibits amplitude changes that are recorded at the reader's receive antenna — this is hinted in Figure 1.2. Having this technique in mind, the following kinds of tags are distinguished:

- **Passive tags** are fully powered by the field emitted by the reader. Aside from powering the tags, the reader's CW is also used for data transmission via backscatter modulation.

- **Semi-passive tags** contain a power source to supply the chip. This helps to increase the read range. Moreover, these tags employ backscatter modulation and, therefore, require the reader's CW to transmit data.

- **Active tags** contain a power source and actively transmit data on their own. These tags do not rely on the reader's CW.

Passive UHF tags with backscatter modulation are the embodiment of RFID, because they can be produced cheaply and in high volumes. Furthermore, their packaging can be small, robust and durable, and they can be read out from several meters of distance under non-line-of-sight conditions.

In order to read out passive or semi-passive tags, the reader has to constantly provide a CW, see Figure 1.2. It is transmitting the CW while simultaneously receiving a weak tag response. The strong, unmodulated CW emitted at the transmitter is also received directly by the receiver, where it is termed leakage [8, 36]. The leakage between transmitter and receiver can significantly limit the reader's receive sensitivity, since the received CW may be significantly stronger than the tag response, i.e., the small amplitude changes a tag imposes on the CW. By a sufficient separation of transmit and receive antenna, the leakage can be reduced. Setups with separated transmit and receive antennas are termed bistatic, as shown in Figure 1.2. The alternative is a monostatic setup, where a single antenna is used for transmission and reception [37]. Methods and setups to combat leakage have been proposed in literature [38].

The *link frequency* refers to the physical transmission rate of information. In UHF RFID according to EPCglobal [39], the link frequency ranges from $40\,\mathrm{kHz}$ to $640\,\mathrm{kHz}$. Note that the data is encoded by FM0 or Miller encoding before transmission [39, 40]. A good overview of the various RFID standards along with their fields of application is provided in [34].

1.1.2 A Survey of Identification Protocols

The predominant application of RFID is the *identification* of objects that are equipped with tags. Most of the standards listed in Table 1.1 employ protocols that can identify multiple tags in read range, NFC being an exception. This thesis focuses on the swift identification of a multitude of tags, a scenario prevalent in UHF or microwave RFID. Therefore, I assume in the sequel that the tags use backscatter modulation as communication method.

Due to the rudimentary nature of passive tags, the wireless communication between reader and tags is rather simplistic. The tags cannot communicate with each other, and they respond (backscatter) on the carrier frequency provided by the reader. Only single tag responses can be handled by a standard reader, multiple tags responding at once produce a *collision* — the tag responses superimpose and "collide" at the reader, and the data is lost. If several tags are in read range and ought to be identified, their responses have to be coordinated by identification protocols.

An overview of the most common protocols is provided in [33, 41]. For better comprehension, I separate the identification process into two subsequent phases:

1. **Acquisition** refers to obtaining the means to communicate with a tag. It establishes a handshake mechanism between reader and tag. The reader acquires a (temporary) identifier that is used to communicate with the tag.

2. **Data read-out** refers to reading out the tag payload via the acquired identifier. The payload typically contains the product code [39] or sensory information [28, 29].

The crucial aspect of the identification protocol is the *acquisition phase*. Before acquisition, the reader has very limited or no knowledge about the tags in read range. In order to avoid collisions during acquisition, the tag responses can be separated in several domains [33, 41], presented here with decreasing popularity.

Time Division Multiple Access (TDMA) is the prevalent technique that separates the tag responses in time domain. The most common protocol is Frame Slotted ALOHA (FSA) that is used in EPCglobal [39]. The tags are randomly scheduled to respond in slots of a frame; besides pseudo-randomly choosing a slot, the tags also randomly generate their response, a 16-bit random number called RN16. The RN16 constitutes the temporary identifier that is later used by the reader to communicate with the tag. The reader acquires the RN16 number of collision-free (singleton) slots, and reschedules tags that have picked a collision slot. This process is illustrated by Figure 1.3. The slot occupancy probability is stated by a binomial distribution [42]. Assume K contending tags and a frame size of F slots; the probability that n tags occupy one slot is

$$P(\text{``}n\text{ tags occupy one slot''}) = p_B\left(n; K, \frac{1}{F}\right) = \binom{K}{n}\left(\frac{1}{F}\right)^n \left(1 - \frac{1}{F}\right)^{K-n}. \quad (1.1)$$
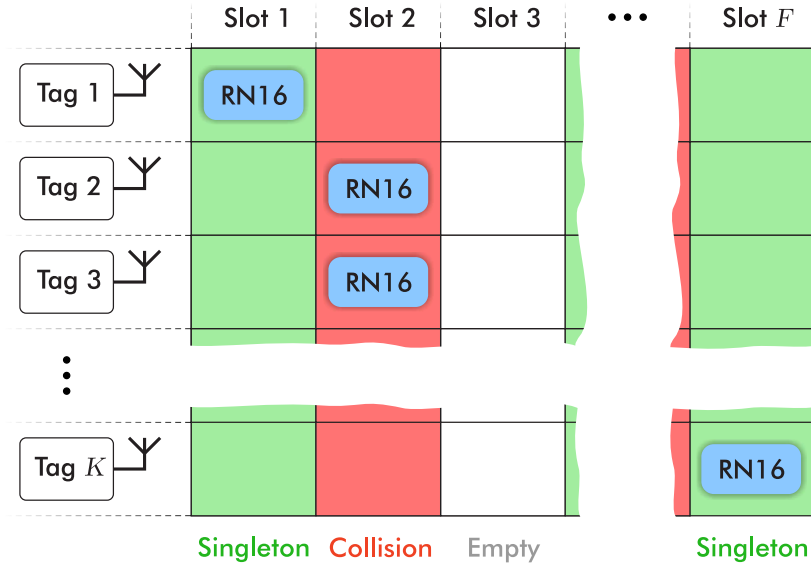
Figure 1.3 FSA: Tags randomly choose slots in a frame.

If the number of slots is set equal to the number of contending tags, the acquisition throughput is maximized, i.e., the probability of a singleton slot is maximized while keeping the collision and empty-slot probabilities low. For large frame sizes $F = K$, the following probabilities are obtained:

$$P(\text{``Empty slot''}) = p_B\left(0; K, \frac{1}{K}\right) = e^{-1} \approx 0.368, \tag{1.2}$$

$$P(\text{``Singleton slot''}) = p_B\left(1; K, \frac{1}{K}\right) = e^{-1} \approx 0.368, \tag{1.3}$$

$$P(\text{``Collision slot''}) = 1 - p_B\left(0; K, \frac{1}{K}\right) - p_B\left(1; K, \frac{1}{F}\right) \approx 0.264. \tag{1.4}$$

This means that in expectation, 36.8% of the tags can be acquired in one round, provided that the number of tags K is known.

Another class of TDMA techniques are tree-based protocols. While FSA is classified as a random and tag-driven protocol, tree-based protocols are deterministic and reader-driven. A subset of tags in read range is queried by the reader to respond in a time slot. If a collision occurs, the first subset is further split into two subsets, until a collision-free slot is obtained. The reader then continues with the next subset, until all subsets are resolved, and all tags are identified. Tree-based approaches have a higher reader-to-tag communication overhead than FSA. However, their deterministic nature bears advantages if a very large number of tags has to be

identified, because identification of all tags is guaranteed. FSA, on the other hand, entails a probabilistic result that holds in expectation; while unlikely, it may happen that several tags collide for a long time.

Code Division Multiple Access (CDMA) techniques separate the tag responses in code domain. In the basic approach, the tags multiply their identifier with a pseudo-random spreading sequence. This is often a spread spectrum technique, i.e., the link frequency (bandwidth) is multiplied by the spreading sequence length. The spreading sequences are chosen to have large auto-correlation and small cross-correlation, facilitating the distinction between tag responses with different spreading codes [43]. Orthogonal Walsh codes are considered as well [44]. CDMA is a power-hungry technique that usually requires semi-passive or active tags [45, 46]. In case of spread spectrum CDMA that entails wide-band transmissions, the receiver structure at the reader becomes more intricate in order to handle multi-path propagation and inter-symbol interference effects. This increases the reader cost.

Space Division Multiple Access (SDMA) refers to the separation of the reading area into sectors. This is accomplished by having several readers with separate antennas, or by beam-forming techniques. In the latter, the reader illuminates only specific sectors of its read range, in which a fraction of the tag population resides. However, sharp beams are only achieved at high frequencies with multiple antennas, rendering the technique expensive.

Frequency Division Multiple Access (FDMA) is a technique where a reader provides several uplink channels on different frequencies for the tags to respond on. The reader requires a dedicated receiver unit for each channel. This entails high hardware requirements and costs. FDMA is very rarely used in RFID.

Combinations of TDMA with one of the other multiplexing approaches are also proposed in literature. The authors in [43] suggest to employ CDMA in order to resolve collisions in FSA. Spatial filtering, i.e., SDMA was proposed in [47] to effectively reduce the number of contending tags in FSA.

After obtaining the tag identifiers with one of the above acquisition methods, the reader *identifies* the tags[1] via *data read-out*. This is usually done by sequentially enquiring the list of identifiers, where the tags respond with their payload. Note that acquisition and data read-out are separated because the identifiers are considerably

---

[1]Note that some CDMA approaches directly identify the tags without separate acquisition.

shorter than the payload. This significantly reduces the total identification time, which can be illustrated by a simple toy example.

Assume an identifier length of 16 bits ("RN16"), and a data payload of $D = 128$ bits. Considering FSA with optimal frame size, the identification of $K = 10$ tags roughly requires

$$\underbrace{16 \cdot e \cdot K}_{\text{Acquisition}} + \underbrace{D \cdot K}_{\text{Data read-out}} \approx 1\,715 \text{ bits}, \tag{1.5}$$

while the direct identification without acquisition of an identifier (replace RN16 in acquisition by data payload D) takes about twice as long:

$$D \cdot e \cdot K \approx 3\,479 \text{ bits}. \tag{1.6}$$

Putting both equations in relation, a speedup is observed as soon as $D > 25$ bits.

Similar experiments can be conducted for CDMA techniques. Assume that the 16-bit identifiers are spread by a Gold code of length 127 [43, 45], where each of the $K = 10$ tags uses a different code instance such that the reader can distinguish the simultaneous responses, and the data payload is $D = 128$ bits. Note that in spread spectrum CDMA, the physical transmission happens at the chip rate that is equal to the spreading factor times the bit rate. Assuming that the unspread data payload is transmitted at chip rate, the identification roughly requires

$$\underbrace{16 \cdot 127}_{\text{Acquisition}} + \underbrace{D \cdot K}_{\text{Data read-out}} = 3\,312 \text{ chips}, \tag{1.7}$$

while the direct spreading of the data payload would take about five times as long:

$$D \cdot 127 = 16\,256 \text{ chips}. \tag{1.8}$$

A rather novel CDMA-related approach utilizes CS techniques to obtain the tag identifiers. The tags respond simultaneously with pseudo-randomly generated sequences, *without spreading* the spectrum or the identifier. The superposition of the responses at the reader is interpreted as a CS measurement that entails an underdetermined linear system of equations, the unknown vector indicates the temporary identifiers. Using CS recovery techniques, the unknown vector is reconstructed and the identifiers are acquired. This concept was first proposed in [48], and it is the core topic of this thesis. The utilization of CS techniques bears significant improvements over the classic CDMA approach, which will become apparent in Section 1.3. Let me first discuss the fundamentals of CS.

1.2   The Emergence of Compressed Sensing

CS was introduced in [13–15] to recover sparse vectors from under-determined systems of linear equations. This is depicted by Figure 1.4. It was shown that a sparse
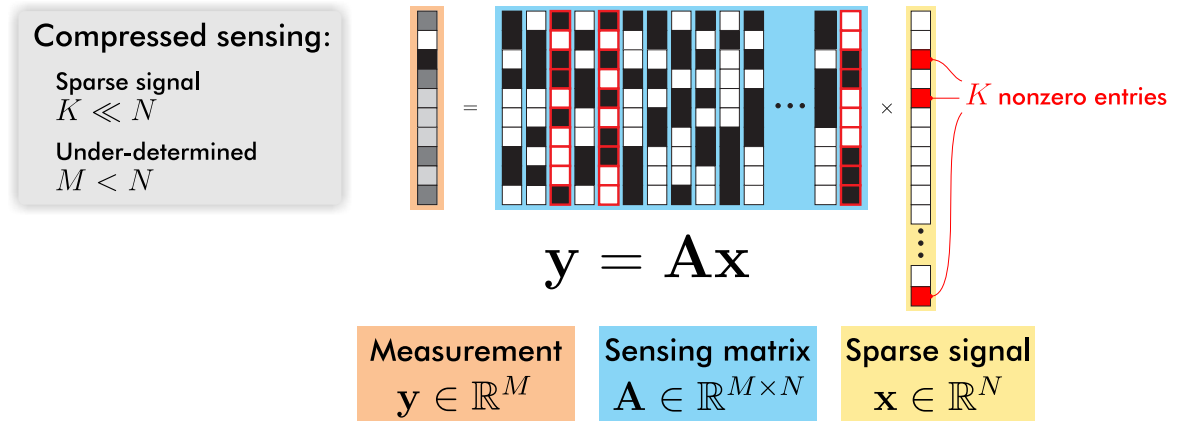
Figure 1.4 CS in a nutshell.

signal vector $\mathbf{x} \in \mathbb{R}^N$ can be recovered from $M < N$ linear measurements $\mathbf{y} \in \mathbb{R}^M$ if sensing matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$ satisfies certain properties that will be discussed in Section 1.2.3. This finding, along with the means to recover $\mathbf{x}$, had a huge impact on the scientific community for various reasons:

- A plethora of applications features data that is sparse or compressible[2] in a certain domain. Think of vectorized images that are dense in image domain but highly compressible after energy compressing transforms such as the fast Fourier transform or the discrete cosine transform. After identifying the sparsity in a problem, CS can be applied.

- Rather than acquiring the whole data first and performing compression afterwards, CS –as the name suggests– allows to perform compression during the sampling process. This is advantageous if one has to deal with a huge amount of data. A prime example is Magnet Resonance Imaging (MRI) [49].

- CS allows to recover signals from incomplete or corrupted data. A typical application is image inpainting [50].

In recent years, CS dominated many areas of research and became a valuable tool. Aside from finding theoretical bounds on how many measurements M are required

---

[2]Compressible data is approximately sparse, i.e., the data is approximately described by a few large entries, while many small entries account for detail that may be omitted.

to recover $\mathbf{x}$ [51, 52], the quest to find optimal recovery algorithms that attain these bounds is still ongoing. In particular, prior information in form of additional signal structure and statistical distributions is incorporated, and computationally efficient algorithms are aspired.

I will now define the CS model in a more rigorous manner as required for this thesis, and then provide an overview of various recovery algorithms.

1.2.1 Sparse Signal Model

In the context of CS, a signal is a sparse vector. A signal vector $\mathbf{x} \in \mathbb{R}^N$ is said to be K-sparse if it features at most K nonzero entries, and K is significantly smaller than the vector dimension N. A customary measure to count the number of nonzero entries is the ℓ_0-"norm"[3] [53]

$$\|\mathbf{x}\|_0 = \lim_{p \to 0} \|\mathbf{x}\|_p^p = \lim_{p \to 0} \sum_{n=1}^{N} |x_n|^p = |\{n : x_n \neq 0\}| . \tag{1.9}$$

Definition 1.2.1 (Support). *The support of a vector $\mathbf{x} \in \mathbb{R}^N$ is a set that contains the indices that correspond to the nonzero entries in $\mathbf{x}$:*

$$\mathcal{S}_{\mathbf{x}} := \{n : x_n \neq 0\}. \tag{1.10}$$

In the *deterministic* setting, a signal vector $\mathbf{x}$ stems from the set of K-sparse signals defined below. This signal model is later used to constrain the problem formulation to search for sparse solutions.

Definition 1.2.2 (Set of K-sparse signals). *The set of K-sparse signals is defined as*

$$\mathcal{X}_K := \left\{ \mathbf{x} \in \mathbb{R}^N : \|\mathbf{x}\|_0 \leq K \right\}. \tag{1.11}$$

In the *random* setting, a signal vector $\mathbf{x}$ is a realization of the random vector x. Each entry of x has a certain probability distribution $f_{\mathsf{x}_n}(x_n)$. In the realm of Bayesian estimation, $f_{\mathsf{x}_n}(x_n)$ is called the *prior distribution* of a signal entry. It is part of the prior knowledge that is required to formulate the estimation problem and, hence, assumed to be (approximately) known. Considering sparse signal vectors that feature entries with value zero, I define the following probability:

---

[3]Note that $\| \cdot \|_0$ is not a valid norm as it does not satisfy the homogeneity property.

Definition 1.2.3 (Zero probability). *The probability that the n-th signal entry is zero is denoted by*

$$\gamma_n := P(\mathsf{x}_n = 0). \tag{1.12}$$

If the number of nonzero entries K is known a priori and all signal entries are equally likely to be zero, the zero probabilities compute as

$$\gamma_n = 1 - \frac{K}{N}, \quad \forall n \in \{1, ..., N\}. \tag{1.13}$$

The nonzero (i.e., "active") entries of $\mathbf{x}$ have a distribution $f_{a_n}(x_n)$, and the overall distribution of a signal entry reads

$$f_{\mathsf{x}_n}(x_n) = \gamma_n \delta(x_n) + (1 - \gamma_n) f_{a_n}(x_n). \tag{1.14}$$

Whether the deterministic or the random signal model is utilized depends on the available prior information.

1.2.2 Measurement Model and Problem Formulation

At the heart of CS lies the *linear measurement model*

$$\mathbf{y} = \mathbf{A}\mathbf{x} \tag{1.15}$$

with *measurement vector* $\mathbf{y} \in \mathbb{R}^M$, *sensing matrix* $\mathbf{A} \in \mathbb{R}^{M \times N}$ and *signal vector* $\mathbf{x} \in \mathbb{R}^N$. The goal is to recover $\mathbf{x}$ from $\mathbf{y}$ in the under-determined case where $M < N$, i.e., from (significantly) fewer samples than the dimension of the unknown vector — this is where the name CS, also known as compressed sampling, originates from.

Considering a full-rank matrix $\mathbf{A}$ in the under-determined case, there exist infinitely many solutions to problem (1.15). A standard approach is the utilization of least squares, yielding the solution with minimal ℓ_2-norm:

$$\widehat{\mathbf{x}}_{\mathrm{LS}}(\mathbf{y}) = \mathbf{A}^{\mathrm{T}}(\mathbf{A}\mathbf{A}^{\mathrm{T}})^{-1}\mathbf{y} = \arg\min_{\widetilde{\mathbf{x}}} \|\widetilde{\mathbf{x}}\|_2 \quad \text{s. t. } \mathbf{y} = \mathbf{A}\widetilde{\mathbf{x}}. \tag{1.16}$$

Assuming a K-sparse vector $\mathbf{x} \in \mathcal{X}_K$, (1.16) may not yield a satisfying solution since the ℓ_2-norm minimizer generally does not promote sparse solutions. A sparsity enforcing formulation reads

$$\widehat{\mathbf{x}}(\mathbf{y}) = \arg\min_{\widetilde{\mathbf{x}}} \|\widetilde{\mathbf{x}}\|_0 \quad \text{s. t. } \mathbf{y} = \mathbf{A}\widetilde{\mathbf{x}}. \tag{1.17}$$

This combinatorial problem, however, entails a very high computational complexity and is infeasible to solve for a large dimension N. It is therefore relaxed by a milder constraint that yields the same result for large dimensions with high probability [13]:

$$\widehat{\mathbf{x}}(\mathbf{y}) = \arg\min_{\widetilde{\mathbf{x}}} \|\widetilde{\mathbf{x}}\|_1 \quad \text{s. t.} \quad \mathbf{y} = \mathbf{A}\widetilde{\mathbf{x}}. \tag{1.18}$$

By replacing the ℓ_0-"norm" by the ℓ_1-norm, a convex problem of reduced complexity is obtained. Problem (1.18) is widely known as *basis pursuit* and has been investigated rigorously in literature [54].

In practice, the CS measurement (1.15) is often distorted by noise. The *noisy linear measurement model* reads

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{w}, \tag{1.19}$$

with *noise vector* $\mathbf{w} \in \mathbb{R}^M$. Discussing the under-determined case ($M < N$) relevant to CS, the least squares approach minimizes the error with respect to the ℓ_2-norm:

$$\widehat{\mathbf{x}}_{\text{LS}}(\mathbf{y}) = \mathbf{A}^{\text{T}}(\mathbf{A}\mathbf{A}^{\text{T}})^{-1}\mathbf{y} = \arg\min_{\widetilde{\mathbf{x}}} \|\mathbf{y} - \mathbf{A}\widetilde{\mathbf{x}}\|_2^2. \tag{1.20}$$

The sparsity enforcing formulation reads

$$\widehat{\mathbf{x}}(\mathbf{y}) = \arg\min_{\widetilde{\mathbf{x}}} \|\widetilde{\mathbf{x}}\|_0 \quad \text{s. t.} \quad \|\mathbf{y} - \mathbf{A}\widetilde{\mathbf{x}}\|_2^2 \leq \epsilon, \tag{1.21}$$

where $\epsilon \geq \|\mathbf{w}\|_2^2$. For feasibility, this combinatorial problem (ℓ_0-"norm") is relaxed into a convex problem (ℓ_1-norm)

$$\widehat{\mathbf{x}}(\mathbf{y}) = \arg\min_{\widetilde{\mathbf{x}}} \|\widetilde{\mathbf{x}}\|_1 \quad \text{s. t.} \quad \|\mathbf{y} - \mathbf{A}\widetilde{\mathbf{x}}\|_2^2 \leq \epsilon, \tag{1.22}$$

which is widely known as *basis pursuit denoising* [54]. An equivalent problem is stated by the Least Absolute Shrinkage and Selection Operator (LASSO) [55]

$$\widehat{\mathbf{x}}(\mathbf{y}; \lambda) = \arg\min_{\widetilde{\mathbf{x}}} \left(\frac{1}{2} \|\mathbf{y} - \mathbf{A}\widetilde{\mathbf{x}}\|_2^2 + \lambda \|\widetilde{\mathbf{x}}\|_1 \right), \tag{1.23}$$

where the solution-sparsity constraint is enforced by a Lagrangian penalty λ.

1.2.3   Sensing Matrix and Recovery Requirements

In many applications, the sensing matrix $\mathbf{A}$ is known and can even be designed. The columns of $\mathbf{A}$ can be interpreted as sampling basis functions, and $\mathbf{A}$ mixes all entries of $\mathbf{x}$ into the measurement vector $\mathbf{y}$. It is crucial to choose $\mathbf{A}$ such that there exists a *unique* recovery solution for $\mathbf{x}$ in the noiseless case (1.15), and such that a *stable* recovery from noisy measurements (1.19) is possible. The following three definitions are used to express recovery guarantees [53, 56].

Definition 1.2.4 (Spark). *The spark of a matrix $\mathbf{A}$, written* $\mathrm{spark}(\mathbf{A})$*, is defined as the smallest number of columns from $\mathbf{A}$ that are linearly dependent.*

Definition 1.2.5 (Mutual coherence). *The mutual coherence of a matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$ is defined as*

$$\mu(\mathbf{A}) := \max_{1 \leq i < j \leq N} \frac{|\mathbf{a}_i^{\mathrm{T}} \mathbf{a}_j|}{\|\mathbf{a}_i\|_2 \|\mathbf{a}_j\|_2}. \tag{1.24}$$

Definition 1.2.6 (Restricted isometry property). *The Restricted Isometry Property (RIP) of a matrix $\mathbf{A}$ is stated by*

$$(1 - \delta_K)\|\mathbf{x}\|_2^2 \leq \|\mathbf{A}\mathbf{x}\|_2^2 \leq (1 + \delta_K)\|\mathbf{x}\|_2^2 \ , \ \forall \mathbf{x} \in \mathcal{X}_K, \tag{1.25}$$

where the restricted isometry constant $\delta_K \in (0, 1)$ is defined as the smallest number that satisfies above equation. A matrix $\mathbf{A}$ is said to have the RIP of order K with constant δ_K if it fulfills (1.25).

The mutual coherence is a measure for how strongly the columns of a matrix are (cor)related, while the RIP states that $\mathbf{A}$ approximately behaves like an orthonormal basis for sparse vectors $\mathbf{x} \in \mathcal{X}_K$.

In order to guarantee a unique solution for $\mathbf{x} \in \mathcal{X}_K$ in the noiseless case (1.15), $\mathbf{A}$ has to satisfy $\mathrm{spark}(\mathbf{A}) > 2K$ [53]. Note, however, that computing the spark is NP-hard, since $\binom{N}{\mathrm{spark}(\mathbf{A})}$ column sets have to be considered. The mutual coherence (1.24) is easier to compute, a unique solution to (1.15) is guaranteed if $\mathbf{A}$ satisfies $\mu(\mathbf{A}) < \frac{1}{2K-1}$ [53]. If $\mathbf{A}$ fulfills the RIP (1.25) with $\delta_K + \delta_{2K} + \delta_{3K} < 1$, then $\mathbf{x} \in \mathcal{X}_K$ can be perfectly recovered from (1.18) [14]. Since the RIP is difficult to compute in practice, it is often bounded by the mutual coherence as $\delta_K < K\mu(\mathbf{A})$ [53].

Advancing to the noisy case (1.19), a stable solution for $\mathbf{x} \in \mathcal{X}_K$ in (1.22) and (1.23) is obtained if $\mathbf{A}$ satisfies the RIP of order $2K$ with a small isometry constant δ_{2K}. This means that $\mathbf{A}$ preserves the Euclidean distance between every pair

of K-sparse vectors up to a small constant δ_{2K} [56]. Consider [14, 56] for details on the constants.

An appropriate sensing matrix $\mathbf{A}$ can be constructed deterministically or randomly. In this work, I pick randomly generated matrices, because it is beneficial for my application. Note that $\mathbf{A}$ is only *generated* randomly, it is fixed for the measurement process, i.e., the CS measurements (1.15) and (1.19) are non-adaptive. It was stated in, e.g., [14] that randomly generated matrices with i.i.d. Gaussian entries with zero mean and variance $\frac{1}{M}$ fulfill the required RIP with overwhelming probability. This is also true for i.i.d. sub-Gaussian distributed entries. An important class of such matrices has entries $\{-\frac{1}{\sqrt{M}}, \frac{1}{\sqrt{M}}\}$ with equal probability — this case will be relevant in my RFID application. Measurement matrices $\mathbf{A}$ with (sub)-Gaussian distributed entries satisfy the RIP of order $2K$ with overwhelming probability, and [14, 56]

$$M = cK \log\left(\frac{N}{K}\right) \tag{1.26}$$

measurements (samples) are required to successfully recover x from (1.22) or (1.23), with $c \in \mathbb{R}$ being a small constant that I call the **measurement multiplier**.

1.2.4   A Survey of Recovery Methods

Many algorithms for CS recovery have been introduced in literature, a good overview is presented in [53, 56, 57]. In order to choose an appropriate recovery algorithm to recover x from (1.19), one has to consider the following points:

- Computational complexity: *"What can I afford, and how long should it take?"*

- Recovery accuracy: *"How much solution fidelity do I need?"*

- Prior knowledge: *"What information do I have?"*

The ideal algorithm has low computational complexity, high recovery accuracy and is able to exploit additional prior knowledge. Most algorithms solve the ℓ_1-regularized problem (1.22), since the combinatorial ℓ_0 problem (1.21) is often infeasible to solve in practice. Let me compare the most prominent approaches to CS recovery.

Convex optimization algorithms such as interior-point methods [15] or projected gradient methods [58] can be applied to the convex problem (1.22) or, equivalently, (1.23). These methods entail reasonable recovery accuracy but high computational complexity. Additional prior knowledge may be included as side constraint. For large signal dimension N, convex optimization is infeasible.

Greedy methods aim at the iterative reconstruction of the signal support set $\mathcal{S}_\mathbf{x}$ and the corresponding signal vector entries. Starting with an empty signal support set $\mathcal{S}_{\widehat{\mathbf{x}}} = \{\emptyset\}$, each iteration adds an index corresponding the column of $\mathbf{A}$ that minimizes the ℓ_2 error $\|\mathbf{y} - \mathbf{A}\widehat{\mathbf{x}}\|_2$, the corresponding signal estimate $\widehat{\mathbf{x}}$ is computed by applying least squares on the support-restricted problem. In case of known K, this procedure is repeated until $|\mathcal{S}_{\widehat{\mathbf{x}}}| = K$. Alternatively, the stopping criterion may be based on the noise variance. The most prominent algorithms are Orthogonal Matching Pursuit (OMP) [59, 60] and Compressive Sampling Matching Pursuit (CoSaMP) [61]. Greedy methods are easy to implement and entail good recovery accuracy. However, a pseudo-inverse (matrix inversion) has to be computed in each iteration, thereby boosting the computational complexity, in particular for large N. Additional structure can be exploited by group-OMP [62]. Greedy methods are efficiently applicable to small or medium signal dimension N.

Iterative thresholding algorithms [63, 64] are very efficient recovery methods that rely on thresholding functions that set signal entries below a certain threshold to zero. Iterative Hard Threshloding (IHT) is based on the ℓ_0-regularized problem

$$\widehat{\mathbf{x}}(\mathbf{y}; \lambda) = \arg \min_{\widetilde{\mathbf{x}}} \left(\frac{1}{2} \|\mathbf{y} - \mathbf{A}\widetilde{\mathbf{x}}\|_2^2 + \lambda \|\widetilde{\mathbf{x}}\|_0 \right), \tag{1.27}$$

while Iterative Soft Threshloding (IST) is based on the ℓ_1-regularized problem (1.23). The thresholds are selected based on the noise variance or the number of nonzero entries K. Both algorithms are easy to implement, have low computational complexity and reasonable recovery accuracy. Additional prior knowledge is not exploited. Iterative thresholding algorithms are applicable to large signal dimension N.

Message passing algorithms [65–68] are based on graphical model concepts [69] and approximately perform belief propagation on a loopy graph [70]. Approximate Message Passing (AMP) iteratively solves the LASSO (1.23), while Bayesian Approximate Message Passing (BAMP) iteratively performs Bayesian estimation based on the prior distribution (1.14). This class of algorithms features the low computational complexity of iterative thresholding algorithms, a simple implementation and good recovery accuracy. Furthermore, prior knowledge can be incorporated naturally, ranging from the signal prior distribution to group- or joint sparsity structure. Message passing algorithms are suitable for large signal dimension N.

For the following reasons, the message passing algorithms AMP and BAMP constitute the optimal choice for my RFID application:

- Low computational complexity facilitates an efficient implementation on the reader hardware [71].

- AMP attains the ℓ_1 phase transition [65] and hence requires the lowest number of measurements for recovery via basis pursuit denoising (1.22).

- BAMP exploits Bayesian prior knowledge about the signal and is able to improve the recovery accuracy, or decrease the number of measurements.

- The algorithms are suitable for large signal dimension N.

Therefore, this thesis is focused on AMP recovery algorithms.

1.3   Motivation: RFID with Compressed Sensing

In many applications envisioned under the IoT paradigm, a massive amount of tags is located in the read range of an RFID reader, waiting to be identified. Several areas of application are subject to a heightened noise level, think about assembly lines or industrial areas that are surrounded by electromagnetic distortions. The readers are often handhelds that are powered by a battery. The optimal identification scheme has the following attributes: *quick, noise robust, reliable, energy efficient*[4].

The key concept of RFID with CS –denoted **CS-RFID** from now on– is based on the following observation: *The number of activated tags that are to be identified by the reader is small compared to the total number of existing tags.* Think of a store where all products are labeled with RFID tags instead of barcodes. A customer brings the shopping cart to the checkout, where the products in the cart are identified by an RFID reader. The products in the cart resemble a very small subset of the total inventory of the store. Similarly in the IoT, a reader at a given time has to cope with a very small subset of tags compared to the total number of existing tags.

This observation puts forward the CS-RFID approach, in which the *tag acquisition is formulated as a CS measurement*. Upon a query from the reader, the activated tags respond *simultaneously*, each with a unique **signature sequence**. Assume that there are N possible sequences that are known to the reader, and that at a given time,

---

[4]*Energy efficiency* implies a short CW transmission time during which the reader transmits the carrier with constant power.
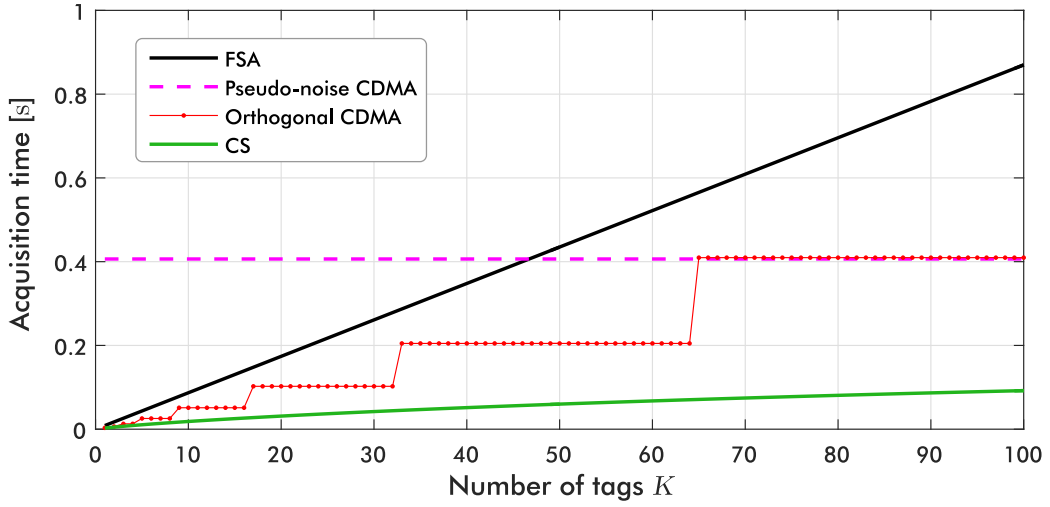
Figure 1.5 Acquisition time for several schemes ($R = 5\,\mathrm{kbit/s}$, $N = 1\,000$).

$K \ll N$ tags respond with their unique sequence. The superposition of sequences received at the reader can be cast as CS measurement similar to (1.19). The N possible signature sequences compose the columns of sensing matrix $\mathbf{A}$, and the indices of the K nonzero entries in $\mathbf{x}$ denote the picked signatures. Using CS recovery algorithms, $\mathbf{x}$ is recovered from $\mathbf{y}$ knowing $\mathbf{A}$. Consequently, the acquisition in CS-RFID takes about M symbols according to (1.26).

In Figure 1.5, the *acquisition time* of CS-RFID is compared to FSA and two CDMA approaches, plotted versus a variable number of activated tags K. To enable a fair comparison, all approaches occupy the same bandwidth and transmit bits or chips at $R = 5\,\mathrm{kbit/s}$.

- FSA chooses the optimal frame size and requires $16 \cdot e \cdot K$ bits for acquisition on average, where 16 bits is the length of the RN16 identifier.

- Pseudo-noise CDMA [43] spreads the 16-bit identifiers with a 127-bit Gold code and requires $16 \cdot 127$ chips for acquisition. It is assumed that all tags choose a different pseudo-noise sequence (Gold code).

- Orthogonal CDMA [44] spreads the 16-bit identifiers with orthogonal Walsh sequences and minimally requires $16 \cdot 2^{\lceil \log_2(K) \rceil}$ chips for acquisition. The number $2^{\lceil \log_2(K) \rceil}$ indicates the minimal sequence length to obtain K orthogonal Walsh sequences. It is assumed that all tags choose a different sequence from the set of orthogonal sequences.

- CS requires $M = c \cdot K \log(N/K)$ bits for acquisition according to (1.26), with $c = 2$ and $N = 1\,000$. A discussion of the parameter selection follows in chapters 2 and 3. The role of the 16-bit identifier is replaced by the signature sequence index, i.e., the column index of $\mathbf{A}$.

Overall, CS-RFID is by far the quickest scheme. Moreover, the CDMA approaches assume that all tags pick different sequences from a small pool of sequences, which is highly unlikely in practice. Let me list the advantages of CS-RFID by already anticipating some results of this thesis:

- Quicker than the widely-used FSA or CDMA schemes.

- Due to the simultaneous tag acquisition, the reader operates energy efficient and does not waste power on empty slots like FSA.

- The pseudo-randomly generated sequences do not require orthogonality or auto-correlation properties like the spreading sequences in CDMA. The spectrum is not spread.

- The approach is not based on random scheduling like FSA, identification is reliable and guaranteed to succeed in finite time.

- The noise robustness is high.

- Prior knowledge such as channel statistics can be utilized.

- The performance can be improved by using several reader receive antennas.

Related Work and Novelty

The CS formulation of the tag acquisition phase was firstly proposed in [48], let me briefly describe the premise: During the acquisition phase, the tags respond simultaneously with pseudo-random signature sequences that are seeded by the tag's temporary identifier, which is a 16 bit random number (i.e., the RN16 number a tag would have picked for FSA). This is formulated as a CS measurement (1.19). Because the total number of possible identifiers is $N = 2^{16}$ and corresponds to the number of columns of the sensing matrix, an efficient recovery of x is infeasible. The scale of the problem is reduced by hashing the identifiers into buckets [48] and eliminating the buckets that contain no energy, thereby strongly reducing the number of possible signatures and, consequently, N. However, this requires knowledge of the number of activated tags K that has to be estimated in a prior step. An improved scale reduction that utilizes a gradient algorithm was introduced in [72]. Another improvement that does not require arbitrary restrictions of the huge initial identifier space was proposed in [73].

In this thesis, I present a different take on the tag acquisition with CS. Where [48] employed a computationally demanding CS recovery algorithm based on convex optimization, I describe how the computationally efficient AMP algorithm can be applied. Different AMP-based algorithms are compared, and a novel algorithm that exploits joint sparsity is introduced — it strongly improves the recovery performance in multiple measurement vector problems, to which CS-RFID appertains. Instead of basing the number of identifiers (columns of the sensing matrix) on the RN16, I introduce two different concepts: *fixed signature assignment* and *random signature assignment*. The former applies to setups with a fixed inventory[5] and is very quick, while the latter is more general and applies to arbitrary inventories. Both concepts are accompanied by identification protocols to ensure reliable identification. Throughout this work, I keep the practical implementation of CS-RFID in mind, culminating in a practical demonstration of the scheme.

---

[5]*Inventory* refers to the set of items or objects that feature an RFID tag for identification.

1.4   Outline

This thesis proposes methods to improve RFID for the IoT and other applications where multiple tags have to be identified quickly and reliably. By utilizing CS and versatile AMP recovery algorithms that are able to exploit prior knowledge and signal structure, **CS-RFID** becomes a very quick, noise robust and energy efficient identification scheme.

Chapter 2: Signal Recovery by Approximate Message Passing describes AMP recovery algorithms for various application cases and a variable amount of prior knowledge. A novel algorithm to exploit joint sparsity is introduced. All algorithms are evaluated numerically with empirical phase transition curves, and analytically with the State Evolution (SE) framework that allows to predict the algorithmic behavior.

Chapter 3: RFID with Compressed Sensing presents CS-RFID in conjunction with AMP and proposes two protocols for reliable identification: The first one is very quick for fixed inventories, the second one is more flexible for arbitrary inventories. The identification performance is compared to FSA.

Chapter 4: Exploiting Joint Sparsity in Tag Acquisition extends CS-RFID to exploit joint sparsity structure. This is particularly relevant to readers with several receive antennas. A support detection scheme for reliable tag acquisition is proposed, and the performance is compared to FSA with collision-recovery capabilities.

Chapter 5: Practical Implementation is devoted to the practical implementation of CS-RFID and demonstrates the feasibility of the approach with proof-of-concept measurements. The impact of detrimental effects –delay and jitter– is measured and evaluated.

Chapter 6: Conclusions summarizes the work, discusses open issues and concludes the thesis.

Chapter 2

Signal Recovery by Approximate Message Passing

In this chapter, the AMP framework that was proposed in [65–68] and the corresponding algorithms that efficiently recover x from (1.19) are reviewed. The presented algorithms allow for seamless incorporation of prior knowledge that improves the recovery performance. In particular, the focus resides on Bayesian estimation that leverages the signal distribution. Moreover, I present a novel algorithm that is able to exploit joint sparsity structure. The presented algorithms are later utilized in chapters 3 to 5 to perform the tag acquisition in CS-RFID.

Based on my work in [74–76], the contributions of this chapter are as follows:

- The AMP and BAMP algorithms are presented.

- Optimal tuning of the AMP algorithm is discussed.

- The Minimum Mean Squared Error (MMSE) estimator function of BAMP is specified for the Bernoulli-Gaussian and Bernoulli-Laplace signal priors.

- To leverage joint sparsity, an extension to the BAMP algorithm, termed BAyesian Structured Signal Approximate Message Passing (BASSAMP), is proposed.

- Empirical phase transition curves are presented, and the recovery performance of the algorithms is compared.

- The SE formalism [65, 77, 78] to predict the behavior of the presented algorithms is reviewed and applied.

2.1 Probabilistic Estimation Setup

I will now derive a Bayesian estimator for $\mathbf{x}$ in (1.19) that utilizes the random signal model from Section 1.2.1. The sensing matrix $\mathbf{A}$ is deterministic (fixed), its entries are assumed to have variance $\frac{1}{M}$, i.e., $\mathbf{A}$ has *normalized columns*. Let me specify the distributions of measurement vector $\mathbf{y} = [\mathsf{y}_1, ..., \mathsf{y}_m, ..., \mathsf{y}_M]^{\mathrm{T}}$, signal vector $\mathbf{x} = [\mathsf{x}_1, ..., \mathsf{x}_n, ..., \mathsf{x}_N]^{\mathrm{T}}$ and noise vector $\mathbf{w} = [\mathsf{w}_1, ..., \mathsf{w}_m, ..., \mathsf{w}_M]^{\mathrm{T}}$. The noise is assumed to be i.i.d. Gaussian with zero mean and covariance $\sigma_{\mathsf{w}}^2 \mathbf{I}_M$, i.e., $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \sigma_{\mathsf{w}}^2 \mathbf{I}_M)$. The noise Probability Density Function (PDF) calculates as

$$f_{\mathbf{w}}(\mathbf{w}) = \prod_{m=1}^{M} f_{\mathsf{w}}(w_m) = \prod_{m=1}^{M} \mathcal{N}(w_m; 0, \sigma_{\mathsf{w}}^2). \tag{2.1}$$

The joint PDF of signal and measurement can be factored according to Bayes' rule:

$$f_{\mathbf{x},\mathbf{y}}(\mathbf{x}, \mathbf{y}) = \underbrace{f_{\mathbf{x}|\mathbf{y}}(\mathbf{x}|\mathbf{y})}_{\text{Posterior}} f_{\mathbf{y}}(\mathbf{y}) = \underbrace{f_{\mathbf{y}|\mathbf{x}}(\mathbf{y}|\mathbf{x})}_{\text{Likelihood}} \underbrace{f_{\mathbf{x}}(\mathbf{x})}_{\text{Prior}}. \tag{2.2}$$

The signal entries are assumed to be independently distributed with PDF $f_{\mathsf{x}_n}(x_n)$, the *prior*, thus, factors as

$$f_{\mathbf{x}}(\mathbf{x}) = \prod_{n=1}^{N} f_{\mathsf{x}_n}(x_n). \tag{2.3}$$

The *likelihood* is characterized by the noise PDF:

$$f_{\mathbf{y}|\mathbf{x}}(\mathbf{y}|\mathbf{x}) = f_{\mathbf{w}}(\mathbf{y} - \mathbf{A}\mathbf{x}) = \prod_{m=1}^{M} f_{\mathsf{w}}\left(y_m - \mathbf{A}_{m,:}\mathbf{x}\right). \tag{2.4}$$

Bayesian estimators of $\mathbf{x}$ rely on the *posterior*

$$f_{\mathbf{x}|\mathbf{y}}(\mathbf{x}|\mathbf{y}) = \frac{1}{f_{\mathbf{y}}(\mathbf{y})} f_{\mathbf{y}|\mathbf{x}}(\mathbf{y}|\mathbf{x}) f_{\mathbf{x}}(\mathbf{x}). \tag{2.5}$$

This is the probabilistic setup that is exploited to infer $\mathbf{x}$ from $\mathbf{y}$. For statistical inference with message passing techniques, the multivariate distribution (2.5) is often factorized into M factors:

$$f_{\mathbf{x}|\mathbf{y}}(\mathbf{x}|\mathbf{y}) = \frac{1}{f_{\mathbf{y}}(\mathbf{y})} \prod_{m=1}^{M} f_{\mathsf{y}_m|\mathbf{x}}(y_m|\mathbf{x}) f_{\mathbf{x}}(\mathbf{x}). \tag{2.6}$$
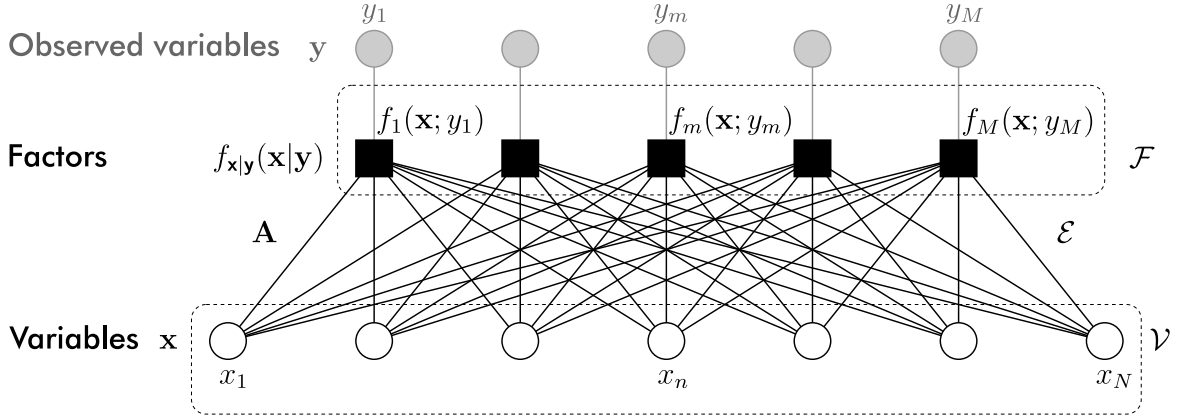
Figure 2.1 Factor graph of measurement (1.19) associated with posterior (2.6).

This factorization is described and visualized by a *factor graph* [79, 80]. The factor graph associated with (2.6) is depicted in Figure 2.1. It consists of the variable nodes $\mathcal{V} = \{x_1, ..., x_N\}$ that encompass the signal of interest $\mathbf{x}$, the factor nodes $\mathcal{F} = \{f_1, ..., f_M\}$ associated with (2.6), and the edges $\mathcal{E} = \{(f_m, x_n) : f_m \in \mathcal{F}, x_n \in \mathcal{V}\}$ that correspond to the relations between $\mathbf{x}$ and $\mathbf{y}$ which are dictated by the entries of $\mathbf{A}$, i.e., a nonzero entry $a_{m,n} \neq 0$ yields an edge between f_m and x_n. The observed (given or measured) variables are hinted in gray.

The MMSE estimator for $\mathbf{x}$ is formulated as [53]

$$\widehat{\mathbf{x}}_{\text{MMSE}}(\mathbf{y}) = \mathbb{E}_{\mathbf{x}}\left[\mathbf{x}|\mathbf{y} = \mathbf{y}\right] = \int_{\mathbb{R}^N} \widetilde{\mathbf{x}}\, f_{\mathbf{x}|\mathbf{y}}(\widetilde{\mathbf{x}}|\mathbf{y})\, d\widetilde{\mathbf{x}}, \tag{2.7}$$

it involves the computation of a high-dimensional integral that is infeasible to solve in practice. It can be approximately[1] solved by employing message passing (belief propagation) and the sum-product algorithm [79–81] on the factor graph. However, a computationally efficient method is only obtained after a series of assumptions and approximations –described in [68]– that yield the AMP algorithm.

---

[1]The considered graph contains cycles which lead to loopy belief propagation that yields an approximate result.

2.2 The AMP Algorithm

AMP was introduced in [65–68] to efficiently solve basis pursuit denoising (1.22), or equivalently, the LASSO problem (1.23). AMP features the low complexity of iterative thresholding algorithms [64] and attains the recovery performance of basis pursuit, see [65].

In literature [66, 68, 69], the AMP derivation begins with the assumption of a posterior distribution that is associated with the factors in Figure 2.1. Considering the basis pursuit denoising case (1.22) that is based on the noisy CS measurement (1.19), the posterior (2.6) is assumed to take the following form:

$$
\begin{aligned}
f_{\mathsf{x}|\mathbf{y}}(\mathbf{x}|\mathbf{y}) &= \frac{1}{f_{\mathbf{y}}(\mathbf{y})} \prod_{m=1}^{M} f_{\mathsf{y}_m|\mathsf{x}}(y_m|\mathbf{x}) \prod_{n=1}^{N} f_{\mathsf{x}_n}(x_n) \\
&= \frac{1}{f_{\mathbf{y}}(\mathbf{y})} \prod_{m=1}^{M} \frac{1}{\sqrt{2\pi\sigma_{\mathsf{w}}^2}} \exp\left(-\frac{(y_m - (\mathbf{Ax})_m)^2}{2\sigma_{\mathsf{w}}^2}\right) \prod_{n=1}^{N} \frac{1}{2\kappa} \exp\left(-\frac{|x_n|}{\kappa}\right),
\end{aligned}
\tag{2.8}
$$

i.e., it features i.i.d. zero-mean Laplace signal entries. This constitutes a Bayesian approach to the LASSO, which is obtained from (2.8) by computing the Maximum Aposteriori Probability (MAP) estimator [53]

$$
\begin{aligned}
\widehat{\mathbf{x}}_{\mathrm{MAP}}(\mathbf{y}) &= \arg\max_{\widetilde{\mathbf{x}}} f_{\mathsf{x}|\mathbf{y}}(\widetilde{\mathbf{x}}|\mathbf{y}) \\
&= \arg\max_{\widetilde{\mathbf{x}}} \left\{ \frac{1}{f_{\mathbf{y}}(\mathbf{y})} \prod_{m=1}^{M} \frac{1}{\sqrt{2\pi\sigma_w^2}} \exp\left(-\frac{(y_m - (\mathbf{A}\widetilde{\mathbf{x}})_m)^2}{2\sigma_w^2}\right) \prod_{n=1}^{N} \frac{1}{2\kappa} \exp\left(-\frac{|\widetilde{x}_n|}{\kappa}\right) \right\} \\
&= \arg\max_{\widetilde{\mathbf{x}}} \left\{ \frac{1}{Z(\mathbf{y})} \exp\left(-\frac{1}{2\sigma_w^2} \sum_{m=1}^{M} (y_m - (\mathbf{A}\widetilde{\mathbf{x}})_m)^2 - \frac{1}{\kappa} \sum_{n=1}^{N} |\widetilde{x}_n|\right) \right\} \\
&= \arg\max_{\widetilde{\mathbf{x}}} \left\{ \frac{1}{Z(\mathbf{y})} \exp\left[-\left(\frac{1}{2\sigma_w^2} \|\mathbf{y} - \mathbf{A}\widetilde{\mathbf{x}}\|_2^2 + \frac{1}{\kappa} \|\widetilde{\mathbf{x}}\|_1\right)\right] \right\} \\
&= \arg\min_{\widetilde{\mathbf{x}}} \left\{ \frac{1}{2\sigma_w^2} \|\mathbf{y} - \mathbf{A}\widetilde{\mathbf{x}}\|_2^2 + \frac{1}{\kappa} \|\widetilde{\mathbf{x}}\|_1 \right\} \\
&= \arg\min_{\widetilde{\mathbf{x}}} \left\{ \frac{1}{2} \|\mathbf{y} - \mathbf{A}\widetilde{\mathbf{x}}\|_2^2 + \lambda \|\widetilde{\mathbf{x}}\|_1 \right\}.
\end{aligned}
\tag{2.9}
$$

In [69], it is explained how the MAP estimator is justified as an alternative to the MMSE estimator, and under which circumstances their solutions coincide. The zero-mean Laplace distribution constitutes a sparsity enforcing prior, i.e., its probability

mass is concentrated around zero for small κ. This is an intuitive explanation for its usage and relation to the LASSO problem.

The Lagrangian parameter λ in the LASSO allows for a trade-off between accuracy with respect to the ℓ_2 observation error $\|\mathbf{y} - \mathbf{A}\widetilde{\mathbf{x}}\|_2$ and the sparsity $\|\widetilde{\mathbf{x}}\|_1$ of the solution. An increase in λ promotes a sparser solution. In AMP, λ is a design parameter. For the proper choice of λ, it was shown in [82, 83] that the fixed point of the AMP solution coincides with the LASSO solution.

For a detailed derivation of AMP that includes several additional steps, the interested reader is referred to [65–68]. In this thesis, I focus on the algorithmic implementation.

2.2.1   Algorithmic Implementation

My implementation of AMP is stated by Algorithm 1, it is based on the implementation in [71]. The algorithm is iterative with iteration index t. Its usage as a superscript indicates the state of the involved variables, i.e., $\mathbf{x}^t$ refers to the signal estimate in the t-th iteration. Note that the algorithm requires $\mathbf{A}$ to have normalized columns, i.e., $\|\mathbf{a}_n\|_2 = 1, \forall n \in \{1, ..., N\}$. Let me discuss the key aspects of the algorithm, starting with soft thresholding.

Revisiting the LASSO problem in the scalar case, i.e.,

$$\widehat{x}_{\text{LASSO}}(u; \lambda) = \arg\min_{\widetilde{x}} \left\{ \frac{1}{2}(u - \widetilde{x})^2 + \lambda|\widetilde{x}| \right\}, \tag{2.10}$$

it is known that the *soft thresholding* function

$$\eta(u; \tau) = \begin{cases} u + \tau & \text{if } u < -\tau \\ 0 & \text{if } -\tau \leq u \leq \tau \\ u - \tau & \text{if } u > \tau \end{cases} \tag{2.11}$$

admits a (possibly optimal) solution to (2.10), see [55]. The soft thresholding function acts as a *denoiser*, i.e., it sets values below a certain threshold τ to zero. This function is also found in line 6 of AMP Algorithm 1, where it is applied entry-wise on the decoupled measurements $\mathbf{u}$, i.e., on $u_n = x_n + \mathbf{a}_n^{\mathrm{T}}\mathbf{r}$. The approximate LASSO solution emerges over several iterations of the algorithm. Note that the LASSO – and hence AMP– does not attain MMSE estimation performance, because it does not take into account the true signal prior distribution, but rather a uniformly good distribution to enforce sparsity [69]. In many cases, the signal prior is not available,

---
Algorithm 1 AMP

1: Initialize $\mathbf{x}^t = \mathbf{0}_{N \times 1}$ and $\mathbf{r}^t = \mathbf{y}$ at $t = 0$
2: **do**
3:      $t \leftarrow t + 1$ ▷ Advance iterations
4:      $\mathbf{u}^{t-1} = \mathbf{x}^{t-1} + \mathbf{A}^{\mathrm{T}} \mathbf{r}^{t-1}$ ▷ Compute decoupled measurements
5:      $\tau^{t-1} = \frac{\lambda}{\sqrt{M}} \|\mathbf{r}^{t-1}\|_2$ ▷ Compute threshold
6:      $\mathbf{x}^t = \eta(\mathbf{u}^{t-1}; \tau^{t-1})$ ▷ Soft thresholding
7:      $\mathbf{r}^t = \mathbf{y} - \mathbf{A}\mathbf{x}^t + \mathbf{r}^{t-1} \frac{1}{M} \|\mathbf{x}^t\|_0$ ▷ Compute residual
8: **while** $\|\mathbf{x}^t - \mathbf{x}^{t-1}\|_2 > \epsilon_{\mathrm{tol}} \|\mathbf{x}^{t-1}\|_2$ and $t < t_{\max}$ ▷ Check stopping criteria
9: return $\widehat{\mathbf{x}} = \mathbf{x}^t$ ▷ Recovered vector

---

and the AMP algorithm constitutes a computationally efficient tool that achieves the performance of basis pursuit, if tuned optimally as described in Section 2.2.3.

