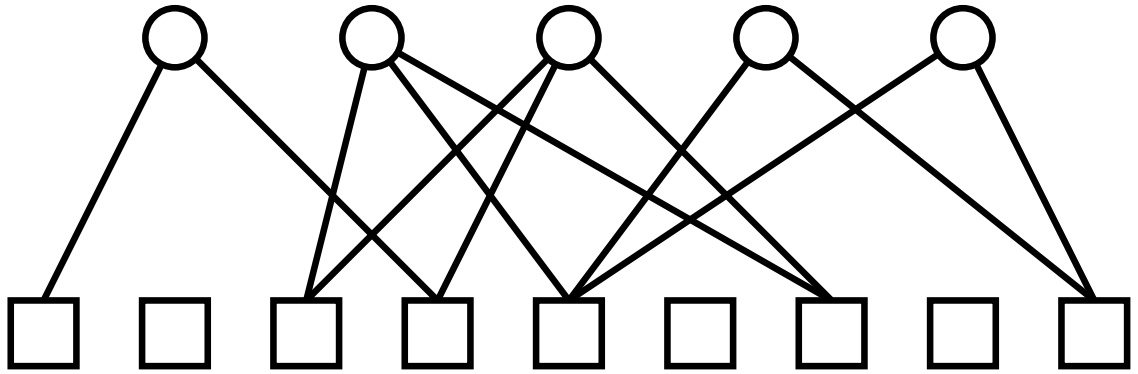


On Frame Asynchronous Coded Slotted ALOHA

Analysis of a Reliable and Low Latency Uncoordinated Multiple Access Technique

ERIK SANDGREN

Department of Signals and Systems
Chalmers University of Technology
Gothenburg, Sweden 2016

Master's thesis EX036/2016

On Frame Asynchronous Coded Slotted ALOHA

Analysis of a Reliable and Low Latency
Uncoordinated Multiple Access Technique

ERIK SANDGREN

CHALMERS

Department of Signals and Systems
Division of Communication Systems
Chalmers University of Technology
Gothenburg, Sweden 2016

On Frame Asynchronous Coded Slotted ALOHA
Analysis of a Reliable and Low Latency Uncoordinated Multiple Access Technique
ERIK SANDGREN

Supervisor and Examiner: Fredrik Brännström, Department of Signals and Systems
Co-supervisor: Alexandre Graell i Amat

Master's Thesis EX036/2016
Department of Signals and Systems
Division of Communication Systems
Chalmers University of Technology
412 96 Gothenburg
Telephone +46 31 772 1000

Cover: A bipartite graph representation of a coded slotted ALOHA system. Check nodes (squares) represent slots and variable nodes (circles) represent users, as detailed in Section 1.2.

Typeset in LaTeX
Gothenburg, Sweden 2016

Abstract

Recently, a new class of uncoordinated multiple access techniques named coded slotted ALOHA (CSA) has emerged and attracted much interest. CSA is based on the well known slotted ALOHA system. The key innovation of CSA is that users replicate each packet over several slots and that decoding is performed iteratively over a sequence of slots using successive interference cancellation. One of the important features of CSA is that it can be represented by means of a bipartite graph. As a result, the performance and analysis of CSA shows a lot of resemblance to that of codes on graphs.

In this work, we consider a *frame asynchronous CSA (FA-CSA)* system, where users joins the system on a slot-by-slot basis according to a Poisson process. A user begins its *local frame* in the slot following the slot it joins, and replicates its message in a number of randomly selected slots of the local frame. This is in contrast with classical frame synchronous CSA (FS-CSA), where a user that joins the system awaits the start of the next *global frame*, in which it replicates its message over a number of randomly selected slots.

The main purpose of this work is to analyze and investigate the performance of FA-CSA. The performance is evaluated in terms of packet loss rate (PLR) and delay. In particular, we derive (approximate) density evolution equations that characterize the asymptotic behavior of iteratively decoded FA-CSA. We show that, if the receiver can monitor the system before anyone starts transmitting, a boundary-effect similar to that of spatially coupled codes occurs, which greatly improves the decoding threshold. Furthermore, we derive tight approximations of the error floor (EF), in the finite frame length regime, based on the probability of occurrence of the most frequent stopping sets. We show that, in general, FA-CSA provides better PLR in both the EF and waterfall regions as compared to frame synchronous CSA. Moreover, we show by simulation that FA-CSA generally outperforms FS-CSA in terms of delay.

Keywords: Coded slotted ALOHA, density evolution, error floor, multiple access, spatial coupling, stopping sets.

Acknowledgments

First of all, I would like to direct many thanks to my two supervisors Fredrik Brännström and Alexandre Graell i Amat for providing me with the opportunity of doing this master's thesis, it has been a great experience working with both of them. Their guidance, helpfulness, and positive attitudes have kept me motivated during this thesis work.

I would also like to thank all my friends in the master thesis room and the MPCOM program for the company and many discussions throughout the year. Last but not least, I want to express gratitude towards my family and my girlfriend Elin, for supporting me through my ups and downs.

Erik Sandgren, Göteborg, June 2016

Acronyms

ARQ	automatic repeat request
BEC	binary erasure channel
CSMA	carrier sense mutiple access
CN	check node
CSA	coded slotted ALOHA
DE	density evolution
EF	error floor
FDMA	frequency division multiple access
FA-CSA	frame asynchronous coded slotted ALOHA
FA-CSA-F	FA-CSA with first slot fixed
FA-CSA-FB	FA-CSA-F with boundary effect
FA-CSA-FNB	FA-CSA-F with no boundary effect
FA-CSA-U	FA-CSA with uniform slot selection
FA-CSA-UB	FA-CSA-U with boundary effect
FA-CSA-UNB	FA-CSA-U with no boundary effect
FS-CSA	frame synchronous coded slotted ALOHA
IC	interference cancellation
LDPC	low-density-parity-check
PLR	packet loss rate
pmf	probability mass function
PA	pure ALOHA
RV	random variable
SA	slotted ALOHA
SC-CSA	spatially coupled CSA
SIC	successive interference cancellation
TDMA	time division multiple access
VN	variable node
WF	waterfall

Symbols

τ	duration of packet [s]
n	frame length [slots]
M	number of active users (RV)
m	number of active users (realization)
g	system load [users/slot]
$\bar{p}$	packet loss rate
n_{RX}	memory size [slots]
$\Lambda(x)$	node-perspective variable node degree distribution
$\lambda(x)$	edge-perspective variable node degree distribution
$\mathrm{P}(x)$	node-perspective check node degree distribution
$\rho(x)$	edge-perspective check node degree distribution
p	probability of passing erasure message over VN edge
q	probability of passing erasure message over CN edge
$p_{i\to j}$	probability of an erasure message from a class-i VN to a class-j CN
$q_{i\to j}$	probability of an erasure message from a class-i CN to a class-j VN
$g^{\star}$	iterative decoding threshold [users/slot]
u	arbitrary user
$\mathcal{S}$	graph representation of a stopping set
$\boldsymbol{v}(\mathcal{S})$	graph profile of $\mathcal{S}$
$v_l(\mathcal{S})$	number of degree l variable nodes in $\mathcal{S}$
$\nu(\mathcal{S})$	number of variable nodes in $\mathcal{S}$
$\mu(\mathcal{S})$	number of check nodes in $\mathcal{S}$
$c(\mathcal{S})$	number of graph isomorphisms of $\mathcal{S}$
$\delta_{\max}$	maximum delay constraint [slots]

Contents

Chapter 1

Introduction

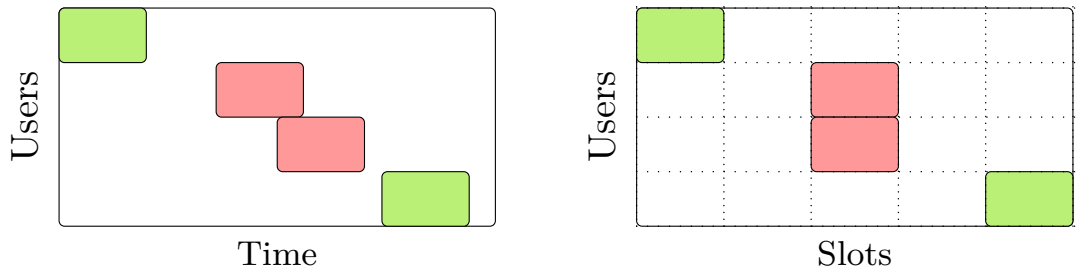
1.1 Background and Motivation

The need for multiple access techniques arises whenever more than one user needs to share common resources in a communication system. Typically, these common resources are time and frequency. The purpose of a multiple access technique is to describe how resources should be allocated amongst users in a system [1]. In many communications systems it is possible to have a coordinated allocation of resources. Coordination is carried out by a central unit, e.g., a base station in a cellular system. Conventional examples of coordinated multiple access techniques are frequency division multiple access (FDMA) and time division multiple access (TDMA) [1]. These techniques are common in, e.g., cellular systems such as the Global System for Mobile (GSM), where TDMA is used, and Digital European Cordless Telephone (DECT), where FDMA is used [2].

In some systems, however, coordination is not possible, for which the reasons can be many. One reason is that the set of users in the system may change too quickly for a coordinator to keep track of. Another reason is that no natural coordinator exists in the system. A third reason is that the coordinated resource allocation itself might be time consuming, and thus, impose unacceptably large latencies on the communication.

Uncoordinated multiple access techniques can be used when coordination is not possible or desired. These techniques let users select independently which resources to use for transmission, in an arbitrary or semi-arbitrary manner. Some of the most frequently used uncoordinated multiple access techniques are based on the ALOHA systems [3, 4], proposed in the 1970s for wireless networks. The ALOHA systems provide packet based communication. In the first adaption of ALOHA, named pure ALOHA (PA), a user simply sends a packet over the shared medium as soon as the packet is generated. A successor of PA is the slotted ALOHA (SA) system, in which time is divided into slots of duration τ and users must send packets within the boundaries of slots.

Example 1. *Toy examples of both of PA and SA are depicted in Fig. 1.1. Green slots correspond to successful transmissions and red slots show packets in collision. In both PA and SA the two middle users are interfering each other and therefore their packets are lost. In PA packets are lost even if there is just a partial collision.*



(a) Pure ALOHA **(b)** Slotted ALOHA

Figure 1.1: Toy examples of pure ALOHA and slotted ALOHA, respectively. Green slots represent successfully received packets while red slots represent packets in collision.

Analysis of the ALOHA systems is commonly performed with the assumptions that users join the system (generates packets) according to a Poisson process with mean g. The parameter g is the average number of users that join per time unit (t.u.), where a t.u. is the duration of a packet or a slot. Furthermore, it is assumed that a packet is successfully received if it is not interfered by any other user packet on the shared medium.

The throughput of a multiple access technique is defined as the fraction of time (or slots) that contains successful packet transmissions. It is straightforward to show that the throughput of PA is $T_{\text{PA}} = ge^{-2g}$, whereas for SA it is $T_{\text{SA}} = ge^{-g}$. The throughput of PA, SA, and TDMA is plotted as a function of the system load in Fig. 1.2a, where the system load is the average number of users per t.u.. The maximum throughput is given by $T^{\star}_{\text{PA}} = 1/(2e)$ and $T^{\star}_{\text{SA}} = 1/e$, for PA and SA respectively. Furthermore, the packet loss rate (PLR), denoted by $\bar{p}$, which is the fraction of unresolved packets such that $\bar{p} = 1 - T/g$, is plotted for the same systems as a function of the system load in Fig. 1.2b. Clearly, the efficiency and reliability of PA and SA is relatively low, i.e., many packets are sent but never resolved. However, PA and SA provide very simple uncoordinated multiple access and we will later see that SA can be extended to provide reliable and low latency uncoordinated multiple access.