Another crucial feature is the addition of the "Onsager term" [69] $\mathbf{r}^{t-1} \frac{1}{M} \|\mathbf{x}^t\|_0$ in the residual computation in line 7. This is the key difference between iterative thresholding algorithms [64] and AMP, and it renders the decoupled measurements Gaussian distributed with mean x_n [69]. The soft thresholder that acts as a denoiser reduces the estimation noise variance over iterations.

The iterations are stopped once the change in the estimated signal is below a certain threshold –controlled by ϵ_{tol}– or the maximal number of iterations $t_{\max}$ is reached. Throughout this work, I choose $\epsilon_{\mathrm{tol}} = 10^{-5}$ in all simulations.

2.2.2   The Complex-valued Case

So far, real-valued signals and measurements were considered. AMP Algorithm 1 can easily be extended to the complex-valued case by replacing the soft thresholding function (2.11) with the *complex soft thresholding* function [84]

$$\eta(u; \tau) = \left(u - \tau \frac{u}{|u|} \right) \cdot \mathbb{I}_{\{|u|^2 > \tau^2\}}, \tag{2.12}$$

with indicator function

$$\mathbb{I}_{\{|u|^2 > \tau^2\}} = \begin{cases} 1 & |u|^2 > \tau^2 \\ 0 & \text{else} \end{cases}. \tag{2.13}$$

This case is important for the deployment of AMP in CS-RFID, where complex-valued signals are encountered. Equipped with (2.12), AMP solves the complex LASSO, see [84].

2.2.3 Optimal Tuning

The AMP Algorithm 1 has a tuning parameter λ that relates to the LASSO (1.23) — this parameter steers the solution sparsity. However, while an increasing λ entails sparser solutions, the accuracy with respect to the ℓ_2 reconstruction error decreases. An optimal tuning prescription that enables AMP to *achieve basis pursuit performance* was presented in [65], it requires knowledge of K. Applying those findings to Algorithm 1, the ℓ_1-optimal parameter reads

$$\lambda_{\ell_1} = \arg\max_{z \geq 0} \underbrace{\frac{1 - \zeta/\delta \left[(1 + z^2) \cdot \int_z^\infty \mathcal{N}(z'; 0, 1)\, dz' - z \cdot \mathcal{N}(z; 0, 1)\right]}{1 + z^2 - \zeta \left[(1 + z^2) \cdot \int_z^\infty \mathcal{N}(z'; 0, 1)\, dz' - z \cdot \mathcal{N}(z; 0, 1)\right]}}_{\rho(z, \delta, \zeta)}, \quad (2.14)$$

where $\delta = \frac{M}{N}$, and $\zeta = 1$ for $x_n \geq 0$, $\zeta = 2$ for $|x_n| \geq 0$. The K-dependency is obtained as follows: λ_{ℓ_1} and the corresponding $\rho(\lambda_{\ell_1}, \delta, \zeta)$ from (2.14) are computed for a wide range of M at fixed N. It holds that $\rho = \frac{K}{M}$ such that $K = \rho(\lambda_{\ell_1}, \delta, \zeta) \cdot M$.

The optimal tuning prescription for $N = 1\,000$ is plotted in Figure 2.2. The left plot shows λ_{ℓ_1} for the full K-range. For sparse signals with $K \ll N$, the LASSO emphasizes the solution sparsity by choosing a large λ. For dense signals, the least squares solution (1.20) is approached. The right plot depicts the CS-relevant sparse case in more detail.
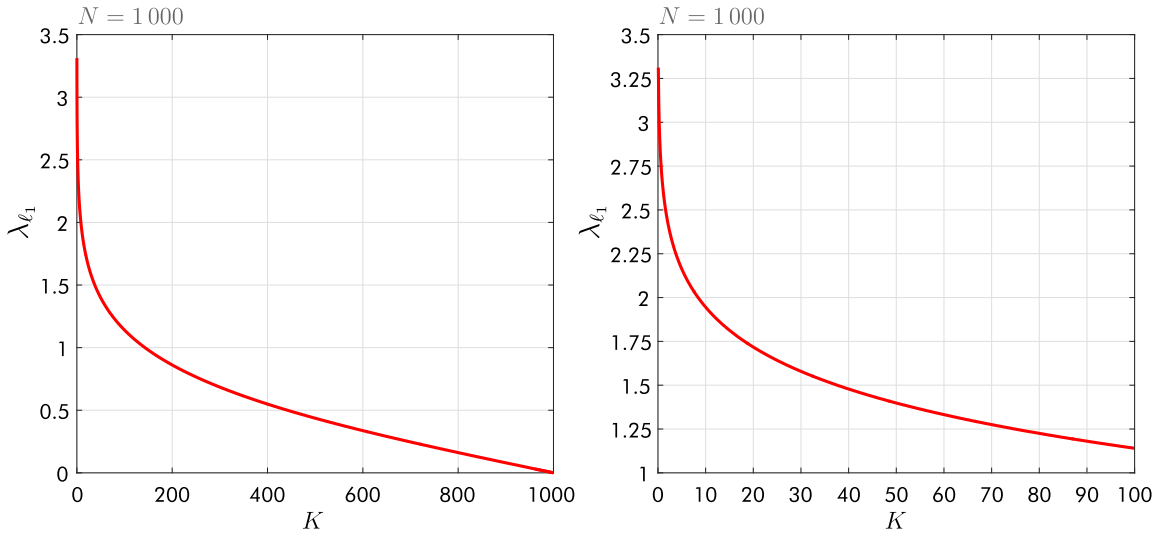
Figure 2.2 ℓ_1-optimal AMP tuning parameter λ depending on K.

The next task is to determine how many measurements M are required to successfully recover a signal vector from (1.19). To that end, the dependency of the
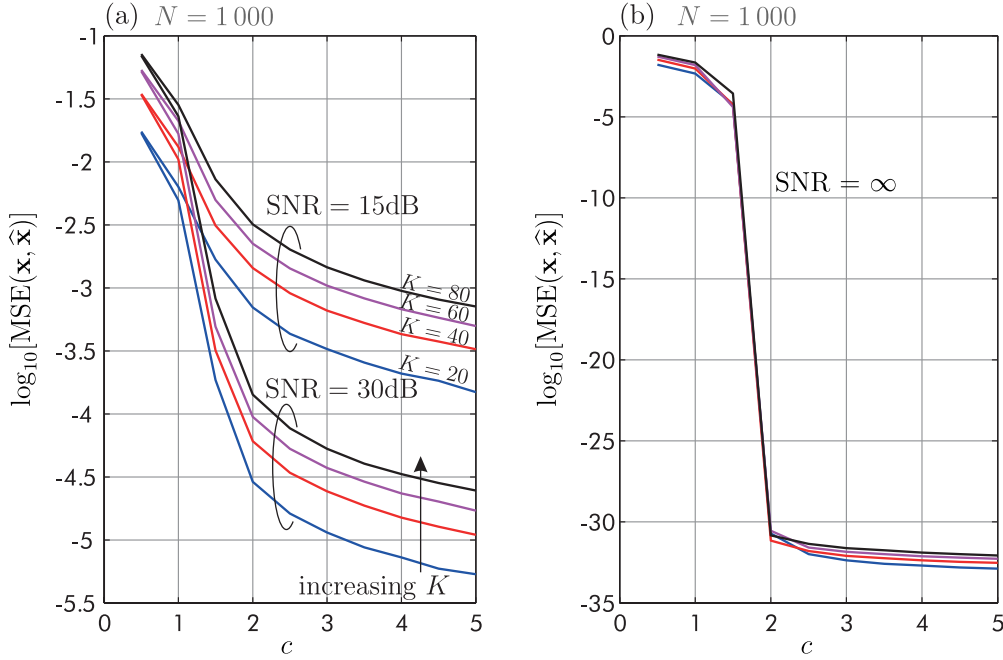
Figure 2.3 $\mathrm{MSE}(\mathbf{x}, \widehat{\mathbf{x}})$ versus measurement multiplier c.

recovery MSE ($\mathrm{MSE}(\mathbf{x}, \widehat{\mathbf{x}})$) on the measurement multiplier c in (1.26) is investigated by simulation. The simulation campaign encompasses $1\,000$ independent random realizations of $\mathbf{A}$, $\mathbf{x}$ and $\mathbf{w}$. Sensing matrix $\mathbf{A}$ features equally likely entries drawn from $\{-\frac{1}{\sqrt{M}}, \frac{1}{\sqrt{M}}\}$ as suggested in Section 1.2.3. The K nonzero entries of signal vector $\mathbf{x}$ are i.i.d. Gaussian with zero mean and unit variance. The noise $\mathbf{w}$ is Gaussian distributed with zero mean and covariance $\sigma_{\mathsf{w}}^2 \mathbf{I}_N$. The noise variance σ_{w}^2 is selected according to the Signal-to-Noise Ratio (SNR) definition

$$\mathrm{SNR} := \frac{\|\mathbf{A}\mathbf{x}\|_2^2}{\mathbb{E}_{\mathbf{w}}\{\|\mathbf{w}\|_2^2\}} = \frac{\|\mathbf{A}\mathbf{x}\|_2^2}{M\sigma_{\mathsf{w}}^2}. \tag{2.15}$$

Figure 2.3 (a) depicts the average MSE versus c, for two instances of SNR. The MSE decreases for an increasing number of measurements. Increasing the number of nonzero entries K in $\mathbf{x}$ increases the MSE, or put differently, more measurements are required to attain the same MSE performance. In the noiseless case depicted in Figure 2.3 (b), a clear transition to a (numerically) perfect recovery is observed at $c = 2$, regardless of K. Takeaway point: In the considered scenario, $c = 2$ in (1.26) results in perfect recovery. In the noisy case, a larger c may be beneficial. These empirical findings are corroborated by [52].

2.3 The BAMP Algorithm

As discussed in [69, 78], AMP algorithms *decouple* the estimation problem associated with CS measurement (1.19) into N uncoupled scalar problems in the asymptotic regime:

Definition 2.3.1 (Asymptotic regime)**.** *The asymptotic regime is defined as*

$$M \to \infty \ \ and \ \ N \to \infty \ \ while \ \ \delta = \frac{M}{N} = \text{constant}. \tag{2.16}$$

Definition 2.3.2 (Decoupling principle)**.** *In the asymptotic regime, the CS measurement (1.19) can be interpreted as N uncoupled (independent) scalar measurements*

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{w} \Leftrightarrow \begin{cases} u_1 = x_1 + \widetilde{w}_1 \\ \vdots \\ u_N = x_N + \widetilde{w}_N \end{cases}, \tag{2.17}$$

where the effective noise *asymptotically obeys* $\widetilde{\mathsf{w}}_n \sim \mathcal{N}(0, \beta)$.

The effective noise accounts for two noise phenomena:

- The *measurement noise* w_m, $m \in \{1, ..., M\}$, is added to the measurements y_m according to (1.19). It is assumed to be Gaussian with zero mean and variance σ_{w}^2.

- The estimation of signal entry x_n depends on the estimation of the other entries x_l, $l \neq n$, which is rooted in the message passing nature of AMP [68]. Many small estimation errors in x_l, $l \neq n$, contribute to the estimation error in x_n. This is referred to as *interference* [77]. The interference noise becomes zero-mean Gaussian in the asymptotic regime.

Consequently, the effective noise $\widetilde{\mathsf{w}}_n$ is approximately Gaussian, and its variance β accounts for the measurement noise and the estimation interference. The interpretation of (1.19) as N uncoupled scalar measurements is a key observation that enables a significantly simplified estimation of $\mathbf{x}$ from $\mathbf{u} = [u_1, ..., u_N]^\mathrm{T}$ by utilizing the decoupled posteriors

$$f_{\mathsf{x}_n|\mathsf{u}_n}(x_n|u_n) = \frac{1}{f_{\mathsf{u}_n}(u_n)} f_{\mathsf{u}_n|\mathsf{x}_n}(u_n|x_n) f_{\mathsf{x}_n}(x_n), \tag{2.18}$$
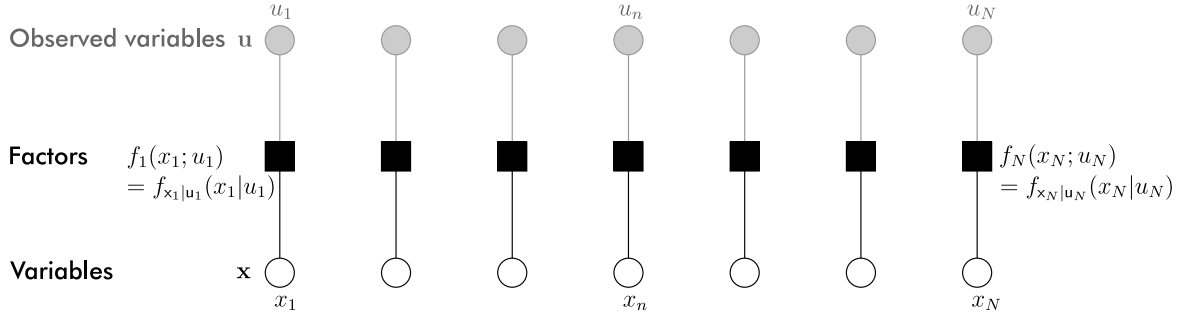
Figure 2.4 Factor graph of decoupled measurements (2.17) associated with (2.18).

where

$$f_{\mathsf{u}_n|\mathsf{x}_n}(u_n|x_n) = f_{\widetilde{\mathsf{w}}}(u_n - x_n) = \mathcal{N}(u_n; x_n, \beta), \qquad (2.19)$$

$$f_{\mathsf{u}_n}(u_n) = \int_{-\infty}^{\infty} f_{\mathsf{u}_n|\mathsf{x}_n}(u_n|\widetilde{x}_n) f_{\mathsf{x}_n}(\widetilde{x}_n) d\widetilde{x}_n. \qquad (2.20)$$

The N uncoupled estimation problems are visualized by the factor graph in Figure 2.4. The graph illustrates the estimation process that occurs in a single BAMP iteration, based on the decoupled measurements in $\mathbf{u}$ that are computed in each BAMP iteration.

BAMP [65–68] leverages prior knowledge in form of the signal prior distribution to perform MMSE estimation based on the decoupled measurements (2.17). Instead of soft thresholding (2.11), BAMP utilizes the following function as denoiser:

$$\begin{aligned}
F(u_n; \beta) &= \mathbb{E}_{\mathsf{x}_n}\{\mathsf{x}_n|\mathsf{u}_n = u_n; \beta\} \\
&= \int_{-\infty}^{\infty} f_{\mathsf{x}_n|\mathsf{u}_n}(\widetilde{x}_n|u_n)\, d\widetilde{x}_n \\
&= \frac{\int_{-\infty}^{\infty} \widetilde{x}_n \cdot f_{\mathsf{u}_n|\mathsf{x}_n}(u_n|\widetilde{x}_n) \cdot f_{\mathsf{x}_n}(\widetilde{x}_n)\, d\widetilde{x}_n}{\int_{-\infty}^{\infty} f_{\mathsf{u}_n|\mathsf{x}_n}(u_n|\widetilde{x}_n) \cdot f_{\mathsf{x}_n}(\widetilde{x}_n)\, d\widetilde{x}_n} \\
&= \frac{\int_{-\infty}^{\infty} \widetilde{x}_n \cdot \mathcal{N}(u_n; \widetilde{x}_n, \beta) \cdot f_{\mathsf{x}_n}(\widetilde{x}_n)\, d\widetilde{x}_n}{\int_{-\infty}^{\infty} \mathcal{N}(u_n; \widetilde{x}_n, \beta) \cdot f_{\mathsf{x}_n}(\widetilde{x}_n)\, d\widetilde{x}_n}.
\end{aligned} \qquad (2.21)$$

This conditional expectation yields the MMSE estimate of x_n given the decoupled measurement u_n. Its derivative will also be encountered in the algorithm:

$$F'(u_n; \beta) = \frac{d}{du_n} F(u_n; \beta). \qquad (2.22)$$

These two functions have to be calculated for a given signal prior. However, not all priors admit closed form expressions. I will later specify (2.21) and (2.22) for a Bernoulli-Gaussian prior and a Bernoulli-Laplace prior. These priors are relevant to my application case. Moreover, they admit closed form expressions that facilitate a proper algorithmic implementation of BAMP.

2.3.1 Algorithmic Implementation

My implementation of BAMP is stated by Algorithm 2. The algorithm is iterative with iteration index t. Note that $\mathbf{A}$ is assumed to have normalized columns, i.e., $\|\mathbf{a}_n\|_2 = 1$, $\forall n \in \{1, ..., N\}$. After initialization, the decoupled measurements (2.17) are computed in line 4:

$$\mathbf{u}^{t-1} = \mathbf{x}^{t-1} + \mathbf{A}^{\mathrm{T}} \mathbf{r}^{t-1}. \tag{2.23}$$

Considering the individual entries of $\mathbf{u}^{t-1}$,

$$u_n^{t-1} = x_n^{t-1} + \underbrace{\mathbf{a}_n^{\mathrm{T}} \mathbf{r}^{t-1}}_{\widetilde{w}_n^{t-1}}, \tag{2.24}$$

the effective noise $\widetilde{w}_n^{t-1}$ that corrupts the decoupled measurements is identified as the inner product of the n-th column of $\mathbf{A}$ with the residual vector $\mathbf{r}^{t-1}$. The effective noise variance β^{t-1} is estimated in line 5. In line 6, the decoupled measurements are denoised by the MMSE estimator (2.21), it is applied entry-wise on $\mathbf{u}^{t-1}$. In line 7, the residual is computed; an important feature of BAMP is the "Onsager term"

$$\mathbf{r}^{t-1} \frac{1}{M} \sum_{n=1}^{N} F'(u_n^{t-1}; \beta^{t-1}), \tag{2.25}$$

it is derived in, e.g., [69], and its inclusion in the residual computation distinguishes AMP-based schemes from iterative thresholding algorithms [64]. As in AMP Algorithm 1, it renders u_n Gaussian distributed with mean x_n and variance β.

The algorithm runs until the change between two subsequent signal estimates becomes smaller than a certain value that is controlled by threshold ϵ_{tol}, or until the maximum number of iterations $t_{\max}$ is reached.

The crucial difference to AMP Algorithm 1 is the exploitation of prior knowledge in form of the signal prior distribution, if available. This awards BAMP with (approximate) MMSE estimation performance that significantly outperforms the basis pursuit performance of optimally tuned AMP [69]. Note that the algorithm uses the

---

Algorithm 2 BAMP

1: Initialize $\mathbf{x}^t = \mathbf{0}_{N\times 1}$ and $\mathbf{r}^t = \mathbf{y}$ at $t = 0$
2: **do**
3:    $t \leftarrow t + 1$ ▷ Advance iterations
4:    $\mathbf{u}^{t-1} = \mathbf{x}^{t-1} + \mathbf{A}^{\mathrm{T}}\mathbf{r}^{t-1}$ ▷ Compute decoupled measurements
5:    $\beta^{t-1} = \frac{1}{M}\|\mathbf{r}^{t-1}\|_2^2$ ▷ Estimate effective noise variance
6:    $\mathbf{x}^t = F(\mathbf{u}^{t-1}; \beta^{t-1})$ ▷ Estimate signal
7:    $\mathbf{r}^t = \mathbf{y} - \mathbf{A}\mathbf{x}^t + \mathbf{r}^{t-1}\frac{1}{M}\sum_{n=1}^{N} F'(u_n^{t-1}; \beta^{t-1})$ ▷ Compute residual
8: **while** $\|\mathbf{x}^t - \mathbf{x}^{t-1}\|_2 > \epsilon_{\mathrm{tol}}\|\mathbf{x}^{t-1}\|_2$ and $t < t_{\max}$ ▷ Check stopping criteria
9: return $\widehat{\mathbf{x}} = \mathbf{x}^t$ ▷ Recovered vector

---

factor graph model from Figure 2.4 in each iteration t to perform iterative MMSE estimation; the "observed" $\mathbf{u}^{t-1}$ that is computed in line 4 is leveraged to compute the signal estimate $\mathbf{x}^t$. The variables are updated in each iteration.

2.3.2   Specification for Relevant Priors

In this section, I specify the MMSE estimator function (2.21) and its derivative (2.22) for two prior distributions that are relevant to CS-RFID.

The **Bernoulli-Gaussian prior** features a PDF

$$f_{\mathsf{x}_n}(x_n) = \gamma_n \delta(x_n) + (1 - \gamma_n)\mathcal{N}(x_n; 0, \sigma_{\mathsf{x}_n}^2), \tag{2.26}$$

i.e., the nonzero entry distribution of the prior (1.14) is Gaussian: $f_{a_n}(x_n) = \mathcal{N}(x_n; 0, \sigma_{\mathsf{x}_n}^2)$. The MMSE estimator function calculates as

$$F(u_n; \beta, \gamma_n) = u_n \cdot \underbrace{\frac{\sigma_{\mathsf{x}_n}^2}{\beta + \sigma_{\mathsf{x}_n}^2}}_{\substack{\text{Gaussian linear} \\ \text{MMSE estimator}}} \cdot \underbrace{\frac{(1 - \gamma_n)\mathcal{N}(u_n; 0, \beta + \sigma_{\mathsf{x}_n}^2)}{\gamma_n \mathcal{N}(u_n; 0, \beta) + (1 - \gamma_n)\mathcal{N}(u_n; 0, \beta + \sigma_{\mathsf{x}_n}^2)}}_{\text{Nonlinear term due to zero component}}$$

$$= u_n \cdot \frac{\sigma_{\mathsf{x}_n}^2}{\beta + \sigma_{\mathsf{x}_n}^2} \cdot \frac{1}{1 + \underbrace{\frac{\gamma_n}{1 - \gamma_n}\sqrt{\frac{\beta + \sigma_{\mathsf{x}_n}^2}{\beta}}\exp\left(-\frac{u_n^2}{2}\frac{\sigma_{\mathsf{x}_n}^2}{\beta(\beta + \sigma_{\mathsf{x}_n}^2)}\right)}_{m(u_n; \beta, \gamma_n)}}, \tag{2.27}$$

its derivative as

$$F'(u_n; \beta, \gamma_n) = \frac{\sigma_{\mathsf{x}_n}^2}{\beta_n + \sigma_{\mathsf{x}_n}^2} \cdot \frac{1}{1 + m(u_n; \beta, \gamma_n)} + \frac{1}{\beta} \cdot m(u_n; \beta, \gamma_n) \cdot F(u_n; \beta, \gamma_n)^2. \tag{2.28}$$

The detailed calculation is provided in Appendix A.1 and Appendix A.2.

The PDF of the **Bernoulli-Laplace prior** reads

$$f_{\mathsf{x}_n}(x_n) = \gamma_n \delta(x_n) + (1 - \gamma_n)\mathcal{L}(x_n; 0, \kappa_n), \tag{2.29}$$

it employs a Laplace distribution with zero mean and variance $2\kappa_n^2$ as the nonzero entry distribution in (1.14): $f_{a_n}(x_n) = \mathcal{L}(x_n; 0, \kappa_n)$. The MMSE estimator function calculates as

$$F(u_n; \beta, \gamma_n) = \frac{\beta \left[h_1(u_n)k_1(u_n) + h_2(u_n)k_2(u_n)\right]}{\frac{\gamma_n}{1-\gamma_n}\frac{4\kappa_n}{\sqrt{2\pi\beta}} + k_1(u_n) + k_2(u_n)} = \frac{p(u_n)}{q(u_n)}, \tag{2.30}$$

with auxiliary functions

$$g_1(u) = \frac{\sqrt{2\beta}}{2\kappa} - \frac{u}{\sqrt{2\beta}}, \tag{2.31}$$

$$g_2(u) = \frac{\sqrt{2\beta}}{2\kappa} + \frac{u}{\sqrt{2\beta}}, \tag{2.32}$$

$$h_1(u) = \frac{u}{\beta} - \frac{1}{\kappa}, \tag{2.33}$$

$$h_2(u) = \frac{u}{\beta} + \frac{1}{\kappa}, \tag{2.34}$$

$$k_1(u) = \mathrm{erfc}\left(g_1(u)\right)\exp\left(g_1(u)^2\right), \tag{2.35}$$

$$k_2(u) = \mathrm{erfc}\left(g_2(u)\right)\exp\left(g_2(u)^2\right), \tag{2.36}$$

the derivative of (2.30) with respect to u_n calculates as

$$F'(u_n; \beta, \gamma_n) = \frac{p'(u_n)q(u_n) - p(u_n)q'(u_n)}{q(u_n)^2}, \tag{2.37}$$

where

$$\begin{aligned} p'(u) = {}& k_1(u) + k_2(u) \\ & + \sqrt{2\beta}\left[h_2(u_n)\left(g_2(u)k_2(u) - \frac{1}{\sqrt{\pi}}\right) - h_1(u_n)\left(g_1(u)k_1(u) - \frac{1}{\sqrt{\pi}}\right)\right], \end{aligned} \tag{2.38}$$

$$q'(u) = \sqrt{\frac{2}{\beta}}\left[g_2(u)k_2(u) - g_1(u)k_1(u)\right]. \tag{2.39}$$

The detailed calculation of (2.30) is provided in Appendix B.1, (2.37) follows from the quotient rule.

2.4 The BASSAMP Algorithm

I will now introduce the BASSAMP algorithm that extends BAMP in order to leverage *joint sparsity*.

Definition 2.4.1 (Joint sparsity). *B sparse signal vectors $\mathbf{x}_b \in \mathcal{X}_K$ with support $\mathcal{S}_{\mathbf{x}_b}$, $b \in \mathcal{B} = \{1, ..., B\}$, exhibit joint sparsity if they have the same support*

$$\mathcal{S}_{\mathbf{x}} \equiv \mathcal{S}_{\mathbf{x}_b}, \forall b \in \mathcal{B}. \tag{2.40}$$

Having B signal vectors, the CS measurement (1.19) is generally extended to

$$\mathbf{y}_b = \mathbf{A}^{(b)} \mathbf{x}_b + \mathbf{w}_b, \tag{2.41}$$

where $\mathbf{y}_b \in \mathbb{R}^M$, $\mathbf{A}^{(b)} \in \mathbb{R}^{M \times N}$ and $\mathbf{w}_b \in \mathbb{R}^M$. The vectors are collected in matrices: $\mathbf{Y} = [\mathbf{y}_1, ..., \mathbf{y}_b, ..., \mathbf{y}_B]$, $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_b, ..., \mathbf{x}_B]$ and $\mathbf{W} = [\mathbf{w}_1, ..., \mathbf{w}_b, ..., \mathbf{w}_B]$. If all B sensing matrices are identical, i.e., $\mathbf{A} \equiv \mathbf{A}^{(b)}, \forall b \in \mathcal{B}$, (2.41) can be rewritten to comprise all vectors:

$$\mathbf{Y} = \mathbf{A}\mathbf{X} + \mathbf{W}. \tag{2.42}$$

A prominent instance of joint sparsity is the multiple measurement vector problem [85, 86]. Typical applications of joint sparsity are MRI [86–88] and direction-of-arrival estimation [89]. In Chapter 4, I demonstrate that joint sparsity is also present in CS-RFID, and that its exploitation strongly improves the tag acquisition process. Methods and guarantees to recover jointly sparse vectors are discussed in [90–94]. My approach combines BAMP with iterative turbo decoding [95–97] in order to leverage the joint sparsity structure. This idea was firstly proposed in [98], where extrinsic information is passed among signal entries that are jointly active (nonzero) or inactive (zero). Based on those insights, I first derive the computationally efficient and approximately MMSE attaining BASSAMP algorithm that exploits Bayesian prior knowledge and the joint sparsity structure. Later, I propose to replace the scalar denoiser function (2.21) of BAMP by a vector denoiser function; this vector denoiser acts on all jointly sparse entries, and I demonstrate that this renders the extrinsic information exchange obsolete, resulting in faster algorithmic convergence.

This section deals with B jointly sparse signal vectors that are collected in a signal matrix $\mathbf{X}$. Similarly, I collect the decoupled measurements in $\mathbf{U} = [\mathbf{u}_1, ..., \mathbf{u}_B]$, the zero probabilities in $\boldsymbol{\Gamma} = [\boldsymbol{\gamma}_1, ..., \boldsymbol{\gamma}_B]$, and the L-values[2] in $\mathbf{L} = [\mathbf{L}_1, ..., \mathbf{L}_B]$. The n-th entry of the b-th column is respectively denoted by $(\mathbf{X})_{n,b} = x_{n,b}$, $(\mathbf{U})_{n,b} = u_{n,b}$, $(\boldsymbol{\Gamma})_{n,b} = \gamma_{n,b}$, and $(\mathbf{L})_{n,b} = L_{n,b}$. Moreover, the effective noise variance is generally different for each signal vector, the variances are collected in a vector $\boldsymbol{\beta} = [\beta_1, ..., \beta_B]^{\mathrm{T}}$.

2.4.1 Algorithmic Implementation for Joint Sparsity

The BASSAMP algorithm for signals with joint sparsity is depicted in Algorithm 3. The key novelties over BAMP are the introduction of the *joint extrinsic update* step and the subsequent *prior update* step. These update functions exploit the joint sparsity structure, i.e., that there are only two cases for the entries of a row in $\mathbf{X}$: Either all entries are zero, or all entries are nonzero. Let me first discuss the key ideas behind the algorithm, and then derive the update functions.

In each iteration t of Algorithm 3, a BAMP iteration is executed independently for all B jointly sparse signals, yielding $u_{n,b}^{t-1}$ and β_b^{t-1}, $\forall b \in \mathcal{B}$. This information is then leveraged to adapt the zero probabilities $\gamma_{n,b}^{t-1}$ for the subsequent BAMP iteration. Note that $\gamma_{n,b}^{t-1}$ is used as prior information in BAMP, it is an input of the MMSE estimator $F(u_{n,b}^{t-1}; \beta_b^{t-1}, \gamma_{n,b}^{t-1})$ and its derivative $F'(u_{n,b}^{t-1}; \beta_b^{t-1}, \gamma_{n,b}^{t-1})$. The zero probabilities are updated by two update functions:

1. The *joint extrinsic update* enforces the joint sparsity structure by accumulating extrinsic information about the b-th vector from the other vectors $l \in \mathcal{B} \backslash b$. The information is based on a Bayesian estimation procedure that takes into account the decoupled measurements $u_{n,l}^{t-1}$ and the effective noise variances β_l^{t-1} that are updated in each iteration. The accumulated information is expressed by log-likelihood ratios $\overline{L}_{n,b}^t$.

2. The *prior update* converts the log-likelihood ratios into probabilities $\gamma_{n,b}^t$. These probabilities are used as a priori information in the subsequent BAMP iteration.

The estimation and adaptation of $\gamma_{n,b}^t$ in each iteration results in quicker algorithmic convergence over BAMP and significantly improved estimation of $\mathbf{X}$, i.e., lower recovery MSE. Furthermore, the information exchange among the vectors leads to a consensus, and the $\gamma_{n,b}^t$ approach one or zero, thereby clearly indicating whether a signal entry was zero or nonzero.

---

[2] L-values are log-likelihood ratios in the context of coding. They are typically used in soft-input channel decoding or in iterative decoding.

Algorithm 3 BASSAMP for signals with joint sparsity

1: Initialize $\mathbf{X}^t = \mathbf{0}_{N \times B}$, $\boldsymbol{\Gamma}^t = \mathbf{1}_{N \times B} - \frac{K}{N}$ and $\mathbf{r}_b^t = \mathbf{y}_b \; \forall b \in \mathcal{B} = \{1, ..., B\}$ at $t = 0$
2: **do**
3:      $t \leftarrow t + 1$
4:      **for** $b = 1$ to B **do**   ▷ BAMP iteration for all signal vectors
5:          $\mathbf{u}_b^{t-1} = \mathbf{x}_b^{t-1} + \mathbf{A}^{(b)\mathrm{T}} \mathbf{r}_b^{t-1}$ ▷ Compute decoupled measurements
6:          $\beta_b^{t-1} = \frac{1}{M} \|\mathbf{r}_b^{t-1}\|_2^2$ ▷ Estimate effective noise variance
7:          $\mathbf{x}_b^t = F(\mathbf{u}_b^{t-1}; \beta_b^{t-1}, \boldsymbol{\gamma}_b^{t-1})$ ▷ Estimate signal
8:          $\mathbf{r}_b^t = \mathbf{y}_b - \mathbf{A}^{(b)} \mathbf{x}_b^t + \mathbf{r}_b^{t-1} \frac{1}{M} \sum_n F'(u_{n,b}^{t-1}; \beta_b^{t-1}, \gamma_{n,b}^{t-1})$ ▷ Compute residual
9:      $\overline{\mathbf{L}}^t = U_J(\mathbf{U}^{t-1}, \boldsymbol{\beta}^{t-1}, \boldsymbol{\Gamma}^0)$ ▷ Joint extrinsic update
10:     $\boldsymbol{\Gamma}^t = U_P(\overline{\mathbf{L}}^t)$ ▷ Prior update
11: **while** $\|\mathbf{X}^t - \mathbf{X}^{t-1}\|_{\mathrm{F}} > \epsilon_{\mathrm{tol}} \|\mathbf{X}^{t-1}\|_{\mathrm{F}}$ and $t < t_{\max}$ ▷ Check stopping criteria
12: **return** $\widehat{\mathbf{X}} = \mathbf{X}^t$ ▷ Recovered vectors

---

Let me begin the algorithmic derivation with the decoupled measurements from (2.17), now extended to matrix indexing. The PDF of $\mathsf{u}_{n,b}$ is obtained by convolving the signal PDF (1.14) with the PDF of the effective noise $\widetilde{\mathsf{w}}_{n,b}$, a Gaussian distribution with zero mean and variance β_b:

$$
\begin{aligned}
f_{\mathsf{u}_{n,b}}(u_{n,b}) &= f_{\mathsf{x}_{n,b}}(u_{n,b}) * f_{\widetilde{\mathsf{w}}_{n,b}}(u_{n,b}) \\
&= \left[\gamma_{n,b} \delta(u_{n,b}) + (1 - \gamma_{n,b}) f_{a_{n,b}}(u_{n,b}) \right] * \mathcal{N}(u_{n,b}; 0, \beta_b) \\
&= \gamma_{n,b} \delta(u_{n,b}) * \mathcal{N}(u_{n,b}; 0, \beta_b) + (1 - \gamma_{n,b}) f_{a_{n,b}}(u_{n,b}) * \mathcal{N}(u_{n,b}; 0, \beta_b) \\
&= \gamma_{n,b} \underbrace{\mathcal{N}(u_{n,b}; 0, \beta_b)}_{f_0(u_{n,b})} + (1 - \gamma_{n,b}) \underbrace{f_{a_{n,b}}(u_{n,b}) * \mathcal{N}(u_{n,b}; 0, \beta_b)}_{f_1(u_{n,b})} \\
&= \gamma_{n,b} f_0(u_{n,b}) + (1 - \gamma_{n,b}) f_1(u_{n,b}).
\end{aligned}
\tag{2.43}
$$

Component $f_0(u_{n,b})$ is the *noise PDF*, component $f_1(u_{n,b})$ is the *signal plus noise PDF*. Inspired by the E-step of the Expectation Maximization (EM) algorithm [99–101], I introduce a *latent binary random variable* $\mathsf{z}_{n,b}$ with the following properties:

Definition 2.4.2 (Latent activity variable)**.** *The latent activity variable is a binary random variable* $\mathsf{z}_{n,b} \in \{0, 1\}$ *with Probability Mass Function (PMF)*

$$
p_{\mathsf{z}_{n,b}}(z_{n,b}) = \gamma_{n,b} \delta(z_{n,b}) + (1 - \gamma_{n,b}) \delta(1 - z_{n,b}).
\tag{2.44}
$$

With the aid of the latent activity variable, (2.43) is reformulated as

$$f_{\mathsf{u}_{n,b}}(u_{n,b}) = \sum_{z_{n,b} \in \{0,1\}} p_{\mathsf{z}_{n,b}}(z_{n,b}) f_{\mathsf{u}_{n,b}|\mathsf{z}_{n,b}}(u_{n,b}|z_{n,b}), \tag{2.45}$$

where

$$f_{\mathsf{u}_{n,b}|\mathsf{z}_{n,b}}(\mathsf{u}_{n,b} = u_{n,b}|\mathsf{z}_{n,b} = 0) = f_0(u_{n,b}), \tag{2.46}$$

$$f_{\mathsf{u}_{n,b}|\mathsf{z}_{n,b}}(\mathsf{u}_{n,b} = u_{n,b}|\mathsf{z}_{n,b} = 1) = f_1(u_{n,b}). \tag{2.47}$$

The goal is to estimate the zero probability $\gamma_{n,b}$ based on the decoupled measurement $u_{n,b}$ in each iteration t. This can be viewed as a classification problem in which I want to determine the probability that $u_{n,b}$ stems from the noise PDF $f_0(u_{n,b})$ rather than the signal plus noise PDF $f_1(u_{n,b})$, i.e., the probability that $z_{n,b} = 0$ rather than $z_{n,b} = 1$. The following statements are equivalent:

$$x_{n,b} \text{ is zero} \Leftrightarrow u_{n,b} \text{ stems from } f_0(u_{n,b}) \Leftrightarrow z_{n,b} = 0,$$

$$x_{n,b} \text{ is nonzero} \Leftrightarrow u_{n,b} \text{ stems from } f_1(u_{n,b}) \Leftrightarrow z_{n,b} = 1.$$

In iteration t, the probability that $x_{n,b}$ is zero based on $u_{n,b}^{t-1}$ and $\gamma_{n,b}^{t-1}$ computes as

$$
\begin{aligned}
P(\mathsf{z}_{n,b} = 0|\mathsf{u}_{n,b} = u_{n,b}^{t-1}) &= \frac{p_{\mathsf{z}_{n,b}}(0) f_{\mathsf{u}_{n,b}|\mathsf{z}_{n,b}}(u_{n,b}^{t-1}|0)}{f_{\mathsf{u}_{n,b}}(u_{n,b}^{t-1})} \\
&= \frac{\gamma_{n,b}^{t-1} f_0(u_{n,b}^{t-1})}{\gamma_{n,b}^{t-1} f_0(u_{n,b}^{t-1}) + (1 - \gamma_{n,b}^{t-1}) f_1(u_{n,b}^{t-1})}.
\end{aligned}
\tag{2.48}
$$

Similarly, the probability that $x_{n,b}$ is nonzero computes as

$$
\begin{aligned}
P(\mathsf{z}_{n,b} = 1|\mathsf{u}_{n,b} = u_{n,b}^{t-1}) &= \frac{p_{\mathsf{z}_{n,b}}(1) f_{\mathsf{u}_{n,b}|\mathsf{z}_{n,b}}(u_{n,b}^{t-1}|1)}{f_{\mathsf{u}_{n,b}}(u_{n,b}^{t-1})} \\
&= \frac{(1 - \gamma_{n,b}^{t-1}) f_1(u_{n,b}^{t-1})}{\gamma_{n,b}^{t-1} f_0(u_{n,b}^{t-1}) + (1 - \gamma_{n,b}^{t-1}) f_1(u_{n,b}^{t-1})}.
\end{aligned}
\tag{2.49}
$$

Now, I introduce soft information in terms of L-values [95, 96] based on the binary latent activity variable:

$$L(\mathsf{z}_{n,b}) = \log \frac{P(\mathsf{z}_{n,b} = 0)}{P(\mathsf{z}_{n,b} = 1)} = \log \frac{\gamma_{n,b}}{1 - \gamma_{n,b}}. \tag{2.50}$$

A large positive value indicates a high probability of $\mathsf{x}_{n,b}$ being zero, a large negative value a high probability of $\mathsf{x}_{n,b}$ being nonzero. Considering the decoupled measurements of iteration t, the conditional L-values read, using (2.48) and (2.49),

$$
\begin{aligned}
L(\mathsf{z}_{n,b}|\mathsf{u}_{n,b} = u_{n,b}^{t-1}) &= \log \frac{P(\mathsf{z}_{n,b} = 0|\mathsf{u}_{n,b} = u_{n,b}^{t-1})}{P(\mathsf{z}_{n,b} = 1|\mathsf{u}_{n,b} = u_{n,b}^{t-1})} \\
&= \log \frac{f_0(u_{n,b}^{t-1})}{f_1(u_{n,b}^{t-1})} + \log \frac{\gamma_{n,b}^{t-1}}{1 - \gamma_{n,b}^{t-1}}.
\end{aligned}
\tag{2.51}
$$

The key feature of BASSAMP is to use the L-values (2.51) of iteration t as *extrinsic a priori information* [96] in the subsequent iteration $t + 1$. To that end, I calculate the L-values that accommodate the *innovation* of the current iteration by subtracting the prior information from the previous iteration:

$$
L_{n,b}^t = L(\mathsf{z}_{n,b}|\mathsf{u}_{n,b} = u_{n,b}^{t-1}) - \underbrace{\log \frac{\gamma_{n,b}^{t-1}}{1 - \gamma_{n,b}^{t-1}}}_{\overline{L}_{n,b}^{t-1}} = \log \frac{f_0(u_{n,b}^{t-1})}{f_1(u_{n,b}^{t-1})}.
\tag{2.52}
$$

These L-values relate the probability mass of $f_0(u_{n,b}^{t-1})$ to the mass of $f_1(u_{n,b}^{t-1})$; a positive L-value indicates that signal entry $x_{n,b}$ was likely to be zero, while a negative L-value indicates that $x_{n,b}$ was likely to be nonzero. Note that the dependency on $\gamma_{n,b}^{t-1}$ has vanished. To enforce and exploit the joint sparsity structure, I define the following procedure:

Definition 2.4.3 (Joint extrinsic update). *The joint extrinsic update is applied on B jointly sparse vectors and performs the following operation that yields $\overline{\mathbf{L}}^t$:*

$$
\overline{L}_{n,b}^t = U_J(\mathbf{U}^{t-1}, \boldsymbol{\beta}^{t-1}, \mathbf{\Gamma}^0) := \overline{L}_{n,b}^0 + \sum_{l \in \mathcal{B} \backslash b} L_{n,l}^t,
\tag{2.53}
$$

$$
\forall n \in \{1, ..., N\}, \forall b \in \mathcal{B} = \{1, ..., B\},
$$

with the static prior information

$$
\overline{L}_{n,b}^0 = \log \frac{\gamma_{n,b}^0}{1 - \gamma_{n,b}^0}.
\tag{2.54}
$$

This can be interpreted as follows: $\overline{L}_{n,b}^0$ is the static prior knowledge about the n-th entry of the b-th signal vector, and $\sum_{l \in \mathcal{B} \backslash b} L_{n,l}^t$ is the extrinsic information from

the other vectors, containing the innovation of the current iteration. If the extrinsic information accumulates to a positive value, entry $x_{n,b}$ becomes more likely to be zero. In case of a negative value, entry $x_{n,b}$ becomes more likely to be nonzero. If the extrinsic information is zero, only the static prior is used, and the algorithm collapses to the BAMP algorithm, individually executed on the B vectors. The zero probabilities $\gamma_{n,b}^0$ of the static prior are initialized according to (1.13).

After the joint extrinsic update, the signal prior is updated accordingly for the subsequent iteration. I define the following procedure based on (2.50):

Definition 2.4.4 (Prior update). *The prior update performs the following operation that yields* $\mathbf{\Gamma}^t = [\boldsymbol{\gamma}_1^t, ..., \boldsymbol{\gamma}_N^t]$:

$$\gamma_{n,b}^t = U_P\left(\overline{L}_{n,b}^t\right) := \frac{1}{1 + \exp\left(-\overline{L}_{n,b}^t\right)}, \tag{2.55}$$

$$\forall n \in \{1, ..., N\}, \forall b \in \mathcal{B} = \{1, ..., B\}.$$

Evidently, the joint extrinsic update (2.53) and the prior update (2.55) can be combined such that both steps are performed in a single update equation:

Definition 2.4.5 (Joint extrinsic prior update). *The joint extrinsic prior update calculates the updated zero probabilities* $\mathbf{\Gamma}^t$ *based on the BAMP variables* $\mathbf{U}^{t-1}$ *and* $\boldsymbol{\beta}^{t-1}$ *and the static prior zero probabilities* $\mathbf{\Gamma}^0$:

$$\begin{aligned}
\gamma_{n,b}^t = U_{JP}(\mathbf{U}^{t-1}, \boldsymbol{\beta}^{t-1}, \mathbf{\Gamma}^0) &= \frac{1}{1 + \exp\left(-\overline{L}_{n,b}^0 - \sum_{l \in \mathcal{B}\backslash b} L_{n,l}^t\right)} \\
&= \frac{1}{1 + \frac{1-\gamma_{n,b}^0}{\gamma_{n,b}^0} \prod_{l \in \mathcal{B}\backslash b} \exp\left(-L_{n,l}^t\right)} \\
&= \frac{1}{1 + \frac{1-\gamma_{n,b}^0}{\gamma_{n,b}^0} \prod_{l \in \mathcal{B}\backslash b} \frac{f_1(u_{n,l}^{t-1})}{f_0(u_{n,l}^{t-1})}},
\end{aligned} \tag{2.56}$$

$$\forall n \in \{1, ..., N\}, \forall b \in \mathcal{B} = \{1, ..., B\}.$$

The implicit dependency on $\boldsymbol{\beta}^{t-1}$ becomes explicit once the innovation L-values $L_{n,b}^t$ are specified for a specific prior.

Derivation with Message Passing on Factor Graph

The BASSAMP algorithm can alternatively be derived from the factor graph in Figure 2.5 that illustrates how the n-th signal entries of B jointly sparse vectors are estimated. The BAMP stage comprises B decoupled measurements, each corresponding to the n-th entry of a signal vector. For convenience in the following, the *vectors denote row vectors* such that, e.g., $\mathbf{x}_n = \mathbf{X}_{n,:} = [x_{n,1}, ..., x_{n,b}]$. The overall distribution associated with this factor graph reads, using Bayes' rule,

$$
\begin{aligned}
f_{\mathbf{x}_n, \mathbf{z}_n | \mathbf{u}_n}(\mathbf{x}_n, \mathbf{z}_n | \mathbf{u}_n) &= \frac{1}{f_{\mathbf{u}_n}(\mathbf{u}_n)} \cdot f_{\mathbf{u}_n | \mathbf{x}_n, \mathbf{z}_n}(\mathbf{u}_n | \mathbf{x}_n, \mathbf{z}_n) \cdot f_{\mathbf{x}_n, \mathbf{z}_n}(\mathbf{x}_n, \mathbf{z}_n) \\
&= \frac{1}{f_{\mathbf{u}_n}(\mathbf{u}_n)} \cdot \underbrace{p_{\mathbf{z}_n}(\mathbf{z}_n)}_{f_n^{\mathrm{E}}(\mathbf{z}_n)} \cdot \prod_{b=1}^{B} \underbrace{f_{\mathbf{u}_{n,b} | \mathbf{x}_{n,b}}(u_{n,b} | x_{n,b})}_{f_{n,b}^{\mathrm{B}}(x_{n,b}; u_{n,b})} \cdot \underbrace{f_{\mathbf{x}_{n,b} | \mathbf{z}_{n,b}}(x_{n,b} | z_{n,b})}_{f_{n,b}^{\mathrm{L}}(x_{n,b}, z_{n,b})} \\
&\propto f_n^{\mathrm{E}}(\mathbf{z}_n) \cdot \prod_{b=1}^{B} f_{n,b}^{\mathrm{B}}(x_{n,b}; u_{n,b}) \cdot f_{n,b}^{\mathrm{L}}(x_{n,b}, z_{n,b}),
\end{aligned}
$$
(2.57)

where $\propto$ stands for equality after scaling to unit area[3]. The key element is the extrinsic exchange via factor node $f_n^{\mathrm{E}}(\mathbf{z}_n)$ that enforces the joint sparsity structure by assuming the following multidimensional Bernoulli prior for $\mathbf{z}_n$:

$$
f_n^{\mathrm{E}}(\mathbf{z}_n) = p_{\mathbf{z}_n}(\mathbf{z}_n) = p_{\mathbf{z}_{n,1}, ..., \mathbf{z}_{n,B}}(z_{n,1}, ..., z_{n,B}) = \gamma_n \prod_{b=1}^{B} \delta(z_{n,b}) + (1 - \gamma_n) \prod_{b=1}^{B} \delta(1 - z_{n,b}).
$$
(2.58)

Either all n-th entries are zero with probability γ_n, or all n-th entries are nonzero with probability $1 - \gamma_n$. Note that this γ_n is not adapted over iterations, it is the static prior according to (1.13). The remaining distributions in (2.57) calculate as

$$
f_{\mathbf{x}_{n,b} | \mathbf{z}_{n,b}}(x_{n,b} | z_{n,b}) = (1 - z_{n,b})\delta(x_{n,b}) + z_{n,b} f_a(x_{n,b}),
$$
(2.59)

$$
f_{\mathbf{u}_{n,b} | \mathbf{x}_{n,b}}(u_{n,b} | x_{n,b}) = \mathcal{N}(u_{n,b}; x_{n,b}, \beta_b).
$$
(2.60)

In each iteration of BASSAMP, the decoupled measurements $\mathbf{u}_n^{t-1}$ are used to estimate $\mathbf{x}_n^t$ and $\mathbf{z}_n^t$. As suggested in [98], the estimation of $\mathbf{x}_n^t$ via BAMP and the estimation of $\mathbf{z}_n^t$ are performed separately. First, a BAMP iteration is executed individually on all B jointly sparse signal vectors, which yields $\mathbf{u}_n^{t-1}$ and $\mathbf{x}_n^t$. Then, message passing and the sum-product algorithm [79–81] are employed to estimate

---

[3]It is convenient to omit the normalization with the joint PDF $f_{\mathbf{u}_n}(\mathbf{u}_n)$. The messages obtained in the process can easily be normalized, if required.
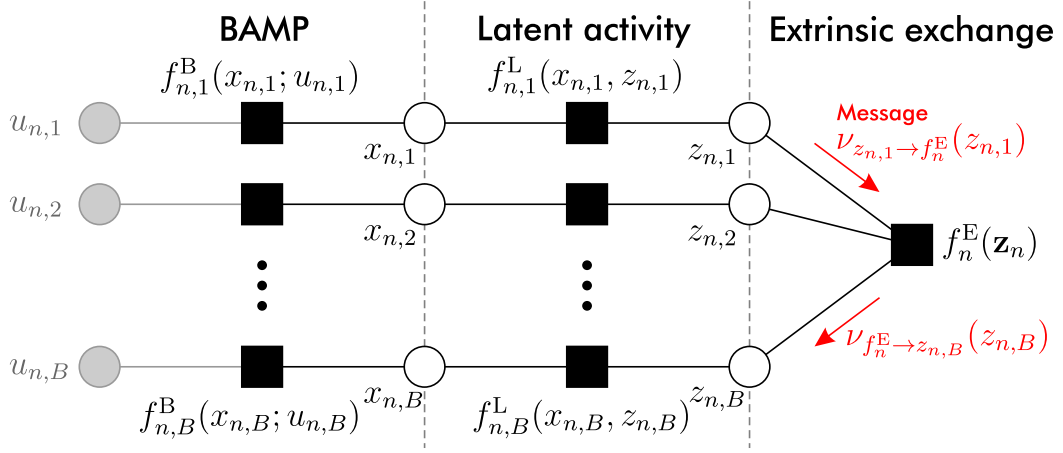
Figure 2.5 Factor graph of BASSAMP enforcing joint sparsity among B signals.

the zero probability $\gamma_{n,b}^t = P(z_{n,b}^t = 0)$. The message from factor $f_{n,b}^{\mathrm{L}}$ to variable $z_{n,b}$ is equal to the message from $z_{n,b}$ to f_n^{E} and computes as

$$
\begin{aligned}
\nu_{z_{n,b} \to f_n^{\mathrm{E}}}(z_{n,b}) &\propto \int_{\mathbb{R}} f_{n,b}^{\mathrm{L}}(x, z_{n,b}) \cdot \nu_{x_{n,b} \to f_{n,b}^{\mathrm{L}}}(x)\, dx \\
&= \int_{\mathbb{R}} f_{\mathsf{x}_{n,b}|\mathsf{z}_{n,b}}(x|z_{n,b}) \cdot f_{\mathsf{u}_{n,b}|\mathsf{x}_{n,b}}(u_{n,b}|x)\, dx \\
&= \int_{\mathbb{R}} [(1 - z_{n,b})\delta(x) + z_{n,b} f_a(x)] \cdot \mathcal{N}(u_{n,b}; x, \beta_b)\, dx \\
&= (1 - z_{n,b}) f_0(u_{n,b}) + z_{n,b} f_1(u_{n,b}).
\end{aligned}
\tag{2.61}
$$

Note that in iteration t, $u_{n,b}^{t-1}$ from the BAMP iteration is plugged in for $u_{n,b}$. This message depends on the latent variable $z_{n,b}$ that is binary, and its L-value

$$
L_{n,b}^t = \log \frac{\nu_{z_{n,b} \to f_n^{\mathrm{E}}}(z_{n,b} = 0)}{\nu_{z_{n,b} \to f_n^{\mathrm{E}}}(z_{n,b} = 1)} = \log \frac{f_0(u_{n,b}^{t-1})}{f_1(u_{n,b}^{t-1})}
\tag{2.62}
$$

contains the full information about $z_{n,b}$. Observe that this is exactly the innovation L-value from (2.52). The message from factor f_n^{E} to variable $z_{n,b}$ computes as

$$
\begin{aligned}
\nu_{f_n^{\mathrm{E}} \to z_{n,b}}(z_{n,b}) &= \sum_{z_l, l \in \mathcal{B} \setminus b} f_n^{\mathrm{E}}(\mathbf{z}_n) \cdot \prod_{l \in \mathcal{B} \setminus b} \nu_{z_{n,l} \to f_n^{\mathrm{E}}}(z_{n,l}) \\
&\propto \gamma_n \delta(z_{n,b}) \prod_{l \in \mathcal{B} \setminus b} f_0(u_{n,l}) + (1 - \gamma_n)\delta(1 - z_{n,b}) \prod_{l \in \mathcal{B} \setminus b} f_1(u_{n,l}).
\end{aligned}
\tag{2.63}
$$

Naturally, this message contains all the extrinsic information about the n-th entry of the b-th signal vector. By computing its L-value in iteration t, the joint extrinsic update (2.53) is obtained:

$$\overline{L}_{n,b}^t = \log \frac{\nu_{f_n^{\mathrm{E}} \to z_{n,b}}(z_{n,b} = 0)}{\nu_{f_n^{\mathrm{E}} \to z_{n,b}}(z_{n,b} = 1)} = \underbrace{\log \frac{\gamma_n}{1 - \gamma_n}}_{\overline{L}_{n,b}^0} + \sum_{l \in \mathcal{B} \backslash b} \underbrace{\log \frac{f_0(u_{n,l}^{t-1})}{f_1(u_{n,l}^{t-1})}}_{L_{n,l}^t}. \tag{2.64}$$

The considered messages are binary distributions that have to sum to probability mass one. Therefore, the normalization constant C of message (2.63) computes as

$$\frac{1}{C} \left[\nu_{f_n^{\mathrm{E}} \to z_{n,b}}(0) + \nu_{f_n^{\mathrm{E}} \to z_{n,b}}(1) \right] = 1$$
$$\Rightarrow C = \gamma_n \prod_{l \in \mathcal{B} \backslash b} f_0(u_{n,l}) + (1 - \gamma_n) \prod_{l \in \mathcal{B} \backslash b} f_1(u_{n,l}), \tag{2.65}$$

and the combined joint extrinsic prior update (2.56) in iteration t is obtained by evaluating this message at $z_{n,b} = 0$:

$$\begin{aligned} \nu_{f_n^{\mathrm{E}} \to z_{n,b}}(0) &= \frac{\gamma_n \prod_{l \in \mathcal{B} \backslash b} f_0(u_{n,l}^{t-1})}{\gamma_n \prod_{l \in \mathcal{B} \backslash b} f_0(u_{n,l}^{t-1}) + (1 - \gamma_n) \prod_{l \in \mathcal{B} \backslash b} f_1(u_{n,l}^{t-1})} \\ &= \frac{1}{1 + \frac{1 - \gamma_n}{\gamma_n} \prod_{l \in \mathcal{B} \backslash b} \frac{f_1(u_{n,l}^{t-1})}{f_0(u_{n,l}^{t-1})}} \\ &= \gamma_{n,b}^t. \end{aligned} \tag{2.66}$$

This message contains the extrinsic belief about the zero probability of the b-th entry.