In systems where PA or SA is employed and reliable communication is desired, the most common solution is to use an automatic repeat request (ARQ) scheme alongside PA or SA [5]. The function of an ARQ scheme is to notify users upon successful transmission of a packet. The intended receiver does so by replying with an acknowledgment whenever it receives a packet successfully. Acknowledgments are typically sent on a separate, collision free, communication channel. If a user does not receive an acknowledgment from the receiver within some time-out duration, it will retransmit the packet. This way, the packet may be transmitted many times before it is successfully received. ARQ schemes allow simple techniques like PA and SA provide reliable multiple access. However, it introduces some problems, such as large delays and possible congestion of the shared medium. Furthermore, ARQ requires a separate channel for acknowledgments. We remark that there are many different types of ARQ schemes and that the one explained here just gives the general idea of its operation.

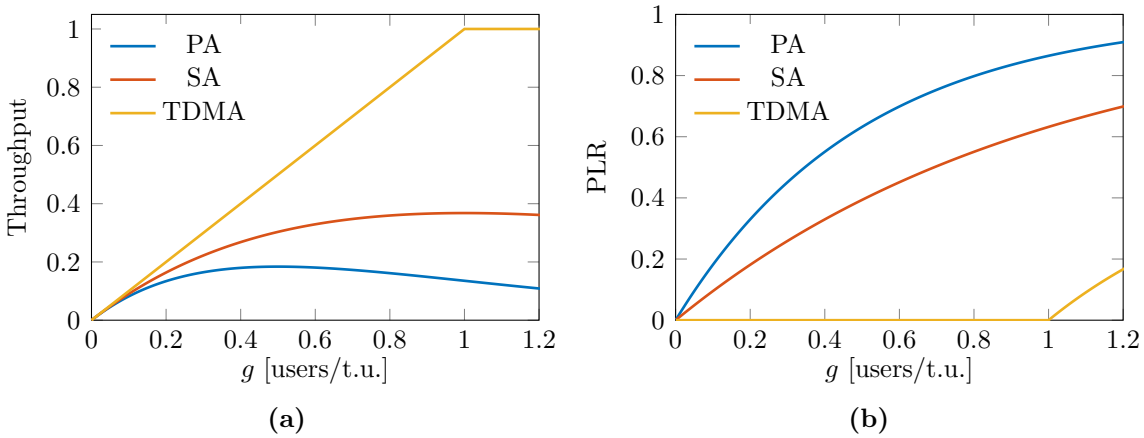


Figure 1.2: Performance comparison of PA, SA, and TDMA.

Another common class of uncoordinated multiple access are carrier sense mutiple access (CSMA) based techniques, which also provide packet based communication [2]. In CSMA, when a user joins the system, it first senses the shared medium to check if it is currently occupied by some other user. If the medium is not busy, the user transmits its packet immediately. However, if the medium is busy, the user will *back off* for some random duration, and then sense the medium again. This process of backing off and sensing may be repeated several times before transmission. Even if CSMA avoids more collisions than PA and SA by sensing the medium, it is most usually employed with an ARQ scheme alongside. One well known standard that uses CSMA with collision avoidance is the IEEE standard 802.11, commonly known as WiFi [6].

Recently, a considerable interest for finding novel solutions to provide reliable, low latency communication in dynamic networks has emerged. The reason for this is that the next generation communication systems, 5G, promise to handle these scenarios [7]. Finding solutions that can provide this type of communication will pave the way for many future technologies such as traffic safety, traffic efficiency, and efficient industrial communication, to mention a few. One essential component of these new systems is, of course, the multiple access technique. Conventional coordinated and uncoordinated multiple access techniques are not well suited for these applications, because they cannot meet the requirements either on reliability, latency, or dynamism. A promising novel uncoordinated multiple access technique for these scenarios is coded slotted ALOHA (CSA).

1.2 Coded Slotted ALOHA

In this section, we introduce the concept of CSA and present some results that are useful in order to understand the work of this thesis. CSA builds on the SA technique and borrows ideas from the field of error correcting codes to provide highly reliable uncoordinated multiple access without the use of an ARQ scheme.

Similarly to SA, time is divided into slots in CSA. Each slot has duration τ and users in the system join the system according to some user model. A user that joins

the system generates a packet of duration τ (including guard intervals), such that a packet can be place within one slot. Two key ingredients of CSA is to let users replicate each packet a number of times within a frame of slots, and to perform iterative decoding of frames using successive interference cancellation (SIC). This idea was first presented in [8], where each user replicated its packet a fixed factor l. Later this idea was generalized so that users instead could pick individual repetition factors from a predefined degree distribution [9]. The degree distribution is given in polynomial form as

$$\Lambda(x) \triangleq \sum_{l} \Lambda_l x^l, \tag{1.1}$$

where Λ_l is the probability that a user selects a repetition factor l and $\Lambda(1) = 1$. A user with repetition factor l is called a degree-l user and transmits l replicas of its packet in l randomly selected slots from a set of n slots. In the context ofCSA n is commonly referred to as the *frame length*. It is also usually assumed that the users are both slot and frame synchronous [8, 9]. Furthermore, in each frame, m users transmit all their replicas in randomly selected slots of that frame. The system load g is defined as

$$g \triangleq \frac{m}{n}. \tag{1.2}$$

More generally, the system load is the average number of users per slot. A slot that contains r replicas is a degree-r slot. Similarly to the degree distribution for users in (1.1), we define a degree distribution for slots as

$$\mathrm{P}(x) \triangleq \sum_{r} \mathrm{P}_r \, x^r, \tag{1.3}$$

where P_r denotes the probability that slot has degree r. $\mathrm{P}(x)$ is a consequence of the selected degree distribution $\Lambda(x)$ for users and the system load g, whereas $\Lambda(x)$ is directly controlled by the system designer.

In the analysis of CSA it is typically assumed that perfect interference cancellation (IC) can be performed to remove the interference of a user's replicas once its packet has been decoded. This assumption makes the analysis of the system more feasible. In [8,9] actual (low complexity) IC was implemented with little performance degradation as compared to perfect IC.