This concludes the BASSAMP algorithm derivation. For $B = 1$, the BASSAMP algorithm collapses to the BAMP algorithm. Note that BASSAMP can easily be extended to group sparsity [75]. In the following, I will discuss special cases of the signal prior, and derive closed-form expressions for the update functions.

2.4.2 Specification for Relevant Priors

I will now specify the BASSAMP update functions (2.53) and (2.56) for the two prior distributions that are relevant to CS-RFID.

Bernoulli-Gaussian Prior

In the Bernoulli-Gaussian case with prior (2.26), the signal plus noise PDF in (2.43) develops into

$$f_1(u_{n,b}) = \mathcal{N}(u_{n,b}; 0, \beta_b + \sigma^2_{\mathsf{x}_{n,b}}). \tag{2.67}$$

Consequently, (2.43) becomes a *Gaussian mixture* [100, 101] with two components:

$$f_{\mathsf{u}_{n,b}}(u_{n,b}) = \gamma_{n,b}\mathcal{N}(u_{n,b}; 0, \beta_b) + (1 - \gamma_{n,b})\mathcal{N}(u_{n,b}; 0, \beta_b + \sigma^2_{\mathsf{x}_{n,b}}). \tag{2.68}$$

The first component is associated with the *zero entries* in the original signal $\mathbf{X}$. The corresponding estimates $u_{n,b}$ solely contain the effective noise $\widetilde{\mathsf{w}}_{n,b} \sim \mathcal{N}(0, \beta_b)$, i.e., $x_{n,b} = 0$. The second component is associated with the *nonzero entries*, where $u_{n,b}$ contains the noisy signal entries, and $x_{n,b} \neq 0$.

The estimate of the zero probability in iteration t becomes, based on (2.48),

$$P(\mathsf{z}_{n,b} = 0|\mathsf{u}_{n,b} = u^{t-1}_{n,b}) = \frac{\gamma^{t-1}_{n,b}\mathcal{N}(u^{t-1}_{n,b}; 0, \beta^{t-1}_b)}{\gamma^{t-1}_{n,b}\mathcal{N}(u^{t-1}_{n,b}; 0, \beta^{t-1}_b) + (1 - \gamma^{t-1}_{n,b})\mathcal{N}(u^{t-1}_{n,b}; 0, \beta^{t-1}_b + \sigma^2_{\mathsf{x}_{n,b}})}. \tag{2.69}$$

In literature [99–101], (2.69) describes the responsibilities that are computed in the E-step of the EM algorithm, i.e., "the responsibility that component $\mathsf{z}_{n,b}$ takes for explaining the decoupled measurement $u^{t-1}_{n,b}$."

The innovation L-values (2.52) boil down to

$$\begin{aligned}
L^t_{n,b} &= \log \frac{\mathcal{N}(u^{t-1}_{n,b}; 0, \beta^{t-1}_b)}{\mathcal{N}(u^{t-1}_{n,b}; 0, \beta^{t-1}_b + \sigma^2_{\mathsf{x}_{n,b}})} \\
&= \log \sqrt{\frac{\beta^{t-1}_b + \sigma^2_{\mathsf{x}_{n,b}}}{\beta^{t-1}_b}} - \frac{1}{2}\frac{(u^{t-1}_{n,b})^2 \sigma^2_{\mathsf{x}_{n,b}}}{\beta^{t-1}_b(\beta^{t-1}_b + \sigma^2_{\mathsf{x}_{n,b}})}.
\end{aligned} \tag{2.70}$$

A detailed derivation of (2.70) is found in Appendix A.3. Based on (2.53) and (2.70), the **Bernoulli-Gaussian joint extrinsic update** reads

$$\overline{L}^t_{n,b} = U_J(\mathbf{U}^{t-1}, \boldsymbol{\beta}^{t-1}, \mathbf{\Gamma}^0) = \log \frac{\gamma^0_{n,b}}{1 - \gamma^0_{n,b}} + \sum_{l \in \mathcal{B} \backslash b} \log \sqrt{\frac{\beta^{t-1}_l + \sigma^2_{\mathsf{x}_{n,l}}}{\beta^{t-1}_l}} - \frac{1}{2}\frac{(u^{t-1}_{n,l})^2 \sigma^2_{\mathsf{x}_{n,l}}}{\beta^{t-1}_l(\beta^{t-1}_l + \sigma^2_{\mathsf{x}_{n,l}})},$$

$$\forall n \in \{1, ..., N\}, \forall b \in \mathcal{B} = \{1, ..., B\}. \tag{2.71}$$

The prior update is the same as defined in (2.55). The **Bernoulli-Gaussian joint extrinsic prior update** becomes

$$
\begin{aligned}
\gamma_{n,b}^t &= U_{JP}(\mathbf{U}^{t-1}, \boldsymbol{\beta}^{t-1}, \boldsymbol{\Gamma}^0) \\
&= \frac{1}{1 + \frac{1-\gamma_{n,b}^0}{\gamma_{n,b}^0} \prod_{l \in \mathcal{B} \backslash b} \sqrt{\frac{\beta_l^{t-1}}{\beta_l^{t-1} + \sigma_{\mathsf{x}_{n,l}}^2}} \exp\left(\frac{1}{2} \cdot \frac{(u_{n,l}^{t-1})^2 \sigma_{\mathsf{x}_{n,l}}^2}{\beta_l^{t-1}(\beta_l^{t-1} + \sigma_{\mathsf{x}_{n,l}}^2)} \right)}, \\
&\quad \forall n \in \{1, ..., N\}, \forall b \in \mathcal{B} = \{1, ..., B\}.
\end{aligned}
\tag{2.72}
$$

Bernoulli-Laplace Prior

The Bernoulli-Laplace prior (2.29) entails the following distribution as signal plus noise PDF in (2.43):

$$
f_1(u_{n,b}) = \frac{1}{4\kappa_{n,b}} \exp\left(-\frac{u_{n,b}^2}{2\beta_b} \right) [k_1(u_{n,b}) + k_2(u_{n,b})],
\tag{2.73}
$$

where $k_1(u_{n,b})$ and $k_2(u_{n,b})$ were defined in (2.35) and (2.36), respectively. The derivation of (2.73) is provided in Appendix B.2.

The estimate of the zero probability in iteration t becomes, based on (2.48),

$$
P(\mathsf{z}_{n,b} = 0 | \mathsf{u}_{n,b} = u_{n,b}^{t-1}) = \frac{\gamma_{n,b}^{t-1} \mathcal{N}(u_{n,b}^{t-1}; 0, \beta_b^{t-1})}{\gamma_{n,b}^{t-1} \mathcal{N}(u_{n,b}^{t-1}; 0, \beta_b^{t-1}) + (1 - \gamma_{n,b}^{t-1}) f_1(u_{n,b})},
\tag{2.74}
$$

and the innovation L-values (2.52) develop into

$$
\begin{aligned}
L_{n,b}^t &= \log \frac{\mathcal{N}(u_{n,b}^{t-1}; 0, \beta_b^{t-1})}{\frac{1}{4\kappa_{n,b}} \exp\left(-\frac{(u_{n,b}^{t-1})^2}{2\beta_b} \right) [k_1(u_{n,b}^{t-1}) + k_2(u_{n,b}^{t-1})]} \\
&= \log \frac{4\kappa_{n,b}}{\sqrt{2\pi\beta_b^{t-1}}} - \log [k_1(u_{n,b}^{t-1}) + k_2(u_{n,b}^{t-1})].
\end{aligned}
\tag{2.75}
$$

The derivation of (2.75) is found in Appendix B.3. Based on (2.53) and (2.75), the **Bernoulli-Laplace joint extrinsic update** reads

$$
\overline{L}_{n,b}^t = U_J(\mathbf{U}^{t-1}, \boldsymbol{\beta}^{t-1}, \boldsymbol{\Gamma}^0) = \log \frac{\gamma_{n,b}^0}{1 - \gamma_{n,b}^0} + \sum_{l \in \mathcal{B} \backslash b} \log \frac{4\kappa_{n,l}}{\sqrt{2\pi\beta_l^{t-1}}} - \log [k_1(u_{n,l}^{t-1}) + k_2(u_{n,l}^{t-1})],
$$
$$
\forall n \in \{1, ..., N\}, \forall b \in \mathcal{B} = \{1, ..., B\}.
\tag{2.76}
$$

The prior update is the same as defined in (2.55). The **Bernoulli-Laplace joint extrinsic prior update** becomes

$$
\gamma_{n,b}^t = U_{JP}(\mathbf{U}^{t-1}, \boldsymbol{\beta}^{t-1}, \boldsymbol{\Gamma}^0)
$$

$$
= \frac{1}{1 + \frac{1-\gamma_{n,b}^0}{\gamma_{n,b}^0} \prod_{l \in \mathcal{B} \backslash b} \frac{\sqrt{2\pi \beta_l^{t-1}}}{4\kappa_{n,l}} \left[k_1(u_{n,l}^{t-1}) + k_2(u_{n,l}^{t-1}) \right]},
\tag{2.77}
$$

$$
\forall n \in \{1, ..., N\}, \forall b \in \mathcal{B} = \{1, ..., B\}.
$$

2.4.3 Extension to Vector Denoiser

The MMSE estimator $F(u_{n,b}; \beta_b, \gamma_{n,b})$ in Algorithm 3 acts as a *scalar denoiser* function on the decoupled measurements $u_{n,b}$. In this section, I extend the MMSE estimator (2.27) of the Bernoulli-Gaussian prior to a *vector denoiser* that operates on a vector input $[u_{n,1}, ..., u_{n,B}]$, i.e., on a row of $\mathbf{U}$. This bears the following advantages:

- The joint sparsity structure is enforced directly — this leads to a faster algorithmic convergence. Moreover, the extrinsic update mechanism becomes obsolete.

- Correlation among the signal entries can be incorporated naturally.

These advantages can be attributed to the usage of a *multivariate prior distribution* for a row of $\mathbf{X}$. For clarity in the exhibition, I omit index n in the remainder of this section:

$$
[u_{n,1}, ..., u_{n,B}] \rightarrow \mathbf{u} = [u_1, ..., u_B],
\tag{2.78}
$$

$$
\underbrace{[x_{n,1}, ..., x_{n,B}]}_{n\text{-th row of } \mathbf{X}} \rightarrow \underbrace{\mathbf{x} = [x_1, ..., x_B]}_{\text{Any row of } \mathbf{X}},
\tag{2.79}
$$

i.e., the vectors in this section refer to rows of the matrices, and the vector denoiser will act on these rows. The *multivariate Bernoulli-Gaussian prior* with joint sparsity assumption reads

$$
f_\mathbf{x}(\mathbf{x}) = \gamma \prod_{b=1}^B \delta(x_b) + (1 - \gamma)\mathcal{N}(\mathbf{x}; \mathbf{0}, \mathbf{Q_x}),
\tag{2.80}
$$

with covariance matrix $\mathbf{Q_x} = \mathbb{E}_\mathbf{x}\{\mathbf{x}^\mathrm{T}\mathbf{x}\}$ ($\mathbf{x}$ is a zero-mean row vector) and multivariate Gaussian distribution

$$\mathcal{N}(\mathbf{x}; \mathbf{0}, \mathbf{Q_x}) = \frac{1}{\sqrt{(2\pi)^B|\mathbf{Q_x}|}} \exp\left(-\frac{1}{2}\mathbf{x}\mathbf{Q_x}^{-1}\mathbf{x}^\mathrm{T}\right). \tag{2.81}$$

The correlations among the signal entries are naturally described by the off-diagonal entries in $\mathbf{Q_x}$. In case of independently distributed entries in $\mathbf{x}$, the off-diagonal entries are zero and $\mathbf{Q_x} = \mathrm{diag}([\sigma_{\mathsf{x}_1}^2, ..., \sigma_{\mathsf{x}_B}^2])$ such that

$$f_\mathbf{x}(\mathbf{x}) = \gamma \prod_{b=1}^{B} \delta(x_b) + (1 - \gamma) \prod_{b=1}^{B} \mathcal{N}(x_b; 0, \sigma_{\mathsf{x}_b}^2). \tag{2.82}$$

Instead of executing B BAMP iterations independently as in Algorithm 3, the BASSAMP algorithm with vector denoiser that is depicted in Algorithm 4 operates on the whole vector. Furthermore, rather than having B different β_b and γ_b –one for each BAMP instance–, the vector case has a single β and γ for each row of $\mathbf{U}$. Note that in the general case with *different sensing matrices* (2.41),

$$\mathbf{A}_\mathrm{all} = [\mathbf{A}^{(1)}, ..., \mathbf{A}^{(B)}] , \quad \mathbf{A}_\mathrm{all}^\mathrm{Tr} = \left[\left(\mathbf{A}^{(1)}\right)^\mathrm{T}, ..., \left(\mathbf{A}^{(B)}\right)^\mathrm{T}\right], \tag{2.83}$$

$$\mathbf{X}_\mathrm{all} = \begin{bmatrix} \mathbf{x}_1 & & \\ & \ddots & \\ & & \mathbf{x}_B \end{bmatrix} , \quad \mathbf{R}_\mathrm{all} = \begin{bmatrix} \mathbf{r}_1 & & \\ & \ddots & \\ & & \mathbf{r}_B \end{bmatrix}, \tag{2.84}$$

where $\mathbf{x}_b$ and $\mathbf{r}_b$ are columns of $\mathbf{X}$ and $\mathbf{R}$, respectively. For *equal sensing matrices* (2.42) as encountered in this thesis,

$$\mathbf{A}_\mathrm{all} = \mathbf{A} , \quad \mathbf{A}_\mathrm{all}^\mathrm{Tr} = \mathbf{A}^\mathrm{T} , \quad \mathbf{X}_\mathrm{all} = \mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_B] , \quad \mathbf{R}_\mathrm{all} = \mathbf{R} = [\mathbf{r}_1, ..., \mathbf{r}_B]. \tag{2.85}$$

By extending the scalar MMSE estimator formulation (2.21) to vectors, the **vector denoiser** calculates as

$$\begin{aligned} F_\mathrm{vec}(\mathbf{u}; \beta) &= \mathbb{E}_\mathbf{x}\{\mathbf{x}|\mathbf{u} = \mathbf{u}; \beta\} \\ &= \frac{\int_{\mathbb{R}^B} \widetilde{\mathbf{x}} \cdot \mathcal{N}(\mathbf{u}; \widetilde{\mathbf{x}}, \mathbf{Q}_{\widetilde{\mathbf{w}}}) \cdot f_\mathbf{x}(\widetilde{\mathbf{x}}) \, d\widetilde{\mathbf{x}}}{\int_{\mathbb{R}^B} \mathcal{N}(\mathbf{u}; \widetilde{\mathbf{x}}, \mathbf{Q}_{\widetilde{\mathbf{w}}}) \cdot f_\mathbf{x}(\widetilde{\mathbf{x}}) \, d\widetilde{\mathbf{x}}}, \end{aligned} \tag{2.86}$$

where the effective noise covariance is assumed to be $\mathbf{Q}_{\widetilde{\mathbf{w}}} = \beta \mathbf{I}_B$ due to i.i.d. $\widetilde{w}_b$, $b \in \mathcal{B}$. For the Bernoulli-Gaussian prior with *independent* signal entries (2.82), the

---

Algorithm 4 BASSAMP for signals with joint sparsity — vector denoiser

---

1: Initialize $\mathbf{X}^t = \mathbf{0}_{N \times B}$, $\gamma = 1 - \frac{K}{N}$ and $\mathbf{r}_b^t = \mathbf{y}_b \ \forall b \in \mathcal{B} = \{1, ..., B\}$ at $t = 0$
2: **do**
3:      $t \leftarrow t + 1$
4:      $\mathbf{U}^{t-1} = \mathbf{X}^{t-1} + \mathbf{A}_{\text{all}}^{\text{Tr}} \mathbf{R}_{\text{all}}^{t-1}$ ▷ Compute decoupled measurements
5:      $\beta^{t-1} = \frac{1}{MB} \|\mathbf{R}^{t-1}\|_{\text{F}}^2$ ▷ Estimate effective noise variance
6:      $\mathbf{X}^t = F_{\text{vec}}(\mathbf{U}^{t-1}; \beta^{t-1})$ ▷ Estimate signal (vector denoiser)
7:      $\mathbf{R}^t = \mathbf{Y} - \mathbf{A}_{\text{all}} \mathbf{X}_{\text{all}}^t + \mathbf{R}^{t-1} \frac{1}{MB} \sum_n \sum_b \left(F'_{\text{vec}}(\mathbf{U}^{t-1}; \beta^{t-1}) \right)_{n,b}$ ▷ Compute residual
8: **while** $\|\mathbf{X}^t - \mathbf{X}^{t-1}\|_{\text{F}} > \epsilon_{\text{tol}} \|\mathbf{X}^{t-1}\|_{\text{F}}$ and $t < t_{\max}$ ▷ Check stopping criteria
9: **return** $\widehat{\mathbf{X}} = \mathbf{X}^t$ ▷ Recovered vectors

---

b-th output of the vector denoiser and its derivative (applied on rows of $\mathbf{U}$) boil down to

$$(F_{\text{vec}}(\mathbf{u}; \beta))_b = u_b \cdot \frac{\sigma_{\mathsf{x}_b}^2}{\beta + \sigma_{\mathsf{x}_b}^2} \cdot \frac{1}{1 + \underbrace{\frac{\gamma}{1 - \gamma} \prod_{l=1}^{B} \sqrt{\frac{\beta + \sigma_{\mathsf{x}_l}^2}{\beta}} \exp\left(-\frac{u_l^2}{2} \frac{\sigma_{\mathsf{x}_l}^2}{\beta(\beta + \sigma_{\mathsf{x}_l}^2)} \right)}_{m_{\text{vec}}(\mathbf{u}; \beta, \gamma)}}, \quad (2.87)$$

$$(F'_{\text{vec}}(\mathbf{u}; \beta))_b = \frac{\sigma_{\mathsf{x}_b}^2}{\beta + \sigma_{\mathsf{x}_b}^2} \cdot \frac{1}{1 + m_{\text{vec}}(\mathbf{u}; \beta, \gamma)} + \frac{1}{\beta} \cdot m_{\text{vec}}(\mathbf{u}; \beta, \gamma) \cdot (F_{\text{vec}}(\mathbf{u}; \beta))_b^2. \quad (2.88)$$

Note the similarities to the scalar denoiser (2.27) and its derivative (2.28); the major difference is the product in $m_{\text{vec}}(\mathbf{u}; \beta, \gamma)$ that involves all vector entries. The derivation of both functions is provided in appendices A.4 and A.5. In case of dependent (correlated) signal entries, (2.86) may be solved numerically.

Algorithm 3 and Algorithm 4 have the same recovery performance in terms of achieved MSE and required number of measurements M — they solve the same estimation problem and utilize the same amount of prior knowledge. Algorithm 3 is based on scalar, independent BAMP denoiser functions, and the joint sparsity structure is enforced by accumulating extrinsic belief about the zero probability. Algorithm 4 employs a vector BAMP denoiser function that directly incorporates prior knowledge about the signal structure by specifying a multivariate signal prior (2.80). The difference becomes obvious when comparing the factor graph of scalar BASSAMP in Figure 2.5 to the factor graph of vector BASSAMP in Figure 2.6; the passed messages involving $\mathbf{x}$ are now vector-valued, and an extrinsic information exchange among scalar entries is no longer required.
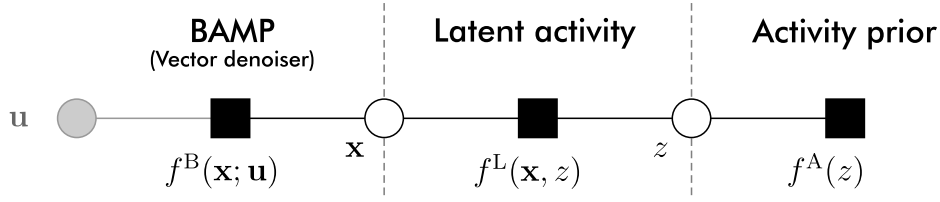
Figure 2.6 Factor graph of BASSAMP with vector denoiser.

The factorization associated with the factor graph in Figure 2.6 is similar to (2.57) but retains the vector-valued form:

$$
\begin{aligned}
f_{\mathbf{x},z|\mathbf{u}}(\mathbf{x}, z|\mathbf{u}) &= \frac{1}{f_{\mathbf{u}}(\mathbf{u})} \cdot f_{\mathbf{u}|\mathbf{x},z}(\mathbf{u}|\mathbf{x}, z) \cdot f_{\mathbf{x},z}(\mathbf{x}, z) \\
&= \frac{1}{f_{\mathbf{u}}(\mathbf{u})} \cdot \underbrace{f_{\mathbf{u}|\mathbf{x}}(\mathbf{u}|\mathbf{x})}_{f^{\mathrm{B}}(\mathbf{x};\mathbf{u})} \cdot \underbrace{f_{\mathbf{x}|z}(\mathbf{x}|z)}_{f^{\mathrm{L}}(\mathbf{x},z)} \cdot \underbrace{p_z(z)}_{f^{\mathrm{A}}(z)} \\
&\propto f^{\mathrm{B}}(\mathbf{x};\mathbf{u}) \cdot f^{\mathrm{L}}(\mathbf{x}, z) \cdot f^{\mathrm{A}}(z).
\end{aligned} \tag{2.89}
$$

Based on (2.58) and (2.80), the factors are specified as

$$
f^{\mathrm{B}}(\mathbf{x};\mathbf{u}) = f_{\mathbf{u}|\mathbf{x}}(\mathbf{u}|\mathbf{x}) = \mathcal{N}(\mathbf{u}; \mathbf{x}, \mathbf{Q}_{\widetilde{\mathbf{w}}}), \tag{2.90}
$$

$$
f^{\mathrm{L}}(\mathbf{x}, z) = f_{\mathbf{x}|z}(\mathbf{x}|z) = (1 - z) \cdot \prod_{b=1}^{B} \delta(x_b) + z \cdot \mathcal{N}(\mathbf{x}; \mathbf{0}, \mathbf{Q}_{\mathbf{x}}), \tag{2.91}
$$

$$
f^{\mathrm{A}}(z) = p_z(z) = \gamma\delta(z) + (1 - \gamma)\delta(1 - z). \tag{2.92}
$$

Note that by using a vector denoiser, the latent activity variable z become obsolete; factor $f^{\mathrm{A}}(z) = p_z(z)$ only contributes the static prior knowledge (1.13) about γ that is now contained in the multivariate signal prior (2.80). It is sufficient to consider the factorization

$$
f_{\mathbf{x}|\mathbf{u}}(\mathbf{x}|\mathbf{u}) = \frac{1}{f_{\mathbf{u}}(\mathbf{u})} \cdot f_{\mathbf{u}|\mathbf{x}}(\mathbf{u}|\mathbf{x}) \cdot f_{\mathbf{x}}(\mathbf{x}). \tag{2.93}
$$

The joint sparsity structure that had to be enforced by $f^{\mathrm{E}}(\mathbf{z}) = p_{\mathbf{z}}(\mathbf{z})$ in scalar BASSAMP is now fully incorporated into the multivariate signal prior $f_{\mathbf{x}}(\mathbf{x})$.

2.4.4 Recovery Stability

In rare cases, Algorithms 3 and 4 run into instabilities. This happens for a low dimension N in the order of 100 or $1\,000$. The AMP-based algorithms are derived for the asymptotic regime as defined in Definition 2.3.1. In finite dimensions, the standalone algorithms AMP and BAMP work very well, but the concatenation of several BAMP instances to exploit joint sparsity structure introduces additional cycles in the graph that lead to occasional instabilities in the BASSAMP algorithms. An increase of $\{K, M, N\}$ while keeping $\delta = \frac{M}{N}$ and $\rho = \frac{K}{M}$ constant reduces the occurrence of instabilities.

I investigated the change in the reconstruction error, which is essentially dictated by the residual $\mathbf{R}^t = [\mathbf{r}_1^t, ..., \mathbf{r}_B^t]$ whose energy relates to the effective noise variance. After observing its behavior over many realizations, I suggest the following heuristic:

- For *stable* recoveries, the residual magnitude $\|\mathbf{R}^t\|_F$ decreases monotonically over iterations t.

- For *unstable* recoveries, the residual magnitude $\|\mathbf{R}^t\|_F$ first decreases and then increases until it oscillates.

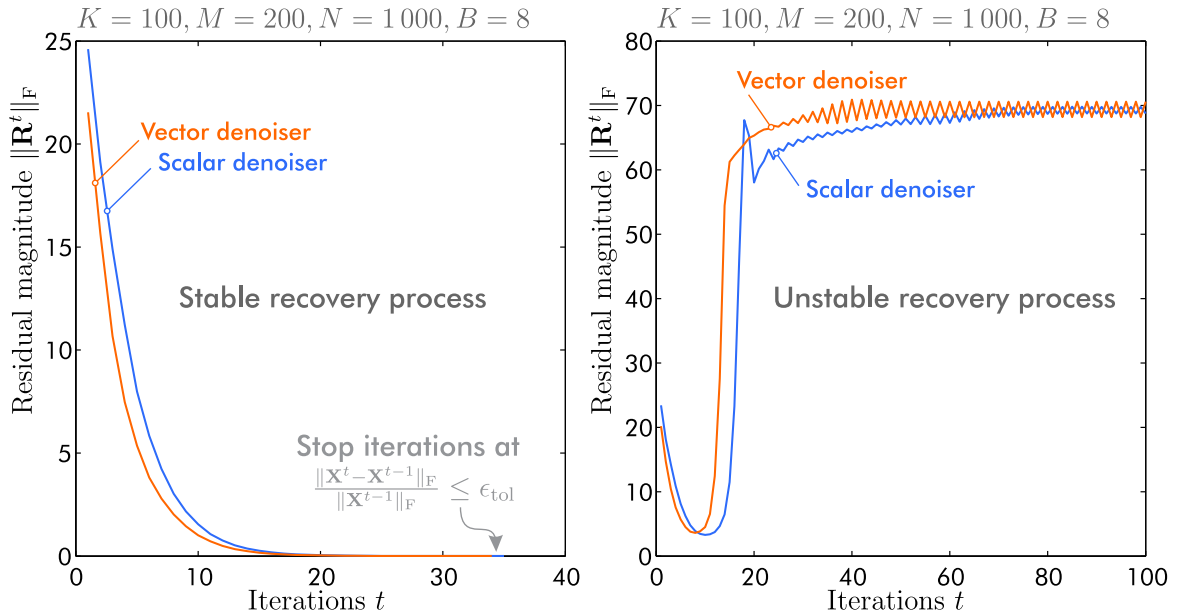This behavior is visualized in Figure 2.7.

Figure 2.7 Residual magnitude versus iterations.

The recovery is denoised by the MMSE estimator as long as the residual magnitude decreases; an increasing magnitude indicates a worsening, unstable recovery. To

combat instabilities, I recommend to stop the algorithm when the residual magnitude starts to increase. This is formally stated by the following condition:

> **if** $\|\mathbf{R}^t\|_{\mathrm{F}} > \|\mathbf{R}^{t-1}\|_{\mathrm{F}}$ **then**
> $\quad \mathbf{X}^t \leftarrow \mathbf{X}^{t-1}$
> $\quad$ break

I add this abort condition between line 8 and 9 of Algorithm 3 (after executing the BAMP iterations), and between line 7 and 8 of Algorithm 4. This condition will be used in all BASSAMP simulations.

2.5 Empirical Phase Transitions

In order to compare the algorithmic performance over a wide range of parameters $\{K, M, N\}$, this section presents the concept of *empirical phase transition curves* [65, 102, 103]. These curves can be used to determine the required number of measurements for successful recovery, based on K and N; they define the sharp boundary of the *under-sampling versus sparsity* trade-off.

To obtain the empirical phase transition curves, I consider an under-sampling $\left(\delta = \frac{M}{N}\right)$ versus sparsity $\left(\rho = \frac{K}{M}\right)$ grid where the values range from 0.05 to 0.95 with step-size 0.05, respectively. At each grid point, 250 realizations of $\mathbf{A} \in \{-\frac{1}{\sqrt{M}}, \frac{1}{\sqrt{M}}\}^{M \times N}$ and $\mathbf{x}$ ($\mathbf{X}$ in case of joint sparsity and $B > 1$) are simulated, the *noise is zero*. The signal dimension is fixed to $N = 1\,000$. The signal recoveries of Algorithm 1, Algorithm 2 and Algorithm 3 are compared — optimal tuning and perfect prior knowledge is assumed. A success indicator for each realization i is defined as

$$S_i = \begin{cases} 1 & \mathrm{NMSE}(\mathbf{x}, \widehat{\mathbf{x}})_i < 10^{-4} \\ 0 & \text{else} \end{cases} . \tag{2.94}$$

The average success at a grid point is obtained as $\overline{S} = \frac{1}{250} \sum_{i=1}^{250} S_i$. The empirical phase transition curves are finally obtained by plotting the 0.5 contour of $\overline{S}$ using the MATLAB[4] function *contour*.

Figure 2.8 depicts the average success $\overline{S}$ on the under-sampling versus sparsity grid for a signal with Bernoulli-Gaussian prior. Key observation: The transition to the success-phase is very sharp. Going from AMP over BAMP to BASSAMP, the success-

---

[4]*MATLAB* is a multi-paradigm numerical computing environment and programming language developed by MathWorks. It is a registered trademark of The MathWorks, inc.
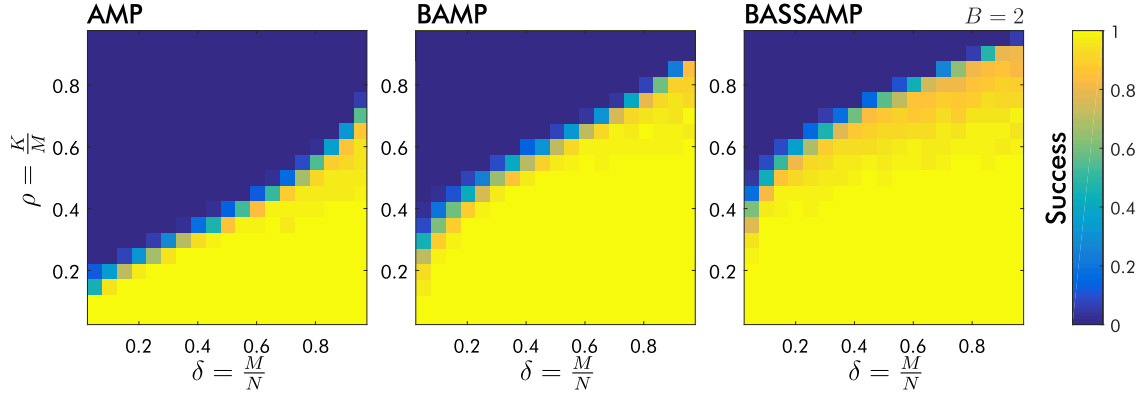
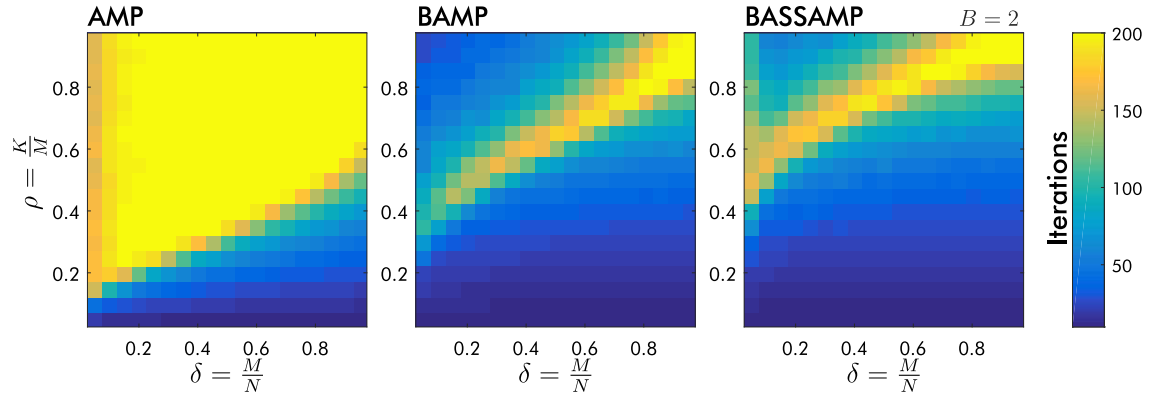Figure 2.8 Empirical phase transition — average success percentage.

Figure 2.9 Empirical phase transition — average number of iterations

ful area becomes larger. This implies that for fixed K and N, fewer measurements M are required.

Figure 2.9 illustrates the average number of iterations at each grid point; note that I set the maximum number to $t_{\max} = 200$. While AMP iterates to the limit in the unsuccessful phase, BAMP and BASSAMP exhibit a large number of iterations only during the phase transition. This plot also demonstrates that within the success-phase, the algorithms typically require very few iterations in the order of 5 to 50. Going from AMP over BAMP to BASSAMP, the number of iterations in the success-phase decreases for a fixed set of parameters $\{K, M, N\}$.

Figure 2.10 finally depicts the empirical phase transition curves, for the two relevant priors (2.26) and (2.29) with $\sigma^2_{\mathsf{x}_n} = 2\kappa^2_n = 1$, and for $B = \{2, 4\}$ jointly sparse vectors in case of BASSAMP. AMP follows the ℓ_1 phase transition of basis-pursuit [102], regardless of the signal prior. BAMP exhibits a strongly improved performance over AMP by exploiting the proper signal prior distribution. By additionally considering joint sparsity, BASSAMP further improves over BAMP.
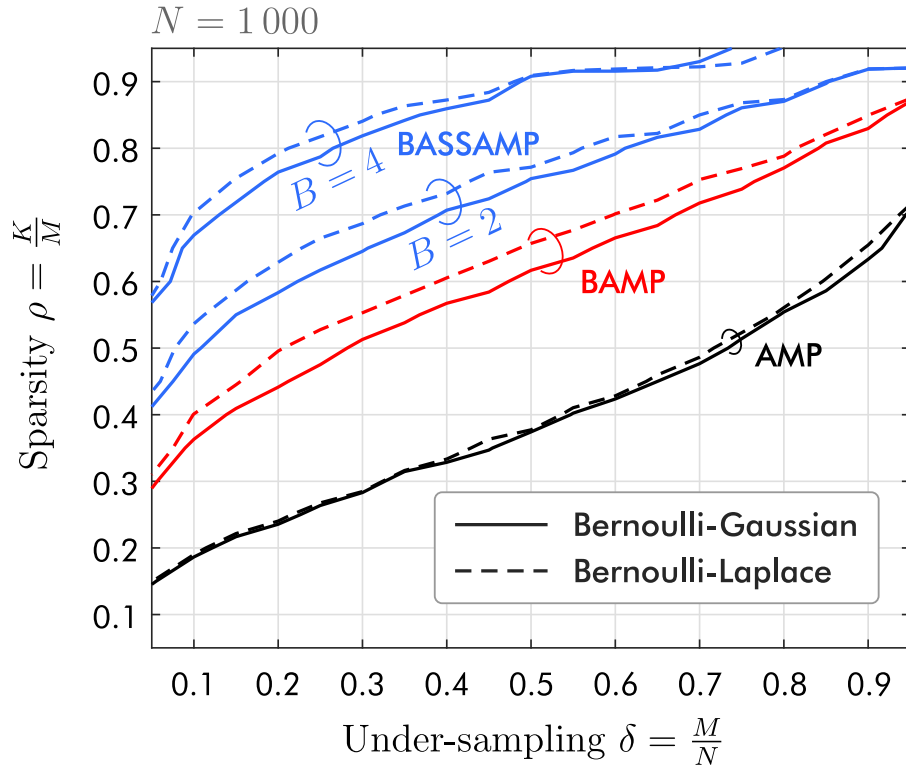
Figure 2.10 Empirical phase transition curves

If noise is present, the sharp phase transition is softened and occurs at lower $\rho = \frac{K}{M}$, i.e., more measurements are required on average for successful recovery. This behavior was investigated in [103].

Overall, the phase transition curves demonstrate that the exploitation of prior knowledge and joint sparsity significantly lowers the number of required measurements for successful recovery. This is an important property that will be leveraged by CS-RFID in Chapter 4.

2.6   State Evolution

A major advantage of the AMP framework over other recovery methods is the State Evolution (SE) formalism [65, 77, 78] that allows to predict the algorithmic behavior *without* computationally expensive Monte Carlo simulations. The prediction becomes exact in the asymptotic regime (see Definition 2.3.1), but is still sufficiently accurate for signal dimensions in the order of $N = 1\,000$, as I will demonstrate in this section. I will utilize the SE formalism for the following tasks:

- Prediction of the recovery MSE for each iteration. This enables to determine the required number of iterations to attain a certain MSE. This is helpful for, e.g., fixed complexity or fixed run-time implementations.

- Investigation of the required number of measurements for AMP.

- Investigation of prior distribution mismatch in BAMP. This allows to quantify the MSE performance gap between a BAMP algorithm whose MMSE estimator was derived for the true prior distribution and another MMSE estimator with a prior distribution mismatch.

- Investigation and prediction of BASSAMP — scalar and vector case.

The SE is based on the decoupling principle (see Definition 2.3.2) and its decoupled measurement model that is inherent to all AMP-based recovery algorithms. In the following, I assume that all entries of a sparse signal vector $\mathbf{x}$ are i.i.d. with distribution (cf. (1.14))

$$f_{\mathsf{x}}(x) = \gamma\delta(x) + (1-\gamma)f_a(x). \tag{2.95}$$

The indices are omitted, since SE yields a macroscopic result that applies to *all entries* of $\mathbf{x}$. Recall the decoupled measurement model (2.17) $\mathsf{u} = \mathsf{x} + \widetilde{\mathsf{w}}$, where $\widetilde{\mathsf{w}} \sim \mathcal{N}(0, \beta)$. In the following, $\widetilde{\mathsf{w}}$ is replaced by $\sqrt{\beta}\mathsf{v}$, $\mathsf{v} \sim \mathcal{N}(0, 1)$, to explicitly visualize the dependency on the effective noise variance β. SE enables the prediction of β over (B)AMP iterations t [78], using the **initialization**

$$\beta^0 = \sigma_{\mathsf{w}}^2 + \frac{1}{\delta}\,\mathbb{E}_{\mathsf{x}}\left\{\mathsf{x}^2\right\}, \tag{2.96}$$

and the **update**

$$\beta^{t+1} = \sigma_{\mathsf{w}}^2 + \frac{1}{\delta}\,\mathbb{E}_{\mathsf{x},\mathsf{v}}\left\{\left[F^t\left(\mathsf{x} + \sqrt{\beta^t}\mathsf{v}\right) - \mathsf{x}\right]^2\right\}, \tag{2.97}$$

where $\delta = \frac{M}{N}$, and $F^t(\cdot)$ is the denoiser function with parameters from iteration t — AMP uses the soft thresholder (2.11), while BAMP employs the MMSE estimator (2.21). The expectation in (2.97) computes the variance of the estimator (denoiser) function. Inserting the signal prior (2.95), its calculation splits up into two parts:

$$
\begin{aligned}
E(\beta^t) &:= \mathbb{E}_{\mathsf{x},\mathsf{v}} \left\{ \left[F^t \left(\mathsf{x} + \sqrt{\beta^t} \mathsf{v} \right) - \mathsf{x} \right]^2 \right\} \\
&= \int_v \int_x \left[F^t \left(x + \sqrt{\beta^t} v \right) - x \right]^2 \cdot f_{\mathsf{v}}(v) \cdot f_{\mathsf{x}}(x) \, dv \, dx \\
&= \gamma \int_v \int_x \left[F^t \left(x + \sqrt{\beta^t} v \right) - x \right]^2 \cdot f_{\mathsf{v}}(v) \cdot \delta(x) \, dv \, dx \\
&\quad + (1-\gamma) \int_v \int_x \left[F^t \left(x + \sqrt{\beta^t} v \right) - x \right]^2 \cdot \mathcal{N}(v; 0, 1) \cdot f_a(x) \, dv \, dx \\
&= \gamma \underbrace{\int_v \left[F^t \left(\sqrt{\beta^t} v \right) \right]^2 \cdot \mathcal{N}(v; 0, 1) \, dv}_{E_0(\beta^t)} \\
&\quad + (1-\gamma) \underbrace{\int_v \int_x \left[F^t \left(x + \sqrt{\beta^t} v \right) - x \right]^2 \cdot \mathcal{N}(v; 0, 1) \cdot f_a(x) \, dv \, dx}_{E_1(\beta^t)} \\
&= \gamma E_0(\beta^t) + (1-\gamma) E_1(\beta^t),
\end{aligned}
\tag{2.98}
$$

where $E_0(\beta^t)$ and $E_1(\beta^t)$ are the estimator variances of the zero respectively nonzero entries in x. They are weighted with their occurrence probability and summed up to obtain the overall estimator variance $E(\beta^t)$. The integrals are solved numerically, since closed form expressions are not available in general. The so obtained values relate to numerical Monte Carlo simulation results as follows, where $\mathbf{u}^t$ stems from simulation, and the MSE is averaged over many realizations of x, $\mathbf{A}$ and $\mathbf{w}$:

$$
\underbrace{\beta^t = \sigma_{\mathsf{w}}^2 + \frac{1}{\delta} E(\beta^{t-1})}_{\text{State evolution}} \approx \underbrace{\text{MSE}(\mathbf{u}^t, \mathbf{x})}_{\text{Simulation}}.
\tag{2.99}
$$

I will now apply this framework to the different AMP-based algorithms encountered in this work.

2.6.1   Required Number of Measurements for AMP

In Section 2.2.3, I discussed the selection of the λ parameter for AMP Algorithm 1, and the proper choice of the measurement multiplier c. Let me investigate the number of required measurements by utilizing SE, λ is chosen according to (2.14).

For Algorithm 1, the threshold of the soft thresholder is given by $\tau^t = \sqrt{\beta^t}\lambda$, and the SE update equation (2.97) becomes

$$\beta^{t+1} = \sigma_{\mathsf{w}}^2 + \frac{1}{\delta}\, \mathbb{E}_{\mathsf{x},\mathsf{v}} \left\{ \left[\eta\left(\mathsf{x} + \sqrt{\beta^t}\mathsf{v};\, \sqrt{\beta^t}\lambda\right) - \mathsf{x} \right]^2 \right\}. \qquad (2.100)$$

In the following experiment, I generate a Bernoulli-Gaussian distributed signal according to (2.26) with $\sigma_{\mathsf{x}}^2 = 100$, there is no measurement noise ($\sigma_{\mathsf{w}}^2 = 0$). The analytical SE (2.100) is compared to a simulation campaign with $1\,000$ realizations of x, $\mathbf{A}$ and w, the MSE is averaged over the realizations.
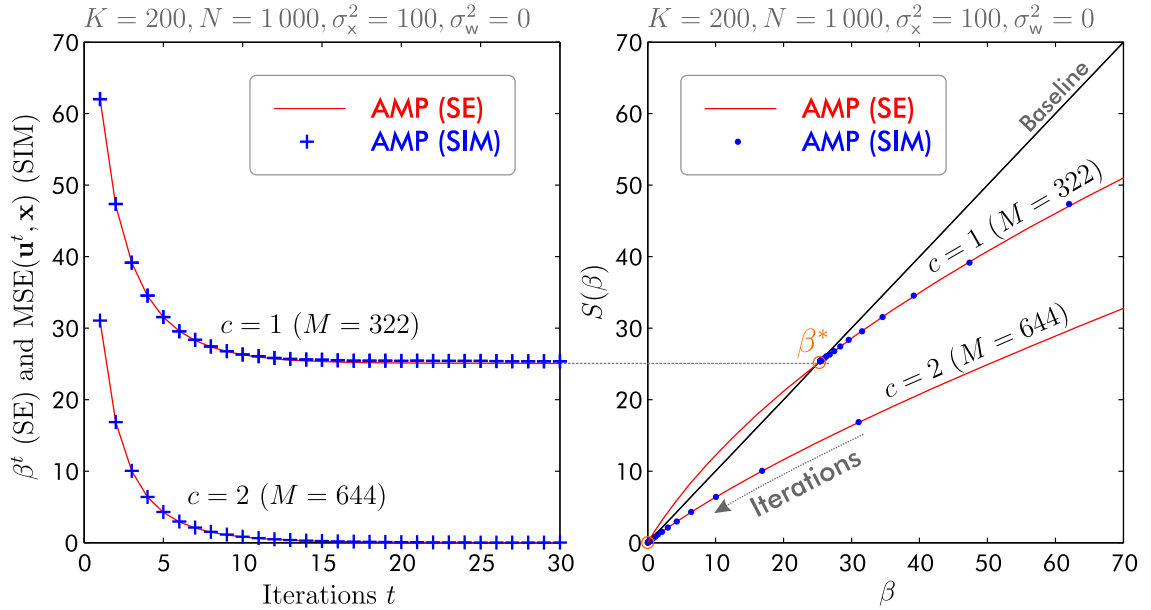

Figure 2.11 SE of AMP for two measurement multipliers $c = \{1, 2\}$.

Figure 2.11 depicts the SE over $t = [1, ..., 30]$ iterations, for two different measurement multipliers $c = \{1, 2\}$, see (1.26). The left figure shows the MSE evolution; SE and simulation are in good agreement. The right plot essentially depicts β^t versus β^{t+1}. Analytically, the red solid lines compute as

$$S(\beta) = \sigma_{\mathsf{w}}^2 + \frac{1}{\delta}\, \mathbb{E}_{\mathsf{x},\mathsf{v}} \left\{ \left[\eta\left(\mathsf{x} + \sqrt{\beta}\mathsf{v};\, \sqrt{\beta}\lambda\right) - \mathsf{x} \right]^2 \right\}, \qquad (2.101)$$

and the dots correspond to the iterations of the simulation, i.e., the tuple

$$[\text{MSE}(\mathbf{u}^t, \mathbf{x}), \text{MSE}(\mathbf{u}^{t+1}, \mathbf{x})] \approx [\beta^t, S(\beta^t)]. \tag{2.102}$$

The black solid line indicates the baseline $S(\beta) = \beta$. Scrutinizing the $c = 1$ evolution, the algorithm converges to the fixed point $\beta^* > 0$ and, hence, indicates unsuccessful denoising. This fixed point marks the first intersection (from the right) between the baseline and the analytic evolution (2.101). For an increased $c = 2$, the algorithm converges to the desired solution $\beta = 0$ that is always a fixed point of $S(\beta)$ (if $\sigma_{\mathsf{w}}^2 = 0$). The smallest c for which $\beta^* = 0$ indicates the *phase transition* of the algorithm at which the denoising becomes successful. This method can be used to analytically determine the minimal number of required measurements, as opposed to the numerical Monte Carlo experiments conducted in Section 2.2.3 and Section 2.5. The theoretical background exceeds the scope of this work, I refer the interested reader to [51, 65, 102, 103].

2.6.2 Prior Mismatch in BAMP

In Section 2.3, several priors for BAMP Algorithm 2 were specified. Let me compare BAMP's denoising performance for different priors by utilizing SE. The SE update equation (2.97) becomes

$$\beta^{t+1} = \sigma_{\mathsf{w}}^2 + \frac{1}{\delta} \mathbb{E}_{\mathsf{x}, \mathsf{v}} \left\{ \left[F\left(\mathsf{x} + \sqrt{\beta^t} \mathsf{v}; \beta^t \right) - \mathsf{x} \right]^2 \right\}. \tag{2.103}$$

In the following experiment, I generate a Bernoulli-Laplace distributed signal according to (2.29) with unit variance, i.e., $2\kappa^2 = 1 \Rightarrow \kappa = 1/\sqrt{2}$. I compare the performance of two BAMP algorithms:

BL-BAMP is Algorithm 2 with the Bernoulli-Laplace denoiser specified in (2.30). This represents the case of perfect prior knowledge.

BG-BAMP is Algorithm 2 with the Bernoulli-Gaussian denoiser specified in (2.27). This represents the case where the exact prior is not known and a Gaussian relaxation is used. The Gaussian signal variance of the denoiser is set to the variance of the Laplace distribution, i.e., $\sigma_{\mathsf{x}}^2 = 2\kappa^2 = 1$.

The analytical SE (2.103) is compared to a simulation campaign with $1\,000$ realizations of $\mathbf{x}$, $\mathbf{A}$ and $\mathbf{w}$, the MSE is averaged over the realizations. Figure 2.12 depicts
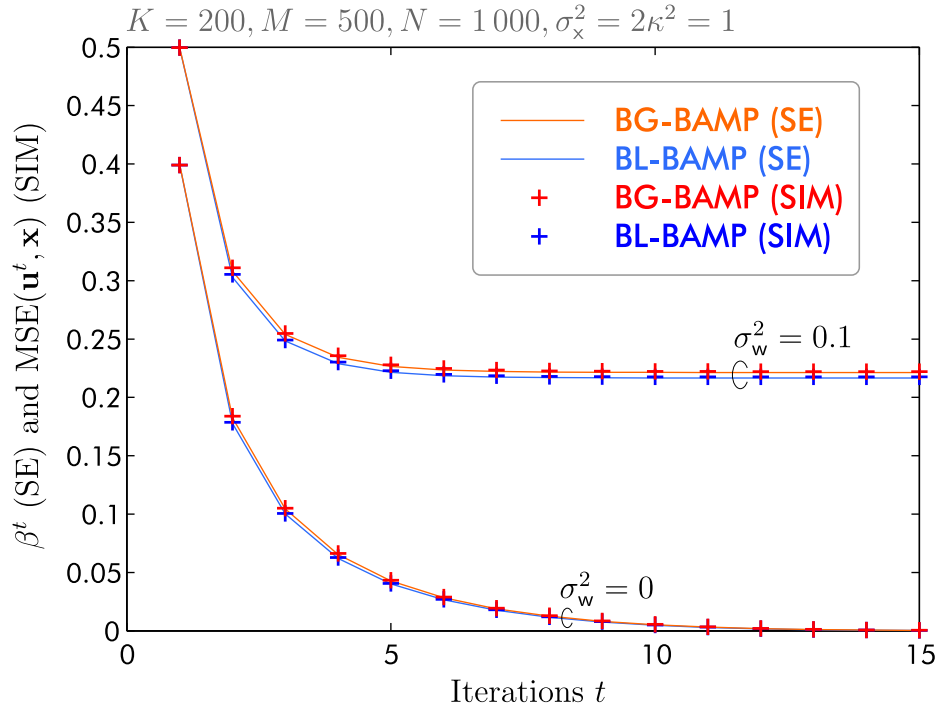
Figure 2.12 SE of two BAMP algorithms with Bernoulli-Laplace signal.

the SE over $t = [1, ..., 15]$ iterations, for two different measurement noise variances $\sigma_w^2 = \{0, 0.1\}$. The BG-BAMP with mismatching prior exhibits a slightly worsened MSE performance. In the noisy case, the remaining MSE after several iterations is slightly larger. In the noiseless case, both algorithms approach numerically perfect recovery. SE is in good agreement with the average simulation MSE.

2.6.3   Prediction of BASSAMP

I will now apply SE to BASSAMP Algorithms 3 and 4. Note that both algorithms feature a different SE, and the prediction of scalar BASSAMP is more challenging due to the extrinsic information exchange. I assume that all signal entries $\mathsf{x}_{n,b}$, $n \in \{1, ..., N\}$, $b \in \mathcal{B} = \{1, ..., B\}$, are i.i.d. *Bernoulli-Gaussian* according to (2.26).

Scalar BASSAMP

In scalar BASSAMP Algorithm 3, the zero and nonzero entries of $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_B]$ have *different state evolutions*, because the zero probabilities $\gamma_{n,b}$ are updated in each iteration. If all n-th entries of the B jointly sparse vectors are zero, the entries become more likely to be zero in each iteration, and $\gamma_{n,b}$ is increased, $\forall b \in \mathcal{B}$. Conversely, if all n-th entries are nonzero, $\gamma_{n,b}$ is decreased over iterations. Considering this in the SE, the individual estimator variances in (2.98) become

$$E_0(\beta^t, \vec{\gamma}_0^t) = \int_v \left[F\left(\sqrt{\beta^t}v; \beta^t, \vec{\gamma}_0^t \right) \right]^2 \cdot \mathcal{N}(v; 0, 1)\, dv, \tag{2.104}$$

$$E_1(\beta^t, \vec{\gamma}_1^t) = \int_v \int_x \left[F\left(x + \sqrt{\beta^t}v; \beta^t, \vec{\gamma}_1^t \right) - x \right]^2 \cdot \mathcal{N}(v; 0, 1) \cdot \mathcal{N}(x; 0, \sigma_{\mathsf{x}}^2)\, dv\, dx. \tag{2.105}$$

Note the dependency on $\vec{\gamma}_0^t$ and $\vec{\gamma}_1^t$ that depict the **zero probability evolution** of the zero entries and the nonzero entries in $\mathbf{X}$, respectively. Considering the signal prior (2.95) and assuming joint sparsity as in Definition 2.4.1, $\gamma \cdot N = K$ entries will follow the zero entry evolution where $x_{n,b} = 0$, $\forall b \in \mathcal{B}$, while $(1 - \gamma) \cdot N = N - K$ entries will follow the nonzero entry evolution where $x_{n,b} \neq 0$, $\forall b \in \mathcal{B}$. The crucial point in BASSAMP's SE is to accurately predict how $\vec{\gamma}_0^t$ and $\vec{\gamma}_1^t$ evolve over iterations t.

In (2.72), the update equation for the zero probabilities $\gamma_{n,b}^t$ was stated. For clarity, let me repeat this equation here with some rearrangements, omitting entry (row) index n as the SE yields a macroscopic result valid for all entries, and retaining vector (column) index b to capture the extrinsic information exchange between the vectors:

$$\gamma_b^t = \frac{1}{1 + \frac{1-\gamma^0}{\gamma^0} \left(\prod_{l \in \mathcal{B} \backslash b} \sqrt{\frac{\beta_l^{t-1}}{\beta_l^{t-1} + \sigma_{\mathsf{x}}^2}} \right) \exp\left(\sum_{l \in \mathcal{B} \backslash b} \frac{1}{2} \cdot \frac{\sigma_{\mathsf{x}}^2}{\beta_l^{t-1}(\beta_l^{t-1} + \sigma_{\mathsf{x}}^2)} \cdot (u_l^{t-1})^2 \right)}. \tag{2.106}$$

Note that I now consider rows of $\mathbf{U}$, and by omitting index n, the following notation holds in the remainder of this section:

$$\underbrace{[u_{n,1}, ..., u_{n,B}]}_{n\text{-th row of }\mathbf{U}} \rightarrow \underbrace{\mathbf{u} = [u_1, ..., u_B]}_{\text{Any row of }\mathbf{U}}. \tag{2.107}$$

Ultimately, I am interested in the expected value of γ_b^t with respect to the decoupled measurements $\mathbf{u}^{t-1}$ that are provided by the BAMP iteration, along with the effective noise variance β^{t-1}. Let me outline the various steps to compute this value.

A decoupled measurement u_l –now interpreted as random variable– is distributed according to (2.68). I use the latent activity variable z_l from (2.44) and formulation (2.45) to obtain

$$f_{\mathsf{u}_l|\mathsf{z}_l}(\mathsf{u}_l = u_l^{t-1}|\mathsf{z}_l = 0) = f_0(u_l^{t-1}) = \mathcal{N}(u_l^{t-1}; 0, \beta_l^{t-1}), \tag{2.108}$$

$$f_{\mathsf{u}_l|\mathsf{z}_l}(\mathsf{u}_l = u_l^{t-1}|\mathsf{z}_l = 1) = f_1(u_l^{t-1}) = \mathcal{N}(u_l^{t-1}; 0, \beta_l^{t-1} + \sigma_{\mathsf{x}}^2). \tag{2.109}$$

- If all entries in a row of $\mathbf{x} = [x_1, ..., x_B]$ are *zero*, the latent activity variables are $\mathbf{z} = [z_1, ..., z_B] = \mathbf{0}_{1 \times B}$, and the individual decoupled measurements u_l are distributed according to (2.108). This marks the *zero entry evolution*.

- If all entries in a row $\mathbf{x} = [x_1, ..., x_B]$ are *nonzero*, the latent activity variables are $\mathbf{z} = [z_1, ..., z_B] = \mathbf{1}_{1 \times B}$, and the individual decoupled measurements u_l are distributed according to (2.109). This marks the *nonzero entry evolution*.

The sum in (2.106) amounts to a new random variable (iteration index t omitted)

$$\bar{\mathsf{v}} = \sum_{l \in \mathcal{B} \backslash b} \frac{1}{2} \cdot \frac{\sigma_{\mathsf{x}}^2}{\beta_l(\beta_l + \sigma_{\mathsf{x}}^2)} \cdot \mathsf{u}_l^2 = \sum_{l \in \mathcal{B} \backslash b} \left(\underbrace{\sqrt{\frac{1}{2} \cdot \frac{\sigma_{\mathsf{x}}^2}{\beta_l(\beta_l + \sigma_{\mathsf{x}}^2)}} \cdot \mathsf{u}_l}_{\mathsf{v}_l} \right)^2. \tag{2.110}$$

- Under the $\mathsf{z}_l = 0$ hypothesis (2.108), v_l has distribution

$$f_{\mathsf{v}_l|\mathsf{z}_l}(v_l|\mathsf{z}_l = 0) = \mathcal{N}\left(v_l; 0, \frac{1}{2} \cdot \frac{\sigma_{\mathsf{x}}^2}{\beta_l + \sigma_{\mathsf{x}}^2}\right). \tag{2.111}$$

- Under the $\mathsf{z}_l = 1$ hypothesis (2.109), v_l has distribution

$$f_{\mathsf{v}_l|\mathsf{z}_l}(v_l|\mathsf{z}_l = 1) = \mathcal{N}\left(v_l; 0, \frac{1}{2} \cdot \frac{\sigma_{\mathsf{x}}^2}{\beta_l}\right). \tag{2.112}$$

All B jointly sparse vectors are expected to undergo the same SE due to the assumption of i.i.d. entries in **X** and **W**. This also implies that the expected effective noise variances are equal due to equal initial values (2.96) such that $\beta_l = \beta$, $\forall l \in \mathcal{B}$. It holds that

$$\overline{v} = \sum_{l=1}^{B-1} v_l^2. \tag{2.113}$$

In order to obtain the distribution of $\overline{v}$, I refer to [104] for the following lemma:

Lemma 2.6.1. *Consider N_V i.i.d. random variables $v_l \sim \mathcal{N}(0, \sigma_v^2)$, $l = \{1, ..., N_V\}$. The sum of the squared variables is Gamma distributed:*

$$\overline{v} = \sum_{l=1}^{N_V} v_l^2 \sim \Gamma\left(\frac{N_V}{2}, 2\sigma_v^2\right). \tag{2.114}$$

By utilizing Lemma 2.6.1 and (2.111)–(2.113), following distribution is obtained, respectively for the *zero entry evolution* and the *nonzero entry evolution*:

$$f_{\overline{v}^{t-1}|\mathbf{z}}(\overline{v}^{t-1}|\mathbf{z} = \mathbf{0}_{1\times B}) = \Gamma\left(\overline{v}^{t-1}; \frac{B-1}{2}, \frac{\sigma_x^2}{\beta^{t-1} + \sigma_x^2}\right), \tag{2.115}$$

$$f_{\overline{v}^{t-1}|\mathbf{z}}(\overline{v}^{t-1}|\mathbf{z} = \mathbf{1}_{1\times B}) = \Gamma\left(\overline{v}^{t-1}; \frac{B-1}{2}, \frac{\sigma_x^2}{\beta^{t-1}}\right). \tag{2.116}$$

Considering above steps in (2.106), the zero probability evolution of the *zero entries* computes as

$$\vec{\gamma}_0^t = \mathbb{E}_{\overline{v}^{t-1}|\mathbf{z}=\mathbf{0}} \left\{ \left(1 + \frac{1-\gamma}{\gamma}\left(\frac{\beta^{t-1}}{\beta^{t-1} + \sigma_x^2}\right)^{\frac{B-1}{2}} \exp\left(\overline{v}^{t-1}\right) \right)^{-1} \right\}$$

$$= \int_{\overline{v}} \Gamma\left(\overline{v}; \frac{B-1}{2}, \frac{\sigma_x^2}{\beta^{t-1} + \sigma_x^2}\right) \cdot \left(1 + \frac{1-\gamma}{\gamma}\left(\frac{\beta^{t-1}}{\beta^{t-1} + \sigma_x^2}\right)^{\frac{B-1}{2}} \exp\left(\overline{v}\right) \right)^{-1} d\overline{v}, \tag{2.117}$$

while the zero probability evolution of the *nonzero entries* computes as

$$\vec{\gamma}_1^t = \mathbb{E}_{\overline{v}^{t-1}|\mathbf{z}=\mathbf{1}} \left\{ \left(1 + \frac{1-\gamma}{\gamma}\left(\frac{\beta^{t-1}}{\beta^{t-1} + \sigma_x^2}\right)^{\frac{B-1}{2}} \exp\left(\overline{v}^{t-1}\right) \right)^{-1} \right\}$$

$$= \int_{\overline{v}} \Gamma\left(\overline{v}; \frac{B-1}{2}, \frac{\sigma_x^2}{\beta^{t-1}}\right) \cdot \left(1 + \frac{1-\gamma}{\gamma}\left(\frac{\beta^{t-1}}{\beta^{t-1} + \sigma_x^2}\right)^{\frac{B-1}{2}} \exp\left(\overline{v}\right) \right)^{-1} d\overline{v}. \tag{2.118}$$

Algorithm 5 SE of scalar BASSAMP for jointly sparse signals

1: Initialize $\beta^t = \sigma_{\mathsf{w}}^2 + \frac{1}{\delta}\sigma_{\mathsf{x}}^2$ and $\vec{\gamma}_0^t = \vec{\gamma}_1^t = \gamma = 1 - \frac{K}{N}$ at $t = 0$, with $\delta = \frac{M}{N}$
2: **for** $t = 1$ to $t_{\max}$ **do**
3:     Compute $E_0(\beta^{t-1}, \vec{\gamma}_0^{t-1})$ according to (2.104)   ▷ Zero entry evolution
4:     Compute $E_1(\beta^{t-1}, \vec{\gamma}_1^{t-1})$ according to (2.105)   ▷ Nonzero entry evolution
5:     $\beta^t = \sigma_{\mathsf{w}}^2 + \frac{1}{\delta}\left(\gamma E_0(\beta^{t-1}, \vec{\gamma}_0^{t-1}) + (1-\gamma)E_1(\beta^{t-1}, \vec{\gamma}_1^{t-1})\right)$ ▷ Total evolution
6:     Compute $\vec{\gamma}_0^t$ according to (2.117)   ▷ Zero prob. evolution of zero entries
7:     Compute $\vec{\gamma}_1^t$ according to (2.118)   ▷ Zero prob. evolution of nonzero entries

---

By combining above insights, I obtain the SE of scalar BASSAMP for jointly sparse signals, outlined in Algorithm 5. Lines 3-5 reproduce the behavior of the BAMP iteration (denoising by MMSE estimator), while lines 6-7 reproduce the joint extrinsic prior update of BASSAMP.

Vector BASSAMP

The vector BASSAMP Algorithm 4 is significantly simpler to handle, because no zero probability evolution has to be tracked. Note that the vector denoiser (2.86) returns a vector output. For the SE, I am interested in a single component of the denoiser output. The individual estimator variances in (2.98) for the b-th output compute as

$$E_0(\beta^t) = \int_{\mathbf{v}\in\mathbb{R}^B} \left[\left(F_{\text{vec}}\left(\mathbf{v};\beta^t\right)\right)_b\right]^2 \cdot \mathcal{N}(\mathbf{v};\mathbf{0},\beta^t\mathbf{I}_B)\,d\mathbf{v}, \tag{2.119}$$

$$E_1(\beta^t) = \int_{\mathbf{v}\in\mathbb{R}^B}\int_{\mathbf{x}\in\mathbb{R}^B} \left[\left(F_{\text{vec}}\left(\mathbf{x}+\mathbf{v};\beta^t\right)-\mathbf{x}\right)_b\right]^2 \cdot \mathcal{N}(\mathbf{v};\mathbf{0},\beta^t\mathbf{I}_B)\cdot\mathcal{N}(\mathbf{x};\mathbf{0},\mathbf{Q}_{\mathbf{x}})\,d\mathbf{v}\,d\mathbf{x}. \tag{2.120}$$

To ease the computational burden of the multidimensional integrals, I suggest to combine several random variables into a super variable according to Lemma 2.6.1. Considering this in case of i.i.d. Bernoulli-Gaussian signal entries based on (2.82) with vector denoiser (2.87), the estimator variances compute more efficiently:

$$E_0(\beta^t) = \int_v \int_{\overline{v}} \left[v \cdot \frac{\sigma_{\mathsf{x}}^2}{\beta^t + \sigma_{\mathsf{x}}^2} \cdot \frac{1}{1 + \frac{\gamma}{1-\gamma}\left(\frac{\beta^t+\sigma_{\mathsf{x}}^2}{\beta^t}\right)^{\frac{N_B}{2}}\exp\left(-\frac{v^2}{2}\frac{\sigma_{\mathsf{x}}^2}{\beta(\beta+\sigma_{\mathsf{x}}^2)}-\overline{v}\right)} \right]^2 \tag{2.121}$$
$$\cdot\,\mathcal{N}\left(v;0,\beta^t\right)\cdot\Gamma\left(\overline{v};\frac{N_B-1}{2},\frac{\sigma_{\mathsf{x}}^2}{\beta^t+\sigma_{\mathsf{x}}^2}\right)\,dv\,d\overline{v},$$

$$
\begin{aligned}
E_1(\beta^t) = \int_v \int_{\overline{v}} \int_x &\left[(x+v) \cdot \frac{\sigma_{\mathsf{x}}^2}{\beta^t + \sigma_{\mathsf{x}}^2} \cdot \frac{1}{1 + \frac{\gamma}{1-\gamma} \left(\frac{\beta^t + \sigma_{\mathsf{x}}^2}{\beta^t} \right)^{\frac{N_B}{2}} \exp\left(-\frac{(x+v)^2}{2} \frac{\sigma_{\mathsf{x}}^2}{\beta(\beta + \sigma_{\mathsf{x}}^2)} - \overline{v} \right)} - x \right]^2 \\
&\cdot \mathcal{N}\left(v; 0, \beta^t \right) \cdot \Gamma\left(\overline{v}; \frac{N_B - 1}{2}, \frac{\sigma_{\mathsf{x}}^2}{\beta^t} \right) \cdot \mathcal{N}\left(x; 0, \sigma_{\mathsf{x}}^2 \right) \, dv \, d\overline{v} \, dx.
\end{aligned}
$$

$$(2.122)$$

This way, the integral in (2.119) is reduced from B to 2 dimensions, and the integral in (2.120) is reduced from $2B$ to 3 dimensions.

Comparison

The SEs of algorithms 3 and 4 are now compared to simulation results in which $\mathrm{MSE}\left(\mathbf{U}^t, \mathbf{X} \right)$ is averaged over $1\,000$ realizations of **X**, **A** and **W**. The results are depicted in Figure 2.13, including a comparison to BAMP Algorithm 2 that operates on single vectors without exploiting the joint sparsity. BASSAMP significantly outperforms BAMP in terms of recovery MSE and denoising performance (steeper slope). Vector BASSAMP has a steeper denoising slope than scalar BASSAMP because no extrinsic information exchange is required. Eventually, both BASSAMP algorithms attain the same MSE performance. For small B, the SE of scalar BASSAMP is too optimistic in the transition phase, i.e., the convergence slope is steeper than in the simulation. The SE of vector BASSAMP is accurate for all B, since no zero probability evolution has to be tracked.

Figure 2.14 illustrates the zero probability evolution of *scalar BASSAMP*, with $B = \{2, 4, 8\}$ jointly sparse signals. The more signals, the higher the confidence in the extrinsic information, and the steeper the slope. The SE prediction becomes better with increasing B; the expected value of (2.106) does not seem to capture the full information exchange for small B.

The proposed SE of scalar BASSAMP should be seen as an approximate result, while the SE of vector BASSAMP is accurate. Both algorithms approach the same recovery MSE.
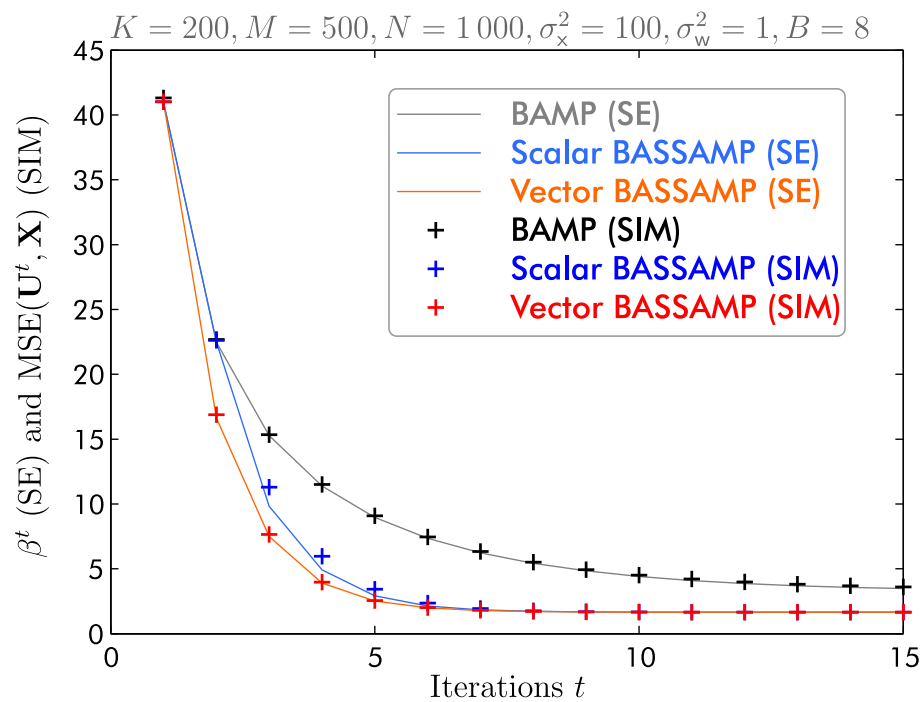
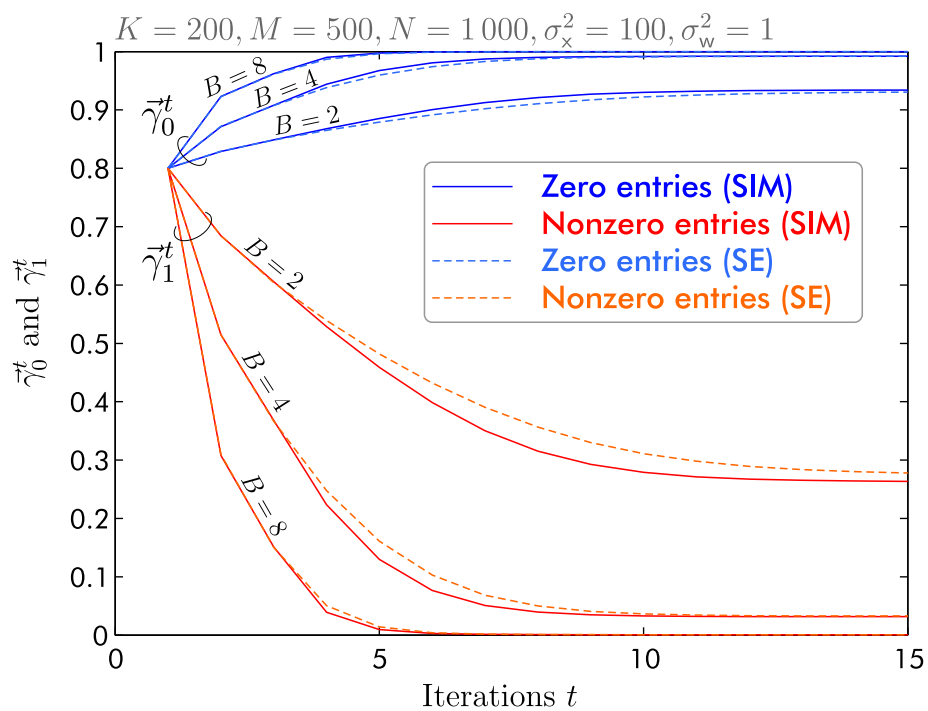Figure 2.13 SE of BASSAMP with i.i.d. Bernoulli-Gaussian signal.

Figure 2.14 Zero probability evolution of scalar BASSAMP.

2.7   Summary

In this chapter, I presented and investigated various AMP-based algorithms that will later be utilized by CS-RFID. The proper choice of the algorithm depends on the amount of available prior knowledge — for convenience, Table 2.1 depicts the prior knowledge that is exploited by the algorithms. The basic AMP algorithm requires

	Sparsity	Bayesian prior	Joint Sparsity
AMP	✓	✗	✗
BAMP	✓	✓	✗
BASSAMP	✓	✓	✓

Table 2.1 Utilization of prior knowledge.

coarse knowledge of the number of nonzero entries K such that the tuning parameter λ can be chosen accordingly, and the proper solution sparsity is promoted. If the Bayesian prior distribution is known, the BAMP algorithm is able to improve the recovery in terms of accuracy and convergence. Additional structure such as joint sparsity in case of multiple measurement vectors further improves the recovery performance. The joint sparsity structure is exploited by the BASSAMP algorithm that I introduced in two flavors: The scalar version performs an extrinsic information exchange among jointly sparse entries, while the vector version enforces the structure by employing a vector denoiser function that is based on a multivariate signal prior. The vector denoiser is also capable of capturing correlation among the signal entries.

The algorithms were compared by empirical phase transition curves based on Monte Carlo simulations, and by the SE formalism that allows to predict the algorithmic behavior analytically. AMP has been demonstrated to be a very versatile tool with excellent CS recovery properties, perfectly suited for practical implementations.

All algorithms are now established and ready to be deployed in CS-RFID.

Chapter 3

RFID with Compressed Sensing

This chapter introduces the fundamental concept of CS-RFID. In the considered RFID setup, a multitude of tags is to be identified by a reader with a single receive antenna. The tag acquisition is formulated as a CS measurement, the tag identifiers are obtained via CS recovery by AMP. In order to guarantee reliable identification of all tags, two identification protocols are proposed.

The following contributions are based on my work in [74, 76, 105–107]:

■ The problem of *tag acquisition* is formulated as CS measurement (1.19). It is described how the efficient AMP Algorithm 1 can be utilized to acquire the tag identifiers.

■ An identification protocol for scenarios with a *fixed inventory* is proposed. It enables very quick and robust tag identification.

■ An identification protocol for scenarios with an *arbitrary inventory* is proposed. This protocol constitutes a more flexible approach to robust tag identification.

■ The performance of both protocols is evaluated analytically and numerically. A comparison to the pervasive FSA protocol demonstrates that the novel protocols in conjunction with CS and AMP are by far quicker and more robust with respect to noise.

3.1   Scenario and Channel Model

I consider a scenario where K tags have to be identified by a single reader. The CS-RFID schemes introduced in this thesis can handle all kinds of tags, i.e., passive, semi-passive or active tags. I focus on passive tags that are powered by the CW carrier signal emitted by the reader, because this is the most challenging case where the tag responses are restricted to binary symbols. Furthermore, this is the pervasive approach used in most applications [8, 10, 12]. Such tags employ backscatter modulation to convey information in the uplink from tag to reader, thereby generating Amplitude Shift Keying (ASK) modulated signals as tag responses. The two resulting ASK symbol levels are henceforth denoted $b_0 \in \mathbb{R}$ and $b_1 \in \mathbb{R}$ corresponding to binary 0 and binary 1, respectively.

In this chapter, the reader employs a single receive antenna. The reader antenna setup can either be monostatic (transmit and receive at same antenna) or bistatic (transmit and receive at separate antennas). I focus on the bistatic case, because it is the more general case when it comes to channel modeling. Such a setup is depicted in Figure 3.1.

The aspired data-rates for RFID range from $40\,\mathrm{kbit/s}$ to $640\,\mathrm{kbit/s}$ [39]. This entails narrow-band transmissions whose physical channels can be modeled by a complex coefficient that characterizes magnitude and phase change. I employ a widely used dyadic channel model [108–110] that consists of a forward channel coefficient from reader to tag, denoted $h_k^{(\mathrm{f})} \in \mathbb{C}$, and a backward channel coefficient from tag to reader, denoted $h_k^{(\mathrm{b})} \in \mathbb{C}$. The total channel from reader to tag k and back is modeled by

$$h_k = h_k^{(\mathrm{f})} h_k^{(\mathrm{b})}. \tag{3.1}$$

The channels are assumed to be static, i.e., to not change significantly throughout a transmission phase. Furthermore, the reader's carrier signal is subject to multipath propagation that causes small-scale fading [111–113]. By changing the tag position within fractions of the wavelength, the propagation components may interfere constructively or destructively. Consequently, the channel coefficient magnitude is subject to spatial fading, which is widely modeled according to a Rayleigh or Rician distribution [108–112]. Based on those findings, I assume Rayleigh distributed channel coefficient magnitudes and uniformly distributed angles (phases). The forward and backward channel coefficients are hence drawn from a circularly symmetric complex-valued Gaussian distribution with zero mean and unit variance. For a bistatic reader setup, forward channel $h_k^{(\mathrm{f})}$ and backward channel $h_k^{(\mathrm{b})}$ are modeled
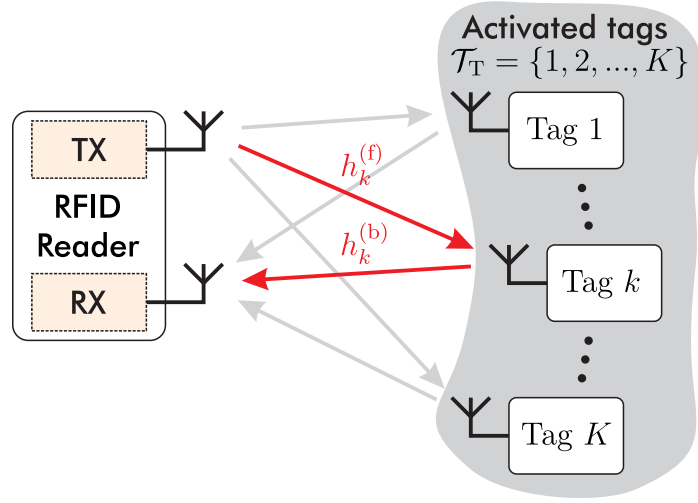
Figure 3.1 A bistatic reader identifies K tags.

by *independently* drawn coefficients. This implies that the reader antennas are sufficiently spaced in order to obtain uncorrelated channels; note that this is an assumption to obtain a simple model and not a requirement for the proposed identification scheme. The monostatic channel model is easily obtained by setting $h_k^{(\mathrm{b})} = \left(h_k^{(\mathrm{f})} \right)^*$.

The carrier-leakage from transmit to receive antenna is assumed to be negligible [38]. Activated tags respond synchronized and with constant symbol duration.

3.2 Tag Acquisition via Compressed Sensing

Tags that are brought close enough to the reader, i.e., within read range, are activated and powered by the reader's CW carrier signal. According to Figure 3.1, K tags are activated and have to be identified by the reader.

Definition 3.2.1 (Set of activated tags). *The set of activated tags encompasses all tags that are activated by the reader for identification. The activated tags are labeled by consecutive integer numbers that comprise the set of activated tags:*

$$\mathcal{T}_{\mathrm{T}} := \{1, ..., k, ..., K\}. \tag{3.2}$$

The problem of *tag acquisition* refers to the acquisition of information about a tag that enables communication with the tag. This information generally does not identify the tag. In FSA, the reader broadcasts a query, and each activated tag randomly picks a slot in a frame and transmits a 16-bit random sequence (RN16).
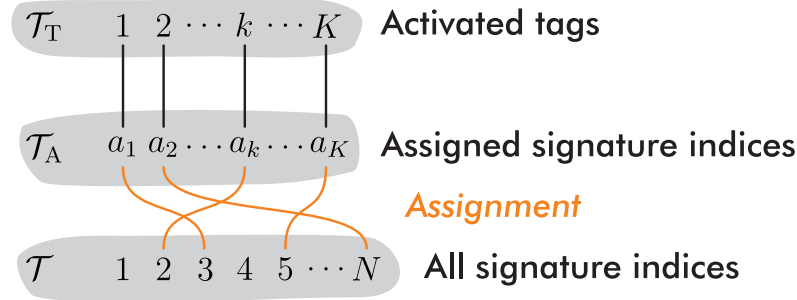
$$|\mathcal{T}_{\mathrm{T}}| = |\mathcal{T}_{\mathrm{A}}| = K \qquad |\mathcal{T}| = N \gg K$$

Figure 3.2 Signature sequence assignment.

The reader acquires the RN16 sequences from collision-free slots and is henceforth able to communicate with these tags. Subsequently, all tags are listening to the reader that broadcasts the acquired RN16 sequences in a sequential manner; if a tag recognizes its RN16 sequence, it responds with the data payload that identifies the tag.

In my proposed CS-RFID approach, all activated tags respond *simultaneously* to a query from the reader. Each tag responds with a *signature sequence*, or briefly, *signature*

Definition 3.2.2 (Set of signature indices)**.** *There are N signatures sequences available in total. The corresponding indices are collected in the set of signature indices*

$$\mathcal{T} := \{1, ..., n, ..., N\}. \tag{3.3}$$

Definition 3.2.3 (Set of assigned signature indices)**.** *Each activated tag $k \in \mathcal{T}_{\mathrm{T}}$ is associated with a signature index $a_k \in \mathcal{T}$, the tag responds with the corresponding signature during tag acquisition. The set of assigned signature indices is*

$$\mathcal{T}_{\mathrm{A}} := \{a_1, ..., a_k, ..., a_K\}. \tag{3.4}$$

Figure 3.2 illustrates above definitions. The two different methods of assignment, fixed or random, will be discussed in sections 3.3 and 3.4, respectively. Let me specify the *signature sequence properties*:

- Activated tag k responds with a signature $\mathbf{s}_{a_k} \in \{b_0, b_1\}^M$ that comprises the ASK symbols $b_0, b_1 \in \mathbb{R}$, i.e., the two amplitudes of backscatter modulation.
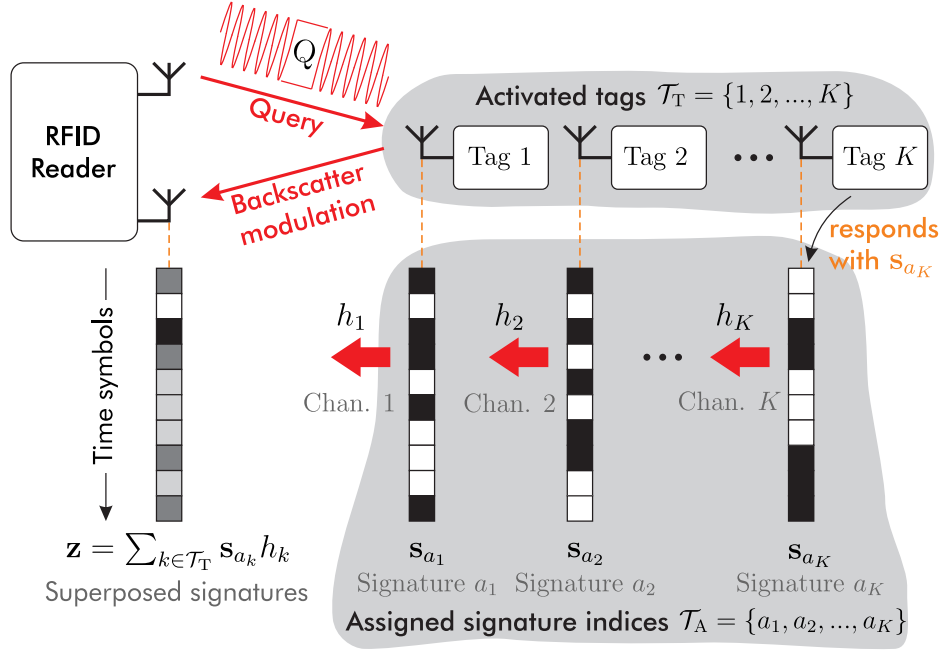
Figure 3.3 Tags respond simultaneously — physical layer illustration.

- The signatures are generated pseudo-randomly, each with a certain seed [114]. Both ASK symbols occur with equal probability $P\{b_0\} = P\{b_1\} = 0.5$.

- The N signature sequences form the columns of signature matrix $\mathbf{S} = [\mathbf{s}_1, ..., \mathbf{s}_N] \in \{b_0, b_1\}^{M \times N}$.

- The reader knows all possible signatures (e.g., by knowing the seeds of the pseudo-random generator). Therefore, it knows $\mathbf{S}$.

For tag acquisition, the reader broadcasts a query and receives a superposition of signature sequences $\mathbf{s}_{a_k}$ that are weighted with the respective channel coefficient h_k, see Figure 3.3. This is formulated as a CS measurement

$$\mathbf{z} = \sum_{k=1}^{K} \mathbf{s}_{a_k} h_k + \mathbf{w} = \mathbf{S}\mathbf{x} + \mathbf{w}, \tag{3.5}$$

where the nonzero entries of $\mathbf{x} \in \mathcal{X}_K \subset \mathbb{C}^N$ store the complex-valued channel coefficients and dictate which columns of $\mathbf{S}$ are selected, and with $\mathbf{w} \in \mathbb{C}^M$ being additive measurement noise with i.i.d. entries from a circularly symmetric complex-valued Gaussian distribution with zero mean and variance σ_{w}^2. The interpretation as CS measurement is visualized by Figure 3.4 (noise omitted).
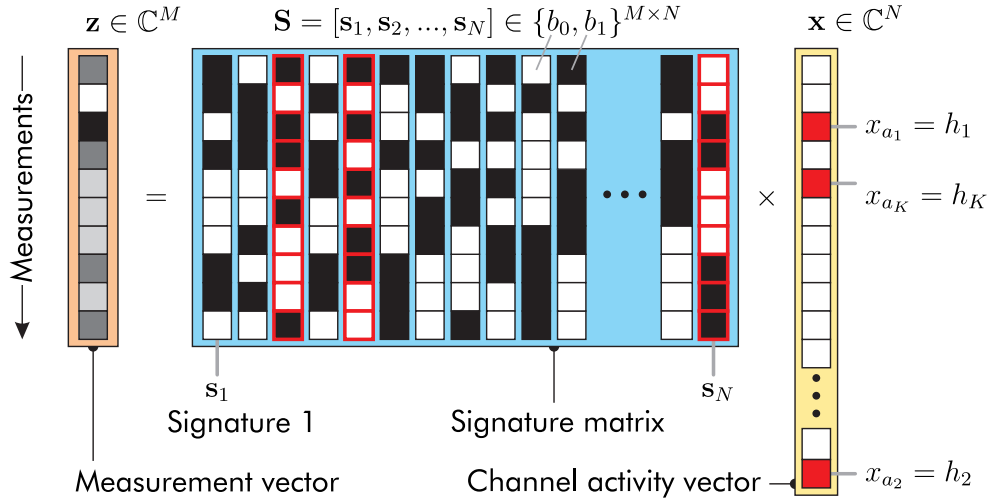
Figure 3.4 Tags respond simultaneously — interpreted as CS measurement.

The K assigned signatures constitute a very small subset of the total amount of signatures N — the reasons for this become clear in sections 3.3 and 3.4. Consequently, the **channel activity vector** $\mathbf{x}$ is sparse with $K \ll N$ nonzero entries. The goal is to recover $\mathbf{x}$ from $\mathbf{z}$ knowing $\mathbf{S}$, because the locations of the nonzero entries in $\mathbf{x}$ indicate which signatures have been chosen by the tags. This is the information used to establish a communication with the tags, i.e., the identifiers in CS-RFID are associated with the signature indices.

In this chapter, I use the AMP Algorithm 1 with complex soft thresholding (2.12) to recover $\mathbf{x}$. This algorithm requires the sensing matrix to have zero mean and normalized columns. However, $\mathbf{S}$ in (3.5) features neither. I will now elaborate how an AMP compliant formulation is obtained; let me first tackle the zero-mean problem.

Since all signatures comprise an equal number of b_0 and b_1 (in expectation or by design), they all have the same mean

$$\overline{s} = \frac{b_0 + b_1}{2}. \tag{3.6}$$

After subtracting the mean from the signatures, they feature binary antipodal symbols $\{-\overline{b}, \overline{b}\}$:

$$b_0 - \overline{s} = -\frac{b_1 - b_0}{2} =: -\overline{b},$$
$$b_1 - \overline{s} = \frac{b_1 - b_0}{2} =: \overline{b}. \tag{3.7}$$

Consequently, (3.5) can be written as

$$\mathbf{z} = \mathbf{S}\mathbf{x} + \mathbf{w} = \left(\overline{\mathbf{A}} + \overline{s} \cdot \mathbf{1}_{M \times N}\right)\mathbf{x} + \mathbf{w} = \underbrace{\overline{\mathbf{A}}\mathbf{x} + \mathbf{w}}_{\mathbf{y}} + \underbrace{\overline{s}\sum_{n=1}^{N} x_n}_{\overline{z}}, \qquad (3.8)$$

where $\overline{\mathbf{A}} \in \{-\overline{b}, \overline{b}\}^{M \times N}$ has zero-mean columns, and $\overline{z}$ is the measurement mean that depends on $\mathbf{x}$. The mean $\overline{z}$ can be computed from $\mathbf{z}$ without knowledge of $\mathbf{x}$:

$$
\begin{aligned}
\frac{1}{M}\sum_{m=1}^{M} z_m &= \frac{1}{M}\sum_{m=1}^{M}\sum_{n=1}^{N} s_{m,n}x_n + \underbrace{\frac{1}{M}\sum_{m=1}^{M} w_m}_{\overline{w}} \\
&= \frac{1}{M}\sum_{m=1}^{M}\sum_{n=1}^{N}(\overline{a}_{m,n} + \overline{s})x_n + \overline{w} \\
&= \frac{1}{M}\sum_{n=1}^{N} x_n \underbrace{\sum_{m=1}^{M}\overline{a}_{m,n}}_{=0} + \overline{s}\underbrace{\sum_{n=1}^{N} x_n}_{=\overline{z}}\underbrace{\sum_{m=1}^{M}\frac{1}{M}}_{=1} + \overline{w} \\
&= \overline{z} + \overline{w}.
\end{aligned}
\qquad (3.9)
$$

The noise has zero mean and therefore vanishes in expectation: $\mathbb{E}_{\mathbf{w}}(\overline{w}) = 0$. Assuming ergodicity and a sufficient number of samples M, $\overline{w}$ approaches zero. The aspired zero-mean formulation of the CS measurement is obtained by

$$\mathbf{y} = \overline{\mathbf{A}}\mathbf{x} + \mathbf{w} \approx \mathbf{z} - \frac{1}{M}\sum_{m=1}^{M} z_m, \qquad (3.10)$$

it becomes exact in the noiseless case, or for large M.

The sensing matrix $\overline{\mathbf{A}}$ in (3.10) features zero-mean columns, which is the first requirement for Algorithm 1. The second requirement –columns with unit ℓ_2-norm– is achieved by scaling:

$$\mathbf{A} = \frac{1}{\overline{b}\sqrt{M}} \cdot \overline{\mathbf{A}}. \qquad (3.11)$$

This yields an $\mathbf{A}$ with Bernoulli-distributed entries that take the values $\{-\frac{1}{\sqrt{M}}, \frac{1}{\sqrt{M}}\}$ with equal probability. As mentioned in Section 1.2.3, such matrices satisfy the RIP with overwhelming probability, and the number of samples –or equivalently, the signature length– M for successful recovery is dictated by (1.26), it depends on c, K and N. In the acquisition phase, the reader advertises M to the tags via query. The

proper choice of the measurement multiplier c in (1.26) was investigated in Section 2.2.3. The number of activated tags K has to be estimated [48, 115]. Considering high and moderate SNR as defined in (2.15) and $c \in [2, 3]$, $\mathbf{x}$ is recovered well by AMP and K can be estimated by sorting the entries in $\widehat{\mathbf{x}}$ with descending magnitude and observing the gap between Correct Detections (CDs) and False Alarms (FAs), see [107]. The ℓ_1-optimal tuning parameter λ_{ℓ_1} is selected according to (2.14). The number of signatures N is known to the reader.

Executing Algorithm 1 with $\mathbf{A}$ and $\mathbf{y}$, an estimate of the channel activity vector, $\widehat{\mathbf{x}}$, is obtained. The **detected set of assigned signature indices** $\widehat{\mathcal{T}}_A = \{\widehat{a}_1, ..., \widehat{a}_K\}$ is assembled by collecting the indices of the K largest entries in $|\widehat{\mathbf{x}}|$. The recovery $\widehat{\mathbf{x}}$ can be multiplied by $\overline{b}\sqrt{M}$ to obtain the original scaling that was lost due to the normalization in (3.11); however, the absolute scaling is not important for tag acquisition.

Tag Acquisition with AMP

1. K tags are put in read range in order to be identified by the reader.

2. The reader estimates or assumes K and chooses $c \in [2, 3]$. It then computes M according to (1.26).

3. The reader broadcasts a query and advertises the signature length M to the activated tags.

4. The tags respond simultaneously, each with its signature sequence. This is cast as CS measurement (3.5). The measurement yields $\mathbf{z}$ which is used to compute $\mathbf{y}$ according to (3.10).

5. CS recovery with AMP is performed by Algorithm 1 with complex soft thresholding (2.12). The algorithm uses $\mathbf{A}$ from (3.11) and $\mathbf{y}$, and the threshold parameter λ is selected according to (2.14). The algorithm yields the signal estimate $\widehat{\mathbf{x}}$.

6. The *detected set of assigned signature indices* $\widehat{\mathcal{T}}_A$ is obtained by collecting the indices of the K largest entries in $|\widehat{\mathbf{x}}|$.

This completes the tag acquisition phase. Subsequently, the reader is ready for the data read-out phase that identifies the tags. Protocols that enable *reliable tag identification* despite noise are discussed in the following sections 3.3 and 3.4.

3.3 Identification with Fixed Signature Assignment (CSF)

Many RFID applications feature a "closed system" were all tags that *could* be activated by the reader are known. I refer to this as the **fixed inventory** of tags. Examples are sensor networks, stores, libraries, offices — virtually all applications where a set of tags is maintained. Maintaining a fixed inventory requires a database that lists all tags and associates them with sensors, products or goods. In view of the tag acquisition scheme proposed in Section 3.2, each tag is uniquely associated with a signature sequence, and the database stores all possible signatures.

Let me lay out an example of a library that uses CS for tag acquisition. Each book features a tag that is uniquely associated with a signature sequence. There are N signatures in total, and all N signatures are uniquely linked to the total inventory of N books (tags). Each column in signature matrix $\mathbf{S}$ uniquely represents a book in the library, as does each entry in channel activity vector $\mathbf{x}$. A customer now brings a small subset of $K \ll N$ books to the checkout, where the tags are activated — this yields $\mathcal{T}_{\mathrm{A}}$, the set of signature indices assigned to those activated tags. After broadcasting the signature query and performing CS measurement (3.5), the reader recovers the channel activity vector $\mathbf{x}$ using AMP. In case of perfect recovery $\widehat{\mathbf{x}} = \mathbf{x}$, the identification is completed since the support of $\widehat{\mathbf{x}}$, given by $\mathcal{T}_{\mathrm{A}}$, corresponds to the signature indices of the K books.

Fixed Signature Assignment

- There is a *fixed inventory* that encompasses N tags.

- Each tag is *uniquely associated* with a signature sequence. Hence, the set of signature indices $\mathcal{T} = \{1, ..., N\}$ corresponds to the set of all tags.

- $K \ll N$ tags are activated by the reader and respond simultaneously with K distinct signatures.

- *Perfect recovery* of channel activity vector $\mathbf{x}$ yields *complete identification*.

3.3.1   Identification Cycle

At low SNR or an insufficient number of measurements M, AMP yields an imperfect recovery $\widehat{x}$ that may contain FAs. This behavior was investigated in [107] for sparse binary vectors. In order to guarantee a reliable recovery nevertheless, I introduce **identification cycles** as depicted in Figure 3.5.
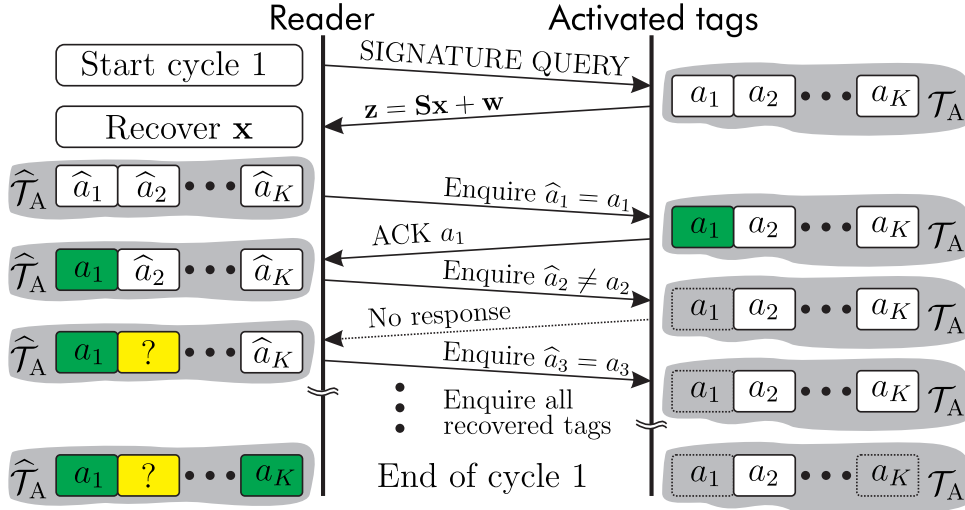

Figure 3.5 Identification cycle for fixed signature assignment.

In the first step, the reader broadcasts a signature query and the CS measurement process (3.5) is triggered. Channel activity vector x is recovered by AMP and the set of assigned signatures is detected, yielding $\widehat{\mathcal{T}}_A$.

In the second step, the reader enquires all signature indices in $\widehat{\mathcal{T}}_A$. A truly activated tag whose signature index was detected correctly acknowledges the enquiry and is, thereby, identified; it is silent in future identification cycles. If a FA is enquired, no tag will respond and the reader continues after a timeout.

Each identification cycle starts with a signature query and, therefore, a CS measurement that encompasses the tags that have previously been missed. The identification cycles are repeated until no tags respond to a query anymore and all tags are identified successfully.

3.3.2   Analytical Performance Investigation

I will now compare the performance of CS-RFID with fixed signature assignment – denoted **CSF** in the sequel– to the widely used FSA protocol. The first figure of merit is the bit overhead for identification.

Definition 3.3.1 (Bit overhead for identification). *The bit overhead for identification is defined as the number of bits required to identify the K activated tags. It is denoted by β.*

Let me begin with the FSA protocol. The K activated tags randomly choose a time-slot in a frame with F slots to transmit a 16-bit random number (RN16). Choosing frame size $F = K$ (equal to the number of activated tags) maximizes the probability of choosing a collision-free slot [42], which turns out to be $e^{-1} \approx 0.368$ according to a binomial distribution. The reader acknowledges (A_{FSA} bits) the successfully received (collision-free) RN16 numbers and reads out the data payload (D bits) of the corresponding tags. Tags with collided RN16 numbers are scheduled to transmit again in a subsequent frame. Omitting signaling overhead, the optimal (lowest possible) bit overhead for identification becomes

$$\beta_{\mathrm{FSA}} = \underbrace{16eK}_{\text{Acquisition}} + \underbrace{K(A_{\mathrm{FSA}} + D)}_{\text{Data read-out}}. \tag{3.12}$$

Typical values from the EPCglobal standard [39] are: $A_{\mathrm{FSA}} = 18$ bits consists of 2 bits ACK command overhead plus RN16 echo to acknowledge the slots, while $D = 48$ bits is the data payload assumed to contain the Electronic Product Code (EPC) and the Cyclic Redundancy Check (CRC). Not knowing the inventory in advance necessitates a data read-out, which is not required in CSF.

In the acquisition phase of CSF, all K activated tags simultaneously transmit a signature of length M according to (1.26). In the noiseless case, the recovery is perfect and the K activated tags are found by enquiry (E bits) and acknowledgment (A_{CSF} bits). The optimal bit overhead for identification reads

$$\beta_{\mathrm{CSF}} = \underbrace{\left\lceil cK \log \frac{N}{K} \right\rceil}_{\text{Acquisition}} + \underbrace{K(E + A_{\mathrm{CSF}})}_{\text{Data read-out}}. \tag{3.13}$$

Let me provide typical values: Since every tag knows its signature index $a_k \in \mathcal{T}_{\mathrm{A}}$, an enquiry contains a tag index $\widehat{a}_k \in \widehat{\mathcal{T}}_{\mathrm{A}}$ in binary representation and therefore requires $E = \lceil \log_2(N) \rceil + 2$ bits, including two-bit command overhead. The ACK response can be very simple and contains $A_{\mathrm{CSF}} = 2$ bits. If additional information is to be read out (e.g., sensor data), a data payload could be added to the ACK response; this is not the case in the identification-only scenario.

Figure 3.6 depicts the theoretically optimal bit overhead for FSA and CSF that is achieved in the noiseless case. From (3.13), it is evident that the bit overhead
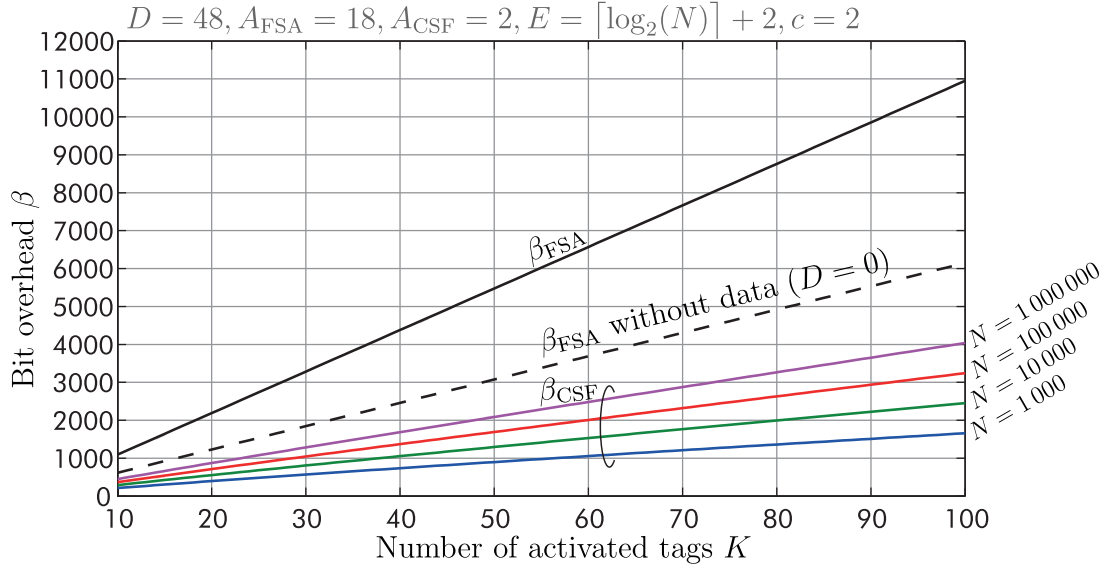
Figure 3.6 Optimal bit overhead for identification: FSA versus CSF.

of CSF depends on the inventory size N. The required signature length increases logarithmically with an increasing inventory, which translates to a longer acquisition phase. It was found that CSF outperforms FSA as long as the inventory size is below $N \approx 10^{15}$ tags, where both overheads are about equal. For reasonably sized inventories, CSF has a significantly lower bit overhead for identification and strongly outperforms FSA. This enables a quicker, more energy-efficient identification of a multitude of tags.

Note that both bit overhead formulas apply to the noiseless case and constitute a lower bound (optimum). The lower the SNR, the more identification cycles are needed in CSF, and the more frames are needed in FSA. A performance evaluation for variable SNR follows in Section 3.5.

3.4 Identification with Random Signature Assignment (CSR)

Fixed signature assignment tackled the case of a fixed tag inventory that was contained within a strictly limited, albeit large set $\mathcal{T}$. A more general approach is aspired for **arbitrary inventories**. The widely employed EPCglobal standard for tag identification [39] foresees tags that randomly choose an RN16 sequence during the acquisition phase. Consequently, the K activated tags randomly pick a sequence out of 2^{16} possible sequences, which renders the K responses distinct with overwhelm-

ing probability. Each tag that obeys the standard can be identified by the reader, no inventory maintenance is required. Note that the EPCglobal standard uses the FSA protocol.

Adopting this idea for CS-RFID, the activated tags randomly pick a signature sequence index from $\mathcal{T}$. The total number of signatures N is now a *design parameter*, the total tag population is not limited to N but is arbitrary.

Let me revisit the library example. The number of books in the library can now be arbitrarily large, a book is identified by the data payload of the tag and not by its signature sequence. The set of signature indices $\mathcal{T}$ contains N signatures and is now independent of the number of books. The user brings $K \ll N$ books (tags) to the checkout, and each activated tag randomly selects a signature index a_k from $\mathcal{T}$ — this constitutes the set of assigned signature indices $\mathcal{T}_A$. After receiving the superimposed signatures, the reader recovers $\mathbf{x}$. In case of perfect recovery, the support of $\widehat{\mathbf{x}}$, i.e., $\widehat{\mathcal{T}}_A$, contains the assigned signature indices. The reader enquires the signature indices; instead of an acknowledgment, the tags respond with the data payload (e.g., product code) for identification.

> **Random Signature Assignment**
>
> - There is an *arbitrary inventory*.
>
> - Each tag *randomly selects* a signature from $\mathcal{T} = \{1, ..., N\}$.
>
> - $K \ll N$ tags are activated by the reader and respond simultaneously with K signatures that are *distinct with high probability*.
>
> - *Perfect recovery* of channel activity vector $\mathbf{x}$ yields the K assigned signature indices.
>
> - The K signature indices are used to identify the K associated tags in an *additional data read-out* phase.

3.4.1 Identification Cycle

The identification cycle for random signature assignment is illustrated by Figure 3.7. Note that random signature assignment faces the problem of repeatedly chosen signatures in the enquiry step — if two or more tags choose the same signature, they will respond simultaneously during data read-out and produce a *collision*. A collision
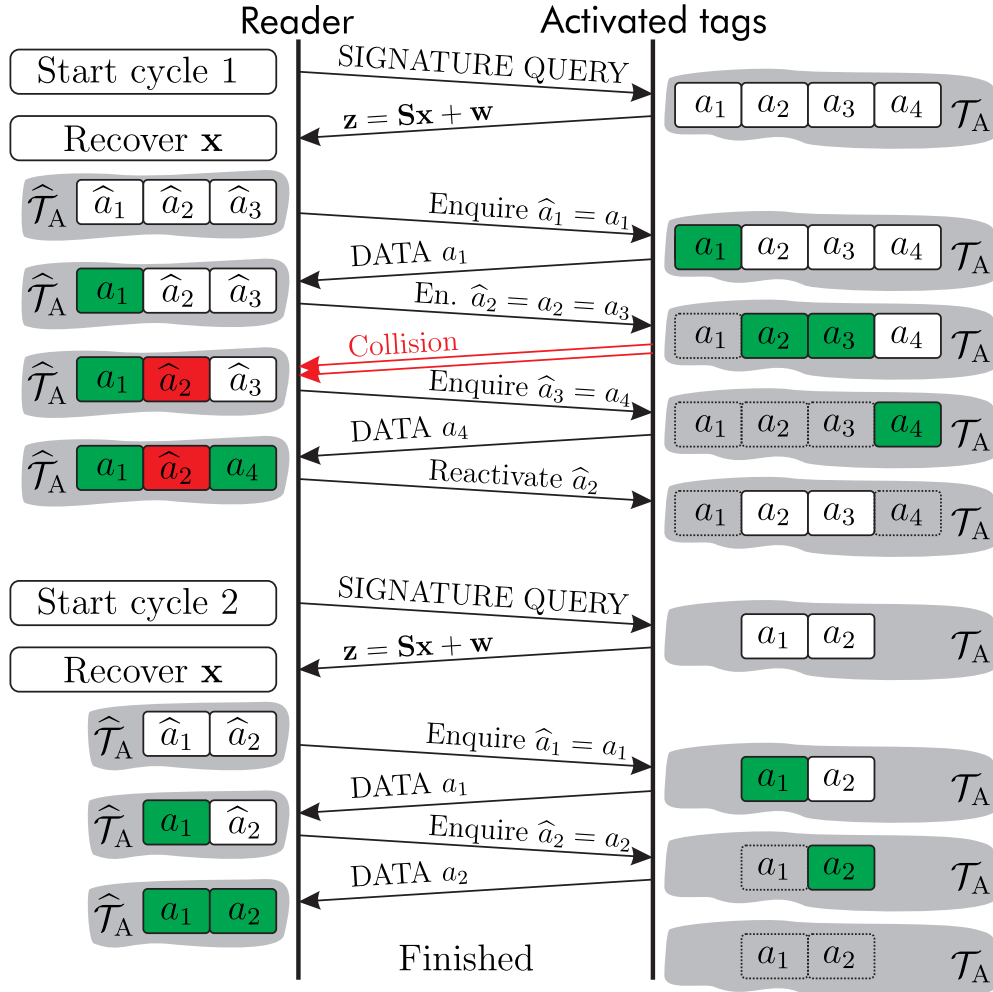
Figure 3.7 Identification cycle for random signature assignment.

can be detected by the reader via, e.g., CRC code. The usage of a CRC code is also helpful in case of erroneous read-out due to noise. If the CRC check fails, the tag(s) associated with the current signature are *reactivated* after the identification cycle. In the subsequent cycle, they again choose a signature and retransmit their data, which is hoped to be error- (and collision-) free now, and again repeated otherwise.

3.4.2 Analytical Performance Investigation

CS-RFID with random signature assignment –denoted **CSR** in the sequel– allows to control the sparsity of $\mathbf{x}$. The number of possible signatures N can be made very large compared to the number of activated tags K, i.e., $N \gg K$. This allows to reduce the probability of a collision in the data read-out. However, the number of

signatures should not be made arbitrarily large, since an increase in N also increases the required number of measurements M according to (1.26). I will now introduce the relevant probabilities and use them to discuss the optimal choice of N that minimizes the bit overhead for identification.

Definition 3.4.1 (Collision probability). *Assume that a specific tag (out of K) has chosen a specific signature (out of N), the probability that m out of the $K-1$ remaining tags choose the same signature is stated by the binomial distribution*

$$p_{\mathrm{B}}\left(m; K-1, \frac{1}{N}\right) = \binom{K-1}{m}\left(\frac{1}{N}\right)^m\left(1-\frac{1}{N}\right)^{K-1-m}. \qquad (3.14)$$

Definition 3.4.2 (Resolvable tag probability). *The probability that the signatures have been assigned uniquely and that the data read-out is collision-free reads*

$$P_R(K, N) := p_{\mathrm{B}}\left(0; K-1, \frac{1}{N}\right) = \left(1-\frac{1}{N}\right)^{K-1}. \qquad (3.15)$$

Definition 3.4.3 (Unresolvable tag probability). *The probability that the signatures have been assigned repeatedly and that the data read-out features one or more collision(s) reads*

$$P_U(K, N) := \sum_{m=1}^{K-1} p_{\mathrm{B}}\left(m; K-1, \frac{1}{N}\right) = 1 - P_R(K, N). \qquad (3.16)$$

The optimal bit overhead for identification with CSF was stated in (3.13). For CSR, the possibility of unresolvable tags in the data read-out due to collision (same signature choice) has to be included. The identification cycle depicted in Figure 3.7 enquires the nonzero entries in the recovered channel activity vector $\widehat{\mathbf{x}}$. Two kinds of nonzero entries in $\mathbf{x}$ (and equivalently, in $\widehat{\mathbf{x}}$ in the noiseless case) are distinguished:

Resolvable entries: These signatures were chosen by one tag only, which happens with probability $P_R(K, N)$. No collision occurs during data read-out. The expected number of resolvable entries is $P_R(K, N) \cdot K$.