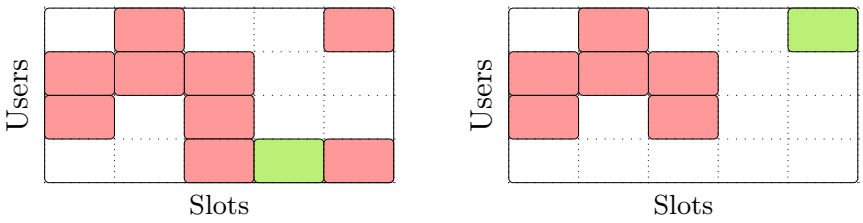
Iterative decoding of a CSA frame using SIC is performed as follows:

1. Find all degree-1 slots in the frame and the corresponding users that transmit within these slots.

2. Decode the packets of the users found in step 1 and cancel the interference of all their replicas using IC.

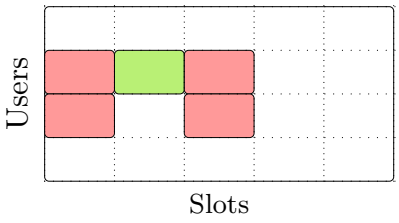
3. If no degree-1 slots were found in step 1 or if a maximum allowed number of iterations has been reached, terminate the decoding. Otherwise, go back to step 1.

To facilitate step 2, all packets in CSA contain pointers to its other replicas. SIC is very powerful because the decoding of one user's packets allows for the interference

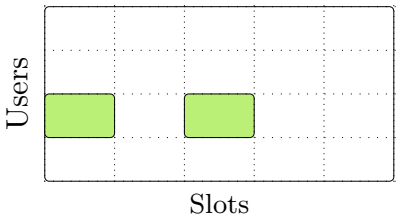


(a) Decoding iteration 1

(b) Decoding iteration 2



(c) Decoding iteration 3



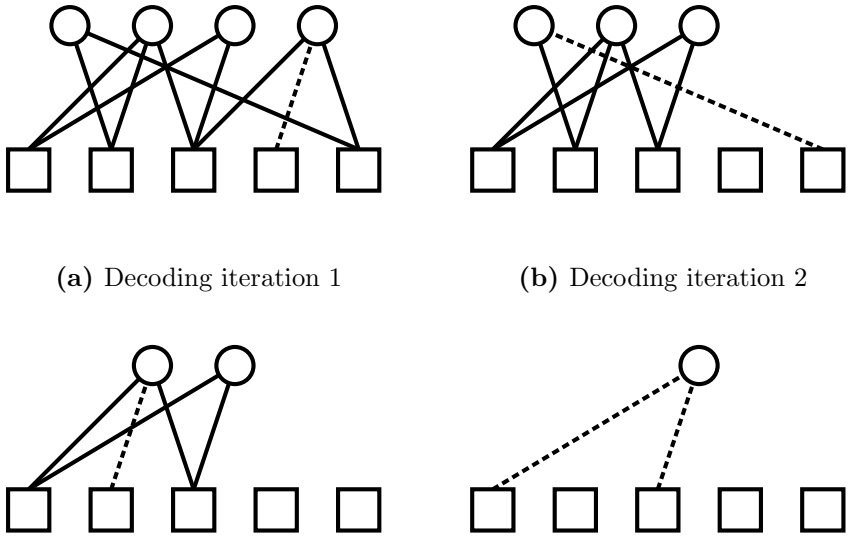
(d) Decoding iteration 4

Figure 1.3: Example of iterative decoding of one frame in CSA. The decoding terminates after 4 iterations, since no unresolved users remain. At each step, slots with degree 1 are colored green and slots with collisions are colored red.

of all its replicas to be removed. This can lead to the exposure of new degree-1 slots and thus, to the decoding of other user's packets.

Example 2. *Iterative decoding of a CSA frame with* $\Lambda(x) = 0.5x^2 + 0.5x^3$, $n = 5$, *and* $m = 4$ *is depicted in Fig. 1.3. Each sub-figure shows how the iterative decoding with SIC proceeds step by step. In iteration 1, user 4 is resolved since slot 4 is of degree 1. The replicas of user 4 in slot 3 and 5 can then also be lifted by IC. In iteration 2, slot 5 is of degree 1 and user 1 can therefore be resolved, and the replica in slot 2 can also be lifted. In iteration 3, slot 2 is of degree 1, user 2 can be resolved, and the replicas lifted. In iteration 4, only user 3 remains and can of course be resolved. We remark that in a CSA system, a new frame would of course take place right after the depicted one, in which 4 new users send their replicas.*

One of the most important results on CSA is the connection to codes on graphs, established in [9]. The operation of a CSA system can be represented by a bipartite graph $\mathcal{G} = \{\mathcal{V}, \mathcal{C}, \mathcal{E}\}$, where $\mathcal{V}$ is the set of variable nodes (VNs), $\mathcal{C}$ is the set of check nodes (CNs) and $\mathcal{E}$ is the set of edges connecting the VNs and CNs. There is an edge $e_{i \to j} \in \mathcal{E}$ from VN i to CN j if user i transmits a packet in slot j. VNs represent users and CNs represent slots, and the terms will be used interchangeably throughout the thesis. Decoding of the CSA system can be viewed as a message passing algorithm over the edges of the underlying graph. This CSA decoder is in fact equivalent to the *peeling decoder* for low-density-parity-check (LDPC) codes over the binary erasure channel (BEC) [10]. We remark that the decoder for CSA (and the peeling decoder) does not always succeed as in Example 2. A collision pattern, or equivalently a graph structure, that makes the decoder fail is called a *stopping set* [10]. More details about stopping sets are given in Section 3.3.



(a) Decoding iteration 1 **(b)** Decoding iteration 2

(c) Decoding iteration 3 **(d)** Decoding iteration 4

Figure 1.4: Bipartite-graph representation of the example in Fig. 1.3. Circles represent VNs and squares represent CNs. Dashed edges show the connection of degree-1 CNs to VNs at each iteration.