Unresolvable entries: These signatures were chosen by two or more tags, which happens with probability $P_U(K, N)$. However, the expected number of unresolvable entries is lower than $P_U(K, N) \cdot K$, since a repeatedly chosen signature occupies only one entry in $\mathbf{x}$, consider the example in Figure 3.8. In Figure 3.8 (a), entry x_2 is unresolvable because three tags chose signature 2. In
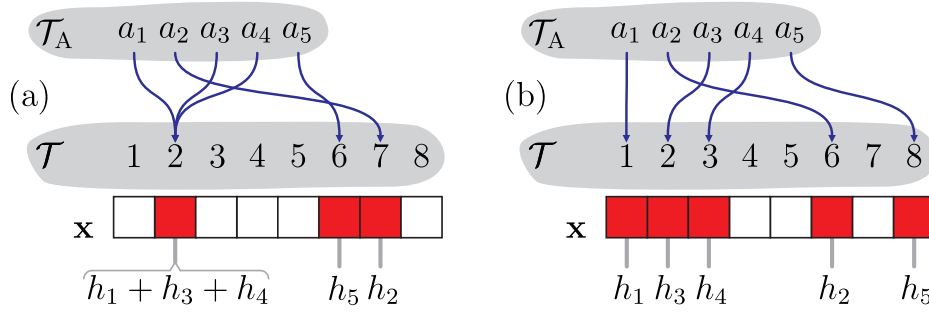
Figure 3.8 Two random signature assignment examples ($K = 5$, $N = 8$).

Figure 3.8 (b), the signatures were chosen uniquely and all entries are resolvable. Channel activity vector $\mathbf{x}$ in (a) features $3 < K$ nonzero entries, while $\mathbf{x}$ in (b) has $5 = K$ nonzero entries. The probability of an unresolvable entry (among K) reads

$$P'_U(K, N) := \underbrace{\left(1 - \left(1 - \frac{1}{N}\right)^K\right)\frac{N}{K}}_{P_T(K,N)} - \underbrace{\left(1 - \frac{1}{N}\right)^{K-1}}_{P_R(K,N)}, \qquad (3.17)$$

consider Appendix C for details. Using (3.17), the expected number of unresolvable entries in $\mathbf{x}$ computes as $P'_U(K, N) \cdot K$.

The occurrence probability of these entries depends on the **ratio between activated tags and available signatures** $\varsigma = \frac{K}{N}$. Figure 3.9 depicts the expected number of nonzero entries in $\mathbf{x}$ normalized by K, i.e., the probability that an entry among K is nonzero. An increase in ς –e.g., by decreasing the number of signatures N– reduces the total amount of nonzero entries, since signatures become more likely to be assigned repeatedly.

Let me introduce the **expected number of unidentified tags in identification cycle** $i = \{1, 2, ...\}$ as $K^{(i)}$, where initially there are $K^{(0)} = K$ unidentified tags. After identification cycle i, $K^{(i)} = P_U(K^{(i-1)}, N) \cdot K^{(i-1)}$ tags remain unidentified (in expectation). Using (1.26) to express M and substituting the number of activated tags K with the expected number of unidentified tags $K^{(i)}$, the optimal bit overhead
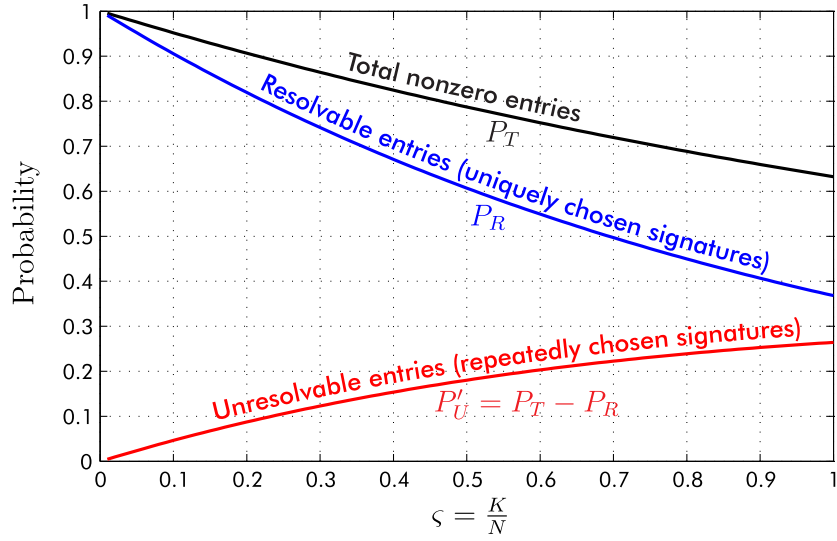
Figure 3.9 Probability that one among K entries in $\mathbf{x}$ is zero.

for identification in cycle i is

$$\beta_{\text{CSR}}^{(i)} = \underbrace{\left[cK^{(i-1)} \log \frac{N}{K^{(i-1)}} \right]}_{\text{Acquisition}} +$$

$$\underbrace{P_R(K^{(i-1)}, N) \cdot K^{(i-1)} \cdot (E + D)}_{\text{Enquiry and data (resolvable entries)}} + \qquad (3.18)$$

$$\underbrace{P_U'(K^{(i-1)}, N) \cdot K^{(i-1)} \cdot (E + D + R)}_{\text{Enquiry, data and reactivation (unresolvable entries)}}.$$

The total optimal bit overhead for identification is obtained by summation over all identification cycles (until all tags are identified):

$$\beta_{\text{CSR}} = \sum_i \beta_{\text{CSR}}^{(i)}. \qquad (3.19)$$

For comparison, the optimal bit overhead for identification in case of collision-free data read-out is given by

$$\widetilde{\beta}_{\text{CSR}} = \underbrace{\left[cK \log \frac{N}{K} \right]}_{\text{Acquisition}} + \underbrace{K(E + D)}_{\text{Enquiry and data}} \leq \beta_{\text{CSR}}, \qquad (3.20)$$

it is attained in case of high sparsity $\varsigma = \frac{K}{N} \to 0$ and, thus, $P_U' \to 0$ (and $P_R \to 1$).
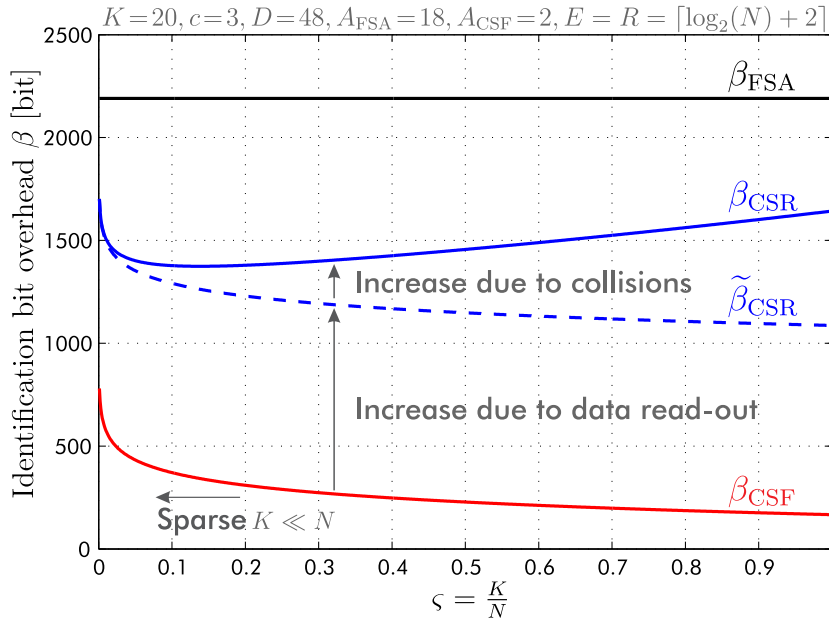
Figure 3.10 Optimal bit overhead for identification in the noiseless case.

Figure 3.10 compares the optimal bit overhead for identification for various choices of ς in the noiseless case. For a fixed activated tag population of $K = 20$, the number of signatures N is varied to realize various ς. The bit overhead of FSA is independent of N, it just depends on K, see (3.12). The bit overhead of CSF decreases for increasing ς due to the signature length dependency (1.26). The bit overhead of CSR first decays according to (1.26), but increases again for larger ς due to the increasing number of unresolvable entries in $\mathbf{x}$ that cause collisions in the data read-out. In the considered scenario, a ς around 0.14 yields the minimal bit overhead and therefore poses the optimal choice to obtain the quickest identification. Note that the minimum is relatively flat such that a well-selected N is (almost) optimal for a wide range of K.

3.5   Performance Comparison

In this section, the identification schemes are hampered by noise, and it is investigated how this affects their performance. The focus of the simulation lies on the *acquisition phase*, i.e., the RN16 sequences in FSA are corrupted by noise, and the CS measurements in CSF and CSR become noisy. The subsequent identification phase is assumed to be ideal, i.e., RN16 collisions in FSA and data read-out collisions in CSR are assumed to be detected perfectly.

The simulation results are averaged over $1\,000$ independent random realizations. Each realization encompasses a newly generated signature matrix $\mathbf{S}$, and newly generated channel coefficients that stay constant over the frames of FSA or identification cycles of CSF and CSR (static scenario). The channel properties were described in Section 3.1. Noise is added according to SNR definition (2.15). The bit overhead of FSA is computed for each SNR value according to $\beta_{\text{FSA}}(\text{SNR}) = \frac{16}{T_{\text{ps(SNR)}}}K + K(A_{\text{FSA}} + D)$, where T_{ps} is the throughput in tags per slot of standard FSA without collision recovery capabilities and a reader that utilizes an MMSE receiver; I use the simulated values from [110]. The bit overhead of CSF is generated by accumulating the bits of all necessary identification cycles until all tags are identified correctly.

All schemes assume to know K in order to facilitate a fair comparison. Consequently, FSA chooses the optimal frame size [42, 110] to maximize the throughput, while CSF and CSR choose the optimal signature length according to (1.26) with $c = 3$, and optimal tuning according to (2.14). All schemes are *adaptive*, i.e., the optimal frame size and optimal signature length are chosen in each frame and each cycle, respectively.

Let me begin by comparing the *bit overhead for identification*. The optimal bit overheads of FSA, CSF, and CSR are respectively defined in (3.12), (3.13), and (3.19). These equations describe the analytical optima that are attained at high SNR. Figure 3.11 shows the simulated bit overheads for various SNR values, the analytic optima are also plotted for $K = 20$. For a fixed number of signatures $N = 1\,000$, the number of activated tags was chosen as $K = \{20, 50, 100\}$, which corresponds to $\varsigma = \{0.02, 0.05, 0.1\}$. FSA has the highest bit overhead and ceases to work below $0\,\text{dB}$ SNR, the corruption of the RN16 sequences is too severe. The lowest bit overhead is attained by CSF, since no data read-out is required for identification. The CSR approach lies somewhere in between. Both CS-RFID approaches outperform FSA and prove to be more robust to noise.

Let me introduce another figure of merit that is widely used to compare the performance of RFID protocols [109, 110].

Definition 3.5.1 (Identification throughput)**.** *The identification throughput is defined as the number of identified tags per invested bit. It relates to the bit overhead for identification as*

$$T := \frac{K}{\beta}. \tag{3.21}$$

The *identification throughput* answers the question: *"How many tags are identified on average by a single bit?"* Figure 3.12 depicts the simulation results. CSF and CSR exhibit a dependency on K according to the operational $\varsigma = \frac{K}{N}$ in Figure 3.10. The throughput of FSA exhibits no variation among choices of K since it neither depends on N nor on K (as long as the optimal frame-size is chosen). The CS-RFID approaches significantly outperform FSA.
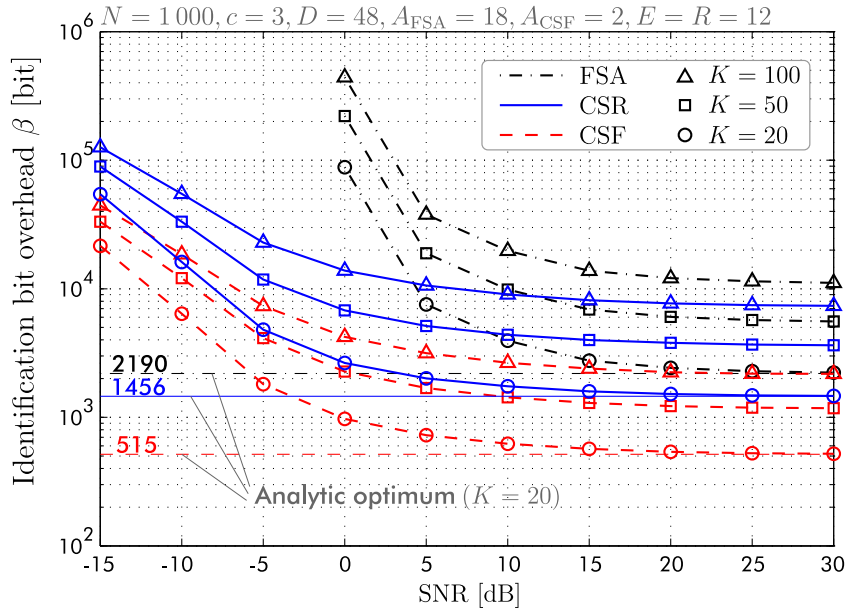

Figure 3.11 Simulated bit overhead versus SNR for various choices of K.
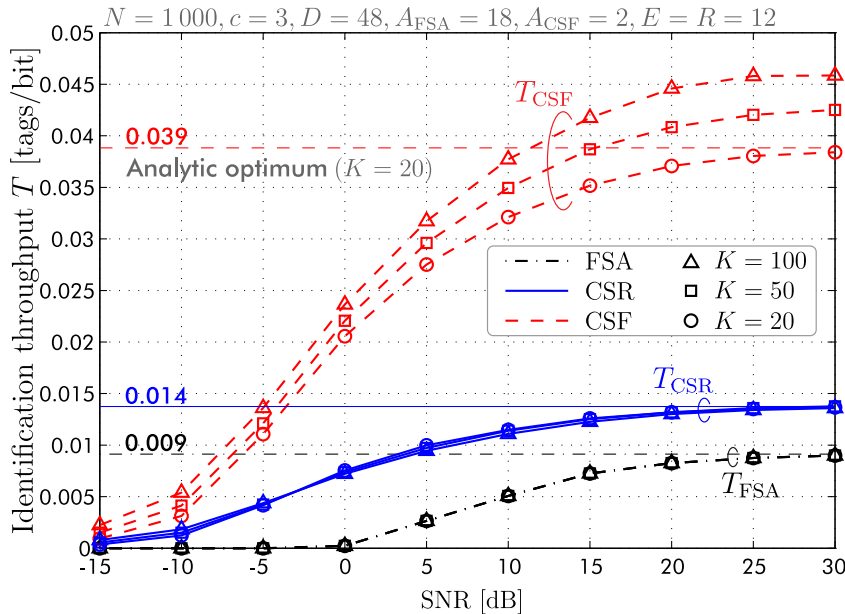
Figure 3.12 Simulated identification throughput versus SNR for various K.

$K = 100, N = 1\,000, c = 3, D = 48, A_{\mathrm{FSA}} = 18, A_{\mathrm{CSF}} = 2, E = R = 12$
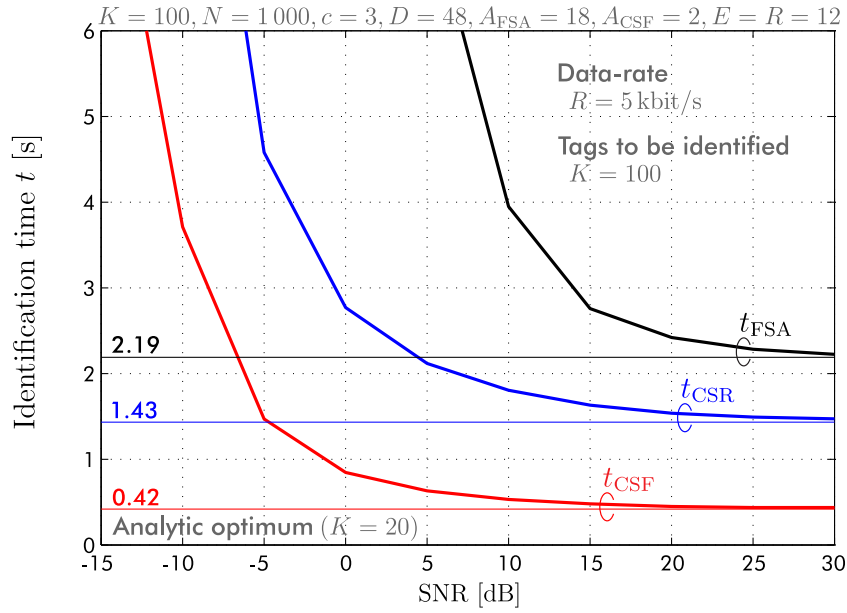
Figure 3.13 Simulated identification time versus SNR.

Let me now compute the *identification time* by using the previous results. Assuming a fixed data-rate[1] R, the identification time computes as

$$t = \frac{\beta}{R} = \frac{K}{R \cdot T}. \tag{3.22}$$

The simulation results are depicted in Figure 3.13 for a fixed $R = 5\,\mathrm{kbit/s}$ and using the bit overhead values from Figure 3.11, or equivalently, the throughput values from Figure 3.12. At $\mathrm{SNR} = 10\,\mathrm{dB}$, CSF is about 7.4 times quicker than FSA, while CSR is still about 2.2 times quicker than FSA. CS-RFID is, thus, by far quicker, particularly at low SNR.

So far, the measurement multiplier that controls the optimal signature length according to (1.26) was chosen "conservatively" large to be $c = 3$. Figure 3.14 shows simulation results that suggest to make c SNR dependent for optimal performance. At high SNR, a $c > 2$ decreases throughput since $c = 2$ already achieves perfect recovery –see Section 2.2.3– in very few identification cycles, in particular one cycle at $\mathrm{SNR} = \infty$. At low SNR, an increasing c leads to higher throughput since the recovery quality improves with additional measurements, and fewer identification cycles are needed. For optimal performance, the choice of c can be made SNR dependent, as

---

[1] Excuse the abuse of notation — R here denotes the data-rate, but previously in this chapter denoted the number of bits for reactivation in CSR. The usage is clear from context.
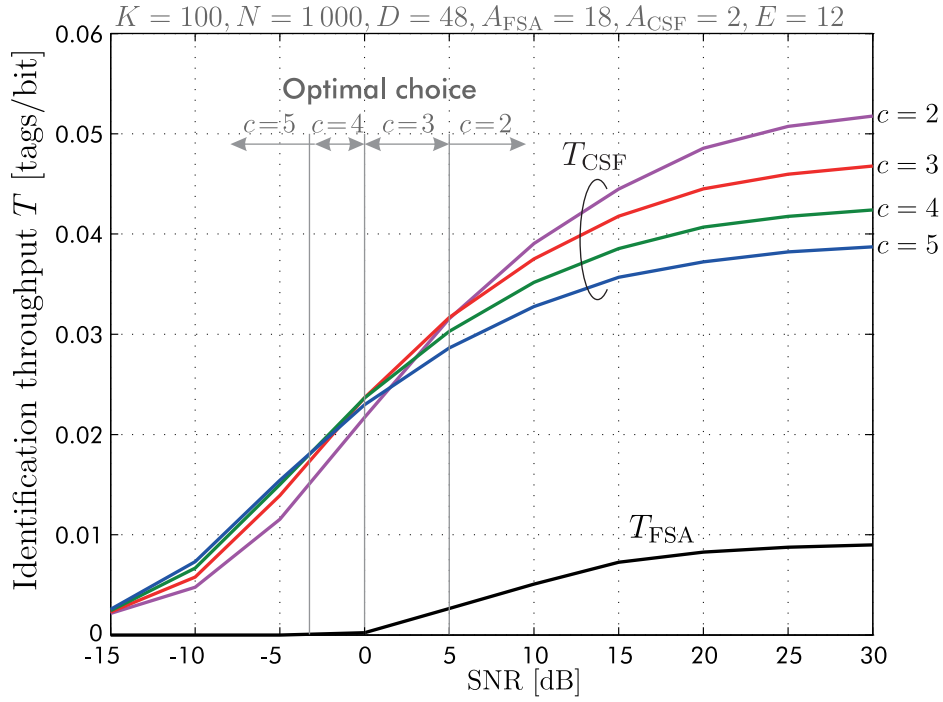
Figure 3.14 Simulated identification throughput for variable c.

hinted in Figure 3.14. However, the performance gain at low SNR is only marginal, and choosing a measurement multiplier $c \in [2, 3]$ yields overall satisfying results.

3.6 Summary

This chapter presented the concept of CS-RFID, where a multitude of tags is identified quickly and reliably by utilizing CS with AMP. The acquisition phase of RFID is formulated as a CS measurement, where the activated tags *respond simultaneously* with an assigned signature sequence — the CS approach *exploits collisions*. It was explained how the physical measurements have to be processed such that the AMP Algorithm 1 can be utilized for CS recovery. The support of the recovered signal vector yields the tag identifiers that are used to communicate with the tags for identification. Two protocols for reliable identification were introduced:

CSF is a protocol for *fixed inventories*. The whole inventory is assumed to be known or maintained by a database. Only a small subset of the inventory is to be identified at a given time — this constitutes the sparsity that is exploited by CS. Each tag (item) has a unique signature sequence; upon query, all activated tags respond simultaneously with this unique identifier. No data read-out is necessary for identification.

CSR is a protocol for *arbitrary inventories*. Similar to the EPCglobal standard, the tags respond with a randomly chosen identifier during the acquisition phase. Each identifier is a pseudo-randomly generated signature sequence; upon query, all activated tags simultaneously respond with a randomly selected identifier. A data read-out is required for identification.

Numerical results suggest that CS-RFID strongly outperforms the pervasive FSA protocol in terms of identification throughput and noise robustness. Furthermore, the acquisition phase is significantly shorter than in FSA, and no reader power is wasted in empty slots; consequently, CS-RFID is highly energy efficient.

Chapter 4

Exploiting Joint Sparsity in Tag Acquisition

In Chapter 3, the AMP Algorithm 1 was employed for CS signal recovery. This algorithm iteratively solves the LASSO problem formulation (1.23). Except for searching for a sparse solution, the AMP algorithm does not exploit any prior knowledge. The BASSAMP algorithm, on the other hand, fully exploits the signal prior distribution and the joint sparsity structure.

In this chapter, CS-RFID is performed on a reader with multiple receive antennas. Such reader receives multiple measurement vectors and facilitates a problem formulation with joint sparsity. The nonzero entries of the jointly sparse signal vectors are distributed according to the channel model. This is an archetype setup for the BASSAMP algorithm that is now employed to improve the CS recovery performance and, in turn, accomplish an even quicker and more reliable tag acquisition.

These are the main contributions of this chapter, based on my work in [76, 116]:

- The origins of joint sparsity in CS-RFID are identified.

- The BASSAMP algorithm is specified for the RFID channel model.

- A Gaussian relaxation of the prior distribution that enables a simple algorithmic implementation without significant performance degradation is proposed.

- A method for support detection based on the BASSAMP algorithm is introduced. In the context of CS-RFID, this relates to an improved detection of the set of assigned signatures $\mathcal{T}_A$, and its cardinality K.

- The performance is compared to FSA with collision recovery capabilities that also exploits several receive antennas at the reader.

4.1   Joint Sparsity in CS-RFID

In Section 2.4, the BASSAMP algorithm to exploit joint sparsity was introduced, and it was demonstrated in sections 2.5 and 2.6 that it significantly outperforms the AMP and BAMP algorithms. Let me list the main benefits for CS-RFID:

- BASSAMP requires fewer measurements M than AMP for successful recovery. Consequently, *shorter signature sequences* can be employed for tag acquisition.

- The acquisition time and the jitter[1] sensitivity are reduced. Furthermore, passive tags require less energy during the acquisition phase.

- The robustness to measurement noise is improved, which facilitates the deployment of CS-RFID in harsh environments.

- The set of assigned signatures, $\mathcal{T}_A$, can be detected more reliably by employing a strongly improved support estimation scheme that exploits soft information from multiple signal vectors. This leads to fewer identification cycles, i.e., fewer repetitions of the acquisition phase, and quicker identification. Furthermore, the number of activated tags K can be estimated implicitly.
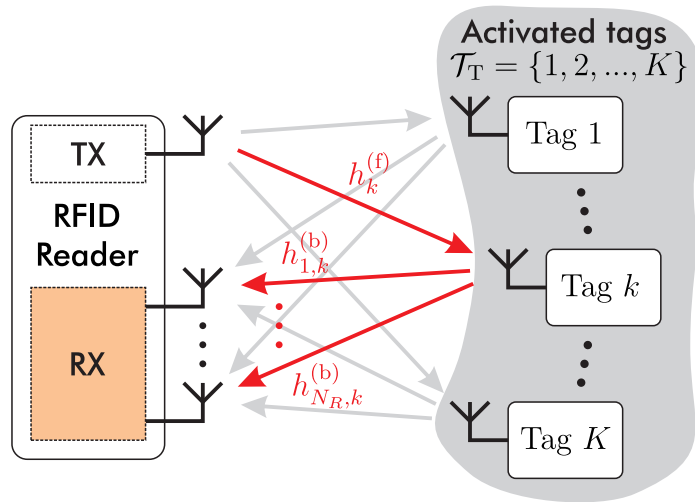
Figure 4.1 A bistatic reader with N_R receive antennas identifies K tags.

It will become apparent presently that joint sparsity in CS-RFID is related to the multiple measurement vector problem, see Section 2.4. This implies that the reader may employ several receive antennas. Figure 4.1 depicts a reader with N_R receive

---

[1]Jitter refers to link frequency deviations among tags. A detailed discussion follows in Chapter 5.

antennas. The total channel from the transmit antenna to the k-th tag and back to the r-th receive antenna is modeled by

$$h_{r,k} = h_k^{(\mathrm{f})} h_{r,k}^{(\mathrm{b})}. \tag{4.1}$$

The channel coefficient distribution will be specified in Section 4.2. Let me list the two mechanisms responsible for joint sparsity in CS-RFID:

Multiple Receive Antennas

Each antenna receives the superposition of signatures (3.5) with different noise realizations and, generally, with different channel coefficient realizations for every tag. The channel coefficient realizations comprise the nonzero entries in the signal vector $\mathbf{x}_r, r \in \{1, ..., N_R\}$. The N_R receive antennas provide N_R jointly sparse signal vectors and, thus, N_R measurement vectors that are generated with the same signature matrix:

$$[\mathbf{z}_1, ..., \mathbf{z}_{N_R}] = \mathbf{S}[\mathbf{x}_1, ..., \mathbf{x}_{N_R}] + [\mathbf{w}_1, ..., \mathbf{w}_{N_R}]. \tag{4.2}$$

Separation into Real and Imaginary Parts

By having complex-valued channel coefficients in $\mathbf{x}_r$ and a real-valued signature matrix $\mathbf{S}$, the measurements can be separated into a real part $\mathbf{z}_r^{(\mathfrak{R})}$ and an imaginary part $\mathbf{z}_r^{(\mathfrak{J})}$, which yields two jointly sparse signal vectors, and two measurement vectors that are generated with the same sensing matrix:

$$[\mathbf{z}_r^{(\mathfrak{R})}, \mathbf{z}_r^{(\mathfrak{J})}] = \mathbf{S}[\mathbf{x}_r^{(\mathfrak{R})}, \mathbf{x}_r^{(\mathfrak{J})}] + [\mathbf{w}_r^{(\mathfrak{R})}, \mathbf{w}_r^{(\mathfrak{J})}]. \tag{4.3}$$

The two variants are *combined* as depicted in Figure 4.2 to obtain

$$B = 2N_R \tag{4.4}$$

jointly sparse vectors. The resulting CS measurement reads

$$\underbrace{\left[\mathbf{z}_1^{(\mathfrak{R})}, \mathbf{z}_1^{(\mathfrak{J})}, ..., \mathbf{z}_{N_R}^{(\mathfrak{R})}, \mathbf{z}_{N_R}^{(\mathfrak{J})}\right]}_{\mathbf{z}} = \mathbf{S} \underbrace{\left[\mathbf{x}_1^{(\mathfrak{R})}, \mathbf{x}_1^{(\mathfrak{J})}, ..., \mathbf{x}_{N_R}^{(\mathfrak{R})}, \mathbf{x}_{N_R}^{(\mathfrak{J})}\right]}_{\mathbf{x}}$$
$$+ \underbrace{\left[\mathbf{w}_1^{(\mathfrak{R})}, \mathbf{w}_1^{(\mathfrak{J})}, ..., \mathbf{w}_{N_R}^{(\mathfrak{R})}, \mathbf{w}_{N_R}^{(\mathfrak{J})}\right]}_{\mathbf{w}}. \tag{4.5}$$
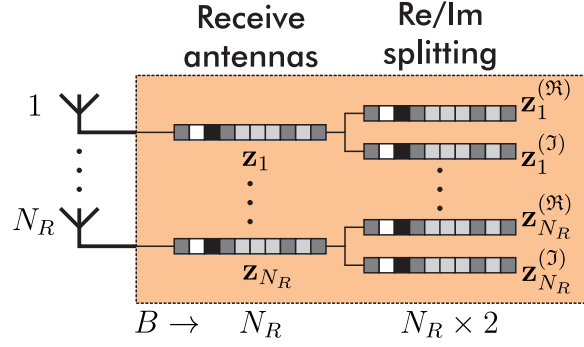
Figure 4.2 Origins of joint sparsity in RFID.

It features a *measurement matrix* $\mathbf{Z} \in \mathbb{R}^{M \times B}$, and a *channel activity matrix* $\mathbf{X} \in \mathbb{R}^{N \times B}$ whose columns have a common support, i.e., the entries of a row in $\mathbf{X}$ are either all zero or all nonzero. Figure 4.3 illustrates the new CS formulation (noise omitted).

I intend to employ BASSAMP for signals with joint sparsity to recover the channel activity matrix $\mathbf{X}$ — its support yields the required information for tag acquisition, i.e., the set of assigned signatures $\mathcal{T}_A$. Similar to AMP in Section 3.2, BASSAMP assumes that the sensing matrix has zero-mean columns. To accomplish this, (3.10) is applied to each measurement vector $\mathbf{z}_b$, $b \in \mathcal{B} = \{1, ..., B\}$:

$$\mathbf{y}_b = \overline{\mathbf{A}}\mathbf{x}_b + \mathbf{w}_b \approx \mathbf{z}_b - \frac{1}{M} \sum_{m=1}^{M} z_{m,b}. \tag{4.6}$$

After normalizing $\overline{\mathbf{A}}$ according to (3.11), the BASSAMP compliant measurement formulation reads

$$\mathbf{Y} = [\mathbf{y}_1, ..., \mathbf{y}_B] = \mathbf{A}\mathbf{X} + \mathbf{W}. \tag{4.7}$$

In order to apply BASSAMP, the functions used in algorithms 3 and 4 have to be specified for the RFID case. In particular, the dyadic channel model that poses as prior for the nonzero entries in $\mathbf{X}$ has to be considered.

4.2   Specification of BASSAMP for CS-RFID

The BASSAMP algorithms 3 and 4 aim at reconstructing $\mathbf{X}$ from $\mathbf{Y}$ in (4.7). They utilize the knowledge of $\mathbf{A}$, the signal prior, and the joint sparsity structure. The signal prior plays a major role in the computation of the BAMP denoiser functions $F(\cdot; \cdot, \cdot)$ and $F'(\cdot; \cdot, \cdot)$, and the BASSAMP update function $U_J(\cdot, \cdot, \cdot)$. Therefore, let me specify the signal prior for the RFID scenario that is dictated by the channel model.
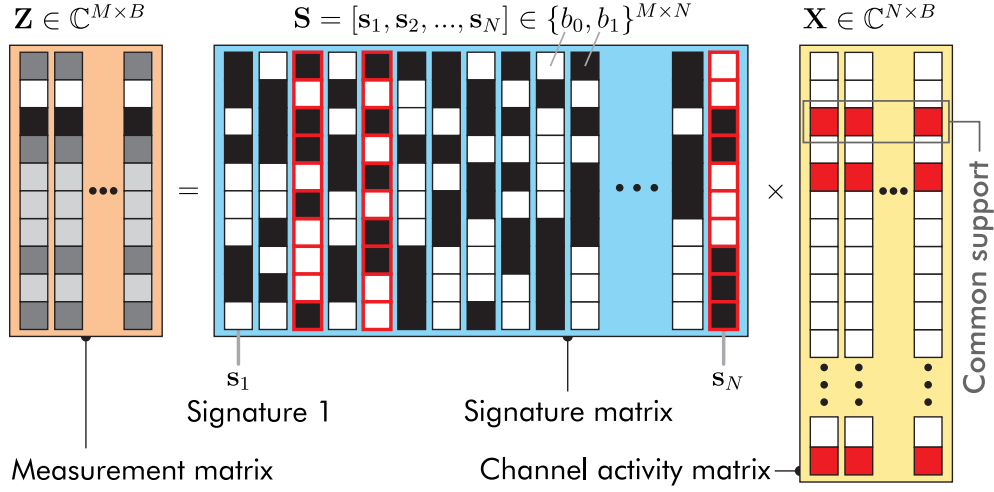
Figure 4.3 CS measurement — visualized with joint sparsity.

4.2.1   Channel Distribution and Signal Prior

In Section 3.1, the dyadic channel model for RFID was introduced. It is adopted in this section to obtain the distribution of the channel coefficients (4.1). Similar to Section 3.1, I assume that the antennas at the reader are sufficiently spaced such that the channels are uncorrelated. I now specify the channel model in more detail, since the full probabilistic description is utilized by the BASSAMP algorithm.

The forward and backward channel coefficients are modeled as independently distributed complex-valued circular symmetric Gaussian random variables such that

$$\mathsf{h}_k^{(\mathrm{f})} \sim \mathcal{CN}\left(0, \sigma^{(\mathrm{f})2}\right), \tag{4.8}$$

$$\mathsf{h}_{r,k}^{(\mathrm{b})} \sim \mathcal{CN}\left(0, \sigma^{(\mathrm{b})2}\right). \tag{4.9}$$

The coefficients are split into the real and imaginary parts according to

$$\mathsf{h}_k^{(\mathrm{f})} = \mathsf{h}_k^{(\mathrm{f},\Re)} + j\mathsf{h}_k^{(\mathrm{f},\Im)}, \tag{4.10}$$

$$\mathsf{h}_{r,k}^{(\mathrm{b})} = \mathsf{h}_{r,k}^{(\mathrm{b},\Re)} + j\mathsf{h}_{r,k}^{(\mathrm{b},\Im)}, \tag{4.11}$$

the real and imaginary parts obey a zero-mean Gaussian distribution with half the original variance, respectively. Thus, the total channel (4.1) is expressed by

$$
\begin{aligned}
\mathsf{h}_{r,k} &= \mathsf{h}_{r,k}^{(\Re)} + j\mathsf{h}_{r,k}^{(\Im)} \\
&= \left(\mathsf{h}_k^{(\mathrm{f},\Re)}\mathsf{h}_{r,k}^{(\mathrm{b},\Re)} - \mathsf{h}_k^{(\mathrm{f},\Im)}\mathsf{h}_{r,k}^{(\mathrm{b},\Im)}\right) + j\left(\mathsf{h}_k^{(\mathrm{f},\Re)}\mathsf{h}_{r,k}^{(\mathrm{b},\Im)} + \mathsf{h}_k^{(\mathrm{f},\Im)}\mathsf{h}_{r,k}^{(\mathrm{b},\Re)}\right).
\end{aligned}
\tag{4.12}
$$

In order to obtain the distribution of the real and imaginary parts of $h_{r,k}$, the following relation is used [117, Proposition 2.2.5]:

Lemma 4.2.1. *Consider four independent random variables* $v_1, v_2 \sim \mathcal{N}(0, \sigma_a^2)$ *and* $v_3, v_4 \sim \mathcal{N}(0, \sigma_b^2)$. *The term* $v = v_1 v_3 - v_2 v_4$ *is Laplace distributed with PDF*

$$f_v(v) = \frac{1}{2\sigma_a \sigma_b} \exp\left(-\frac{1}{\sigma_a \sigma_b}|v|\right). \tag{4.13}$$

The same holds for $v = v_1 v_3 + v_2 v_4$.

By using (4.8)–(4.12) and Lemma 4.2.1, I arrive at the following theorem:

Theorem 4.2.1 (Channel distribution). *Consider a forward channel coefficient with complex-valued circular symmetric Gaussian distribution with zero mean and variance* $\sigma^{(f)^2}$, *and a backward channel coefficient with complex-valued circular symmetric Gaussian distribution with zero mean and variance* $\sigma^{(b)^2}$. *The total channel coefficient PDF reads*

$$f_h(h) = \frac{1}{\sigma^{(f)}\sigma^{(b)}} \exp\left(-\frac{2}{\sigma^{(f)}\sigma^{(b)}}|h|\right), \tag{4.14}$$

where h is a placeholder for the real part $h_{r,k}^{(\mathfrak{R})}$ or imaginary part $h_{r,k}^{(\mathfrak{J})}$ of the total channel.

Comparing this to a standard Laplace distribution, the real and imaginary parts of the total channel are Laplace distributed with scale parameter $\kappa_n = \frac{1}{2}\sigma^{(f)}\sigma^{(b)}$, i.e., with variance $2\kappa_n^2 = \frac{1}{2}\sigma^{(f)^2}\sigma^{(b)^2}$.

The BASSAMP algorithm features a Bayesian MMSE estimator (2.21) that fully exploits the prior distribution of the signal entries $x_{n,b}$. In (4.14), the distribution of the *nonzero* entries in the random signal matrix $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_B]$ is specified. A realization thereof yields the channel activity matrix $\mathbf{X}$, its nonzero entries are the channel coefficient realizations. Considering the random signal model as introduced in (1.14), the prior distribution of a random signal entry $x_{n,b}$ reads

$$f_{x_{n,b}}(x_{n,b}) = \gamma_{n,b}\delta(x_{n,b}) + (1 - \gamma_{n,b})f_h(x_{n,b}), \tag{4.15}$$

where $\gamma_{n,b}$ is the probability that the n-th signal entry of the b-th vector is zero. This is the *Bernoulli-Laplace prior* –with $\kappa_n = \frac{1}{2}\sigma^{(f)}\sigma^{(b)}$– for which BAMP and scalar BASSAMP were specified in Section 2.3.2 and Section 2.4.2, respectively.

4.2.2 Gaussian Relaxation

The BASSAMP algorithm is executed on the reader hardware. A simple, computationally efficient and numerically stable algorithm is aspired. Scrutinizing the MMSE estimator function in (2.30), the auxiliary functions (2.35) and (2.36) turn out to be numerically challenging. Assuming, e.g., $\sigma^{(f)} = \sigma^{(f)} = 1$, $\beta = 0.2$ and $\gamma = 0.9$, the auxiliary function $k_1(u)$ outputs values in the order of 10^{100} for an input $u = 10$. A single precision computer outputs $k_1(u) = \infty$ for $u > 6.33$. While the majority of the u values are smaller and concentrated around zero due to the PDF of u, these cases still occur. Executing operations on such large numbers introduces numerical errors. In order to alleviate these numerical challenges, I propose to approximate the channel coefficient prior distribution (4.14) by a Gaussian distribution, which I refer to as the *Gaussian relaxation*.

The Laplace distribution (4.14) has zero mean and variance $\frac{1}{2}\sigma^{(f)^2}\sigma^{(b)^2}$. The approximating Gaussian distribution is assumed to have the same parameters, i.e., zero mean and variance $\sigma_\mathsf{x}^2 = \frac{1}{2}\sigma^{(f)^2}\sigma^{(b)^2}$. Therefore, the *relaxed* version of the channel coefficient PDF (4.14) is obtained as

$$f_\mathsf{h}^\mathrm{R}(h) = \frac{1}{\sqrt{\pi\sigma^{(f)^2}\sigma^{(b)^2}}} \exp\left(-\frac{h^2}{\sigma^{(f)^2}\sigma^{(b)^2}}\right). \tag{4.16}$$

Consequently, the signal prior distribution (4.15) is relaxed from a Bernoulli-Laplace distribution into a Bernoulli-Gaussian distribution. This relaxation lowers the numerical complexity because simpler functions are used, see sections 2.3.2 and 2.4.2.

In Section 2.6.2, I demonstrated by SE that this particular prior mismatch hardly affects the recovery performance. To further validate this claim for a wide range of parameters, I compute the *empirical phase transition curves* in the noiseless case. I reside to the same numerical simulation setup as presented in Section 2.5. A complex-valued signal vector $\mathbf{x} \in \mathbb{C}^N$ is considered, i.e., $N_R = 1$, and $B = 2$ after separation into real and imaginary parts. The nonzero entries of $\mathbf{x}$ are drawn from the channel distribution (4.14) with $\sigma^{(f)} = \sigma^{(b)} = 1$, i.e., the signal entries are Bernoulli-Laplace distributed.

The empirical phase transition curves are depicted in Figure 4.4. Two instances of BASSAMP are compared; one employs the *true* Bernoulli-Laplace prior with channel distribution (4.14), the other employs the Bernoulli-Gaussian relaxation with channel distribution (4.16), both assume $\sigma^{(f)} = \sigma^{(b)} = 1$. For comparison, the AMP Algorithm 1 that operates on the complex-valued signal vector is also considered — its phase transition obeys the complex LASSO [84]. The Gaussian relaxation exhibits
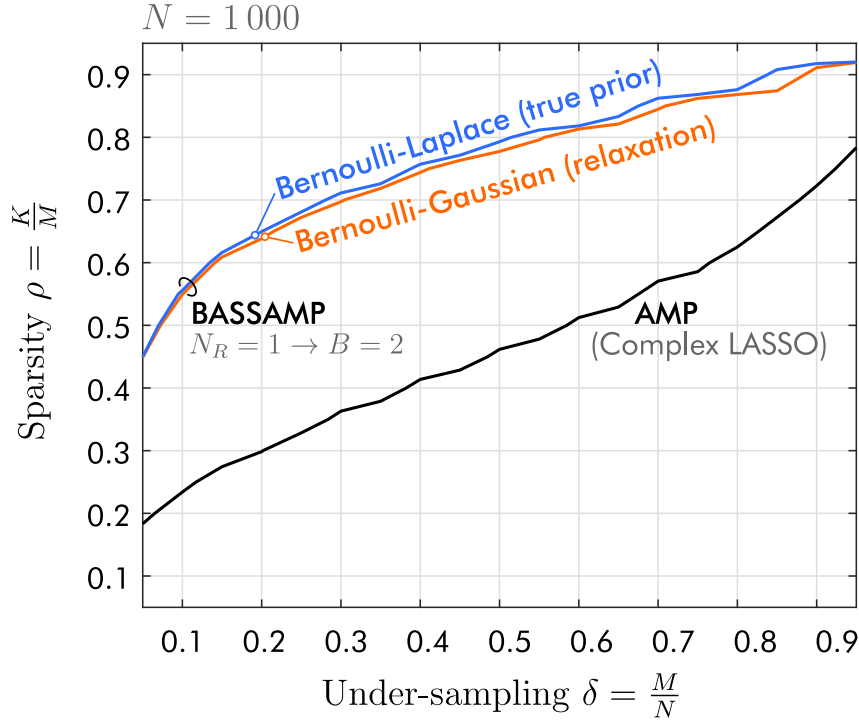
Figure 4.4 Phase transition of Gaussian relaxation.

a marginal decline in performance that is irrelevant in practice. Furthermore, the BASSAMP algorithm strongly outperforms the AMP algorithm, i.e., it consistently requires fewer measurements. In the sequel, I employ the Gaussian relaxation.

4.2.3   Choice of Parameters

The choice of the channel model depends on the location of the reader antennas, the environment (scatterers and reflectors), and the effective read range. The variances $\sigma^{(f)2}$ and $\sigma^{(b)2}$ describe the strength of the spatial fading of the forward and backward link, respectively. In order to estimate them, one would have to measure the forward and backward links separately for many prospective tag positions.

In practice, one would rather measure the total channel (4.1), or avoid measuring the channels and determining their distribution entirely. The AMP framework allows to perform prior estimation during recovery, i.e., the recovery algorithm can be adapted to estimate the prior over iterations. One such algorithmic extension was proposed in [118], and it was shown that the performance degradation due to unknown prior is negligible in practice. Moreover, measurements in Chapter 5 suggest that the Gaussian relaxation with a coarse assumption of the variances is sufficient.

4.3 Support Detection

In this section, I present a support detection method that extends the findings in [116]. The support detection is crucial for quick tag identification as it yields

- the detected set of assigned signatures $\widehat{\mathcal{T}}_{\mathrm{A}}$,

- the estimated number of activated tags $\widehat{K}$.

In particular, $\widehat{\mathcal{T}}_{\mathrm{A}}$ is of vital importance for quick identification:

- In CSF, this set directly identifies the activated tags. The reader enquires the signature indices in order to confirm the identification. An erroneous set prolongs this enquiry phase and leads to a repetition of the acquisition phase.

- In CSR, the indices of the assigned signatures are used to communicate with the tags for data read-out. Again, an erroneous set prolongs the enquiry phase and leads to a repetition of the acquisition phase.

For all schemes, a wrongly detected set of assigned signatures prolongs the identification and increases the reader-to-tag communication overhead.

In Chapter 3, I assumed to (approximately) know the number of activated tags K, i.e., $\widehat{K} = K$. Utilizing this knowledge, the detected set of assigned signatures $\widehat{\mathcal{T}}_{\mathrm{A}}$ was dictated by the K largest entries of the AMP signal recovery $|\widehat{\mathbf{x}}|$. The newly employed BASSAMP algorithms enable robust support estimation by combining the soft information of all recovered vectors $\widehat{\mathbf{x}}_b$, $b \in \mathcal{B} = \{1, ..., B\}$. Let me derive the support detection scheme that is based on Bayesian estimation. Note that I base my derivation on scalar BASSAMP Algorithm 3, and explain at the end how the findings are applicable to the vector version in Algorithm 4.

4.3.1 Probabilistic Derivation

The first ingredient is the specification of the distribution of the decoupled measurements (2.17), which was stated in (2.43). For the considered RFID scenario, I obtain

$$f_{\mathsf{u}_{n,b}}(u_{n,b}) = \gamma_{n,b} f_0(u_{n,b}) + (1 - \gamma_{n,b}) f_1(u_{n,b}) \tag{4.17}$$

with –using (2.35), (2.36), (2.73), and (4.14)–

$$
\begin{aligned}
f_0(u_{n,b}) &= \frac{1}{\sqrt{2\pi\beta_b}} \exp\left(-\frac{u_{n,b}^2}{2\beta_b}\right), \\
f_1(u_{n,b}) &= \frac{1}{2\sigma^{(\mathrm{f})}\sigma^{(\mathrm{b})}} \exp\left(-\frac{u_{n,b}^2}{2\beta_b}\right) \left[k_1(u_{n,b}) + k_2(u_{n,b})\right].
\end{aligned}
\tag{4.18}
$$

If the Gaussian relaxation with nonzero entry PDF (4.16) is used, $f_1(u_{n,b})$ has to be replaced by

$$f_1^{\mathrm{R}}(u_{n,b}) = \frac{1}{\sqrt{2\pi(\beta_b + \sigma_{\mathsf{x}}^2)}} \exp\left(-\frac{u_{n,b}^2}{2(\beta_b + \sigma_{\mathsf{x}}^2)}\right). \tag{4.19}$$

The second ingredient is the utilization of the latent activity variables that were introduced in Definition 2.4.2. A signal entry $x_{n,b}$ is zero if the latent variable $z_{n,b} = 0$, and it is nonzero if $z_{n,b} = 1$. Now, the responsibilities similar to (2.48) and (2.49) are obtained and interpreted as follows:

- The posterior probability that signal entry $\mathsf{x}_{n,b} = 0$ given the decoupled measurement $u_{n,b}$ reads

$$
\begin{aligned}
P(\mathsf{z}_{n,b} = 0 | \mathsf{u}_{n,b} = u_{n,b}; \gamma_{n,b}) &= \frac{P(\mathsf{z}_{n,b} = 0) \cdot f_{\mathsf{u}_{n,b}|\mathsf{z}_{n,b}}(u_{n,b}|\mathsf{z}_{n,b} = 0)}{f_{\mathsf{u}_{n,b}}(u_{n,b})} \\
&= \frac{\gamma_{n,b} \cdot f_0(u_{n,b})}{f_{\mathsf{u}_{n,b}}(u_{n,b})}.
\end{aligned}
\tag{4.20}
$$

- The posterior probability that signal entry $\mathsf{x}_{n,b} \neq 0$ given the decoupled measurement $u_{n,b}$ reads

$$
\begin{aligned}
P(\mathsf{z}_{n,b} = 1 | \mathsf{u}_{n,b} = u_{n,b}; \gamma_{n,b}) &= \frac{P(\mathsf{z}_{n,b} = 1) \cdot f_{\mathsf{u}_{n,b}|\mathsf{z}_{n,b}}(u_{n,b}|\mathsf{z}_{n,b} = 1)}{f_{\mathsf{u}_{n,b}}(u_{n,b})} \\
&= \frac{(1 - \gamma_{n,b}) \cdot f_1(u_{n,b})}{f_{\mathsf{u}_{n,b}}(u_{n,b})}.
\end{aligned}
\tag{4.21}
$$

The third ingredient is the combination of this soft information. Due to the joint sparsity structure, there are only two relevant cases: Either $x_{n,b} = 0$, $\forall b \in \mathcal{B}$, or $x_{n,b} \neq 0$, $\forall b \in \mathcal{B}$. I intend to calculate the overall probability that the n-th entry of *all* B jointly sparse vectors $\mathbf{x}_b$ is zero respectively nonzero. Based on (4.20) and (4.21), the posterior probability that all n-th signal entries are zero, given $u_{n,b}$ and $\gamma_{n,b}$, $\forall b \in \mathcal{B}$, computes as

$$\epsilon_n^{(0)} := \prod_{b=1}^{B} P(\mathsf{z}_{n,b} = 0 | u_{n,b}; \gamma_{n,b}) = \frac{1}{d} \prod_{b=1}^{B} \gamma_{n,b} \cdot f_0(u_{n,b}), \qquad (4.22)$$

while the posterior probability that all n-th signal entries are nonzero computes as

$$\epsilon_n^{(1)} := \prod_{b=1}^{B} P(\mathsf{z}_{n,b} = 1 | u_{n,b}; \gamma_{n,b}) = \frac{1}{d} \prod_{b=1}^{B} (1 - \gamma_{n,b}) \cdot f_1(u_{n,b}), \qquad (4.23)$$

where d is a common partition factor.

Finally, the detected signal support is equivalent to the detected set of assigned signatures and is obtained by comparing these probabilities:

$$\widehat{\mathcal{T}}_{\mathrm{A}} = \left\{ n \in \mathcal{T} : \frac{\epsilon_n^{(1)}}{\epsilon_n^{(0)}} = \frac{\prod_{b=1}^{B}(1 - \gamma_{n,b}) \cdot f_1(u_{n,b})}{\prod_{b=1}^{B} \gamma_{n,b} \cdot f_0(u_{n,b})} > 1 \right\}. \qquad (4.24)$$

The estimated number of activated tags is defined as the cardinality of this set:

$$\widehat{K} = |\widehat{\mathcal{T}}_{\mathrm{A}}|. \qquad (4.25)$$

Note that the support estimation is performed **after executing** BASSAMP Algorithm 3; it considers the values $u_{n,b}$, $\gamma_{n,b}$ and β_b after the last iteration t. If vector BASSAMP Algorithm 4 is employed, all entries have the same β, and $\gamma_{n,b}$ is not directly available. It can be computed according to the joint extrinsic prior update equation (2.56) that was specified for the Bernoulli-Gaussian prior in (2.72).

Regardless of the algorithm, the prior probabilities are initialized with a coarse assumption of K, termed K^0:

$$\gamma_{n,b}^0 = 1 - \frac{K^0}{N}. \qquad (4.26)$$

The following numerical results demonstrate that this assumption can, indeed, be very coarse.

4.3.2 Numerical Performance Evaluation

Let me demonstrate the performance of the support detection scheme by simulation. The goal is to demonstrate that the initial assumption K^0 in (4.26) can be very coarse, i.e., that exact knowledge of K is not required. Note that this assumption affects the BASSAMP algorithm, and the output of the algorithm affects the support estimation. I therefore investigate the conjunction of BASSAMP and the support estimation.

The simulation campaign encompasses $1\,000$ realizations of $\mathbf{A}$, $\mathbf{X}$ and $\mathbf{W}$ in (4.7), and the dyadic channel model is employed. Furthermore, $K = 100$ tags are activated, and $N = 1\,000$. The figures of merit are the number of CDs and the number of FAs [107] that partition the detected set of assigned signatures $\widehat{\mathcal{T}}_{\mathrm{A}}$ (4.24). An index in $\widehat{\mathcal{T}}_{\mathrm{A}}$ either refers to a CD or a FA. The simulation results are averaged over the realizations. Since the signals can now be matrices, I extend the SNR definition (2.15) to

$$\mathrm{SNR} := \frac{\|\mathbf{A}\mathbf{X}\|_{\mathrm{F}}^2}{\mathbb{E}_{\mathbf{W}}\{\|\mathbf{W}\|_{\mathrm{F}}^2\}} = \frac{\|\mathbf{A}\mathbf{X}\|_{\mathrm{F}}^2}{MB\sigma_{\mathbf{w}}^2}. \tag{4.27}$$

Figure 4.5 shows the average number of CDs and FAs versus variable M in the noiseless case. The BASSAMP algorithm is plotted for $N_R = 1$ (left) and $N_R = 4$ (right), and AMP is plotted for $N_R = 1$ for comparison. Note that AMP recovers a single complex-valued signal vector $\mathbf{x}$, while BASSAMP exploits joint sparsity. Furthermore, AMP assumes to know K and $\mathcal{T}_{\mathrm{A}}$ is detected by collecting the indices corresponding to the K largest entries in $|\widehat{\mathbf{x}}|$, while BASSAMP employs the support estimation scheme (4.24). Three different initializations of K^0 are investigated:

- A strong under-assumption $K^0 = 50$,

- the true assumption $K^0 = 100$,

- a strong over-assumption $K^0 = 150$.

The results suggest that the support estimation in conjunction with the BASSAMP algorithm is very tolerant regarding the initial value of K. A strong over- or under-assumption of K^0 respectively results in a larger or smaller number of CDs and FAs during the phase transition, but has no effect after the phase transition where perfect recovery is observed, i.e., where only CDs are observed and $\widehat{\mathcal{T}}_{\mathrm{A}} = \mathcal{T}_{\mathrm{A}}$. Since the algorithm is typically operated at signature lengths M that result in successful recoveries that are perfect in the noiseless case, the considered assumptions of K have no impact in the noiseless case. In comparison to the AMP approach used in Chapter 3, the proposed scheme is superior in several ways:
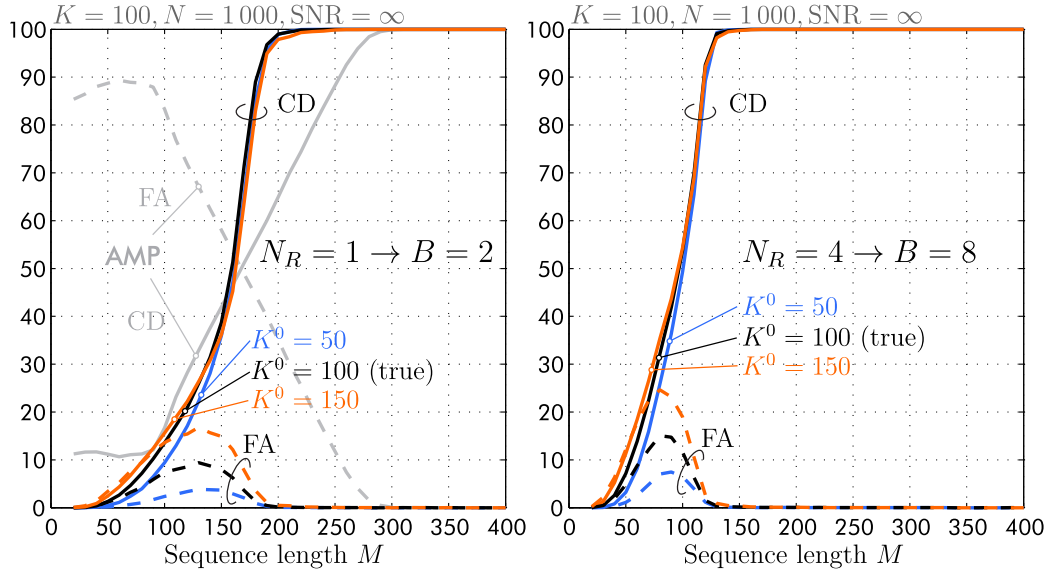
Figure 4.5 Support detection for variable M.

- (Accurate) knowledge of K is not required.

- Perfect detection is achieved with a reduced M. This entails shorter signatures for tag acquisition with CS-RFID.

- For a suboptimal choice of M that leads to a suboptimal recovery, the support detection features very few FAs. This reduces the reader-to-tag communication overhead, because fewer tags are wrongly enquired. In comparison, AMP features many FAs.

An increased number of receive antennas N_R leads to an earlier phase transition, i.e., fewer measurements M are required for perfect detection. This is illustrated by the right plot in Figure 4.5. To put it in numbers: In the considered scenario, perfect support detection with BASSAMP in the noiseless case is achieved for $M \geq 210$ with $N_R = 1$, and for $M \geq 140$ with $N_R = 4$.

Figure 4.6 shows the average number of CDs and FAs versus variable SNR at fixed $M = 400$, i.e., in the regime where AMP and BASSAMP enable perfect recovery in the noiseless case. Similar to the previous results, an over- or under-assumption of K^0 affects the outcome of the support detection only marginally. The noise robustness of the support detection increases with an increasing number of receive antennas N_R, this is observed by comparing the left with the right plot in Figure 4.6. In the considered example, $N_R = 4$ receive antennas result in perfect support detection above $\mathrm{SNR} = 15\,\mathrm{dB}$, which enables immediate identification after only one cycle.
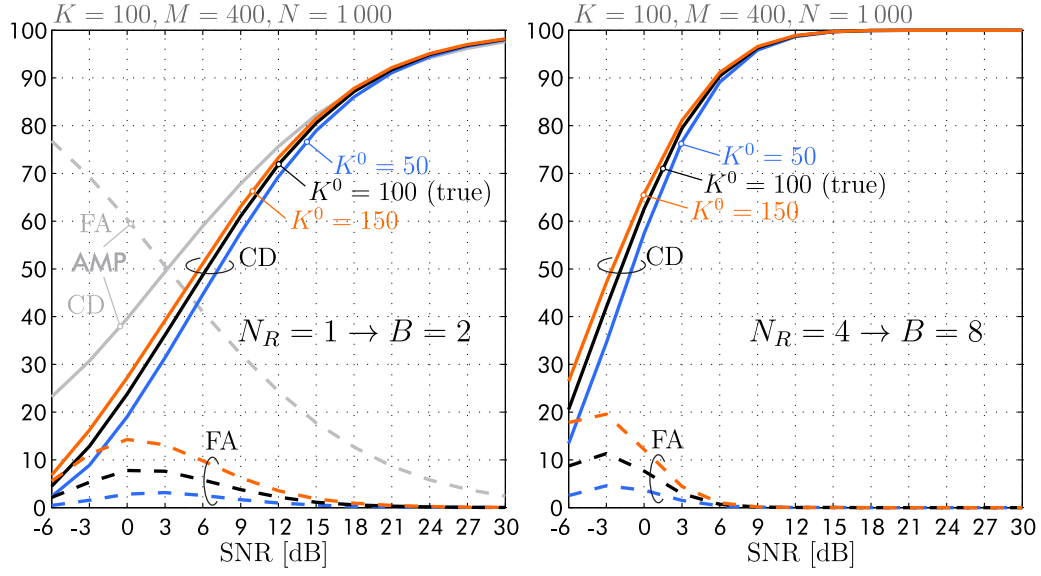
Figure 4.6 Support detection for variable SNR.

4.4 Performance Comparison

I now compare the BASSAMP-based approach that exploits joint sparsity to the AMP-based approach from Chapter 3, and to FSA-based schemes. Note that the contributions of this chapter *improve the tag acquisition phase*. I therefore introduce the following figure of merit:

Definition 4.4.1 (Bit overhead for tag acquisition). *The bit overhead for tag acquisition is defined as the number of bits required to acquire the K activated tags. It is denoted by $\beta^{(A)}$.*

This bit overhead definition excludes the identification phase and allows to compare all CS-RFID schemes whose acquisition is formulated as (3.5). It is straightforward to compute the relevant pendant to the identification throughput:

Definition 4.4.2 (Acquisition throughput). *The acquisition throughput is defined as the number of acquired tags per invested bit. It relates to the bit overhead for tag acquisition as*

$$T^{(A)} := \frac{K}{\beta^{(A)}}. \tag{4.28}$$

4.4.1 Improvement of Acquisition Phase — Perfect Conditions

Let me begin with a comparison under perfect conditions, i.e., in the noiseless case. In the following, I only consider the number of bits transmitted by the tags and omit protocol overhead and commands from the reader. The baseline is the widely employed FSA protocol, where tags are randomly scheduled to transmit in slots of a frame in order to avoid collisions. It features following bit overhead for tag acquisition:

$$\beta_{\mathrm{FSA}}^{(\mathrm{A})} = \frac{16}{T_{\mathrm{ps}}} K, \tag{4.29}$$

where T_{ps} is the throughput per slot [110], i.e., the number of tags acquired per slot, and the number 16 refers to the RN16 sequences utilized during acquisition. If the number of activated tags is known, the optimal choice of the frame size leads to a maximum average throughput of $T_{\mathrm{ps}} = e^{-1} \approx 0.368$ [42]. In [109, 110, 119], collision recovery schemes have been proposed that allow shortened frame sizes and, thus, increased throughput numbers. A reader with N_R receive antennas can resolve up to $2N_R$ collisions [110] in a slot. Assuming perfect channel knowledge and knowledge of the number of activated tags K, a reader with $N_R = 1$ receive antenna features a maximal theoretical throughput of $T_{\mathrm{ps}} = 0.841$, while a reader with $N_R = 4$ receive antennas achieves a maximal theoretical throughput of $T_{\mathrm{ps}} = 4.479$ [110].

On the other hand, there is CS-RFID in which all activated tags respond simultaneously with sequences of length M during the acquisition phase. This can be cast as a CS measurement of the type (1.19) that needs to be solved for $\mathbf{x}$ utilizing the M measurements in $\mathbf{y}$. Therefore, the optimal bit overhead for tag acquisition reads, cf. (1.26),

$$\beta_{\mathrm{CS}}^{(\mathrm{A})} = M = \left\lceil cK \log \frac{N}{K} \right\rceil. \tag{4.30}$$

In Section 2.2.3, a measurement multiplier of $c = 2$ was demonstrated to yield perfect recovery results in the noiseless case. It was demonstrated in Figure 4.4 and Section 4.3.2 that the BASSAMP-based scheme requires fewer measurements to achieve the same MSE performance, thereby allowing for $c < 2$. Scrutinizing Figure 4.5, $M \approx 210$ ($c \approx 0.9$) for $N_R = 1$ and $M \approx 140$ ($c \approx 0.6$) for $N_R = 4$ lead to perfect recovery, respectively.

Figure 4.7 depicts the bit overhead versus K for various schemes *under perfect conditions*, i.e., in the noiseless case where the channels are known to the FSA-based collision recovery scheme, and all schemes know the number of activated tags K.
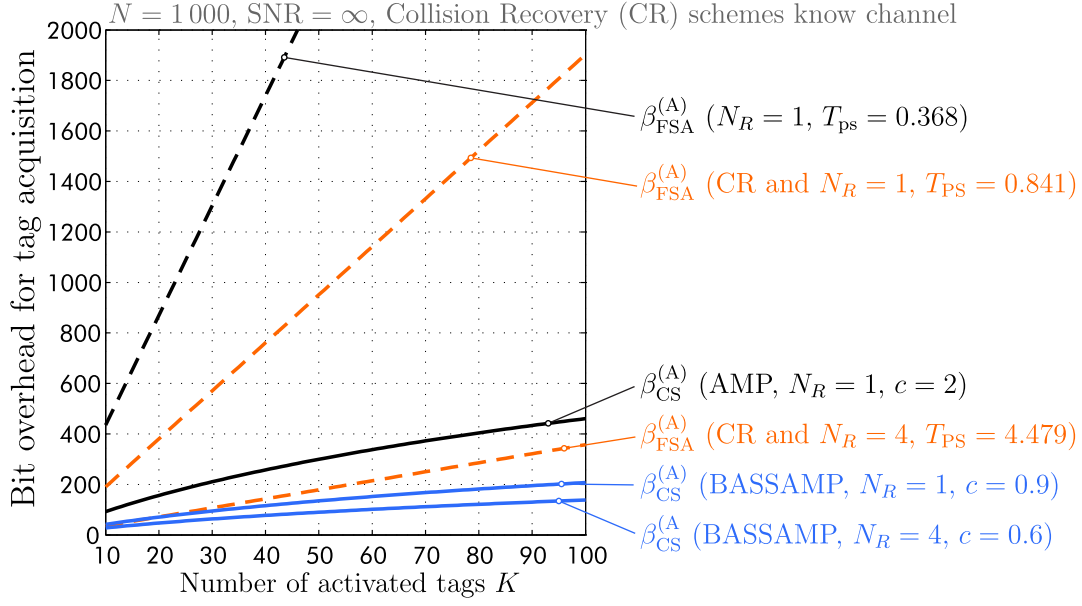
Figure 4.7 Ideal bit overhead for acquisition.

For CS-RFID, I assume $N = 1\,000$. CS-RFID strongly outperforms FSA, despite its collision recovery capabilities, and BASSAMP reduces the bit overhead over the AMP-based scheme by a factor of 2.2 ($N_R = 1$) respectively 3.3 ($N_R = 4$).

By utilizing the channel statistics and the joint sparsity among the signal vectors, a strong improvement over previous approaches is observed. Furthermore, the BASSAMP-based scheme already shows a significant improvement for a reader that employs only $N_R = 1$ receive antenna.

4.4.2   Improvement of Acquisition Phase — Imperfect Conditions

I will now advance to the noisy case. In particular, the tag responses of FSA are corrupted by noise, and the collision recovery scheme [110] faces channel estimation errors. CS-RFID has to deal with noisy CS measurements (1.19). Here are the simulation guidelines.

FSA: For the legacy FSA approach [39], the reader features one receive antenna and no collision recovery capabilities. The advanced FSA approach features collision recovery capabilities as introduced in [110]. There, the channels have to be estimated by using a set of orthogonal sequences that are transmitted prior to the RN16 sequences — note that I omit the bit overhead for the channel estimation sequences, although they will have a significant overhead in practice. The frame size is adjusted in each cycle in order to maximize the throughput

[42, 110], depending on the number of remaining tags. It is assumed that the schemes know the number of tags K. The average acquisition throughput is obtained as

$$T_{\text{FSA}}^{(\text{A})} = \frac{K}{\beta_{\text{FSA}}^{(\text{A})}} = \frac{T_{\text{ps}}}{16}. \tag{4.31}$$

I used the T_{ps} values from [110].

CS: For the AMP-based approach, the reader features one receive antenna, and AMP is employed as recovery algorithm that estimates a single complex-valued signal vector. The improved approach features the BASSAMP algorithm that was specified for the dyadic channel model. I simulated $1\,000$ random realizations of **A**, **x** and **w** and averaged the results. In each realization, CS measurements of the form (4.7) are performed in cycles, see Section 3.3.1. Each cycle features a CS tag acquisition, and the cycles are repeated until all tags are identified. In each cycle, the number of measurements (sequence length) M is set according to (1.26) based on the remaining number of unidentified tags. AMP is assumed to know K, and $\widehat{\mathcal{T}}_{\text{A}}$ is composed of the indices that correspond to the K largest entries in the recovered signal vector. BASSAMP utilizes the support estimation (4.24). The average acquisition throughput is obtained as

$$T_{\text{CS}}^{(\text{A})} = \frac{K}{\beta_{\text{CS}}^{(\text{A})}}. \tag{4.32}$$

The simulated bit overhead $\beta_{\text{CS}}^{(\text{A})}$ in (4.32) may include several cycles, whereas (4.30) refers to the optimal bit overhead of only one cycle.

Figure 4.8 depicts the average acquisition throughput versus SNR for $K = 100$ and $N = 1\,000$. The reader employs $N_R = 4$ receive antennas, and for the BASSAMP-based scheme the signature length M is varied by $c = \{0.5, 1, 2, 3\}$. By exploiting joint sparsity and knowledge of the channel statistics, BASSAMP allows to *shorten the signatures* and thereby *drastically increases the acquisition throughput* of CS-RFID. Compared to AMP with $N_R = 1$, the BASSAMP scheme with $N_R = 4$ features an *increased noise robustness* for the same signature length ($c = 3$), i.e., the noise robustness increases with increasing N_R. An advantage over collision recovery schemes is the fact that CS-RFID does not require channel knowledge, only coarse knowledge of the channel statistics. Overall, the BASSAMP approach significantly improves the acquisition phase of CS-RFID and strongly outperforms other schemes such as FSA with collision recovery.
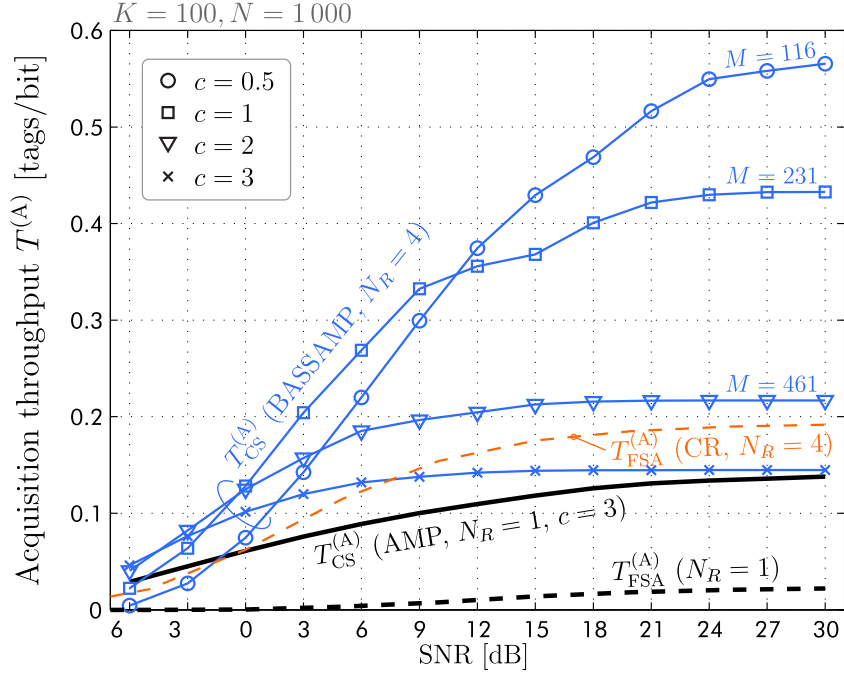
Figure 4.8 Simulated acquisition throughput vs. SNR — parameterized on c.

Let me list the achieved improvements (at high SNR) with $N_R = 4$ receive antennas at the reader. By employing BASSAMP, the tag acquisition in CS-RFID is

- 4.3 times quicker than the AMP-based approach with a single receive antenna,

- 3 times quicker than FSA with collision recovery,

- 26 times quicker than legacy FSA without collision recovery capability, and a reader with a single receive antenna.

4.4.3   Improvement of Identification

To complete the performance investigation, the simulation results of Section 4.4.2 are evaluated using the identification throughput measure from Chapter 3. To enable a realistic comparison of the schemes, the FSA-based collision recovery scheme is assumed to employ pilot sequences for channel estimation that consist of 8 bits [110]. Consequently, the bit overhead for identification reads

$$\beta_{\text{FSA}} = \underbrace{\frac{(16 + 8)}{T_{\text{ps}}} K}_{\text{Acquisition}} + \underbrace{K(A_{\text{FSA}} + D)}_{\text{Data read-out}}. \tag{4.33}$$
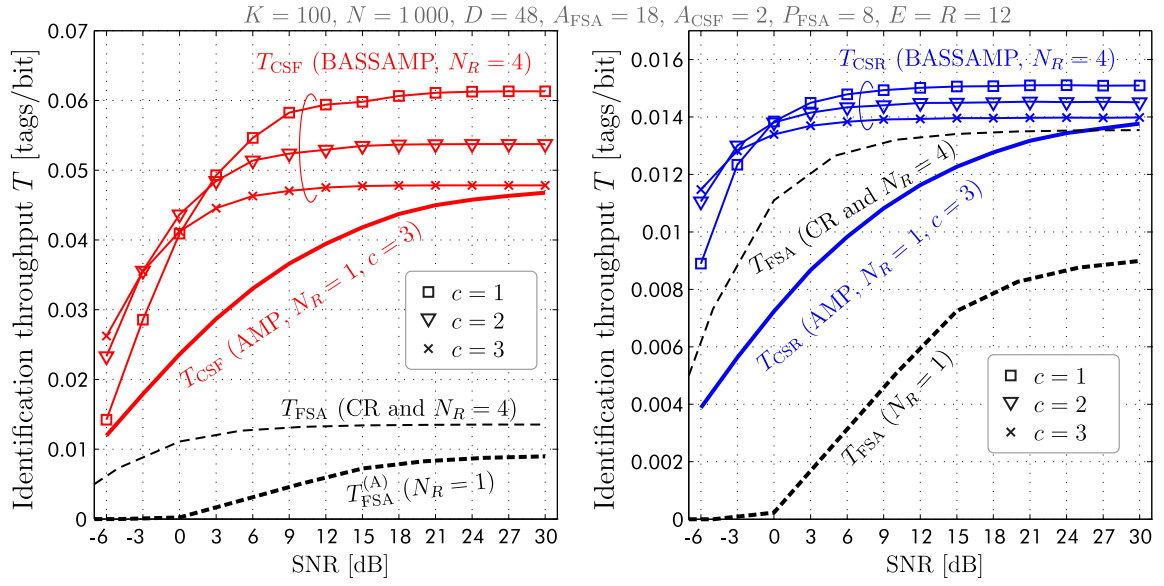
Figure 4.9 Simulated identification throughput vs. SNR — parameterized on c.

Figure 4.9 shows the simulated identification throughput over variable SNR, for CSF (left plot) and CSR (right plot). CSF and CSR exhibits a strong improvement over FSA, despite collision recovery capabilities. By employing BASSAMP, the signature length can be shortened significantly, and the noise robustness is increased.

4.5 Summary

In this chapter, I demonstrated how Bayesian prior knowledge and joint sparsity structure can be exploited in CS-RFID. Several jointly sparse vectors are obtained by employing multiple receive antennas at the reader, and by separating the signals into real and imaginary parts. The BASSAMP algorithm was specified for a dyadic channel model that is widely used in RFID; the signal entries were shown to be Bernoulli-Laplace distributed. It was verified that the relaxation to a Bernoulli-Gaussian prior reduces the performance only marginally. In practice, one would use the Bernoulli-Gaussian prior with a coarse assumption of the signal (channel) variance based on some initial measurements.

A support estimation scheme based on the BASSAMP algorithm was presented and evaluated. The detected set of assigned signatures is obtained more reliably than with standard AMP, the number of activated tags K is estimated implicitly.

By exploiting the prior knowledge about the signals from multiple receive antennas at the reader, the acquisition phase of CS-RFID is strongly improved. Here are the main benefits:

- Shorter signature sequences during tag acquisition.

- Quicker and more reliable tag acquisition.

- Higher noise robustness and lower jitter sensitivity.

The proposed CS-RFID scheme with multiple receive antennas significantly outperforms FSA with collision recovery capabilities that also utilizes several receive antennas. A big advantage over such collision recovery schemes is the fact that no channel knowledge is required in CS-RFID. Note that while the channel coefficients are estimated inherently during the acquisition phase of CS-RFID, the acquisition itself just requires binary information, i.e., whether a signal entry was zero or nonzero.

Chapter 5

Practical Implementation

To complete the thesis, this chapter demonstrates the practical feasibility of CS-RFID. After discussing the aspects and challenges of a practical implementation, I introduce a flexible measurement setup that is capable of testing non-standard RFID schemes. Measurement results verify the practical feasibility of CS-RFID with off-the-shelf hardware, and the impacts of detrimental effects are evaluated and discussed.

Based on my work in [106, 120], this chapter encompasses the following contributions:

- The hardware requirements of CS-RFID are discussed.

- The practical challenges –delay and jitter– are identified, and mitigation techniques are proposed.

- A flexible measurement setup is introduced. This setup allows to control and synchronize the responses of several UHF tags.

- Proof-of-concept measurements are performed, and the feasibility of CS-RFID with off-the-shelf hardware is demonstrated.

- The impacts of delay and jitter are evaluated by measurement.

5.1   Implementation Aspects

Let me briefly discuss the hardware requirements for CS-RFID. Passive RFID tags are typically limited in memory and do not admit the storage of a plethora of signatures; I propose the following concepts for implementation, based on the two identification protocols that were introduced in Chapter 3:

CSF: In case of *fixed signature assignment*, a tag has to produce a unique binary signature of variable length. Essentially, this signature sequence can be generated by a pseudo-random generator [114] that is seeded with the unique tag identifier, while the aspired length is advertised by the reader. Since the possible seeds are known to the reader, it can construct the sensing matrix $\mathbf{A}$. Another possibility is to store the signature directly in memory, which is typically feasible for a single signature.

CSR: In case of *random signature assignment*, a tag has to be able to produce N distinct signatures. A possible implementation could foresee two pseudo-random generators. The first one randomly selects one out of N possible numbers, i.e., selects a signature index from $\mathcal{T} = \{1, ..., N\}$. This number is then used as a seed for the second pseudo-random generator that generates the signature sequence. In this manner, there is no need to store the whole pool of signatures.

Considering today's tag requirements based on the EPCglobal standard [39], my proposed concepts result in similar complexity of the chip. The somewhat problematic issue that will be addressed in this chapter is the tag's on-board oscillator that directly affects the backscatter link frequency. CS-RFID assumes synchronized responses during tag acquisition; oscillator frequencies that vary from tag to tag hamper the acquisition process. In order to determine the actual hardware requirements, Section 5.3 introduces a flexible measurement setup that allows to control the oscillator frequencies, among other things.

On the reader side, CS-RFID features a relatively light-weight implementation. The sample-processing steps described in Section 3.2 entail simple operations. The AMP recovery algorithms proposed in Chapter 2 are of low computational complexity, an investigation of an AMP hardware implementation was performed in [71]. Considering a portable reader, the battery drainage is reduced by CS-RFID compared to EPCglobal and FSA. This is due to the significantly shorter acquisition phase that makes CS-RFID more energy efficient.

5.2   Practical Challenges and Mitigation Techniques

For many applications, RFID tags are produced in high volumes with cheap on-board oscillators — think of passive tags that replace the bar-code stickers in product identification. This introduces component tolerances and results in variations of the oscillator frequencies between the tags, thereby rendering a synchronization of tag responses a challenge. The two major issues are:

Delay: Tags respond at different times, i.e., the responses feature different delays.

Jitter: Tags respond with different link frequencies (data rate), i.e., the responses have different symbol length.

In EPCglobal with FSA [39], these issues are usually not relevant, since the slots have guard intervals, and only single responses are received in a slot. In the tag acquisition phase of CS-RFID, the activated tags respond simultaneously with signature sequences. These sequences correspond to the columns of the sensing matrix in CS. If unknown delay or jitter are present, the true physical sensing matrix is different from the assumed one that is used for CS recovery — the difference is visualized by Figure 5.1. In case of delay, the columns are shifted in time, typically by fractions of the symbol duration. In case of jitter, the symbol length varies, and the whole sequence length is affected.
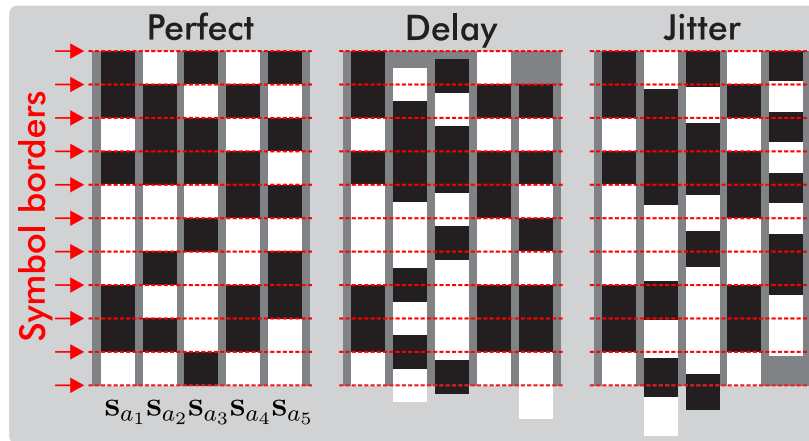
Figure 5.1 Practical challenges: delay and jitter.

In order to achieve synchronicity, the tags could either tune their on-board oscillator according to a reference beacon from the reader, or they are equipped with more accurate oscillators. Both approaches increase the hardware costs, and it is crucial to determine the actual requirements in order to balance the costs. This can

precisely be done with the measurement setup that will be introduced in Section 5.3; it allows to individually control delay and jitter of each tag. A measurement-based evaluation follows in Section 5.4.

The ramifications of delay and jitter depend on the receiver structure in the reader, i.e., how the received symbols are obtained from the samples. A widely used receiver structure is the homodyne receiver that directly converts the received signal to baseband [8]. The downconversion is performed by the same local oscillator that is used to generate the CW transmit signal. The baseband signal is then integrated, either in analog fashion by an integrator circuit, or digitally by summing up samples. Depending on the accumulated energy, a symbol decision is made. The measurement setup that I will propose in Section 5.3 acts as a homodyne receiver. The signal analyzer records baseband samples at a sampling frequency of $10\,\mathrm{MHz}$. Assuming a link frequency of $f_L = 40\,\mathrm{kHz}$, the symbol duration is $T_L = 25\,\mu\mathrm{s}$, and every symbol comprises $\xi_0 = 250$ samples.

In order to mitigate the impact of delay, I propose to employ **symbol center sampling** as illustrated by Figure 5.2. Instead of integrating over the whole symbol, only the samples around the symbol center are considered. The parameter α determines the **percentage of retained samples**, e.g., $\alpha = 0.2$ with symbol duration $\xi_0 = 250$ samples results in $\alpha\xi_0 = 50$ retained samples. This way, the inter-symbol interference due to delay is mitigated. The symbol center can be determined in various ways:

- By utilizing a preamble that is equal for all tags, e.g., $[0, 1, 0]$. The accumulated responses will exhibit an area with high energy where the 1 symbol of all responses overlapped. The center of this area represents the symbol center. Several such areas will occur if randomly generated sequences are received, the preamble is therefore not mandatory.

- By choosing the symbol center based on statistical data from observations, i.e., by trusting the expected delay value for a certain type of tag.

- By detecting the start of the tag responses, and adding half of the symbol duration $T_L/2$. This will put the reference on the earliest tag(s).

Delay puts a constant phase shift on all symbols, its effect is independent of the sequence length. The impact of jitter, on the other hand, is particularly severe for long signature sequences. Assume that Tag 1 has a link frequency of $f_{L_1} = 40\,\mathrm{kHz}$, and Tag 2 backscatters on $f_{L_2} = 40.4\,\mathrm{kHz}$, i.e., has a jitter of 1% with respect to
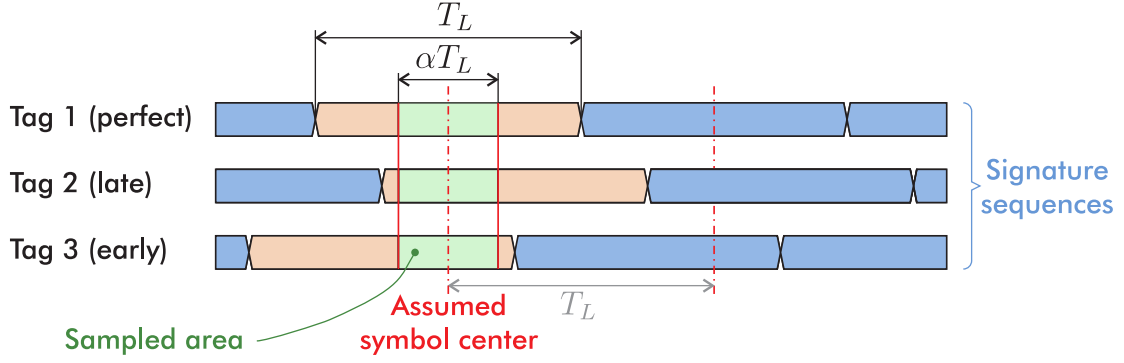
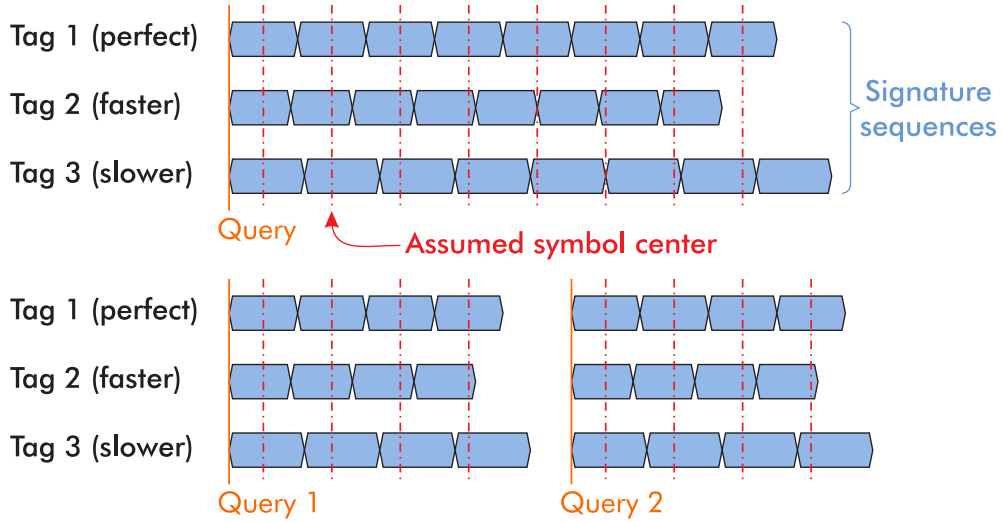Figure 5.2 Delay mitigation by symbol center sampling.

Figure 5.3 Jitter mitigation by resynchronization via query.

Tag 1. After $M = 100$ symbols, the response of Tag 1 lasts one symbol longer than the response of Tag 2. If one integrates over the whole symbol duration, this leads to severe inter-symbol interference. While the symbol center sampling technique from Figure 5.2 helps up to a certain point, it is no remedy for symbols that are delayed for more than one symbol duration. Therefore, I propose to employ **resynchronization** of the responses via query as illustrated by Figure 5.3. Instead of responding with the whole signature sequence at once, the sequence is separated into several chunks, and each chunk is triggered by a separate query. Assuming that the initial response delays have a minor impact, each query resynchronizes the sequences, and the effect of inter-symbol interference by jitter is reduced.

Note that these techniques are not optimal but rather quick and simple approaches; by sampling only the symbol center, information is discarded. A further investigation of delay- and jitter-mitigation techniques exceeds the scope of this thesis.

5.3   Flexible Measurement Setup

I will now introduce a flexible measurement setup that allows to determine the hardware requirements of passive UHF tags for CS-RFID. These are its key capabilities:

- Programmable tags that allow to test non-standard protocols are employed.

- The tags are passive and use backscatter modulation like commercial tags.

- Clock and data can be controlled individually for each tag via PC and MATLAB.

- Synchronization of several tags is achievable.

The measurement setup depicted in Figure 5.4 (product images from [121]) satisfies above criteria. It features the following components:
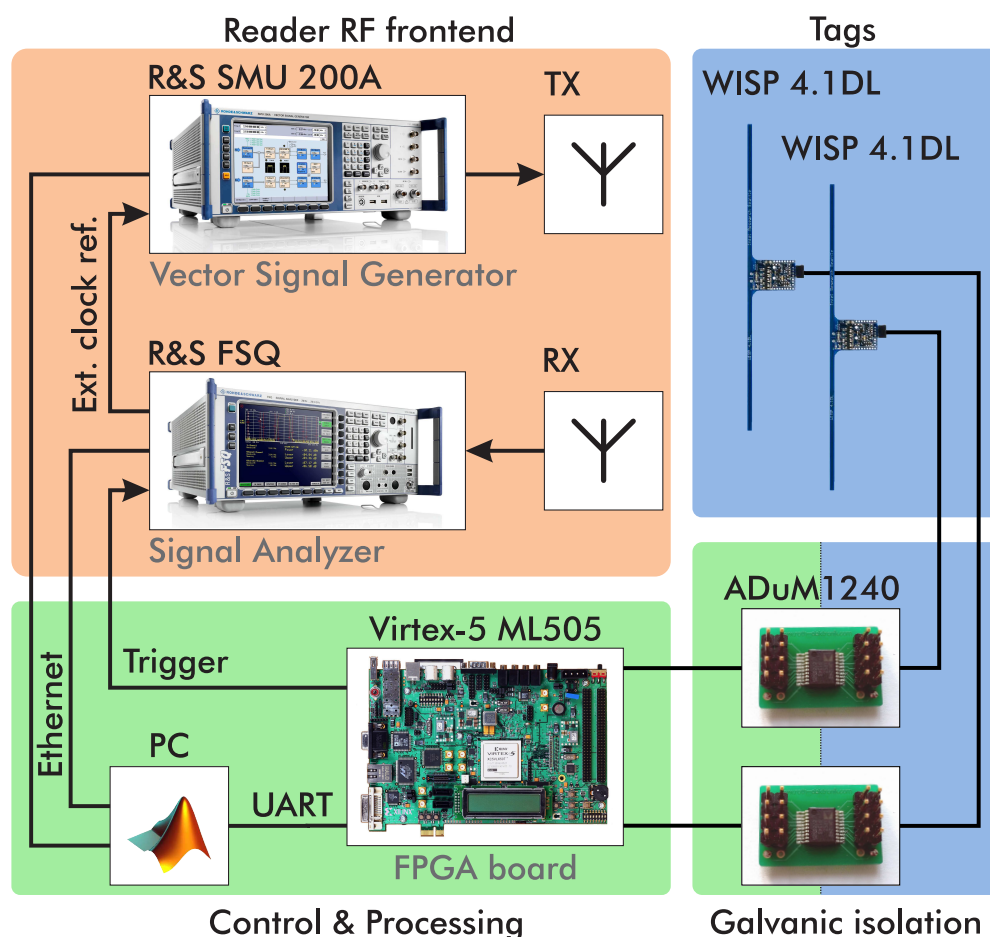
Figure 5.4 Flexible measurement setup — overview.

Reader RF frontend: To provide a constant carrier signal that powers the tags, a Rohde & Schwarz SMU 200A vector signal generator [122] (denoted SMU in the sequel) is employed. Without additional amplifier, the output power is adjustable up to $1\,\mathrm{W}$ ($30\,\mathrm{dBm}$). The RF output is connected to a circularly polarized UHF patch antenna (TX) with a gain of $9\,\mathrm{dBi}$. Signals are received by a separate antenna (RX) of the same type, the setup is bistatic. The samples are recorded by a Rohde & Schwarz FSQ signal analyzer [123] (denoted FSQ). The FSQ provides the clock reference for the SMU — this enables phase coherent operation of the two devices, a homodyne receiver is realized.

Control & Processing: The RF frontend devices (SMU & FSQ) are connected to a PC via Ethernet and are controlled by MATLAB with National Instruments VISA drivers. Via UART, the PC is connected to a ML505 experimental board that features a Virtex-5 Field-Programmable Gate Array (FPGA). This board is used to generate and control synchronous clock references for several tags. Furthermore, data that is synchronous with the respective clock reference is generated here, individually for each tag. Data refers to the tag response sequence. Clock and data are later fed to the tags.

Tags: The role of passive RFID tags is assigned to the Wireless Identification and Sensing Platform (WISP) [124–126]. The WISP[1] constitutes a programmable wireless sensing platform that operates in the UHF band. Out of the box, the matching frequency lies around $860\,\mathrm{MHz}$ and can be increased to about $960\,\mathrm{MHz}$ by trimming its dipole antenna. An energy harvesting stage allows fully passive operation, and data is transferred to the reader by backscatter modulation as used in general purpose RFID tags. The platform features a fully programmable MSP430 micro-controller operating at $1.8\,\mathrm{V}$, the current consumption ranges from $1\mu\mathrm{A}$ (low power mode) to about $100\,\mu\mathrm{A}$ (active mode). In firmware, duty cycling between the active and low power modes is utilized in order to reduce the average power consumption, which extends the possible read-out range to several meters [124, 127]. Furthermore, the platform features sensing capabilities such as a temperature sensor and an accelerometer.

Several iterations of the WISP are available, the two most recent designs are the WISP 5LRG [126] and the WISP 4.1DL [125]. Measurements conducted in [127] show that the power harvesting front end of the WISP 5LRG does not

---

[1]At this point, I would like to thank the Sensor Systems Laboratory of the University of Washington in Seattle for providing several WISPs.
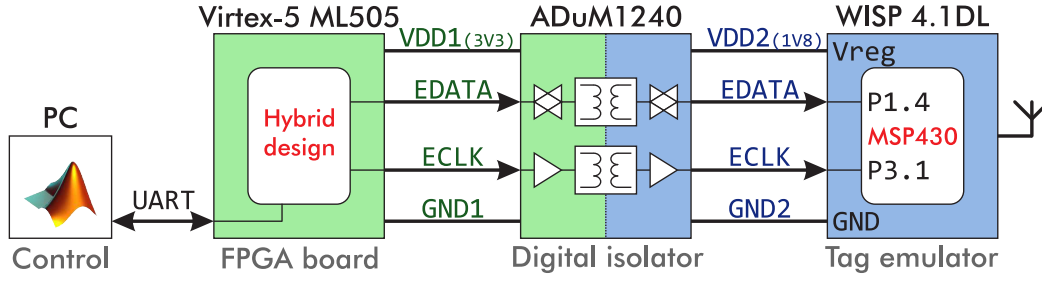
Figure 5.5 Galvanic isolation between FPGA and WISP.

generate a constant supply voltage from the carrier signal but features periodic voltage drops. This is due to the S-882Z charge pump that enables high voltage gains at the expense of a limited tag-response duration. I decided to stick to the older WISP 4.1DL whose rectifier stage provides a constant supply voltage as long as the rectified voltage lies above $1.8\,\mathrm{V}$. This enables long, uninterrupted tag responses as long as the WISP is close enough to the transmit antenna of the reader. A comprehensive measurement-based evaluation of the WISP that utilizes components of the proposed setup is provided by [127]. In order to be powered continuously by the SMU that transmits the CW at an output power of $25\,\mathrm{dBm}$, the WISP has to be within a meter of the TX antenna.

Galvanic isolation: The clock and data signals generated on the FPGA are fed to the WISP over a digital isolator as illustrated by Figure 5.5 (VDD refers to positive supply voltage, GND to ground). This enables passive tag operation since the FPGA hardware and the WISP hardware are galvanically separated. As digital isolator, an ADuM1240 is employed. This device drains an additional current of about $4\,\mu\mathrm{A}$ from the WISP if ECLK (external clock from FPGA) and EDATA (external data from FPGA) are operated at a (link) frequency of $40\,\mathrm{kHz}$. Figure 5.6 depicts the practical realization of the connection.

Before a measurement, the PC initializes the measurement devices (SMU & FSQ). Then, the PC configures the FPGA platform, the following settings can be adjusted *for each WISP* individually:

- The *link frequency* f_{L_k} (period time T_{L_k}) of the k-th WISP can be adjusted as

$$f_{L_k} = \frac{100\,\mathrm{MHz}}{2o_k}, \tag{5.1}$$

where o_k is a clock divider that ranges from 1 to 2047. The link frequency is associated with ECLK in Figure 5.5.
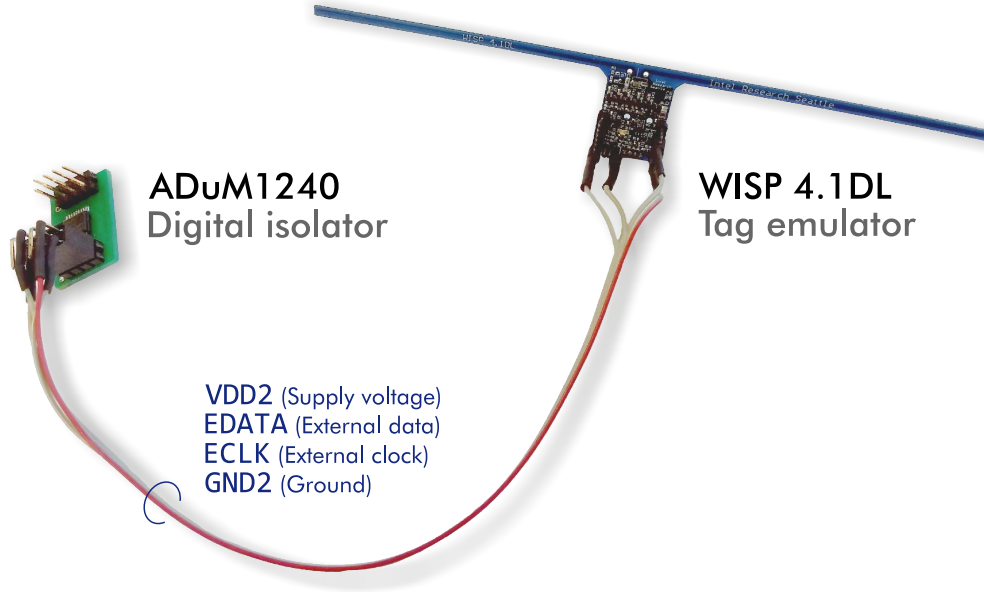
Figure 5.6 Photo of digital isolator and WISP.

- The *data symbols* that compose the tag response can be set in the FPGA. The symbols of the k-th WISP are stored in a binary vector $\mathbf{d}_k$ whose maximum length depends on the allocated FPGA memory. The data is associated with EDATA in Figure 5.5. Considering the k-th WISP, EDATA_k is synchronous with ECLK_k.

- The *phase delay* of the k-th WISP can be adjusted as

$$\phi_k = \frac{p_k}{100\,\text{MHz}} = p_k \times 10\,\text{ns}, \tag{5.2}$$

where ECLK_k and EDATA_k are delayed with respect to a common origin by ϕ_k seconds, and p_k ranges from 0 to 2047.

- The FPGA features a *continuous mode* and a *single mode* for output. In the continuous mode, ECLK_k and EDATA_k are outputted repeatedly until stopped, for all activated tags $k \in \{1, ..., K\}$. In the single mode, ECLK_k and EDATA_k are outputted once for the length of the respective data vectors $\mathbf{d}_k$, for all activated tags $k \in \{1, ..., K\}$.

- The FPGA can output a *trigger signal* in a defined time period before outputting ECLK and EDATA — this trigger signal is used to initiate the sampling at the FSQ signal analyzer.

5.4   Measurements

The measurement setup is now utilized to synchronize the WISP responses, perform a CS-RFID proof of concept, and to investigate the impact of delay and jitter.

5.4.1   Synchronization of Tag Responses

In this section, I demonstrate how the WISP responses can be synchronized by the proposed setup. Note that the Virtex-5 ML505 FPGA board has a system clock that runs at $100\,\mathrm{MHz}$ — this value determines the accuracy (granularity) of the clock (ECLK) and data (EDATA) adjustments.

Ideally, ECLK would be used by the WISP as an external clock reference. However, this is not possible with the WISP 4.1DL. To control synchronicity, each rising edge of ECLK triggers an interrupt subroutine that wakes the MSP430 controller on the WISP from a low power mode, makes it read the data at pin 1.4, and outputs the data at the antenna port (backscatter modulation). Then, the MSP430 returns to low power mode. Since these operations run on the internal clock that varies among tags, an initial adjustment of the phase delay ϕ_k is necessary, which is depicted in Figure 5.7. The MSP430 features a Digitally Controlled Oscillator (DCO) that can only be programmed for frequency ranges rather than exact values. I compared two WISPs whose DCOs have been programmed according to data-sheet to run in a range of $[1.6, 3]\,\mathrm{MHz}$. The measured DCO frequencies were $f_{\mathrm{DCO1}} = 2.7624\,\mathrm{MHz}$ and $f_{\mathrm{DCO2}} = 3.0211\,\mathrm{MHz}$, i.e., notably different. This varies the length of the interrupt subroutines and introduces a delay that can be compensated by the FPGA. Considering our setup, ECLK_2 and EDATA_2 have to be delayed by a certain $\phi_2 > 0$ while $\phi_1 = 0$ in order to guarantee synchronous responses, as illustrated by Figure 5.7.

The measurement setup[2] in accordance with Figure 5.4 is depicted in Figure 5.8. WISP1 and WISP2 respectively respond with the orthogonal sequences

$$\mathbf{d}_1 = [1, -1, -1, 1, -1, 1, 1, -1, 1, -1, -1, 1, -1, 1, 1, -1],$$
$$\mathbf{d}_2 = [1, -1, -1, 1, -1, 1, -1, 1, -1, 1, 1, -1, 1, -1, 1, -1].$$

Such sequences are used for channel estimation in multiple-response scenarios [119, 110]. The FSQ records **in-phase samples** $\mathbf{r}^{(\mathrm{I})}$ and **quadrature samples** $\mathbf{r}^{(\mathrm{Q})}$ in the baseband, the complex-valued **receive samples** are obtained as $\mathbf{r} = \mathbf{r}^{(\mathrm{I})} + j\mathbf{r}^{(\mathrm{Q})}$.

---

[2]The WISPs are mounted on an industrial grade Rohacell® 31 fixture (commercial name of poly-methacrylimide foam) that has a permittivity of $\epsilon_r = 1.05$ and a dissipation factor of $\tan(\delta) = 0.0003$ at $2.5\,\mathrm{GHz}$ and, thus, has electromagnetic properties similar to air.
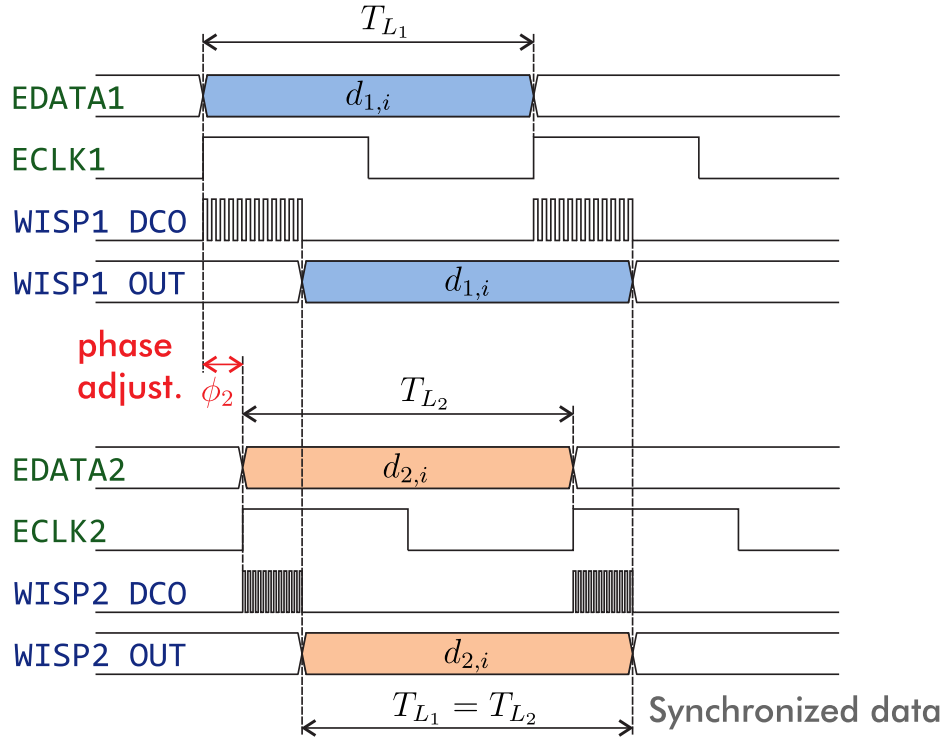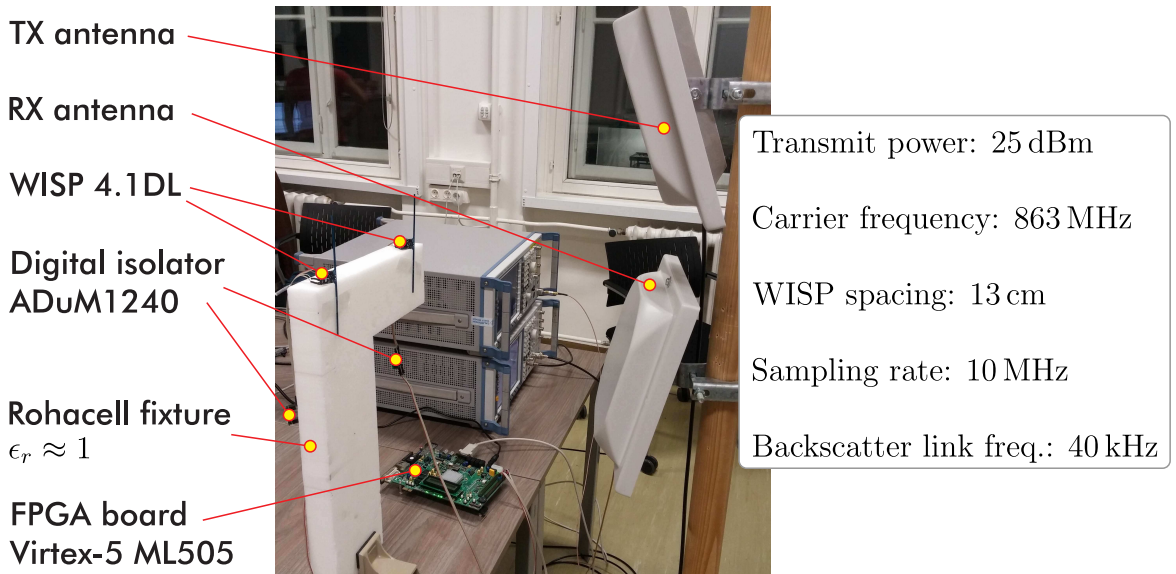
Figure 5.7 Timing diagram — synchronization of i-th WISP-data symbol.

Figure 5.8 Measurement setup for synchronization.

Figure 5.9 depicts the magnitude of unsynchronized WISP responses where two WISPs respond to a query without external control; note that a '1' symbol has been added to mark the sequence end. The DCO variation among the WISPs results in strong asynchronicity (significant delay and jitter). The FSQ records samples at a rate of $10\,\mathrm{MHz}$ which results in a measurement resolution of $100\,\mathrm{ns}$. Figure 5.10 shows two WISP responses that are synchronized by the setup. The synchronization is perfect up to the measurement resolution.
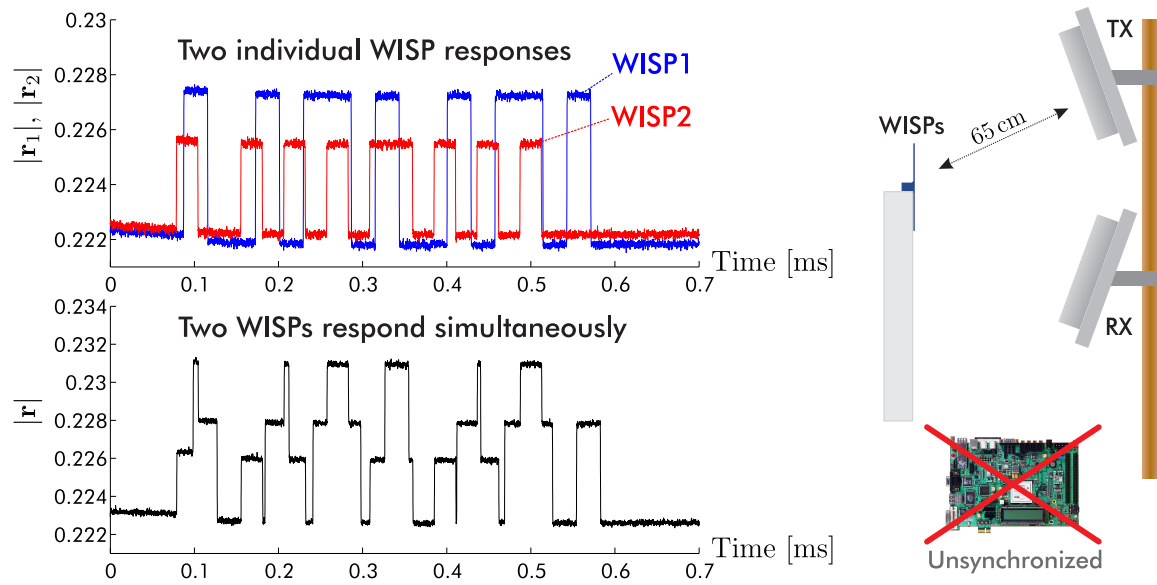
Figure 5.9 Measured WISP responses — unsynchronized case.
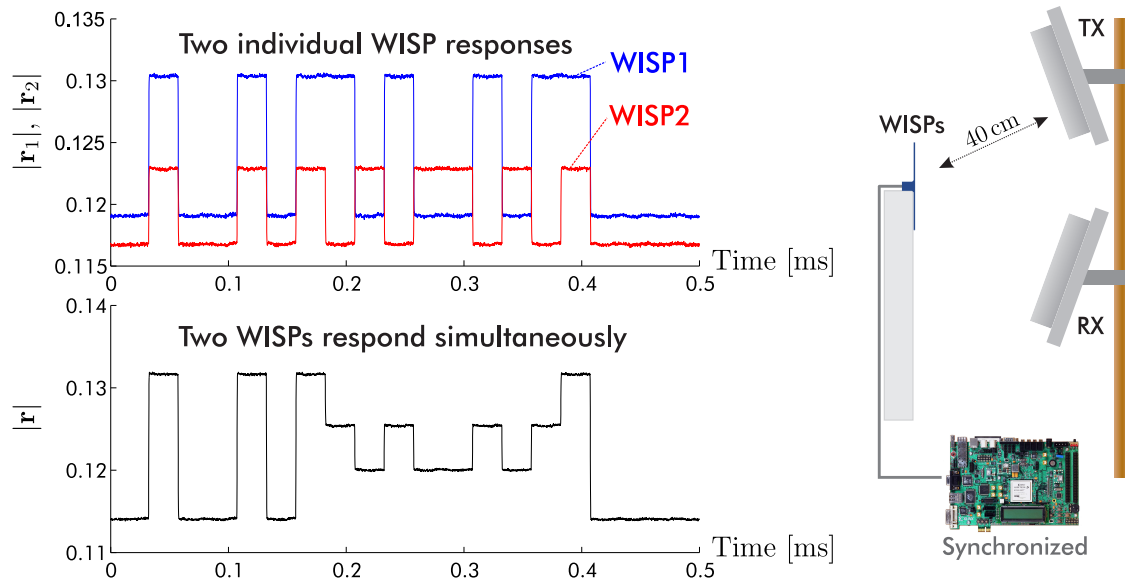
Figure 5.10 Measured WISP responses — synchronized by FPGA.