Example 3. *In Fig. 1.4, a graph representation of the decoding process previously depicted in Fig. 1.3 is shown. Circles represent VNs (users) and squares represent CNs (slots). Decoding using the graph is performed by iteratively peeling of all edges of VNs with at least one edge connected to a degree-1 CN. Edges connected to degree-1 CNs at each iteration are dashed in Fig. 1.4.*

A common analytical tool for iteratively decoded error correcting codes is density evolution (DE). DE is used to predict the average asymptotic behavior of LDPC code ensembles with certain degree distributions. It does so by mimicking the iterative decoder in a probabilistic manner. DE tracks the evolution, over decoding iterations, of the probability densities of messages passed in the iterative message passing decoder [10, 11]. This is especially simple for codes over the BEC, where a message passed in the decoder is binary, i.e., either an erasure message or a non-erasure message. Therefore, it suffices to track a single parameter per node in the DE, i.e., the probability of passing an erasure message. Because of the similarities between the decoding of CSA and the peeling decoder for LDPC codes over the BEC, it makes sense that it is also possible to derive DE equations that characterizes the asymptotic performance of CSA [9]. The derivation of the DE equations assumes that the underlying graph is cycle-free, which is true for CSA when the frame length tends to infinity.

We now derive the DE equations for CSA, as first derived in [9]. For the derivation, let p_k denote the probability that an erasure message is passed over the edges from a VN at iteration k, and denote by q_k the probability that an erasure message is passed over the edges from a CN at iteration k. A degree-l VN passes an erasure message over an edge if all the incoming messages on its other edges in the previous

iteration were erasures, thus,

$$p_k = q_{k-1}^{l-1}. \tag{1.4}$$

A degree-r CN does *not* pass an erasure message over an edge if at least one of the incoming messages on its other edges in the previous iteration was a non-erasure message, thus,

$$q_k = 1 - (1 - p_k)^{r-1}. \tag{1.5}$$

We remark that (1.4) and (1.5) follow from the properties of the CSA decoder.

Equations (1.4) and (1.5) were derived assuming a certain degree (l or r) for VNs and CNs, respectively. In general, VNs and CNs in the system will have different degrees. Therefore, we generalize (1.4) and (1.5) by computing the expectation with respect to the probability that an edge is connected to a degree-l VN and a degree-r CN, respectively. To facilitate this computation we define the *edge-perspective* distributions for VNs and CNs as

$$\lambda(x) \triangleq \sum_l \lambda_l x^{l-1} \quad \text{and} \quad \rho(x) \triangleq \sum_r \rho_r x^{r-1}, \tag{1.6}$$

where λ_l is the probability that an edge is connected to a degree-l VN and ρ_r is the probability that an edge is connected to a degree-r CN. Furthermore,

$$\lambda_l = \frac{\Lambda_l l}{\Lambda'(1)} \quad \text{and} \quad \rho_r = \frac{\mathrm{P}_r\, r}{\mathrm{P}'(1)}, \tag{1.7}$$

where f' denotes the derivative of the function f.

Now, (1.4) is generalized to

$$p_k = \sum_l \lambda_l q_{k-1}^{l-1} = \lambda(q_{k-1}), \tag{1.8}$$

and (1.5) to

$$q_k = 1 - \sum_r \rho_r (1 - p_k)^{r-1} = 1 - \rho(1 - p_k), \tag{1.9}$$

where $\rho(x) = e^{-g\Lambda'(1)(1-x)}$ as $m \to \infty$ and $n \to \infty$ and the ratio $g = m/n$ is kept constant [9]. This gives

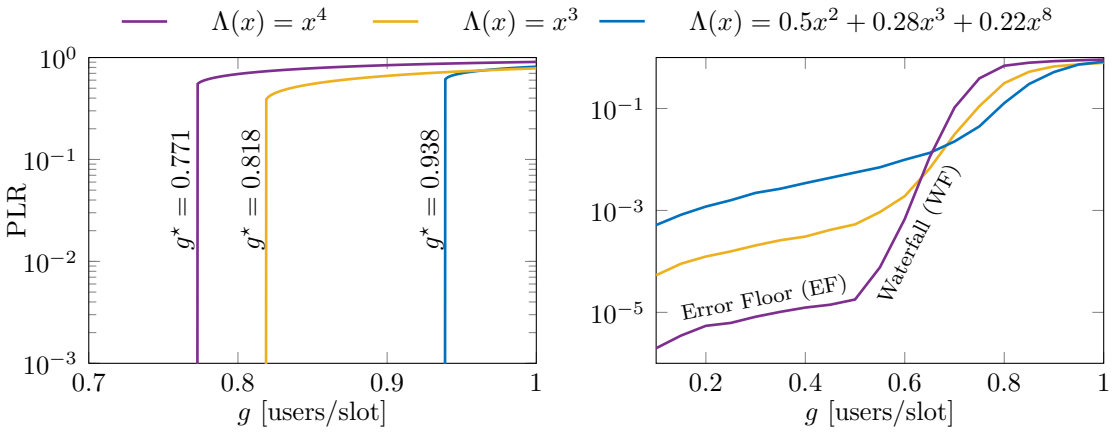
$$q_k \overset{(a)}{=} 1 - e^{-g\Lambda'(1)\lambda(q_{k-1})} \overset{(b)}{=} 1 - e^{-g\Lambda'(q_{k-1})}, \tag{1.10}$$

where in (a) it was used that $p_k = \lambda(q_{k-1})$ and in (b) that $\lambda(x) = \Lambda'(x)/\Lambda'(1)$.

DE is now performed by iteratively updating (1.10), with q_0 initialized to 1, until a fixed point is reached. The PLR at the kth iteration, $\bar{p}_k$, can be computed as

$$\bar{p}_k = \Lambda(q_k). \tag{1.11}$$

In the asymptotic regime, CSA typically exhibits a threshold behavior, i.e., when the system operates below a threshold, denoted by $g^\star$, all users can be resolved. However, if the system operates above the threshold, the PLR will be bounded away from zero. The threshold $g^\star$ can be predicted by the DE equation (1.10). It



(a) PLR obtained by DE with $n \to \infty$ **(b)** PLR obtained by simulation with $n = 100$

Figure 1.5: Typical PLR performance of CSA in the asymptotic and finite frame length regimes.

can be found by searching for the maximum load g for which q converges to 0 by iteratively updating (1.10).

Asymptotic performance curves for CSA obtained by DE are depicted in Fig. 1.5a. The figure clearly depicts the threshold in the asymptotic PLR performance of CSA and additionally gives the values of the thresholds $g^\star$ for the three distributions. The VN-degree distribution $\Lambda(x) = 0.50x^2 + 0.28x^3 + 0.22x^8$ with $g^\star = 0.938$ was found in [9] by a threshold optimization over VN degree distributions with maximum degree 8. In [12] it was shown that CSA (asymptotically) can provide packet loss free transmission at a load of one user per slot, by letting user repetition degrees follow an *ideal soliton distribution*. The ideal soliton distribution, however, is not very practical since it has maximum degree equal to the frame length n. Fig. 1.5b depicts instead simulated finite frame length performance with $n = 100$. We observe that the finite frame length performance typically shows two regions of interest, the waterfall (WF) region and the error floor (EF) region, both indicated in Fig. 1.5b for $\Lambda(x) = x^4$. In the WF region the PLR curve steeply changes from low to high. This WF is related to the asymptotic threshold, whereas in the EF region the PLR is low and governed by some dominant stopping sets. In [13] and [14] a framework for predicting the finite frame length EF was established. We remark that CSA can provide much better reliability (PLR) for reasonable system loads as compared with SA and PA, compare Fig. 1.5b and Fig. 1.2b.

Before we move on to the contribution of this thesis, we first give a short summary of some of the other relevant works on the topic of CSA.

In [15] a generalized CSA scheme was proposed where packets are split into smaller segments and a randomly selected local component code is used to encode the packet. Furthermore, a spatially coupled CSA (SC-CSA) was proposed in [16]. A large improvement of the iterative decoding threshold was observed for SC-CSA as compared to standard CSA. This improvement is similar and related to the thresholds improvement resulting from spatial coupling in LDPC codes [17].

To improve the delay performance, a frame asynchronous coded slotted ALOHA (FA-CSA) system was proposed in [18]. In FA-CSA a user that joins the system begins its *local frame* in the following slot. Furthermore, the user will select slots for

its replicas randomly from the slots of its local frame. This is in contrast with classical frame synchronous coded slotted ALOHA (FS-CSA) where users that join the system await the start of the next *global frame*, in which they become active and will select slots for their replicas randomly from the slots of that global frame. Simulation results in [18] show that, in addition to improve the average delay, FA-CSA also outperforms FS-CSA in terms of throughput. A fully asynchronous (both frame- and slot- asynchronous) CSA system was later proposed in [19]. The performance of this fully asynchronous CSA in terms of PLR, also considering physical layer aspects, was evaluated in accordance with the framework introduced in [20] and by simulations.

1.3 Aim and Outline

In this thesis, we consider FA-CSA systems, where users join the system on a slot-by-slot basis according to a Poisson process. Our ambition is to characterize the performance of FA-CSA in terms of reliability and delay.

We derive approximate DE equations that predict the PLR performance of FA-CSA as the frame length tends to infinity. Several models are considered for the initialization of the system. We show that, if there is a *boundary effect*, i.e., if the CNs at the boundary of the system have lower expected degree than other CNs, the iterative decoding threshold is greatly improved as compared to the case with no boundary effect. This threshold improvement is similar to the effect of spatial coupling in LDPC codes, an idea that has previously been applied to CSA with SC-CSA [16]. Furthermore, we derive tight analytical approximations of the EF in the finite frame length regime. This analysis is based on the framework established in [13] and [14], where the probability of occurrence of minimal stopping sets is computed and used to approximate the EF in CSA systems. In order to evaluate the performance of FA-CSA, we compare it with FS-CSA and SC-CSA in terms of threshold, EF, and delay. The evaluation is based on the derived DE equations, EF-predictions, and simulations.

The work in this thesis has resulted in an accepted conference paper [21], to be presented at the 9th International Symposium on Turbo Codes & Iterative Information Processing, Brest, France, September 2016, and a submitted journal paper [22]. Note that the content of [22] completely covers what is presented in this thesis.

The remainder of this thesis is organized as follows. In Chapter 2 we give a detailed system model for FA-CSA. In Chapter 3 we analytically derive the CN degree distributions, approximate DE equations, and predictions of the EF in the finite frame length regime, for FA-CSA. In Chapter 4 we give numerical results on and compare the performance of FA-CSA to FS-CSA and SC-CSA. Finally, Chapter 5 concludes the thesis.

Chapter 2

System Model

A model for the systems considered in this thesis is presented in this chapter. The systems do inherit many properties from the CSA system described in Section 1.2, however, all properties of the considered systems are stated for completeness.

We consider a CSA system where time is divided into slots, each of duration τ, and where users are slot-synchronized. A user that joins the system generates a message and selects a repetition factor l randomly according to a predefined degree distribution [9]. The message is then mapped into a physical layer packet of duration τ (including guard intervals), such that one packet can be sent within one slot. The user transmits l copies (called replicas) of the packet in randomly selected slots. A user that repeats its packet l times is called a degree-l user and similarly a slot containing r replicas from different users is called a degree-r slot. It is assumed that a common receiver captures all transmission in the system, and that a packet can always be decoded if at least one of its replicas is in a degree-1 slot. Furthermore, to facilitate decoding, each replica of a packet contains pointers to all other replicas of that packet.

We assume that users join the system on a slot basis according to a Poisson process and let K denote the number of users that join in a slot. Then K is a Poisson distributed random variable (RV) with mean g, $K \sim \mathsf{Po}(g)$, where g is the average system load in users per slot. The probability that k users join in a given slot is thus,

$$\Pr(K = k) = \frac{e^{-g} g^k}{k!}. \tag{2.1}$$

This is a common user model for multiple access techniques, used, e.g., in the original ALOHA systems [3, 4].