5.4.2 CS-RFID Proof of Concept

Equipped with the flexible setup, I am able to perform measurements were multiple tags respond simultaneously and synchronously with defined data. This constitutes the basis for the CS-RFID proof of concept. Note that a different antenna arrangement is used, the setup is depicted in Figure 5.11. I consider a scenario where $K = 2$ WISPs (tags) respond with signature sequences $\mathbf{s}_k \in \{b_0, b_1\}^M$, $k \in \{1, 2\}$, of length $M = 40$. The sequences feature an equal number of b_0 and b_1. These two sequences are the first two columns of a big sensing matrix $\mathbf{S}$ with $N = 1\,000$ columns.
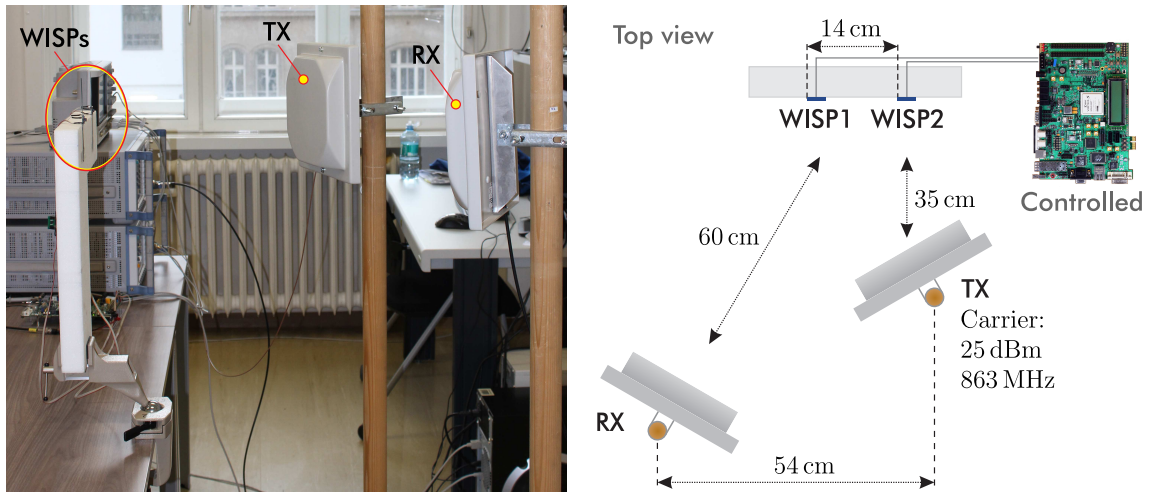
Figure 5.11 Measurement setup — antenna arrangement for proof of concept.

The **measurement methodology** foresees a response design with *three stages*, as depicted in Figure 5.12. In the first stage, both WISPs respond simultaneously with their signature sequence; this yields the samples $\mathbf{r}_{12}$. In the second and third stage, only WISP1 respectively WISP2 responds with its sequence, yielding $\mathbf{r}_1$ and $\mathbf{r}_2$. Based on the discussion in Section 3.2, the FSQ receive samples $\mathbf{r} = \mathbf{r}^{(\mathrm{I})} + j\mathbf{r}^{(\mathrm{Q})}$ are processed as follows:

1. Considering a sampling rate of $10\,\mathrm{MHz}$ and a link frequency of $40\,\mathrm{kHz}$, the symbol duration amounts to $\xi_0 = 250$ samples per symbol. The receive symbols in $\mathbf{z} = [z_1, ..., z_m, ..., z_M]^{\mathrm{T}}$ are obtained by integrating the samples $\mathbf{r} = [r_1, ..., r_l, ..., r_{M\xi_0}]^{\mathrm{T}}$ over the symbol duration ξ_0:

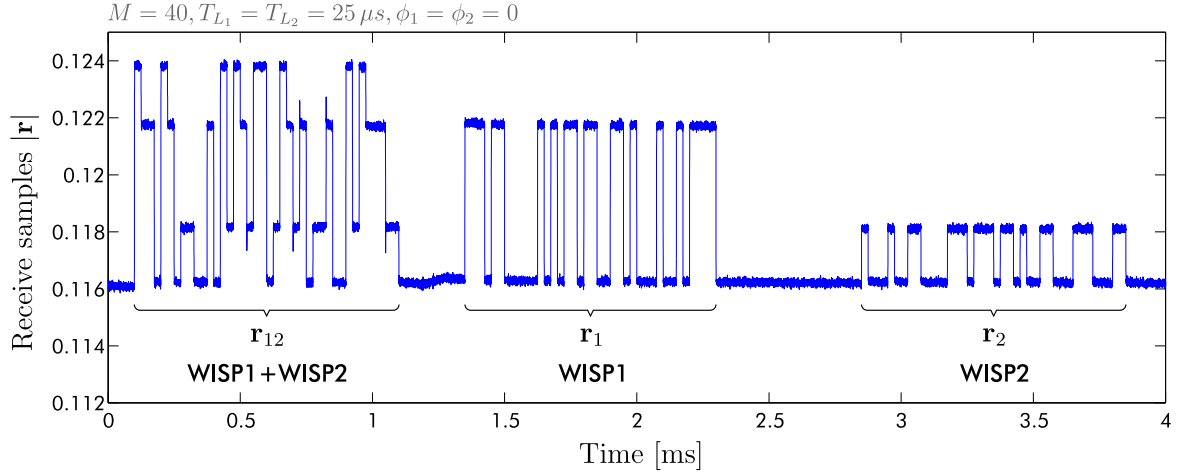$$z_m = \frac{1}{\xi_0} \sum_{l=(m-1)\xi_0+1}^{m\xi_0} r_l. \tag{5.3}$$

Figure 5.12 Measurement methodology — three stages.

If symbol center sampling as shown in Figure 5.2 is employed, the symbols are obtained as

$$z_m = \frac{1}{\alpha \xi_0} \sum_{l=[m-\frac{1}{2}(1+\alpha)]\xi_0+1}^{[m-\frac{1}{2}(1-\alpha)]\xi_0} r_l. \tag{5.4}$$

2. The mean is removed according to (3.10):

$$\mathbf{y} = \mathbf{z} - \frac{1}{M} \sum_{m=1}^{M} z_m. \tag{5.5}$$

After sample processing, the measurement vectors $\mathbf{y}_{12}$, $\mathbf{y}_1$ and $\mathbf{y}_2$ are obtained. The single-response stages allow to estimate the individual channel coefficients via inner product of the measurement vectors with the known sequences:

$$\widehat{h}_k = \langle \mathbf{y}_k, \bar{\mathbf{s}}_k \rangle = \bar{\mathbf{s}}_k^{\mathrm{H}} \mathbf{y}_k, k \in \{1, 2\}, \tag{5.6}$$

where $\bar{\mathbf{s}}_k \in \{-\bar{b}, \bar{b}\}^M$ denotes the signatures after mean removal (3.7). I assume $\bar{b} = \frac{1}{\sqrt{M}}$; this way, the channel estimates have the same scaling as the AMP or BASSAMP recoveries. Based on (5.6), I construct the **aspired channel activity vector**

$$\mathbf{x}^* = \begin{bmatrix} \widehat{h}_1 \\ \widehat{h}_2 \\ \mathbf{0}_{N-2} \end{bmatrix} \tag{5.7}$$
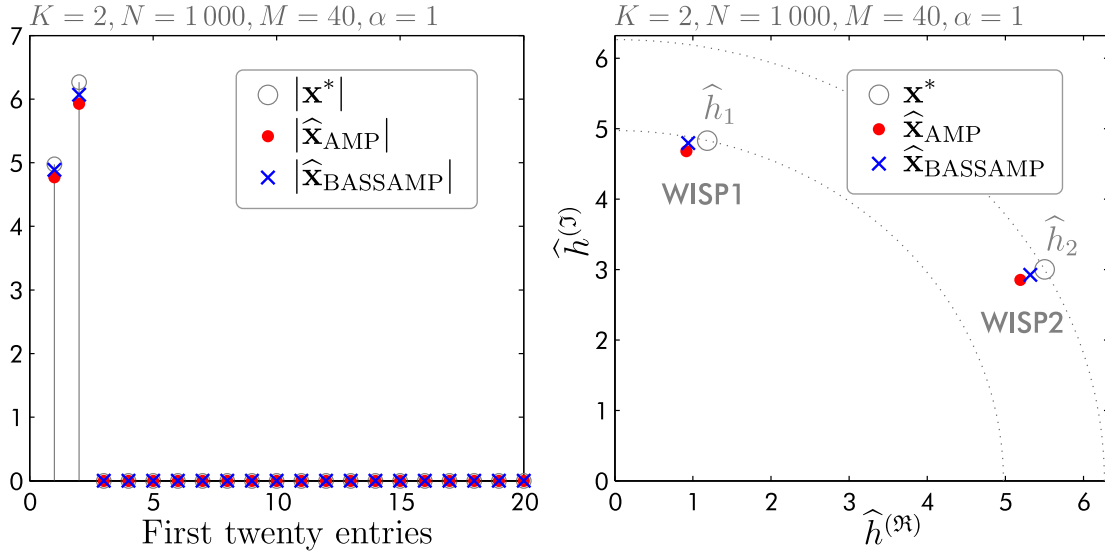
Figure 5.13 Proof of concept — recovery (left) and channel estimates (right).

that has the true support, and channel coefficients that are estimated well from the single tag responses without interference from other tags.

The measurement vector $\mathbf{y}_{12}$ from the first stage (simultaneous responses) are fed to the AMP and BASSAMP algorithms, and the recoveries $\widehat{\mathbf{x}}_{\text{AMP}}$ and $\widehat{\mathbf{x}}_{\text{BASSAMP}}$ are obtained. Note that AMP works on the complex-valued vector, while BASSAMP works on two jointly sparse vectors after real and imaginary part separation. AMP is tuned with $\lambda_{\ell_1} = 2.43$ according to (2.14). BASSAMP uses $\gamma^0 = 1 - \frac{K}{N} = 0.998$, and a Bernoulli-Gaussian prior with $\sigma_{\times}^2 = 100$. This variance is set according to the channel estimates — their magnitude is in the order of $|\widehat{h}| \approx 10$ and directly relates to the standard deviation of the nonzero entries that are assumed to be Gaussian distributed (Gaussian relaxation).

The aspired channel activity vector is compared to the estimates from the first stage. Since I chose the first two signatures of the sensing matrix, only the first two entries of the recovered vector $\widehat{\mathbf{x}}$ should be nonzero, containing the channel coefficients. The left plot in Figure 5.13 depicts the magnitude of the CS recoveries and the aspired channel activity vector. Both CS recovery algorithms produce an estimate very close to $\mathbf{x}^*$ and –most importantly for tag acquisition– the support is detected perfectly. The right plot in Figure 5.13 illustrates the channel coefficients in the complex plane — it is verified that the phase is estimated well. This result demonstrates that the proposed tag acquisition techniques of chapters 3 and 4 work on the physical layer in a real setting, which concludes my proof of concept and shows the feasibility of CS-RFID with off-the-shelf hardware.

5.4.3   Influence of Delay and Jitter

The flexible measurement setup is capable of systematically adjusting delay and jitter of several WISPs. I will now investigate the impact of such detrimental effects, considering two WISPs. Initially, both WISPs have the same link frequency of $f_{L_1} = f_{L_2} = 40\,\text{kHz}$ (symbol duration $T_{L_1} = T_{L_2} = 250\,\mu\text{s}$), and the phase delay is set to $\phi_1 = \phi_2 = 0$. WISP1 is kept at these values, while WISP2 is controlled to change the phase delay (variable delay) or the frequency (variable jitter).

The setup and methodology from Section 5.4.2 are adopted. For each value of delay or jitter, 100 frames containing $\mathbf{r}_{12}$, $\mathbf{r}_1$ and $\mathbf{r}_2$ are recorded. For each frame, the channel estimates (5.6) and the corresponding $\mathbf{x}^*$ as well as $\widehat{\mathbf{x}}_{\text{AMP}}$ and $\widehat{\mathbf{x}}_{\text{BASSAMP}}$ are computed. Moreover, the support detection and the NMSE between the aspired channel activity vector $\mathbf{x}^*$ and and the CS recoveries are computed. The support detection of AMP is defined as

$$\widehat{\mathcal{T}}_{\text{A}} = \left\{ n \in \mathcal{T} : |\widehat{x}_{\text{AMP},n}| > \epsilon_{\text{D}} \right\}, \tag{5.8}$$

where the threshold is heuristically set to $\epsilon_{\text{D}} = 0.1$ (one order of magnitude below the expected channel magnitude). BASSAMP uses the support estimation (4.24). All results are *averaged* over the 100 received frames per adjustment.

Variable Delay

The delay is adjusted as a percentage of the symbol duration by increasing $\phi_2 \geq \phi_1 = 0$, i.e., WISP2 is delayed with respect to WISP1. Figure 5.14 depicts the NMSE (left), and the channel estimates in the complex plane (right). Considering full symbol sampling ($\alpha = 1$) and an increasing delay, the NMSE increases, and the channel coefficient of WISP2 decreases in magnitude. The symbol center sampling method from Section 5.2 with $\alpha = 0.2$ is capable of restraining the delay until 40% of the symbol duration where the inter-symbol interference with the shifted signature begins. Figure 5.15 and Figure 5.16 illustrate the support detection, respectively for $\alpha = 1$ and $\alpha = 0.2$. With increasing delay, the AMP recovery is flooded with FAs. The number of FAs could be reduced by increasing the threshold ϵ_{D} in (5.8), but the probability that a true tag is missed increases as well. BASSAMP, on the other hand, provides a reliable support detection with very few FAs.

Both signatures are correctly detected until the delay of WISP2 becomes too severe, which happens at a shift by more than 50% of the symbol duration. The signature of WISP1 is fully received at all time, while the signature of WISP2 is

slowly shifted — with increasing delay, the inter-symbol interference becomes more severe. Note that WISP1 is always detected correctly, while the shifted signature of WISP2 causes the CS recovery algorithms to pick other signatures (FAs) that best explain the measurements. In other words, the recovery algorithms distribute the energy of the shifted signature among other signatures to obtain a recovery that satisfies the optimization task. Symbol center sampling can restrain this effect for reasonable delays below 50% of the symbol duration.

Variable Jitter

For this measurement, the sequences in $\mathbf{r}_{12}$ *start at the same time,* but WISP2 has a higher link frequency than WISP1:

$$f_{L_2} = (1 + \Delta f_L)f_{L_1}. \tag{5.9}$$

I chose $f_{L_1} = 40\,\text{kHz}$, and f_{L_2} is varied as $\Delta f_L \in [0, 6)\%$. The ramifications of jitter on the link frequency and the symbol duration are depicted in Table 5.1. The channel estimates from the single-response stages are computed according to the true link frequency to get the best estimates possible, i.e., the symbol duration for integration of the WISP2 response is adapted according to Table 5.1.

$\Delta f_L[\%]$	0	0.40	0.81	1.21	1.63	2.04	2.46	2.88	3.31	3.73	4.17	4.60	5.04	5.49	5.93
$f_{L_2}[\text{kHz}]$	40.00	40.16	40.32	40.49	40.65	40.82	40.98	41.15	41.32	41.49	41.67	41.84	42.02	42.19	42.37
$T_{L_2}[\mu\text{s}]$	25.00	24.90	24.80	24.70	24.60	24.50	24.40	24.30	24.20	24.10	24.00	23.90	23.80	23.70	23.60
$\xi_{0,L_2}[\text{samp.}]$	250	249	248	247	246	245	244	243	242	241	240	239	238	237	236

Table 5.1 Variable jitter — f_{L_2} is varied, $f_{L_1} = 40\,\text{kHz}$.

Figure 5.17 depicts the NMSE (left), and the channel estimates in the complex plane (right). A similar behavior to the delay case is observed. Symbol center sampling is able to alleviate the impact of relatively small jitter. The decline in the channel coefficient magnitude of WISP2 can be explained with the same argument as before, i.e., the inter-symbol interference becomes more severe for larger jitter.

Figure 5.18 and Figure 5.19 respectively depict the support detection with full symbol sampling ($\alpha = 1$) and 20% symbol center sampling ($\alpha = 0.2$). Remember that the sequences are $M = 40$ symbols long. At $\Delta f_L = 2.5\%$, the response of WISP2 is by one symbol shorter than the response of WISP1. Both sequences are detected correctly until that point; for larger jitter, WISP2 is not detected correctly anymore in all cases. Again, BASSAMP exhibits a better support detection than AMP, keeping the number of FAs very low.
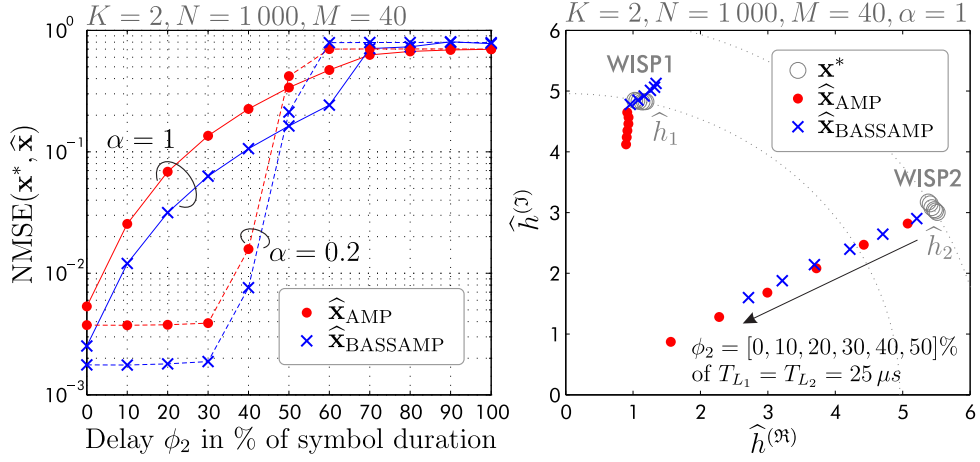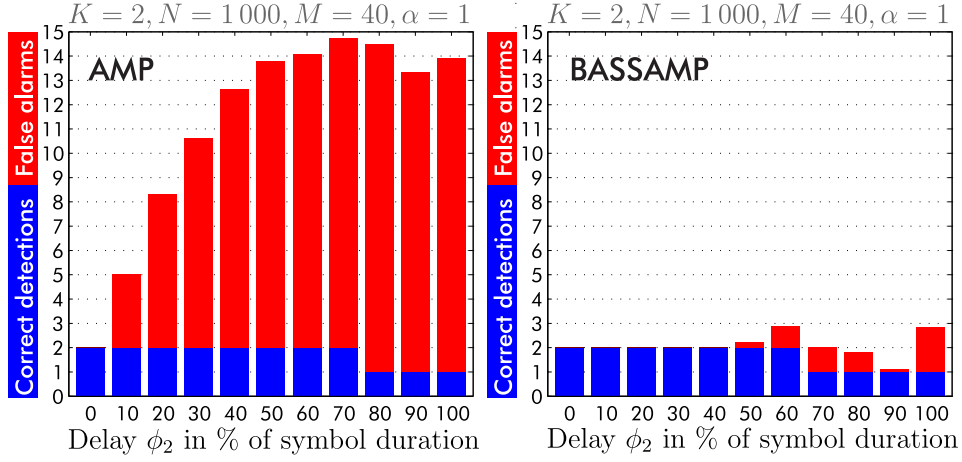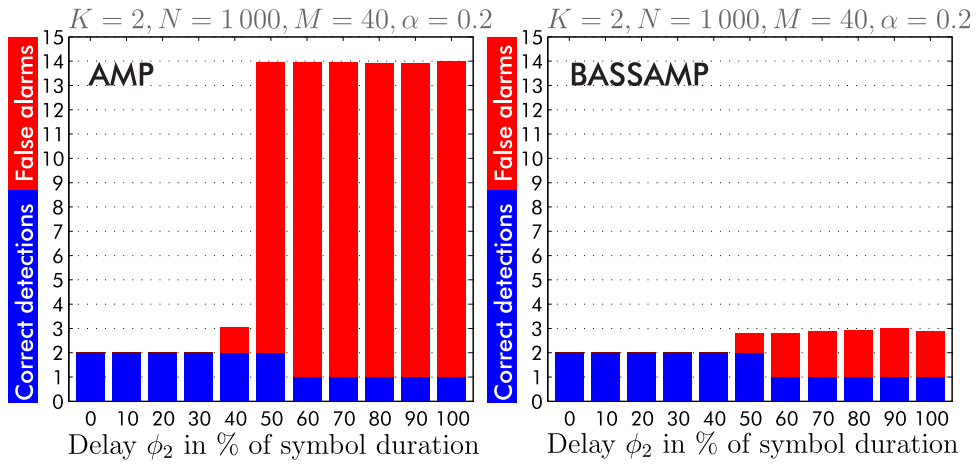
Figure 5.14 Variable delay — NMSE and channel estimates.

Figure 5.15 Variable delay — support for full symbol sampling ($\alpha = 1$).

Figure 5.16 Variable delay — support for 20% symbol sampling ($\alpha = 0.2$).
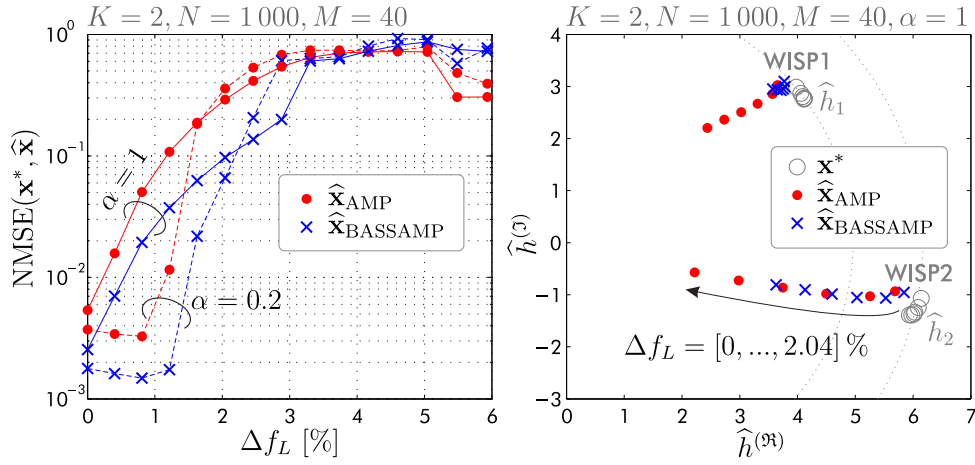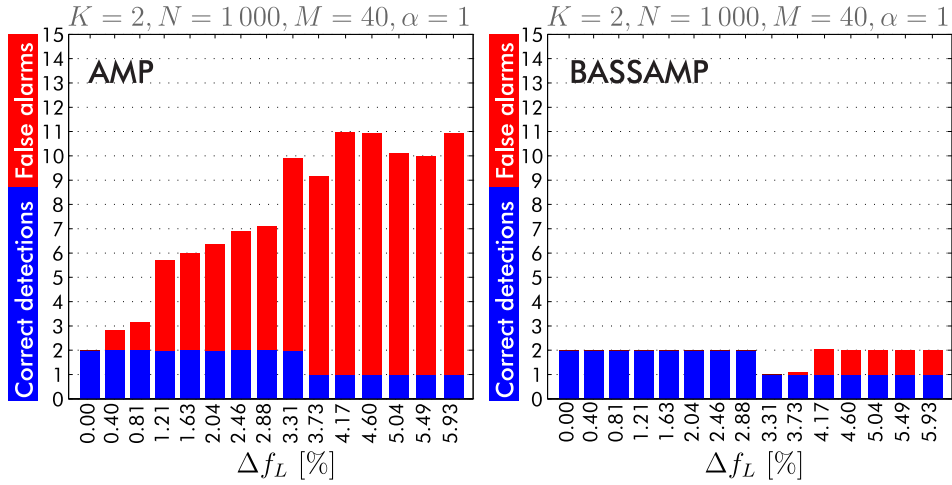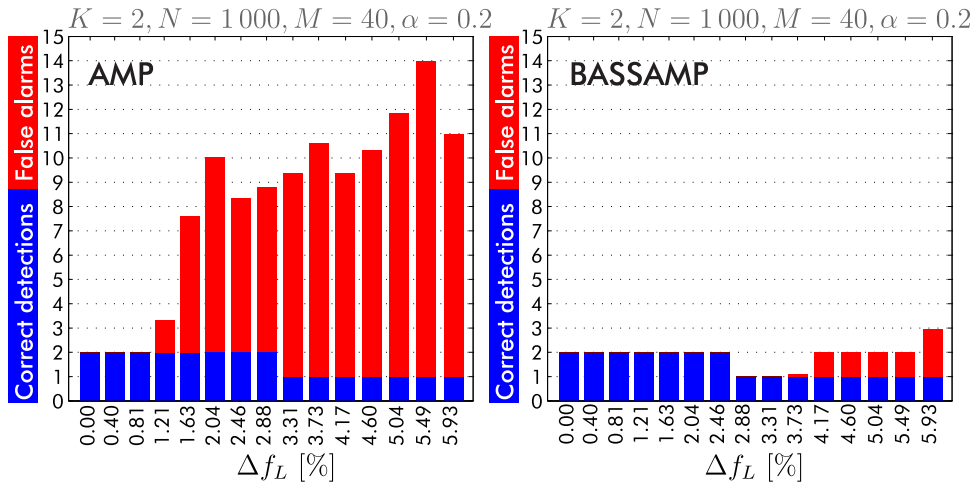
Figure 5.17 Variable jitter — NMSE and channel estimates.

Figure 5.18 Variable jitter — support for full symbol sampling ($\alpha = 1$).

Figure 5.19 Variable jitter — support for 20% symbol sampling ($\alpha = 0.2$).

Further Observations and Remarks

Based on the measurements, I collect my findings to deduce the following points:

- The impact of delay becomes problematic if the affected tag is delayed by more than 50% of the expected symbol duration.

- The impact of jitter becomes problematic if the sequence of the affected tag differs by one symbol or more from the expected sequence length.

- In case of AMP, the number of FAs can be steered with the support detection threshold ϵ_D, or by the LASSO parameter λ. The ℓ_1-optimal λ may not be the support-optimal choice; a heuristic could be found.

- In case of BASSAMP, the controllable parameter is the assumed signal variance σ_x^2. The standard deviation σ_x is set according to the expected size of the channel magnitudes. I tried several values in the range of $\sigma_x \in [5, 100]$ — the recoveries were only affected marginally, but larger values tend to reduce the NMSE and the number of FAs in the support detection.

- Despite loosing CDs in case of severe delay, all tags will be identified eventually. Since the responses that have the assumed delay are identified in all cases, the delayed instances are identified in a subsequent identification cycle, in which the delayed version of the response is the new normal. This can be seen as successive inter-symbol interference cancellation, since the detected tags are silent in a subsequent round, and their contributions are no longer in the superposition of responses. A similar argument holds for jitter, as long as it is not too severe.

- While symbol center sampling with $\alpha < 1$ improves the performance for reasonable delays below 50% of the symbol duration, it degrades the performance for larger delays where the number of detected FAs is often increased. If inter-symbol interference is unavoidable, it is better to sample the whole symbol. This is also true for large jitter.

5.5   Summary

The tag-hardware requirements of CS-RFID are comparable to the ones imposed by the EPCglobal standard, while the readers that use CS-RFID will operate more energy efficient due to the shortened tag acquisition phase.

CS-RFID assumes synchronized responses during the tag acquisition, and real-world tags exhibit variations in the on-board oscillator frequency that lead to *delay* and *jitter* — these are the main practical challenges. In order to mitigate the impact of delay, I suggested symbol center sampling that reduces the inter-symbol interference of delayed tag responses. To mitigate the impact of jitter, I proposed to transmit the signature sequences in chunks, and to resynchronize the beginning of each chunk via query. Note that these issues concern passive tags; active tags could use a phase-locked loop to tune their oscillator frequency according to a reference beacon from the reader.

I presented a flexible measurement setup to systematically control delay, jitter and data of several UHF RFID tags. This setup allows to test non-standard protocols such as CS-RFID. A proof-of-concept measurement demonstrated the practical feasibility of the scheme, followed by an investigation of the detrimental effects. Delay becomes problematic if it exceeds 50% of the symbol duration, while jitter strongly affects the recovery as soon as the total signature length among tag responses differs by one symbol or more.

In a practical setting, the AMP recoveries exhibit many FAs if detrimental effects are present. The BASSAMP recoveries, on the other hand, entail good support estimation with very few FAs that only occur in case of severe delay or jitter. Remember that each FA results in an additional enquiry during the identification cycle, thereby prolonging the identification. I suggest to use BASSAMP in practice — measurement results advocate the usage of the Gaussian relaxation in a real-world setting. The standard deviation of the channel distribution is selected to have the same order of magnitude than the magnitude of the channel coefficients. In a given setup, the channel coefficient magnitudes are determined by measuring the channel for a few tag positions, the exact or "true" variance is not required.

Chapter 6

Conclusions

The rising tide of the IoT carries a tremendous amount of new applications in RFID, and our society is slowly realizing the potential that accompanies this technology. The proliferation of RFID will strongly increase in the upcoming years, and new schemes that improve upon the current standard will unlock new, previously inconceivable applications.

In this thesis, I proposed novel approaches to the rapid identification of multiple RFID tags by a single reader device. These approaches rely on CS recovery algorithms, and I picked the class of computationally efficient and versatile AMP algorithms to do the heavy lifting. The combination of RFID and CS with AMP recovery reveals a strong new scheme termed CS-RFID that exhibits many benefits over the current state of the art. Let me list them by summarizing the contributions of this work. I will then address some open issues, and finally conclude the thesis.
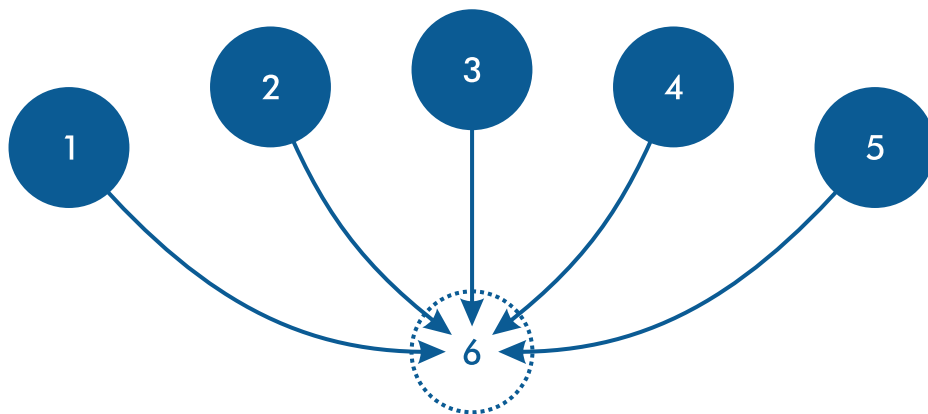
6.1   Summary of Contributions

Chapter 1 first discussed the principles of RFID, and then introduced the current state-of-the-art identification schemes, in particular FSA that is used in the prevalent EPCglobal standard. The fundamentals of CS were presented, and the classes of recovery algorithms were listed. AMP was selected because it is efficient and applicable to large signal dimensions; furthermore, a variable amount of prior knowledge that improves the recovery can be incorporated.

Chapter 2 examined various AMP recovery algorithms. The standard AMP algorithm solves the LASSO (basis pursuit) problem, the BAMP algorithm approaches MMSE estimation performance by utilizing the signal prior distribution. I specified the BAMP denoiser functions for the Bernoulli-Gaussian and the Bernoulli-Laplace signal priors. BAMP was extended to BASSAMP, a novel algorithm that incorporates joint sparsity, i.e., it exploits that several signal vectors have the same support. This algorithm came in two flavors. BASSAMP with scalar denoiser performs an extrinsic information exchange between jointly sparse signal vectors in order to reach a consensus about the support and to improve the MMSE estimation. BASSAMP with vector denoiser directly enforces the joint sparsity structure by using a multivariate signal prior, this prior can also include signal correlations. Both algorithms have the same asymptotic performance, but the vector denoiser converges in fewer iterations. All algorithms were vetted numerically by empirical phase transition curves and analytically with the SE framework. The more prior knowledge is available, the fewer measurements are required for successful recovery, and the higher is the noise robustness.

Chapter 3 introduced the concept of CS-RFID, in which the tag acquisition phase is cast as a CS measurement, and the acquisition itself is a CS recovery problem. It was outlined how AMP can be applied to solve the recovery. During acquisition, all tags respond simultaneously, each with an assigned signature sequence. The signatures are the columns of a signature matrix, and its multiplication with the channel activity vector yields the measurement vector. The number of responding tags is small compared to the total number of signatures (possible tags), the problem of identifying the activated tags is hence equivalent to detecting the support of the channel activity vector. Due to its sparsity, the signature sequences can be made shorter than the total number of possible signatures (possible tags). To ensure reliable identification, I introduced two identification protocols. The very quick CSF involves fixed

signature assignment and is applicable to fixed inventories, while the more general CSR entails random signature assignment and applies to arbitrary inventories. The performance was investigated and compared to the widely used FSA protocol, and CS-RFID was found to strongly outperform it in terms of identification throughput and noise robustness.

Chapter 4 augmented CS-RFID to the multiple measurement vector problem, in which the BASSAMP algorithm is employed to exploit joint sparsity. The origins of joint sparsity are the separation into real and imaginary parts, and the employment of several receive antennas at the reader. A Bayesian support detection scheme based on BASSAMP was introduced — it combines the soft information from several signal vectors. Tremendous gains over the standard AMP scheme were identified. Furthermore, CS-RFID was demonstrated to strongly outperform FSA collision recovery schemes that also exploit several receive antennas at the reader.

Chapter 5 investigated the practical implementation of CS-RFID. Delay and jitter among the tag responses were identified as detrimental effects that hamper the tag acquisition phase. I proposed symbol center sampling and resynchronization via query as simple remedies for delay and jitter, respectively. A flexible measurement setup that is capable of testing non-standard RFID protocols was introduced. It allows to control clock and data of several WISP UHF RFID tags, and it was shown how several WISPs can be synchronized, without delay and jitter. A proof-of-concept measurement demonstrated that CS-RFID works on the physical layer and is, hence, feasible in practice. The detrimental impacts of delay and jitter were scrutinized, and conclusions regarding the required tag-oscillator accuracy were drawn.

6.2   Open Issues

In this thesis, I mainly considered a static scenario in which the hardware or its surroundings do not move significantly during identification. This is reflected by channel coefficients that are constant throughout the acquisition phase. In some scenarios, the reader or the tags are moved significantly. This implies that the channels change during the acquisition due to small-scale fading — the CS recovery algorithms have to be adapted accordingly to grasp this behavior. An AMP scheme that successfully copes with time-varying signals was proposed in [85]; it is also capable of learning the signal (channel) parameters.

Aside from channel changes, it may also occur that tags "appear" or "disappear" during acquisition. This happens to tags that are moved into or out of the read range, or to tags that do not receive enough power from the reader and sporadically power off. I investigated the case of sporadically powered tags in [106]. It turned out that the negative impact of sporadically powered tags is relatively small compared to the impact of delay and jitter.

When it comes to the hardware implementation, the flexible setup from Chapter 5 demonstrates that CS-RFID works on the physical layer. To draw a comprehensive conclusion, the next step is to implement the *protocols* in hardware, thereby demonstrating a real-time proof of concept. Furthermore, measurements with several receive antennas at the reader are of interest. The impact of channel correlation could be determined by measurement, it depends on the antenna spacing. The vector BASSAMP Algorithm 4 is theoretically capable of incorporating such correlation among the jointly sparse signal entries, further investigation is required.

Considering the practical setup in Figure 5.4, the WISP currently limits the read range; it has to be within a meter of the transmit antenna of the reader in order to harvest enough energy for continuous backscatter transmission. If short, non-continuous responses are sufficient, larger read ranges are possible, since the micro-controller on the WISP can go into low power mode between transmissions. However, the response probability is lowered with increasing distance [127]. Future work foresees the following *alterations of the WISP*:

- Improved energy harvesting stage for larger read ranges in passive operation.

- Optional battery that enables semi-passive operation and extensive read ranges.

- The WISP 4.1 runs with the internal frequency provided by its on-board DCO. A newer version of the MSP430 micro-controller enables external clock sources. The whole chip could, therefore, run with the external clock provided by the FPGA, and the synchronization scheme from Section 5.4.1 that is based on interrupts becomes obsolete. This enables higher accuracy and higher backscatter link frequencies that are currently limited by the duration of the interrupt subroutine.

6.3   Conclusion

This thesis demonstrates a compelling case for CS-RFID — whenever the current EPCglobal standard reaches its limits, CS-RFID is capable of improving the identification throughput, the noise robustness and the energy efficiency. These improvements come at the price of a heightened sensitivity to delay and jitter in the tag responses. However, the benefits outweigh the burden, and the continuous technological advancement will alleviate these issues in the years to come.

The utilization of the AMP framework for CS recovery is an integral point of this thesis. It renders the recovery computationally efficient and feasible for a hardware implementation on the reader. Moreover, the seamless incorporation of prior knowledge allows to tailor the algorithms to specific application cases. I demonstrated that the exploitation of joint sparsity and prior knowledge greatly improves the recovery performance, which translates to quicker, more robust tag acquisition in CS-RFID.

In conclusion, I strongly believe that new, demanding applications that arise under the IoT paradigm can benefit significantly from CS-RFID.

Appendix A

Specification for Bernoulli-Gaussian Prior

Nontrivial integrals are solved with the aid of WolframAlpha®. The following identities are used, where $a, b \in \mathbb{R}$:

$$\int_{-\infty}^{\infty} \exp\left(-\widetilde{x}^2 a + \widetilde{x}^2 b\right) \, d\widetilde{x} = \left[\frac{1}{2}\sqrt{\frac{\pi}{a}} \exp\left(\frac{b^2}{4a}\right) \mathrm{erf}\left(\frac{2a\widetilde{x} - b}{2\sqrt{a}}\right)\right]_{\widetilde{x}=-\infty}^{\infty}$$

$$= \sqrt{\frac{\pi}{a}} \exp\left(\frac{b^2}{4a}\right),$$

$$\int_{-\infty}^{\infty} \widetilde{x} \cdot \exp\left(-\widetilde{x}^2 a + \widetilde{x}^2 b\right) \, d\widetilde{x} = \left[\frac{1}{4a}\sqrt{\frac{\pi}{a}} \exp\left(\frac{b^2}{4a}\right) \mathrm{erf}\left(\frac{2a\widetilde{x} - b}{2\sqrt{a}}\right) - \frac{\exp(-a\widetilde{x}^2 + b\widetilde{x})}{2a}\right]_{\widetilde{x}=-\infty}^{\infty}$$

$$= \frac{b}{2a}\sqrt{\frac{\pi}{a}} \exp\left(\frac{b^2}{4a}\right).$$

A.1 MMSE Estimator

Calculation of $F(u_n; \beta)$: To calculate (2.27), the prior (2.26) is plugged into (2.21):

$$F(u_n; \beta) = \frac{\int_{-\infty}^{\infty} \widetilde{x}_n \cdot \mathcal{N}(u_n; \widetilde{x}_n, \beta) \cdot f_{\mathsf{x}_n}(\widetilde{x}_n) d\widetilde{x}_n}{\int_{-\infty}^{\infty} \mathcal{N}(u_n; \widetilde{x}_n, \beta) \cdot f_{\mathsf{x}_n}(\widetilde{x}_n) d\widetilde{x}_n} = \frac{F^{\mathrm{N}}(u_n; \beta)}{F^{\mathrm{D}}(u_n; \beta)}.$$

Let me begin with the numerator integral (index n omitted):

$$
\begin{aligned}
F^{\mathrm{N}}(u; \beta) &= \int_{-\infty}^{\infty} \widetilde{x} \cdot \frac{1}{\sqrt{2\pi\beta}} \exp\left(-\frac{(u-\widetilde{x})^2}{2\beta}\right) \cdot \left(\gamma\delta(\widetilde{x}) + (1-\gamma_n)\mathcal{N}(\widetilde{x}; 0, \sigma_{\mathsf{x}}^2)\right) d\widetilde{x} \\
&= \frac{\gamma}{\sqrt{2\pi\beta}} \underbrace{\int_{-\infty}^{\infty} \widetilde{x} \cdot \delta(\widetilde{x}) \cdot \exp\left(-\frac{(u-\widetilde{x})^2}{2\beta}\right) d\widetilde{x}}_{=0} \\
&\quad + \frac{1}{\sqrt{2\pi\beta}} \frac{1-\gamma}{\sqrt{2\pi\sigma_{\mathsf{x}}^2}} \int_{-\infty}^{\infty} \widetilde{x} \cdot \exp\left(-\frac{\widetilde{x}^2}{2\sigma_{\mathsf{x}}^2}\right) \cdot \exp\left(-\frac{(u-\widetilde{x})^2}{2\beta}\right) d\widetilde{x} \\
&= \frac{1}{\sqrt{2\pi\beta}} \frac{1-\gamma}{\sqrt{2\pi\sigma_{\mathsf{x}}^2}} \int_{-\infty}^{\infty} \widetilde{x} \cdot \exp\left(-\frac{(u-\widetilde{x})^2}{2\beta} - \frac{\widetilde{x}^2}{2\sigma_{\mathsf{x}}^2}\right) d\widetilde{x} \\
&= \frac{\exp\left(-\frac{u^2}{2\beta}\right)}{\sqrt{2\pi\beta}} \frac{1-\gamma}{\sqrt{2\pi\sigma_{\mathsf{x}}^2}} \int_{-\infty}^{\infty} \widetilde{x} \cdot \exp\left(-\widetilde{x}^2 \underbrace{\left(\frac{\sigma_{\mathsf{x}}^2 + \beta}{2\sigma_{\mathsf{x}}^2 \beta}\right)}_{a} + \widetilde{x} \underbrace{\left(\frac{u}{\beta}\right)}_{b}\right) d\widetilde{x} \\
&= \frac{\exp\left(-\frac{u^2}{2\beta}\right)}{\sqrt{2\pi\beta}} \frac{1-\gamma}{\sqrt{2\pi\sigma_{\mathsf{x}}^2}} \frac{b}{2a} \sqrt{\frac{\pi}{a}} \exp\left(\frac{b^2}{4a}\right) \\
&= \mathcal{N}(u; 0, \beta) \frac{1-\gamma}{\sqrt{2\pi\sigma_{\mathsf{x}}^2}} \frac{u\sigma_{\mathsf{x}}^2}{\beta + \sigma_{\mathsf{x}}^2} \sqrt{\frac{\pi}{a}} \exp\left(\frac{b^2}{4a}\right).
\end{aligned}
$$

Now the denominator integral:

$$
\begin{aligned}
F^{\mathrm{D}}(u;\beta) &= \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\beta}} \exp\left(-\frac{(u-\widetilde{x})^2}{2\beta}\right) \cdot \left(\gamma\delta(\widetilde{x}) + (1-\gamma)\mathcal{N}(\widetilde{x};0,\sigma_{\mathsf{x}}^2)\right) d\widetilde{x} \\
&= \frac{\gamma}{\sqrt{2\pi\beta}} \int_{-\infty}^{\infty} \delta(\widetilde{x}) \cdot \exp\left(-\frac{(u-\widetilde{x})^2}{2\beta}\right) d\widetilde{x} \\
&\quad + \frac{1}{\sqrt{2\pi\beta}} \frac{1-\gamma}{\sqrt{2\pi\sigma_{\mathsf{x}}^2}} \int_{-\infty}^{\infty} \exp\left(-\frac{\widetilde{x}^2}{2\sigma_{\mathsf{x}}^2}\right) \cdot \exp\left(-\frac{(u-\widetilde{x})^2}{2\beta}\right) d\widetilde{x} \\
&= \gamma\frac{\exp\left(-\frac{u^2}{2\beta}\right)}{\sqrt{2\pi\beta}} + \frac{\exp\left(\frac{-u^2}{2\beta}\right)}{\sqrt{2\pi\beta}} \frac{1-\gamma}{\sqrt{2\pi\sigma_{\mathsf{x}}^2}} \int_{-\infty}^{\infty} \exp\left(-\widetilde{x}^2 a + \widetilde{x} b\right) d\widetilde{x} \\
&= \gamma\mathcal{N}(u;0,\beta) + \mathcal{N}(u;0,\beta)\frac{1-\gamma}{\sqrt{2\pi\sigma_{\mathsf{x}}^2}} \sqrt{\frac{\pi}{a}} \exp\left(\frac{b^2}{4a}\right).
\end{aligned}
$$

Now I combine the results:

$$
\begin{aligned}
F(u;\beta) &= \frac{\mathcal{N}(u;0,\beta)\frac{1-\gamma}{\sqrt{2\pi\sigma_{\mathsf{x}}^2}}\frac{u\sigma_{\mathsf{x}}^2}{\beta+\sigma_{\mathsf{x}}^2}\sqrt{\frac{\pi}{a}}\exp\left(\frac{b^2}{4a}\right)}{\gamma\mathcal{N}(u;0,\beta) + \mathcal{N}(u;0,\beta)\frac{1-\gamma}{\sqrt{2\pi\sigma_{\mathsf{x}}^2}}\sqrt{\frac{\pi}{a}}\exp\left(\frac{b^2}{4a}\right)} \\
&= u \cdot \frac{\sigma_{\mathsf{x}}^2}{\beta+\sigma_{\mathsf{x}}^2} \cdot \frac{\frac{1-\gamma}{\sqrt{2\pi\sigma_{\mathsf{x}}^2}}\sqrt{\frac{\pi}{a}}\exp\left(\frac{b^2}{4a}\right)}{\gamma + \frac{1-\gamma}{\sqrt{2\pi\sigma_{\mathsf{x}}^2}}\sqrt{\frac{\pi}{a}}\exp\left(\frac{b^2}{4a}\right)} \\
&= u \cdot \frac{\sigma_{\mathsf{x}}^2}{\beta+\sigma_{\mathsf{x}}^2} \cdot \frac{\frac{1}{\sqrt{2\pi\sigma_{\mathsf{x}}^2}}\sqrt{\frac{2\pi\sigma_{\mathsf{x}}^2\beta}{\beta+\sigma_{\mathsf{x}}^2}}\exp\left(\frac{u^2}{2}\frac{\sigma_{\mathsf{x}}^2}{\beta(\beta+\sigma_{\mathsf{x}}^2)}\right)}{\frac{\gamma}{1-\gamma} + \frac{1}{\sqrt{2\pi\sigma_{\mathsf{x}}^2}}\sqrt{\frac{2\pi\sigma_{\mathsf{x}}^2\beta}{\beta+\sigma_{\mathsf{x}}^2}}\exp\left(\frac{u^2}{2}\frac{\sigma_{\mathsf{x}}^2}{\beta(\beta+\sigma_{\mathsf{x}}^2)}\right)} \\
&= u \cdot \frac{\sigma_{\mathsf{x}}^2}{\beta+\sigma_{\mathsf{x}}^2} \cdot \frac{1}{1 + \frac{\gamma}{1-\gamma}\sqrt{\frac{\beta+\sigma_{\mathsf{x}}^2}{\beta}}\exp\left(-\frac{u^2}{2}\frac{\sigma_{\mathsf{x}}^2}{\beta(\beta+\sigma_{\mathsf{x}}^2)}\right)} \qquad \square
\end{aligned}
$$

A.2   MMSE Estimator Derivative

Calculation of $F'(u_n; \beta)$: In the sequel, I use following substitutions for clarity:

$$q = \frac{\sigma_{\mathsf{x}}^2}{\beta},$$

$$\alpha = \frac{\gamma}{1 - \gamma}\sqrt{1 + q}.$$

Let me demonstrate how (2.28) is calculated. Firstly, the product rule is applied (index n omitted for clarity):

$$F'(u; \beta) = \frac{d}{du}\left[u\frac{q}{1 + q} \cdot \frac{1}{1 + \alpha \exp\left(-\frac{u^2}{2\beta}\frac{q}{1+q}\right)}\right]$$

$$= \underbrace{\frac{q}{1 + q} \cdot \frac{1}{1 + \alpha \exp\left(-\frac{u^2}{2\beta}\frac{q}{1+q}\right)}}_{=M(u,\gamma,q)} + u\frac{q}{1 + q} \cdot \frac{d}{du}\left[\frac{1}{1 + \alpha \exp\left(-\frac{u^2}{2\beta}\frac{q}{1+q}\right)}\right]$$

Secondly, the quotient rule is applied:

$$F'(u; \beta) = M(u, \gamma, q) + u\frac{q}{1 + q} \cdot \frac{d}{du}\left[\frac{1}{1 + \alpha \exp\left(-\frac{u^2}{2\beta}\frac{q}{1+q}\right)}\right]$$

$$= M(u, \gamma, q) + u\frac{q}{1 + q} \cdot \frac{\alpha\frac{u}{\beta}\frac{q}{1+q}\exp\left(-\frac{u^2}{2\beta}\frac{q}{1+q}\right)}{\left(1 + \alpha \exp\left(-\frac{u^2}{2\beta}\frac{q}{1+q}\right)\right)^2}$$

$$= M(u, \gamma, q) + \frac{1}{\beta}\underbrace{\alpha\exp\left(-\frac{u^2}{2\beta}\frac{q}{1+q}\right)}_{m(u;\gamma,q)}\underbrace{\left(u\frac{q}{1 + q}\frac{1}{1 + \alpha \exp\left(-\frac{u^2}{2\beta}\frac{q}{1+q}\right)}\right)^2}_{F(u;\beta)^2}$$

$$= M(u, \gamma, q) + \frac{1}{\beta}m(u; \gamma, q) \cdot F(u; \beta)^2 \quad \square$$

A.3   Innovation L-Values

Calculation of L_n^t: The innovation L-values (2.70) are computed as follows, where it is used that $P(\mathsf{z}_{n,b} = 0|\mathsf{u}_{n,b} = u_{n,b}^{t-1}) = 1 - P(\mathsf{z}_{n,b} = 1|\mathsf{u}_{n,b} = u_{n,b}^{t-1})$:

$$
\begin{aligned}
L_n^t &= L(\mathsf{z}_{n,1}|\mathsf{u}_n = u_n^{t-1}) - \log \frac{\gamma_n^{t-1}}{1 - \gamma_n^{t-1}} \\
&= \log \frac{P(\mathsf{z}_{n,b} = 0|\mathsf{u}_{n,b} = u_{n,b}^{t-1})}{P(\mathsf{z}_{n,b} = 1|\mathsf{u}_{n,b} = u_{n,b}^{t-1})} \frac{1 - \gamma_n^{t-1}}{\gamma_n^{t-1}} \\
&= \log \frac{\frac{\gamma_n^{t-1} f_0(u_n^{t-1})}{\gamma_n^{t-1} f_0(u_n^{t-1}) + (1-\gamma_n^{t-1}) f_1(u_n^{t-1})}}{1 - \frac{\gamma_n^{t-1} f_0(u_n^{t-1})}{\gamma_n^{t-1} f_0(u_n^{t-1}) + (1-\gamma_n^{t-1}) f_1(u_n^{t-1})}} \frac{1 - \gamma_n^{t-1}}{\gamma_n^{t-1}} \\
&= \log \frac{\gamma_n^{t-1} f_0(u_n^{t-1})}{\gamma_n^{t-1} f_0(u_n^{t-1}) + (1 - \gamma_n^{t-1}) f_1(u_n^{t-1}) - \gamma_n^{t-1} f_0(u_n^{t-1})} \frac{1 - \gamma_n^{t-1}}{\gamma_n^{t-1}} \\
&= \log \frac{f_0(u_n^{t-1})}{f_1(u_n^{t-1})} \\
&= \log \frac{\mathcal{N}(u_n^{t-1}; 0, \beta^{t-1})}{\mathcal{N}(u_n^{t-1}; 0, \beta^{t-1} + \sigma_{\mathsf{x}_n}^2)} \\
&= \log \frac{\frac{1}{\sqrt{2\pi\beta^{t-1}}} \exp\left(-\frac{(u_n^{t-1})^2}{2\beta^{t-1}}\right)}{\frac{1}{\sqrt{2\pi(\beta^{t-1}+\sigma_{\mathsf{x}_n}^2)}} \exp\left(-\frac{(u_n^{t-1})^2}{2(\beta^{t-1}+\sigma_{\mathsf{x}_n}^2)}\right)} \\
&= \frac{1}{2} \log \frac{\beta^{t-1} + \sigma_{\mathsf{x}_n}^2}{\beta^{t-1}} - \frac{1}{2} \frac{(u_n^{t-1})^2 \sigma_{\mathsf{x}_n}^2}{\beta^{t-1}(\beta^{t-1} + \sigma_{\mathsf{x}_n}^2)} \quad \square
\end{aligned}
$$

A.4   Vector MMSE Estimator (Vector Denoiser)

Calculation of $F_{\text{vec}}(\mathbf{u}; \beta)$: The vector denoiser function (2.87) is computed according to (2.86):

$$F_{\text{vec}}(\mathbf{u}; \beta) = \frac{\int_{\mathbb{R}^B} \widetilde{\mathbf{x}} \cdot \mathcal{N}(\mathbf{u}; \widetilde{\mathbf{x}}, \mathbf{Q}_{\widetilde{\mathbf{w}}}) \cdot f_{\mathbf{x}}(\widetilde{\mathbf{x}}) \, d\widetilde{\mathbf{x}}}{\int_{\mathbb{R}^B} \mathcal{N}(\mathbf{u}; \widetilde{\mathbf{x}}, \mathbf{Q}_{\widetilde{\mathbf{w}}}) \cdot f_{\mathbf{x}}(\widetilde{\mathbf{x}}) \, d\widetilde{\mathbf{x}}} = \frac{F_{\text{vec}}^{\text{N}}(\mathbf{u}; \beta)}{F_{\text{vec}}^{\text{D}}(\mathbf{u}; \beta)}.$$

Assuming i.i.d. effective noise, the vector-likelihood is given by

$$\mathcal{N}(\mathbf{u}; \mathbf{x}, \beta \mathbf{I}_B) = \prod_{l=1}^{B} \frac{1}{\sqrt{2\pi\beta}} \exp\left(-\frac{(u_l - x_l)^2}{2\beta}\right).$$

Its multiplication with the multivariate Bernoulli-Gaussian signal prior with independently distributed entries (2.82) gives

$$\mathcal{N}(\mathbf{u}; \mathbf{x}, \beta \mathbf{I}_B) \cdot f_{\mathbf{x}}(\mathbf{x}) = \gamma \prod_{l=1}^{B} \frac{1}{\sqrt{2\pi\beta}} \delta(x_l) \exp\left(-\frac{(u_l - x_l)^2}{2\beta}\right)$$

$$+ (1-\gamma) \prod_{l=1}^{B} \frac{1}{\sqrt{2\pi\sigma_{\mathsf{x}_l}^2}} \frac{1}{\sqrt{2\pi\beta}} \exp\left(-\frac{(u_l - x_l)^2}{2\sigma_{\mathsf{x}_l}^2} - \frac{x_l^2}{2\sigma_{\mathsf{x}_l}^2}\right).$$

The numerator integral of the vector denoiser calculates as

$$F_{\text{vec}}^{\text{N}}(\mathbf{u}; \beta) = \gamma \underbrace{\int_{\mathbb{R}^B} \widetilde{\mathbf{x}} \cdot \prod_{l=1}^{B} \frac{1}{\sqrt{2\pi\beta}} \cdot \delta(\widetilde{x}_l) \cdot \exp\left(-\frac{(u_l - \widetilde{x}_l)^2}{2\beta}\right) d\widetilde{\mathbf{x}}}_{=\mathbf{0}}$$

$$+ (1-\gamma) \int_{\mathbb{R}^B} \widetilde{\mathbf{x}} \cdot \prod_{l=1}^{B} \frac{1}{\sqrt{2\pi\beta}} \frac{1}{\sqrt{2\pi\sigma_{\mathsf{x}_l}^2}} \exp\left(-\frac{(u_l - \widetilde{x}_l)^2}{2\sigma_{\mathsf{x}_l}^2} - \frac{\widetilde{x}_l^2}{2\sigma_{\mathsf{x}_l}^2}\right) d\widetilde{\mathbf{x}}$$

$$= (1-\gamma) \int_{\mathbb{R}^B} \widetilde{\mathbf{x}} \cdot \prod_{l=1}^{B} \frac{1}{\sqrt{2\pi\beta}} \frac{1}{\sqrt{2\pi\sigma_{\mathsf{x}_l}^2}} \exp\left(-\widetilde{x}_l^2 \underbrace{\frac{\sigma_{\mathsf{x}_l}^2 + \beta}{2\sigma_{\mathsf{x}_l}^2\beta}}_{a_l} + \widetilde{x}_l \underbrace{\frac{u_l}{\beta}}_{b_l} - \underbrace{\frac{u_l^2}{2\beta}}_{c_l}\right) d\widetilde{\mathbf{x}}.$$

The b-th entry of its vector output calculates as

$$
\begin{aligned}
\left(F_{\text{vec}}^{\text{N}}(\mathbf{u}; \beta)\right)_b &= (1 - \gamma) \int_{\mathbb{R}^B} \widetilde{x}_b \cdot \prod_{l=1}^{B} \frac{1}{\sqrt{2\pi\sigma_{\mathsf{x}_l}^2}} \frac{1}{\sqrt{2\pi\beta}} \exp\left(-\widetilde{x}_l^2 a_l + \widetilde{x}_l b_l - c_l\right) \, d\widetilde{\mathbf{x}} \\
&= (1 - \gamma) \int_{\mathbb{R}} \frac{1}{\sqrt{2\pi\sigma_{\mathsf{x}_b}^2}} \cdot \widetilde{x}_b \cdot \exp\left(-\widetilde{x}_b^2 a_b + \widetilde{x}_b b_b\right) \, d\widetilde{x}_b \\
&\quad \cdot \left[\prod_{l \backslash b} \int_{\mathbb{R}} \frac{1}{\sqrt{2\pi\sigma_{\mathsf{x}_l}^2}} \cdot \exp\left(-\widetilde{x}_l^2 a_l + \widetilde{x}_l b_l\right) \, d\widetilde{x}_l\right] \cdot \left[\prod_{l=1}^{B} \frac{1}{\sqrt{2\pi\beta}} \exp\left(-c_l\right)\right] \\
&= (1 - \gamma) \cdot \frac{1}{\sqrt{2\pi\sigma_{\mathsf{x}_b}^2}} \cdot \frac{b_b}{2a_b} \sqrt{\frac{\pi}{a_b}} \exp\left(\frac{b_b^2}{4a_b}\right) \\
&\quad \cdot \left[\prod_{l \backslash b} \frac{1}{\sqrt{2\pi\sigma_{\mathsf{x}_l}^2}} \cdot \sqrt{\frac{\pi}{a_l}} \exp\left(\frac{b_l^2}{4a_l}\right)\right] \cdot \left[\prod_{l=1}^{B} \frac{1}{\sqrt{2\pi\beta}} \exp\left(-c_l\right)\right] \\
&= (1 - \gamma) \cdot \frac{b_b}{2a_b} \cdot \left[\prod_{l=1}^{B} \frac{1}{\sqrt{2\pi\sigma_{\mathsf{x}_l}^2}} \cdot \sqrt{\frac{\pi}{a_l}} \exp\left(\frac{b_l^2}{4a_l}\right)\right] \cdot \left[\prod_{l=1}^{B} \frac{1}{\sqrt{2\pi\beta}} \exp\left(-c_l\right)\right].
\end{aligned}
$$

The denominator integral calculates as

$$
\begin{aligned}
F_{\text{vec}}^{\text{D}}(\mathbf{u}; \beta) &= \gamma \cdot \prod_{l=1}^{B} \int_{\mathbb{R}} \frac{1}{\sqrt{2\pi\beta}} \cdot \delta(\widetilde{x}_l) \cdot \exp\left(-\frac{(u_l - \widetilde{x}_l)^2}{2\beta}\right) \, d\widetilde{x}_l \\
&\quad + (1 - \gamma) \cdot \prod_{l=1}^{B} \int_{\mathbb{R}} \frac{1}{\sqrt{2\pi\sigma_{\mathsf{x}_l}^2}} \frac{1}{\sqrt{2\pi\beta}} \cdot \exp\left(-\frac{(u_l - \widetilde{x}_l)^2}{2\sigma_{\mathsf{x}_l}^2} - \frac{\widetilde{x}_l^2}{2\sigma_{\mathsf{x}_l}^2}\right) \, d\widetilde{x}_l \\
&= \gamma \prod_{l=1}^{B} \frac{1}{\sqrt{2\pi\beta}} \exp\left(-c_l\right) + (1 - \gamma) \prod_{l=1}^{B} \frac{1}{\sqrt{2\pi\sigma_{\mathsf{x}_l}^2}} \frac{1}{\sqrt{2\pi\beta}} \cdot \sqrt{\frac{\pi}{a_l}} \exp\left(\frac{b_l^2}{4a_l}\right) \exp\left(-c_l\right) \\
&= \left[\gamma + (1 - \gamma) \cdot \prod_{l=1}^{B} \frac{1}{\sqrt{2\pi\sigma_{\mathsf{x}_l}^2}} \cdot \sqrt{\frac{\pi}{a_l}} \exp\left(\frac{b_l^2}{4a_l}\right)\right] \cdot \left[\prod_{l=1}^{B} \frac{1}{\sqrt{2\pi\beta}} \exp\left(-c_l\right)\right].
\end{aligned}
$$

Combining the results, the b-th output of the vector denoiser boils down to

$$
\begin{aligned}
(F_{\text{vec}}(\mathbf{u};\beta))_b &= \frac{\left(F_{\text{vec}}^{\text{N}}(\mathbf{u};\beta)\right)_b}{F_{\text{vec}}^{\text{D}}(\mathbf{u};\beta)} \\
&= \frac{(1-\gamma)\cdot\frac{b_b}{2a_b}\cdot\prod_{l=1}^{B}\frac{1}{\sqrt{2\pi\sigma_{\mathsf{x}_l}^2}}\cdot\sqrt{\frac{\pi}{a_l}}\exp\left(\frac{b_l^2}{4a_l}\right)}{\gamma+(1-\gamma)\cdot\prod_{l=1}^{B}\frac{1}{\sqrt{2\pi\sigma_{\mathsf{x}_l}^2}}\cdot\sqrt{\frac{\pi}{a_l}}\exp\left(\frac{b_l^2}{4a_l}\right)} \\
&= \frac{b_b}{2a_b}\cdot\frac{1}{1+\frac{\gamma}{1-\gamma}\prod_{l=1}^{B}\cdot\sqrt{2a_l\sigma_{\mathsf{x}_l}^2}\exp\left(-\frac{b_l^2}{4a_l}\right)} \\
&= u_b\cdot\frac{\sigma_{\mathsf{x}_b}^2}{\beta+\sigma_{\mathsf{x}_b}^2}\cdot\frac{1}{1+\frac{\gamma}{1-\gamma}\prod_{l=1}^{B}\sqrt{\frac{\beta+\sigma_{\mathsf{x}_l}^2}{\beta}}\exp\left(-\frac{u_l^2}{2}\frac{\sigma_{\mathsf{x}_l}^2}{\beta(\beta+\sigma_{\mathsf{x}_l}^2)}\right)} \qquad \square
\end{aligned}
$$

A.5 Vector MMSE Estimator Derivative

Calculation of $F'_{\text{vec}}(\mathbf{u}; \beta)$: The vector denoiser derivative (2.88) computes as

$$
(F'_{\text{vec}}(\mathbf{u}; \beta))_b := \frac{\partial \left(F_{\text{vec}}(\mathbf{u}; \beta) \right)_b}{\partial u_b}.
$$

Firstly, the product rule is applied:

$$
(F'_{\text{vec}}(\mathbf{u}; \beta))_b = \frac{\partial}{\partial u_b} \left[u_b \frac{\sigma_{\mathsf{x}_b}^2}{\beta + \sigma_{\mathsf{x}_b}^2} \right] \cdot \frac{1}{1 + \frac{\gamma}{1-\gamma} \prod_{l=1}^{B} \sqrt{\frac{\beta + \sigma_{\mathsf{x}_l}^2}{\beta}} \exp\left(-\frac{u_l^2}{2} \frac{\sigma_{\mathsf{x}_l}^2}{\beta(\beta + \sigma_{\mathsf{x}_l}^2)} \right)}
$$

$$
+ u_b \frac{\sigma_{\mathsf{x}_b}^2}{\beta + \sigma_{\mathsf{x}_b}^2} \cdot \underbrace{\frac{\partial}{\partial u_b} \left[\frac{1}{1 + \frac{\gamma}{1-\gamma} \prod_{l=1}^{B} \sqrt{\frac{\beta + \sigma_{\mathsf{x}_l}^2}{\beta}} \exp\left(-\frac{u_l^2}{2} \frac{\sigma_{\mathsf{x}_l}^2}{\beta(\beta + \sigma_{\mathsf{x}_l}^2)} \right)} \right]}_{M_b(\mathbf{u})}.
$$

$M_b(\mathbf{u})$ is computed via quotient rule:

$$
M_b(\mathbf{u}) = \frac{0 - \left(-\frac{u_b \sigma_{\mathsf{x}_b}^2}{\beta(\beta + \sigma_{\mathsf{x}_b}^2)} \right) \frac{\gamma}{1-\gamma} \prod_{l=1}^{B} \sqrt{\frac{\beta + \sigma_{\mathsf{x}_l}^2}{\beta}} \exp\left(-\frac{u_l^2}{2} \frac{\sigma_{\mathsf{x}_l}^2}{\beta(\beta + \sigma_{\mathsf{x}_l}^2)} \right)}{\left[1 + \frac{\gamma}{1-\gamma} \prod_{l=1}^{B} \sqrt{\frac{\beta + \sigma_{\mathsf{x}_l}^2}{\beta}} \exp\left(-\frac{u_l^2}{2} \frac{\sigma_{\mathsf{x}_l}^2}{\beta(\beta + \sigma_{\mathsf{x}_l}^2)} \right) \right]^2}
$$

$$
= \frac{\frac{1}{\beta} \cdot u_b \frac{\sigma_{\mathsf{x}_b}^2}{\beta + \sigma_{\mathsf{x}_b}^2} \cdot \frac{\gamma}{1-\gamma} \prod_{l=1}^{B} \sqrt{\frac{\beta + \sigma_{\mathsf{x}_l}^2}{\beta}} \exp\left(-\frac{u_l^2}{2} \frac{\sigma_{\mathsf{x}_l}^2}{\beta(\beta + \sigma_{\mathsf{x}_l}^2)} \right)}{\left[1 + \frac{\gamma}{1-\gamma} \prod_{l=1}^{B} \sqrt{\frac{\beta + \sigma_{\mathsf{x}_l}^2}{\beta}} \exp\left(-\frac{u_l^2}{2} \frac{\sigma_{\mathsf{x}_l}^2}{\beta(\beta + \sigma_{\mathsf{x}_l}^2)} \right) \right]^2}.
$$

With $M_b(\mathbf{u})$ and the substitution

$$
m_{\text{vec}}(\mathbf{u}; \beta, \gamma) \equiv \frac{\gamma}{1 - \gamma} \prod_{l=1}^{B} \sqrt{\frac{\beta + \sigma_{\mathsf{x}_l}^2}{\beta}} \exp\left(-\frac{u_l^2}{2} \frac{\sigma_{\mathsf{x}_l}^2}{\beta(\beta + \sigma_{\mathsf{x}_l}^2)} \right),
$$

the derivative of the b-th vector denoiser output becomes

$$
(F'_{\text{vec}}(\mathbf{u}; \beta))_b = \frac{\sigma_{\mathsf{x}_b}^2}{\beta + \sigma_{\mathsf{x}_b}^2} \cdot \frac{1}{1 + m_{\text{vec}}(\mathbf{u}; \beta, \gamma)}
$$

$$
+ \frac{1}{\beta} \cdot m_{\text{vec}}(\mathbf{u}; \beta, \gamma) \cdot \left[u_b \frac{\sigma_{\mathsf{x}_b}^2}{\beta + \sigma_{\mathsf{x}_b}^2} \cdot \frac{1}{1 + m_{\text{vec}}(\mathbf{u}; \beta, \gamma)} \right]^2
$$

$$
= \frac{\sigma_{\mathsf{x}_b}^2}{\beta + \sigma_{\mathsf{x}_b}^2} \cdot \frac{1}{1 + m_{\text{vec}}(\mathbf{u}; \beta, \gamma)} + \frac{1}{\beta} \cdot m_{\text{vec}}(\mathbf{u}; \beta, \gamma) \cdot (F_{\text{vec}}(\mathbf{u}; \beta))_b^2 \quad \square
$$

Appendix B

Specification for Bernoulli-Laplace Prior

B.1   MMSE Estimator

Calculation of $F(u_n; \beta)$: To calculate (2.30), the prior (2.29) is plugged into (2.21):

$$F(u_n; \beta) = \frac{\int_{-\infty}^{\infty} \widetilde{x}_n \cdot \mathcal{N}(u_n; \widetilde{x}_n, \beta) \cdot f_{\mathsf{x}_n}(\widetilde{x}_n) d\widetilde{x}_n}{\int_{-\infty}^{\infty} \mathcal{N}(u_n; \widetilde{x}_n, \beta) \cdot f_{\mathsf{x}_n}(\widetilde{x}_n) d\widetilde{x}_n} = \frac{F^{\mathrm{N}}(u_n; \beta)}{F^{\mathrm{D}}(u_n; \beta)}.$$

Let me begin with the numerator integral (index n omitted):

$$\begin{aligned}
F^{\mathrm{N}}(u; \beta) &= \int_{-\infty}^{\infty} \widetilde{x} \cdot \frac{1}{\sqrt{2\pi\beta}} \exp\left(-\frac{(u-\widetilde{x})^2}{2\beta}\right) \cdot (\gamma\delta(\widetilde{x}) + (1-\gamma_n)\mathcal{L}(\widetilde{x}; 0, \kappa)) \, d\widetilde{x} \\
&= \frac{\gamma}{\sqrt{2\pi\beta}} \underbrace{\int_{-\infty}^{\infty} \widetilde{x} \cdot \delta(\widetilde{x}) \cdot \exp\left(-\frac{(u-\widetilde{x})^2}{2\beta}\right) d\widetilde{x}}_{=0} \\
&\quad + \frac{1-\gamma}{\sqrt{2\pi\beta}} \frac{1}{2\kappa} \int_{-\infty}^{\infty} \widetilde{x} \cdot \exp\left(-\frac{(u-\widetilde{x})^2}{2\beta}\right) \cdot \exp\left(-\frac{|\widetilde{x}|}{\kappa}\right) d\widetilde{x} \\
&= \frac{1-\gamma}{\sqrt{2\pi\beta}} \frac{1}{2\kappa} \int_{-\infty}^{\infty} \widetilde{x} \cdot \exp\left(-\frac{(u-\widetilde{x})^2}{2\beta} - \frac{|\widetilde{x}|}{\kappa}\right) d\widetilde{x}
\end{aligned}$$

$$= \frac{1-\gamma}{\sqrt{2\pi\beta}}\frac{1}{2\kappa}\left(\underbrace{\int_{-\infty}^{0} \widetilde{x} \cdot \exp\left(-\frac{(u-\widetilde{x})^2}{2\beta} + \frac{\widetilde{x}}{\kappa} \right) d\widetilde{x}}_{A(u)} \right.$$

$$\left. + \underbrace{\int_{0}^{\infty} \widetilde{x} \cdot \exp\left(-\frac{(u-\widetilde{x})^2}{2\beta} - \frac{\widetilde{x}}{\kappa} \right) d\widetilde{x}}_{B(u)} \right).$$

The integral $A(u)$ calculates as

$$A(u) = \int_{-\infty}^{0} \widetilde{x} \cdot \exp\left(-\frac{(u-\widetilde{x})^2}{2\beta} + \frac{\widetilde{x}}{\kappa} \right) d\widetilde{x}$$

$$= \int_{-\infty}^{0} \widetilde{x} \cdot \exp\left(-\widetilde{x}^2 \underbrace{\left(\frac{1}{2\beta} \right)}_{a} + \widetilde{x} \underbrace{\left(\frac{u}{\beta} + \frac{1}{\kappa} \right)}_{b_1} - \underbrace{\left(\frac{u^2}{2\beta} \right)}_{c} \right) d\widetilde{x}$$

$$= \frac{\exp(-c)}{4a^{\frac{3}{2}}}\left[\sqrt{\pi}b_1 \exp\left(\frac{b_1^2}{4a} \right)\operatorname{erf}\left(\frac{2a\widetilde{x} - b_1}{2\sqrt{a}} \right) - 2\sqrt{a}\exp\left(\widetilde{x}(b_1 - a\widetilde{x}) \right) \right]_{\widetilde{x}=-\infty}^{0}$$

$$= \frac{\exp(-c)}{4a^{\frac{3}{2}}}\left(\sqrt{\pi}b_1 \exp\left(\frac{b_1^2}{4a} \right)\operatorname{erf}\left(\frac{-b_1}{2\sqrt{a}} \right) - 2\sqrt{a} + \sqrt{\pi}b_1 \exp\left(\frac{b_1^2}{4a} \right) \right)$$

$$= \frac{\exp(-c)}{4a^{\frac{3}{2}}}\left(\sqrt{\pi}b_1 \exp\left(\frac{b_1^2}{4a} \right)\left(\operatorname{erf}\left(\frac{-b_1}{2\sqrt{a}} \right) + 1 \right) - 2\sqrt{a} \right)$$

$$= \frac{\exp(-c)}{4a^{\frac{3}{2}}}\left(\sqrt{\pi}b_1 \exp\left(\frac{b_1^2}{4a} \right)\left(1 + \operatorname{erfc}\left(\frac{b_1}{2\sqrt{a}} \right) \right) - 2\sqrt{a} \right).$$

Similarly, integral $B(u)$ calculates as

$$
B(u) = \int_0^\infty \widetilde{x} \cdot \exp\left(-\frac{(u-\widetilde{x})^2}{2\beta} - \frac{\widetilde{x}}{\kappa}\right) d\widetilde{x}
$$

$$
= \int_0^\infty \widetilde{x} \cdot \exp\left(-\widetilde{x}^2 \underbrace{\left(\frac{1}{2\beta}\right)}_{a} + \widetilde{x} \underbrace{\left(\frac{u}{\beta} - \frac{1}{\kappa}\right)}_{b_2} - \underbrace{\left(\frac{u^2}{2\beta}\right)}_{c}\right) d\widetilde{x}
$$

$$
= \frac{\exp(-c)}{4a^{\frac{3}{2}}}\left[\sqrt{\pi}b_2 \exp\left(\frac{b_2^2}{4a}\right)\mathrm{erf}\left(\frac{2a\widetilde{x} - b_2}{2\sqrt{a}}\right) - 2\sqrt{a}\exp\left(\widetilde{x}(b_2 - a\widetilde{x})\right)\right]_{\widetilde{x}=0}^{\infty}
$$

$$
= \frac{\exp(-c)}{4a^{\frac{3}{2}}}\left(\sqrt{\pi}b_2 \exp\left(\frac{b_2^2}{4a}\right) - \left(\sqrt{\pi}b_2 \exp\left(\frac{b_2^2}{4a}\right)\mathrm{erf}\left(\frac{-b_2}{2\sqrt{a}}\right) - 2\sqrt{a}\right)\right)
$$

$$
= \frac{\exp(-c)}{4a^{\frac{3}{2}}}\left(\sqrt{\pi}b_2 \exp\left(\frac{b_2^2}{4a}\right)\left(1 - \mathrm{erf}\left(\frac{-b_2}{2\sqrt{a}}\right)\right) + 2\sqrt{a}\right)
$$

$$
= -\frac{\exp(-c)}{4a^{\frac{3}{2}}}\left(\sqrt{\pi}b_2 \exp\left(\frac{b_2^2}{4a}\right)\left(1 + \mathrm{erfc}\left(\frac{b_2}{2\sqrt{a}}\right)\right) - 2\sqrt{a}\right).
$$

Now I plug $A(u)$ and $B(u)$ back into the numerator formulation

$$
F^{\mathrm{N}}(u;\beta) = \frac{1-\gamma}{\sqrt{2\pi\beta}}\frac{1}{2\kappa}\left(A(u) + B(u)\right)
$$

$$
= \frac{1-\gamma}{\sqrt{2\pi\beta}}\frac{1}{2\kappa}\sqrt{\pi}\frac{\exp(-c)}{4a^{\frac{3}{2}}}
$$

$$
\cdot\left[b_1 \exp\left(\frac{b_1^2}{4a}\right)\left(1 + \mathrm{erfc}\left(\frac{b_1}{2\sqrt{a}}\right)\right) - b_2 \exp\left(\frac{b_2^2}{4a}\right)\left(1 + \mathrm{erfc}\left(\frac{b_2}{2\sqrt{a}}\right)\right)\right].
$$

After re-substituting a, b_1, b_2 and c, and performing some rearrangements, the numerator expression becomes

$$
F^{\mathrm{N}}(u;\beta) = (1-\gamma)\frac{\beta}{4\kappa}\exp\left(-\frac{u^2}{2\beta}\right)
$$

$$
\cdot\left[\underbrace{\left(\frac{u}{\beta} - \frac{1}{\kappa}\right)}_{h_1(u)}\cdot\mathrm{erfc}\underbrace{\left(\frac{\sqrt{2\beta}}{2\kappa} - \frac{u}{\sqrt{2\beta}}\right)}_{g_1(u)}\cdot\exp\underbrace{\left(\left(\frac{u}{\sqrt{2\beta}} - \frac{\sqrt{2\beta}}{2\kappa}\right)^2\right)}_{g_1(u)^2}\right.
$$

$$
\left. + \underbrace{\left(\frac{u}{\beta} + \frac{1}{\kappa}\right)}_{h_2(u)}\cdot\mathrm{erfc}\underbrace{\left(\frac{\sqrt{2\beta}}{2\kappa} + \frac{u}{\sqrt{2\beta}}\right)}_{g_2(u)}\cdot\exp\underbrace{\left(\left(\frac{u}{\sqrt{2\beta}} + \frac{\sqrt{2\beta}}{2\kappa}\right)^2\right)}_{g_2(u)^2}\right].
$$

Let me now tackle the denominator:

$$F^{\mathrm{D}}(u;\beta) = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\beta}} \exp\left(-\frac{(u-\widetilde{x})^2}{2\beta}\right) \cdot (\gamma\delta(\widetilde{x}) + (1-\gamma_n)\mathcal{L}(\widetilde{x};0,\kappa))\, d\widetilde{x}$$

$$= \frac{\gamma}{\sqrt{2\pi\beta}} \int_{-\infty}^{\infty} \delta(\widetilde{x}) \cdot \exp\left(-\frac{(u-\widetilde{x})^2}{2\beta}\right) d\widetilde{x}$$

$$+ \frac{1-\gamma}{\sqrt{2\pi\beta}} \frac{1}{2\kappa} \int_{-\infty}^{\infty} \exp\left(-\frac{(u-\widetilde{x})^2}{2\beta}\right) \cdot \exp\left(-\frac{|\widetilde{x}|}{\kappa}\right) d\widetilde{x}$$

$$= \gamma \frac{1}{\sqrt{2\pi\beta}} \exp\left(-\frac{u^2}{2\beta}\right) + (1-\gamma)\frac{1}{4\kappa} \exp\left(\frac{2\beta}{4\kappa^2}\right)$$

$$\cdot \left[\mathrm{erfc}\,(g_1(u)) \exp\left(-\frac{u}{\kappa}\right) + \mathrm{erfc}\,(g_2(u)) \exp\left(\frac{u}{\kappa}\right)\right].$$

Combining the numerator and denominator and performing some rearrangements, I finally obtain the MMSE estimator function

$$F(u;\beta) = \frac{F^{\mathrm{N}}(u;\beta)}{F^{\mathrm{D}}(u;\beta)}$$

$$= \frac{\beta\left[h_1(u) \cdot \underbrace{\mathrm{erfc}\,(g_1(u)) \cdot \exp\left(g_1(u)^2\right)}_{k_1(u)} + h_2(u) \cdot \underbrace{\mathrm{erfc}\,(g_2(u)) \cdot \exp\left(g_2(u)^2\right)}_{k_2(u)}\right]}{\frac{\gamma}{1-\gamma} \cdot \frac{4\kappa}{\sqrt{2\pi\beta}} + \left[\underbrace{\mathrm{erfc}\,(g_1(u)) \cdot \exp\left(g_1(u)^2\right)}_{k_1(u)} + \underbrace{\mathrm{erfc}\,(g_2(u)) \cdot \exp\left(g_2(u)^2\right)}_{k_2(u)}\right]}$$

$$= \frac{\beta\left[h_1(u)k_1(u) + h_2(u)k_2(u)\right]}{\frac{\gamma}{1-\gamma} \frac{4\kappa}{\sqrt{2\pi\beta}} + k_1(u_n) + k_2(u)} \qquad \square$$

B.2 Signal Plus Noise Distribution

Calculation of $f_1(u_n)$: Let me lay out the calculation of (2.73), starting from the definition of $f_1(u_n)$ in (2.43):

$$
\begin{aligned}
f_1(u_n) &= f_{a_n}(x_n) * \mathcal{N}(u_n; 0, \beta) \\
&= \mathcal{L}(u_n; 0, \kappa) * \mathcal{N}(u_n; 0, \beta) \\
&= \frac{1}{2\kappa} \exp\left(-\frac{|u_n|}{\kappa}\right) * \frac{1}{\sqrt{2\pi\beta}} \exp\left(-\frac{u_n^2}{2\beta}\right) \\
&= \frac{1}{2\kappa} \frac{1}{\sqrt{2\pi\beta}} \int_{-\infty}^{\infty} \exp\left(-\frac{|\tau|}{\kappa}\right) \cdot \exp\left(-\frac{(u_n - \tau)^2}{2\beta}\right) d\tau \\
&= \frac{1}{4\kappa_n} \exp\left(-\frac{u_n^2}{2\beta}\right) \left[\underbrace{\operatorname{erfc}\left(g_1(u)\right) \cdot \exp\left(g_1(u)^2\right)}_{k_1(u)} + \underbrace{\operatorname{erfc}\left(g_2(u)\right) \cdot \exp\left(g_2(u)^2\right)}_{k_2(u)}\right] \\
&= \frac{1}{4\kappa_n} \exp\left(-\frac{u_n^2}{2\beta}\right) [k_1(u_n) + k_2(u_n)] \quad \square
\end{aligned}
$$

B.3 Innovation L-Values

Calculation of L_n^t: The innovation L-values (2.75) are computed as follows:

$$
\begin{aligned}
L_n^t &= L(\mathsf{z}_{n,1}|\mathsf{u}_n = u_n^{t-1}) - \log\frac{\gamma_n^{t-1}}{1 - \gamma_n^{t-1}} \\
&= \log\frac{P(\mathsf{z}_{n,b} = 0|\mathsf{u}_{n,b} = u_{n,b}^{t-1})}{P(\mathsf{z}_{n,b} = 1|\mathsf{u}_{n,b} = u_{n,b}^{t-1})}\frac{1 - \gamma_n^{t-1}}{\gamma_n^{t-1}} \\
&= \log\frac{\frac{\gamma_n^{t-1} f_0(u_n^{t-1})}{\gamma_n^{t-1} f_0(u_n^{t-1}) + (1 - \gamma_n^{t-1})f_1(u_n^{t-1})}}{1 - \frac{\gamma_n^{t-1} f_0(u_n^{t-1})}{\gamma_n^{t-1} f_0(u_n^{t-1}) + (1 - \gamma_n^{t-1})f_1(u_n^{t-1})}}\frac{1 - \gamma_n^{t-1}}{\gamma_n^{t-1}} \\
&= \log\frac{\gamma_n^{t-1} f_0(u_n^{t-1})}{\gamma_n^{t-1} f_0(u_n^{t-1}) + (1 - \gamma_n^{t-1})f_1(u_n^{t-1}) - \gamma_n^{t-1} f_0(u_n^{t-1})}\frac{1 - \gamma_n^{t-1}}{\gamma_n^{t-1}} \\
&= \log\frac{f_0(u_n^{t-1})}{f_1(u_n^{t-1})} \\
&= \log\frac{\frac{1}{\sqrt{2\pi\beta^{t-1}}} \exp\left(-\frac{u_n^{t-1}}{2\beta^{t-1}}\right)}{\frac{1}{4\kappa_n} \exp\left(-\frac{u_n^{t-1}}{2\beta^{t-1}}\right) [k_1(u_n^{t-1}) + k_2(u_n^{t-1})]} \\
&= \log\frac{4\kappa_n}{\sqrt{2\pi\beta^{t-1}}} - \log\left[k_1(u_n^{t-1}) + k_2(u_n^{t-1})\right] \quad \square
\end{aligned}
$$

Appendix C

Expected Number of Nonzero Entries

Let me derive (3.17). Consider the following occupancy problem: K balls are randomly distributed among N holes. The probability that a certain hole (out of N) is *not* occupied by any of the K balls is given by $\left(1 - \frac{1}{N}\right)^K$. The probability that a certain hole (out of N) is occupied is given by $1 - \left(1 - \frac{1}{N}\right)^K$. Considering all N holes, the expected number of occupied holes reads

$$O(K, N) = \left(1 - \left(1 - \frac{1}{N}\right)^K\right) N.$$

In the context of RFID with CS, the N holes correspond to the total number of signatures, and the K balls are the activated tags that choose among the N signatures.

The expected number of uniquely chosen holes (among the $O(K, N)$ occupied holes) computes as $P_R(K, N) \cdot K$, see (3.15). The expected number of repeatedly chosen holes is then given by $O(K, N) - P_R(K, N) \cdot K$. Inserting the probabilistic expressions and normalizing the expected number of repeatedly chosen holes by K, I obtain the probability of an unresolvable entry (among K) in the RFID context as (3.17).

List of Abbreviations

AMP Approximate Message Passing

ASK Amplitude Shift Keying

BAMP Bayesian Approximate Message Passing

BASSAMP BAyesian Structured Signal Approximate Message Passing

CD Correct Detection

CDMA Code Division Multiple Access

CoSaMP Compressive Sampling Matching Pursuit

CRC Cyclic Redundancy Check

CS Compressed Sensing

CSF CS-RFID with fixed signature assignment

CSR CS-RFID with random signature assignment

CW Continuous Wave

DCO Digitally Controlled Oscillator

ECLK External Clock (FPGA to WISP)

EDATA External Data (FPGA to WISP)

EM Expectation Maximization

EPC Electronic Product Code

FA False Alarm

FDMA Frequency Division Multiple Access

FPGA Field-Programmable Gate Array

FSA Frame Slotted ALOHA

FSQ Signal Analyzer by Rohde & Schwarz

HF High Frequency

IHT Iterative Hard Threshloding

IoT Internet of Things

IST Iterative Soft Threshloding

LASSO Least Absolute Shrinkage and Selection Operator

LF Low Frequency

M2M Machine-to-Machine

MAP Maximum Aposteriori Probability

MMSE Minimum Mean Squared Error

MRI Magnet Resonance Imaging

MSE Mean Squared Error

NFC Near Field Communication

NMSE Normalized Mean Squared Error

OMP Orthogonal Matching Pursuit

PC Personal Computer

PDF Probability Density Function

PMF Probability Mass Function

RF Radio Frequency

RFID Radio Frequency IDentification

RIP Restricted Isometry Property

RN16 16-bit random number (tag identifier)

SDMA Space Division Multiple Access

SE State Evolution

SMU Vector Signal Generator by Rohde & Schwarz

SNR Signal-to-Noise Ratio

TDMA Time Division Multiple Access

UART Universal Asynchronous Receiver/Transmitter (interface)

UHF Ultra High Frequency

WISP Wireless Identification and Sensing Platform

List of Figures

References

[1] E. Brynjolfsson and A. McAfee, *The second machine age: Work, progress, and prosperity in a time of brilliant technologies*. WW Norton & Company, 2014.

[2] S. Lohr, "The age of big data," *New York Times*, vol. 11, 2012.

[3] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[4] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[5] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[6] S. Li, L. Da Xu, and X. Wang, "Compressed sensing signal and data acquisition in wireless sensor networks and internet of things," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 4, pp. 2177–2186, 2013.

[7] R. Want, "An introduction to RFID technology," *IEEE Pervasive Computing*, vol. 5, no. 1, pp. 25–33, 2006.

[8] D. M. Dobkin, *The RF in RFID, Second Edition: UHF RFID in Practice*. Newnes, 2012.

[9] K. Domdouzis, B. Kumar, and C. Anumba, "Radio-Frequency Identification (RFID) applications: A brief introduction," *Advanced Engineering Informatics*, vol. 21, no. 4, pp. 350–355, 2007.

[10] X. Jia, Q. Feng, T. Fan, and Q. Lei, "RFID technology and its applications in Internet of Things (IoT)," in *2nd IEEE International Conference on Consumer Electronics, Communications and Networks (CECNet)*, pp. 1282–1285, 2012.

[11] D. HEMALATHA and B. E. AFREEN, "Development in RFID (Radio Frequency Identification) Technology in Internet of Things (IOT)," *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, vol. 4, no. 11, 2015.

[12] M. Robshaw and T. Williamson, "RAIN RFID and the Internet of Things: Industry Snapshot and Security Needs," *NIST Lightweigth Cryptography Workshop*, 2015.

[13] D. L. Donoho, "Compressed Sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[14] E. J. Candes, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on pure and applied mathematics*, vol. 59, no. 8, pp. 1207–1223, 2006.

[15] E. J. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[16] P. Demestichas, A. Georgakopoulos, D. Karvounas, K. Tsagkaris, V. Stavroulaki, J. Lu, C. Xiong, and J. Yao, "5G on the horizon: key challenges for the radio-access network," *IEEE Vehicular Technology Magazine*, vol. 8, no. 3, pp. 47–53, 2013.

[17] W. H. Chin, Z. Fan, and R. Haines, "Emerging technologies and research challenges for 5G wireless networks," *IEEE Wireless Communications*, vol. 21, no. 2, pp. 106–112, 2014.

[18] J. Landt, "The history of RFID," *IEEE Potentials*, vol. 24, no. 4, pp. 8–11, 2005.

[19] R. Weinstein, "RFID: a technical overview and its application to the enterprise," *IT professional*, vol. 7, no. 3, pp. 27–33, 2005.

[20] University of Jyväskylä via Statistica, "Projected size of the global market for RFID tags from 2010 to 2020." http://www.statista.com/statistics/299966/size-of-the-global-rfid-market/, 2016.

[21] M. Brettel, N. Friederichsen, M. Keller, and M. Rosenberg, "How virtualization, decentralization and network building change the manufacturing landscape: An Industry 4.0 Perspective," *International Journal of Mechanical, Industrial Science and Engineering*, vol. 8, no. 1, pp. 37–44, 2014.

[22] F. Michahelles, F. Thiesse, A. Schmidt, and J. R. Williams, "Pervasive RFID and near field communication technology," *IEEE Pervasive Computing*, no. 3, pp. 94–96, 2007.

[23] S. S. Saab and Z. S. Nakad, "A standalone RFID indoor positioning system using passive tags," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 5, pp. 1961–1970, 2011.

[24] P. S. Hall and Y. Hao, "Antennas and propagation for body centric communications," in *First European Conference on Antennas and Propagation (EuCAP)*, pp. 1–7, 2006.

[25] K. Koski, A. Vena, L. Sydanheimo, L. Ukkonen, and Y. Rahmat-Samii, "Design and implementation of electro-textile ground planes for wearable UHF RFID patch tag antennas," *IEEE Antennas and Wireless Propagation Letters*, vol. 12, pp. 964–967, 2013.

[26] K. Koski, L. Sydanheimo, Y. Rahmat-Samii, and L. Ukkonen, "Fundamental characteristics of electro-textiles in wearable UHF RFID patch antennas for body-centric sensing systems," *IEEE Transactions on Antennas and Propagation*, vol. 62, no. 12, pp. 6454–6462, 2014.

[27] O. Boric-Lubecke, V. M. Lubecke, B. Jokanovic, A. Singh, E. Shahhaidar, and B. Padasdao, "Microwave and wearable technologies for 5G," in *12th International Conference on Telecommunication in Modern Satellite, Cable and Broadcasting Services (TELSIKS)*, pp. 183–188, 2015.

[28] R. Want, "Enabling ubiquitous sensing with RFID," *Computer*, vol. 37, no. 4, pp. 84–86, 2004.

[29] M. Buettner, B. Greenstein, A. Sample, J. R. Smith, and D. Wetherall, "Revisiting smart dust with RFID sensor networks," in *Proceedings of the 7th ACM Workshop on Hot Topics in Networks (HotNets-VII)*, 2008.

[30] H. Aubert, "RFID technology for human implant devices," *Comptes Rendus Physique*, vol. 12, no. 7, pp. 675–683, 2011.

[31] D. He and S. Zeadally, "An analysis of RFID authentication schemes for internet of things in healthcare environment using elliptic curve cryptography," *IEEE Internet of Things Journal*, vol. 2, no. 1, pp. 72–83, 2015.

[32] M. Blowers, *Evolution of Cyber Technologies and Operations to 2035*. Springer, 2015.

[33] D. K. Klair, K.-W. Chin, and R. Raad, "A Survey and Tutorial of RFID Anti-Collision Protocols," *IEEE Communications Surveys & Tutorials*, vol. 12, no. 3, pp. 400–421, 2010.

[34] EBV Elektronik and partners, "RFID Selection Guide." https://www.adafruit.com/datasheets/rfid%20guide.pdf, 2010.

[35] C. Boyer and S. Roy, "—Invited Paper—Backscatter Communication and RFID: Coding, Energy, and MIMO Analysis," *IEEE Transactions on Communications*, vol. 62, no. 3, pp. 770–785, 2014.

[36] K. Penttila, L. Sydanheimo, and M. Kivikoski, "Implementation of Tx/Rx isolation in an RFID reader," *International Journal of Radio Frequency Identification Technology and Applications*, vol. 1, no. 1, pp. 74–89, 2006.

[37] P. V. Nikitin and K. Rao, "Antennas and propagation in UHF RFID systems," *challenge*, vol. 22, p. 23, 2008.

[38] G. Lasser, R. Langwieser, and C. F. Mecklenbräuker, "Automatic leaking carrier canceller adjustment techniques," *EURASIP Journal on Embedded Systems*, vol. 2013, no. 1, pp. 1–15, 2013.

[39] *EPC Radio-Frequency Identity Protocols Generation-2 UHF RFID*. http://www.gs1.org/gsmp/kc/epcglobal/uhfc1g2, April 2015.

[40] M. Mohaisen, H. Yoon, and K. Chang, "Radio transmission performance of EPCglobal Gen-2 RFID system," in *10th International Conference on Advanced Communication Technology (ICACT)*, vol. 2, pp. 1423–1428, 2008.

[41] D.-H. Shih, P.-L. Sun, D. C. Yen, and S.-M. Huang, "Taxonomy and survey of RFID anti-collision protocols," *Computer communications*, vol. 29, no. 11, pp. 2150–2166, 2006.

[42] M. H. Nazir, Y. Xu, and A. Johansson, "Optimal Dynamic Frame-Slotted Aloha," in *7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM)*, pp. 1–4, 2011.

[43] C. Mutti and C. Floerkemeier, "CDMA-based RFID Systems in Dense Scenarios: Concepts and Challenges," in *Proc. of IEEE International Conference on RFID*, pp. 215–222, 2008.

[44] J. Y. Maina, M. H. Mickle, M. R. Lovell, and L. A. Schaefer, "Application of CDMA for anti-collision and increased read efficiency of multiple RFID tags," *Journal of Manufacturing Systems*, vol. 26, no. 1, pp. 37–43, 2007.

[45] A. Loeffler, *Using CDMA as Anti-Collision Method for RFID-Research & Applications*. INTECH Open Access Publisher, 2011.

[46] G. Mazurek, "Active RFID system with spread-spectrum transmission," *IEEE Transactions on Automation Science and Engineering*, vol. 6, no. 1, pp. 25–32, 2009.

[47] J. Kaitovic and M. Rupp, "RFID physical layer collision recovery receivers with spatial filtering," in *IEEE International Conference on RFID Technology and Applications (RFID-TA)*, pp. 39–44, IEEE, 2015.

[48] J. Wang, H. Hassanieh, D. Katabi, and P. Indyk, "Efficient and Reliable Low-Power Backscatter Networks," *ACM SIGCOMM Computer Communication Review*, vol. 42, no. 4, pp. 61–72, 2012.

[49] M. Lustig, D. Donoho, and J. M. Pauly, "Sparse MRI: The application of compressed sensing for rapid MR imaging," *Magnetic resonance in medicine*, vol. 58, no. 6, pp. 1182–1195, 2007.

[50] Q. Li, Y. Han, and J. Dang, "Image decomposing for inpainting using compressed sensing in DCT domain," *Frontiers of Computer Science*, vol. 8, no. 6, pp. 905–915, 2014.

[51] F. Krzakala, M. Mézard, F. Sausset, Y. Sun, and L. Zdeborová, "Probabilistic reconstruction in compressed sensing: algorithms, phase diagrams, and threshold achieving matrices," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2012, no. 08, p. P08009, 2012.

[52] D. L. Donoho, A. Javanmard, and A. Montanari, "Information-theoretically optimal compressed sensing via spatial coupling and approximate message passing," *IEEE Transactions on Information Theory*, vol. 59, no. 11, pp. 7434–7464, 2013.

[53] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. Springer New York, 2010.

[54] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic Decomposition by Basis Pursuit," *SIAM journal on scientific computing*, vol. 20, no. 1, pp. 33–61, 1998.

[55] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

[56] Y. C. Eldar and G. Kutyniok, *Compressed sensing: theory and applications*. Cambridge University Press, 2012.

[57] G. Pope, "Compressive Sensing: A Summary of Reconstruction Algorithms." http://e-collection.library.ethz.ch/eserv/eth:41464/eth-41464-01.pdf, Master Thesis, Department of Computer Science, ETH Zürich, 2009.

[58] M. A. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 4, pp. 586–597, 2007.

[59] Y. C. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Conference Record of The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers*, pp. 40–44, 1993.

[60] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

[61] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2009.

[62] G. Swirszcz, N. Abe, and A. C. Lozano, "Grouped orthogonal matching pursuit for variable selection and prediction," in *Advances in Neural Information Processing Systems*, pp. 1150–1158, 2009.

[63] T. Blumensath and M. E. Davies, "Iterative thresholding for sparse approximations," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5-6, pp. 629–654, 2008.

[64] A. Maleki and D. L. Donoho, "Optimally tuned iterative reconstruction algorithms for compressed sensing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 330–341, 2010.

[65] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proceedings of the National Academy of Sciences*, vol. 106, no. 45, pp. 18914–18919, 2009.

[66] D. L. Donoho, A. Maleki, and A. Montanari, "Message Passing Algorithms for Compressed Sensing: I. Motivation and Construction," in *Proceedings of IEEE Information Theory Workshop (ITW)*, pp. 1–5, 2010.

[67] A. Maleki, "Approximate message passing algorithms for compressed sensing." http://www.ece.rice.edu/~mam15/thesis.pdf, PhD Thesis, Department of Electrical Engineering, Stanford University, 2011.

[68] D. L. Donoho, A. Maleki, and A. Montanari, "How to Design Message Passing Algorithms for Compressed Sensing," *preprint*, 2011.

[69] A. Montanari, "Graphical Models Concepts in Compressed Sensing," *Compressed Sensing: Theory and Applications*, pp. 394–438, 2012.

[70] Y. Weiss and W. T. Freeman, "Correctness of belief propagation in Gaussian graphical models of arbitrary topology," *Neural computation*, vol. 13, no. 10, pp. 2173–2200, 2001.

[71] P. Maechler, C. Studer, D. E. Bellasi, A. Maleki, A. Burg, N. Felber, H. Kaeslin, and R. G. Baraniuk, "VLSI Design of Approximate Message Passing for Signal Restoration and Compressive Sensing," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 2, no. 3, pp. 579–590, 2012.

[72] G. Lai, Y. Liu, X. Lin, and L. Zhang, "Collision-based Radio frequency identification using compressive sensing," in *15th IEEE International Conference on Communication Technology (ICCT)*, pp. 759–763, 2013.

[73] M. Kaneko, W. Hu, K. Hayashi, and H. Sakai, "Compressed Sensing-based Tag Identification Protocol for a Passive RFID System," *IEEE Communication Letters*, vol. 18, no. 11, pp. 2023–2026, 2014.

[74] M. Mayer, N. Goertz, and J. Kaitovic, "RFID Tag Acquisition via Compressed Sensing," in *Proceedings of IEEE RFID Technology and Applications Conference (RFID-TA)*, pp. 26–31, 2014.

[75] M. Mayer and N. Goertz, "Bayesian Optimal Approximate Message Passing to Recover Structured Sparse Signals," *arXiv preprint arXiv:1508.01104*, 2015.

[76] M. Mayer, G. Hannak, and N. Goertz, "Exploiting Joint Sparsity in Compressed Sensing-based RFID," *EURASIP Journal on Embedded Systems*, vol. 2016, no. 1, pp. 1–15, 2016.

[77] D. L. Donoho, A. Maleki, and A. Montanari, "Message Passing Algorithms for Compressed Sensing: II. Analysis and Validation," in *Proceedings of IEEE Information Theory Workshop (ITW)*, 2010.

[78] M. Bayati and A. Montanari, "The dynamics of message passing on dense graphs, with applications to compressed sensing," *IEEE Transactions on Information Theory*, vol. 57, no. 2, pp. 764–785, 2011.

[79] J. S. Yedidia, "Message-passing algorithms for inference and optimization," *Journal of Statistical Physics*, vol. 145, no. 4, pp. 860–890, 2011.

[80] H.-A. Loeliger, J. Dauwels, J. Hu, S. Korl, L. Ping, and F. R. Kschischang, "The Factor Graph Approach to Model-Based Signal Processing," *Proceedings of the IEEE*, vol. 95, no. 6, pp. 1295–1322, 2007.

[81] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 498–519, 2001.

[82] A. Mousavi, A. Maleki, and R. G. Baraniuk, "Asymptotic Analysis of LAS-SOs Solution Path with Implications for Approximate Message Passing," *arXiv preprint arXiv:1309.5979*, 2013.

[83] S. Rangan, P. Schniter, E. Riegler, A. Fletcher, and V. Cevher, "Fixed points of generalized approximate message passing with arbitrary matrices," in *IEEE International Symposium on Information Theory Proceedings (ISIT)*, pp. 664–668, 2013.

[84] A. Maleki, L. Anitori, Z. Yang, and R. G. Baraniuk, "Asymptotic analysis of complex LASSO via complex approximate message passing (CAMP)," *IEEE Transactions on Information Theory*, vol. 59, no. 7, pp. 4290–4308, 2013.

[85] J. Ziniel and P. Schniter, "Efficient high-dimensional inference in the multiple measurement vector problem," *IEEE Transactions on Signal Processing*, vol. 61, no. 2, pp. 340–354, 2013.

[86] S. F. Cotter, B. D. Rao, K. Engan, and K. Kreutz-Delgado, "Sparse solutions to linear inverse problems with multiple measurement vectors," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2477–2488, 2005.

[87] I. F. Gorodnitsky, J. S. George, and B. D. Rao, "Neuromagnetic source imaging with FOCUSS: a recursive weighted minimum norm algorithm," *Electroencephalography and clinical Neurophysiology*, vol. 95, no. 4, pp. 231–251, 1995.

[88] D. Liang, L. Ying, and F. Liang, "Parallel MRI Acceleration Using M-FOCUSS," in *3rd International Conference on Bioinformatics and Biomedical Engineering (ICBBE)*, pp. 1–4, IEEE, 2009.

[89] G. Tzagkarakis, D. Milioris, and P. Tsak, "Multiple-measurement Bayesian compressed sensing using GSM priors for DOA estimation," in *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, pp. 2610–2613, 2010.

[90] J. Chen and X. Huo, "Theoretical results on sparse representations of multiple-measurement vectors," *IEEE Transactions on Signal Processing*, vol. 54, no. 12, pp. 4634–4643, 2006.

[91] D. P. Wipf and B. D. Rao, "An empirical Bayesian strategy for solving the simultaneous sparse approximation problem," *IEEE Transactions on Signal Processing*, vol. 55, no. 7, pp. 3704–3716, 2007.

[92] G. Obozinski, M. J. Wainwright, and M. I. Jordan, "Support union recovery in high-dimensional multivariate regression," *The Annals of Statistics*, pp. 1–47, 2011.

[93] Y. C. Eldar and H. Rauhut, "Average case analysis of multichannel sparse recovery using convex relaxation," *IEEE Transactions on Information Theory*, vol. 56, no. 1, pp. 505–519, 2010.

[94] T. Wimalajeewa and P. K. Varshney, "OMP based joint sparsity pattern recovery under communication constraints," *IEEE Transactions on Signal Processing*, vol. 62, no. 19, pp. 5059–5072, 2014.

[95] J. Hagenauer, "Source-controlled channel decoding," *IEEE Transactions on Communications*, vol. 43, no. 9, pp. 2449–2457, 1995.

[96] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Transactions on Information Theory*, vol. 42, no. 2, pp. 429–445, 1996.

[97] J. Hagenauer, "The EXIT chart-introduction to extrinsic information transfer in iterative processing," in *Proc. of 12th European Signal Processing Conference (EUSIPCO)*, pp. 1541–1548, Citeseer, 2004.

[98] P. Schniter, "Turbo reconstruction of structured sparse signals," in *44th Annual Conference on Information Sciences and Systems (CISS)*, pp. 1–6, 2010.

[99] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

[100] C. M. Bishop, *Pattern recognition and machine learning*, vol. 4. springer New York, 2006.

[101] L. Xu and M. I. Jordan, "On convergence properties of the EM algorithm for Gaussian mixtures," *Neural computation*, vol. 8, no. 1, pp. 129–151, 1996.

[102] D. Donoho and J. Tanner, "Observed universality of phase transitions in high-dimensional geometry, with implications for modern data analysis and signal processing," *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 367, no. 1906, pp. 4273–4293, 2009.

[103] D. L. Donoho, A. Maleki, and A. Montanari, "The noise-sensitivity phase transition in compressed sensing," *Information Theory, IEEE Transactions on*, vol. 57, no. 10, pp. 6920–6941, 2011.

[104] C. Forbes, M. Evans, N. Hastings, and B. Peacock, "Chi-Squared Distribution," *Statistical Distributions, Fourth Edition*, pp. 69–73, 2010.

[105] M. Mayer and N. Goertz, "RFID Tag Acquisition via Compressed Sensing: Flexibility by Random Signature Assignment," in *Proceedings of the 5th International EURASIP Workshop on RFID Technology*, pp. 53–58, 2015.

[106] M. Mayer and N. Goertz, "RFID Tag acquisition via compressed sensing: fixed vs. random signature assignment," *IEEE Transactions on Wireless Communications*, vol. 15, no. 3, pp. 2118–2129, 2016.

[107] M. Mayer and N. Goertz, "Improving Approximate Message Passing Recovery of Sparse Binary Vectors by Post Processing," in *Proceedings of 10th International ITG Conference on Systems, Communications and Coding (SCC 2015)*, 2015.

[108] J. Griffin and G. Durgin, "Gains For RF Tags Using Multiple Antennas," *IEEE Transactions on Antennas and Propagation*, vol. 56, pp. 563–570, Feb 2008.

[109] C. Angerer, R. Langwieser, and M. Rupp, "RFID Reader Receivers for Physical Layer Collision Recovery," *IEEE Transactions on Communications*, vol. 58, no. 12, pp. 3526–3537, 2010.

[110] J. Kaitovic, R. Langwieser, and M. Rupp, "A smart collision recovery receiver for RFIDs," *EURASIP Journal on Embedded Systems*, vol. 2013, no. 1, pp. 1–19, 2013.

[111] D. Kim, M.-A. Ingram, and J. Smith, W.W., "Measurements of small-scale fading and path loss for long range RF tags," *IEEE Transactions on Antennas and Propagation*, vol. 51, pp. 1740–1749, Aug 2003.

[112] L. W. Mayer, M. Wrulich, and S. Caban, "Measurements and channel modeling for short range indoor UHF applications," in *First European Conference on Antennas and Propagation (EuCAP)*, pp. 1–5, IEEE, 2006.

[113] A. Lázaro, D. Girbau, and D. Salinas, "Radio link budgets for UHF RFID on multipath environments," *IEEE Transactions on Antennas and Propagation*, vol. 57, no. 4, pp. 1241–1251, 2009.

[114] P. H. Bardell, W. H. McAnney, and J. Savir, *Built-in Test for VLSI: Pseudorandom Techniques*. New York, NY, USA: Wiley-Interscience, 1987.

[115] Y. Zheng and M. Li, "ZOE: Fast cardinality estimation for large-scale RFID systems," in *Proceedings of IEEE INFOCOM*, pp. 908–916, 2013.

[116] G. Hannak, M. Mayer, G. Matz, and N. Goertz, "An Approach to Complex Bayesian-optimal Approximate Message Passing," *arXiv preprint arXiv:1511.08238*, 2015.

[117] S. Kotz, T. Kozubowski, and K. Podgorski, *The Laplace distribution and generalizations: a revisit with applications to communications, economics, engineering, and finance.* Springer Science & Business Media, 2001.

[118] C. Guo and M. Davies, "Near Optimal Compressed Sensing Without Priors: Parametric SURE Approximate Message Passing," *IEEE Transactions on Signal Processing*, vol. 63, no. 8, pp. 2130–2141, 2015.

[119] J. Kaitovic, M. Simko, R. Langwieser, and M. Rupp, "Channel Estimation in Tag Collision Scenarios," in *IEEE International Conference on RFID (IEEE RFID)*, pp. 74–80, 2012.

[120] M. Mayer, B. Ramos Elbal, W. Gartner, R. Langwieser, and J. Kaitovic, "A Flexible Setup to Determine RFID Tag Requirements for Multiple-Response Scenarios," in *Proceedings of 10th IEEE International Conference on RFID (IEEE RFID)*, 2016.

[121] "SMU and FSQ product images by courtesy of Rohde & Schwarz and ML505 product image." Respectively retrieved from https://cdn.rohde-schwarz.com/pws/dl_downloads/dl_common_library/dl_manuals/gb_1/s/smu200a_1/SMU200A_OperatingManual_en_19.pdf, https://cdn.rohde-schwarz.com/pws/dl_downloads/dl_common_library/dl_manuals/gb_1/f/fsq_1/FSQ_OpMa_en_02.pdf, http://www.fpgadeveloper.com/en/ml50567-and-xupv5-boards/ml505.JPG, Accessed June 2016.

[122] Rohde & Schwarz GmbH & Co. KG, *R&S SMU200A Vector Signal Generator Operating Manual.* https://cdn.rohde-schwarz.com/pws/dl_downloads/dl_common_library/dl_manuals/gb_1/s/smu200a_1/SMU200A_OperatingManual_en_19.pdf, 2012.

[123] Rohde & Schwarz GmbH & Co. KG., *R&S FSQ Signal Analyzer Operating Manual.* https://cdn.rohde-schwarz.com/pws/dl_downloads/dl_common_library/dl_manuals/gb_1/f/fsq_1/FSQ_OpMa_en_02.pdf, 2014.

[124] A. P. Sample, D. J. Yeager, P. S. Powledge, A. V. Mamishev, and J. R. Smith, "Design of an RFID-based battery-free programmable sensing platform," *IEEE Transactions on Instrumentation and Measurement*, vol. 57, no. 11, pp. 2608–2615, 2008.

[125] "WISP 4.1 Wiki." https://wisp.wikispaces.com/.

[126] "WISP 5 Wiki." http://wisp5.wikispaces.com/WISP+Home/.

[127] B. Ramos Elbal, "Measurement Based Evaluation of the Wireless Identification and Sensing Platform." http://publik.tuwien.ac.at/files/PubDat_240043.pdf, Master Thesis, Faculty of Electrical Engineering and Information Technology, TU Wien, 2015.