We define the two most important performance measures for the considered systems.

Definition 1. *The PLR, denoted by $\bar{p}$, is the average probability that an arbitrary user's packet is never resolved.*

Definition 2. *The delay of a resolved user's packet is the number of slots between the slot a user joins the system and the slot following the decoding of its packet.*

The PLR measures a systems reliability whereas the delay measures the latency performance.

2.1 Frame Synchronous Coded Slotted ALOHA

In FS-CSA, communication takes place during *global frames* consisting of n slots each. A degree-l user that joins the system waits until the next frame, and transmits its l replicas in randomly chosen slots of that frame. We then say that the user is *active* the whole duration of the frame. We denote by $M \sim \mathsf{Po}(ng)$ the RV representing the number of active users per frame. Note that the active users in a frame are all the users that joined the system during the previous frame.

Decoding of FS-CSA is performed on a slot-by-slot basis. Assume the decoding of slot i. First, the interference caused by packets for which replicas in previous slots have already been decoded is canceled from the slot. The receiver then checks if slot i is a degree-1 slot and, if not, the decoding of slot i is stopped. Otherwise, the packet in slot i is decoded and the interference from all its replicas canceled from the corresponding past slots. The receiver then proceeds to iteratively find any degree-1 slots in its memory, decode the packets in these slots, and cancel the interference of all replicas of the decoded packets. This process continues until no new degree-1 slots are found or a maximum number of iterations is reached.

FS-CSA is similar to the CSA described in Section 1.2, with the differences being the Poisson user model and slot-by-slot decoding. This model for FS-CSA is introduced here because it is more similar and comparable to the FA-CSA system that we will describe next.

2.2 Frame Asynchronous Coded Slotted ALOHA

In FA-CSA, when a degree-l user joins the system it transmits a first replica in the following slot. The remaining $l-1$ replicas are distributed uniformly within the $n-1$ subsequent slots. Similarly to FS-CSA, n is the frame length of FA-CSA. However, contrary to FS-CSA, slots are not arranged in global frames. We say that a user is active during the n slots following the slot where it joins the system and, accordingly, we call the n slots where a user is active its local frame. Because a user always transmits in the first slot of its local frame, we call this system FA-CSA with first slot fixed (FA-CSA-F).

Decoding of FA-CSA is performed in a similar way as for FS-CSA, with the only difference that the receiver needs to consider not only the slots of a current frame, but all slots of the entire history of the system. In practice, the receiver cannot consider infinitely many slots and has a finite memory. We denote by n_{RX} the size of the receiver memory in number of slots. It was shown in [18] that increasing n_{RX} beyond $5n$ does not improve performance in general. A finite n_{RX} creates the notion of a *sliding-window* decoder.

We let

$$M_i \sim \mathsf{Po}(\mu_i) \tag{2.2}$$

denote the number of active users in the ith slot of FA-CSA, which is Poisson distributed with mean μ_i. The active users in slot i are all users that joined in the slots $[i-n, i-1]$. We consider two different models for the initialization of the system, i.e., for $1 \leq i \leq n$. The first model assumes that there are no active users

at $i = 0$. In this case

$$\mu_i = \begin{cases} ig & \text{for } 1 \leq i < n \\ ng & \text{for } i \geq n \end{cases}, \tag{2.3}$$

and we say that a *boundary effect* is present for this model. Effectively, this means that the $n - 1$ first slots of the system have lower average degree than the rest.

The second model assumes that there are already $M \sim \mathsf{Po}(ng)$ active users at $i = 0$. Thus,

$$\mu_i = ng \quad \text{for all } i \geq 1. \tag{2.4}$$

For this model, all considered slots have the same average degree.

The system with boundary effect corresponds to a system where the receiver is present at the very start of communication, or potentially, a system with periods of low load. A practical example is road side infrastructures in a vehicular network as the intended receivers. On the other hand, the model with no boundary effect is useful for systems where the receiver joins an already ongoing communication, e.g., a vehicle in a local vehicular network as the intended receiver. A vehicle will join and leave local networks with ongoing communication as it is moving.

In addition to the initialization models described above, we introduce another model for the selection of slots for transmission aside from FA-CSA-F. We consider a system where a degree-l user selects all l slots for transmission randomly from the local frame and call this FA-CSA with uniform slot selection (FA-CSA-U). FA-CSA-U is useful because it is more similar to FS-CSA and provides a simplified analysis in some cases, however we remark that FA-CSA-F is more practical and performs better in general, therefore, the emphasis of this work is on FA-CSA-F.

In all, we consider four models for FA-CSA, i.e., FA-CSA-F with boundary effect (FA-CSA-FB), FA-CSA-F with no boundary effect (FA-CSA-FNB), FA-CSA-U with boundary effect (FA-CSA-UB) and FA-CSA-U with no boundary effect (FA-CSA-UNB). The terminology boundary effect will become clearer in Section 3.1 and 3.2.

Example 4. *An example of FS-CSA and FA-CSA-FB is depicted in Fig. 2.1, with* $n = 6$, $g = 0.5$ *and* $\Lambda(x) = 0.5x^2 + 0.5x^3$. *Gray areas show the frames of FS-CSA and local frames for each user in FA-CSA-FB. Slots filled with green represent packets of degree-2 users and slots filled with blue represents packets of degree-3 users. The delay of each user is indicated by a number in the slot in which it is decoded. Furthermore, the four striped green slots in the second frame of the FS-CSA example, collide in such a way that the packets in these slots cannot be decoded. Such collision patterns are called stopping sets.*

As described in Section 1.2, CSA can be represented by a bipartite graph. This is of course also true for the systems considered here. In Fig. 2.2, the equivalent graph representation of Fig. 2.1 is depicted.

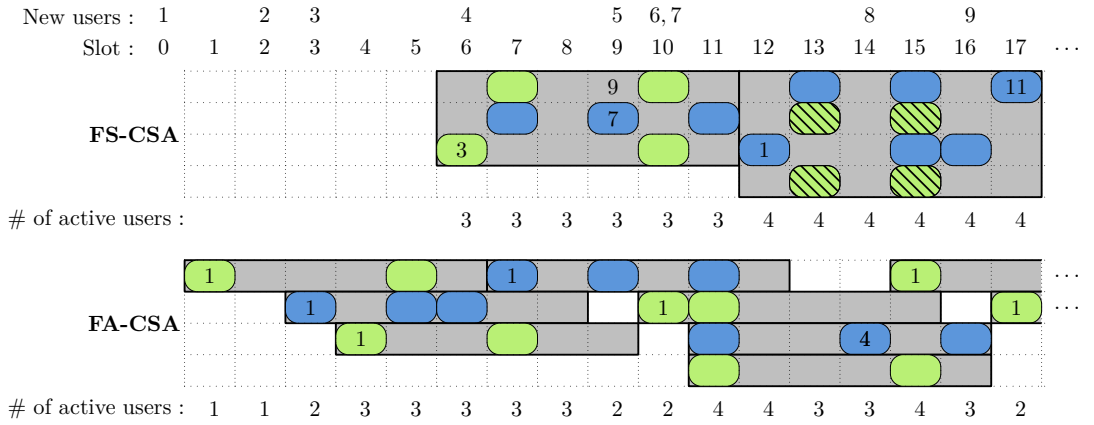


Figure 2.1: An illustration of both FS-CSA and FA-CSA-FB. Both systems have the same new users joining. Green and blue slots represent replicas of degree-2 and 3 users, respectively. The four striped slots of FS-CSA constitutes a stopping set and cannot be resolved by the iterative decoder.

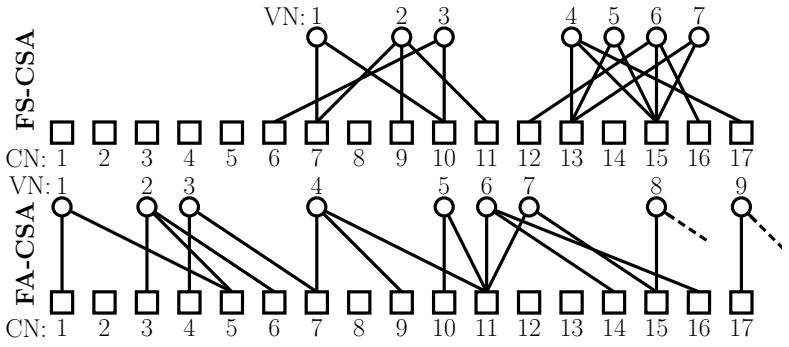


Figure 2.2: Equivalent graph representations of the FS-CSA and FS-CSA systems depicted in Fig. 2.1. VNs (users) are represented by circles and CNs (slots) by squares.

Chapter 3

Analysis

3.1 Degree Distributions

In this section, we derive the VN- and CN-degree distributions for FA-CSA systems. These degree distributions will later be used in the DE analysis. We define the *node-perspective* VN- and CN-degree distribution as in (1.1) and (1.3), i.e.,

$$\Lambda(x) \triangleq \sum_l \Lambda_l x^l \quad \text{and} \quad \mathrm{P}(x) \triangleq \sum_r \mathrm{P}_r\, x^r, \tag{3.1}$$

respectively, where Λ_l is the probability that an arbitrary VN has degree l and P_r is the probability that a CN has degree r. $\Lambda(x)$ is under the control of the system designer and is subject to optimization. We introduce also the *edge-perspective* VN- and CN-degree distributions as in (1.6),

$$\lambda(x) \triangleq \sum_l \lambda_l x^{l-1} \quad \text{and} \quad \rho(x) \triangleq \sum_r \rho_r x^{r-1}, \tag{3.2}$$

where λ_l denotes the probability that an edge is connected to a degree-l VN and ρ_r denotes the probability that an edge is connected to a degree-r CN. The probabilities λ_l and ρ_r are given by

$$\lambda_l = \frac{l\Lambda_l}{\sum_d d\Lambda_d} \quad \text{and} \quad \rho_r = \frac{r\,\mathrm{P}_r}{\sum_d d\,\mathrm{P}_d}, \tag{3.3}$$

i.e., $\lambda(x) = \Lambda'(x)/\Lambda'(1)$ and $\rho(x) = \mathrm{P}'(x)/\,\mathrm{P}'(1)$, where f' denotes the derivative of the function f.

3.1.1 Frame Asynchronous CSA with First Slot Fixed

For CSA systems with boundary effect, the first n CNs all have distinct degree distributions. This gives rise to different *classes* of CNs and VNs. We call a CN at position i (slot i) a class-i CN. Similarly, a VN that joins the system at position $i-1$ is a class-i VN. Additionally, in FA-CSA-F a degree-l class-i VN always has one connection to a class-i CN, i.e., a fixed edge, and it has $l-1$ connections to randomly selected CNs of classes

$$\mathcal{J}_i \triangleq [i+1, i+n-1]. \tag{3.4}$$

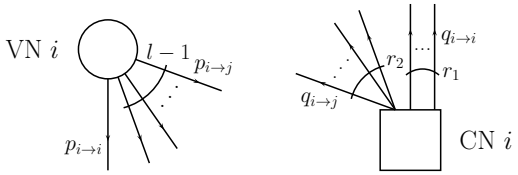


Figure 3.1: Class-i VN and CN and their corresponding connectivity.

The node connectivity for class-i VNs and CNs of FA-CSA-F is depicted in Fig. 3.1. Accordingly, we define the node-perspective VN-degree distributions for FA-CSA-F

$$\begin{aligned}\Lambda^{i\to i}(x) &= x,\\ \Lambda^{i\to\mathcal{J}_i}(x) &= \sum_l \Lambda_l^{i\to\mathcal{J}_i} x^l \overset{(a)}{=} \sum_l \Lambda_l x^{l-1},\end{aligned} \tag{3.5}$$

where $\Lambda^{i\to i}(x)$ represents the fixed connection. $\Lambda_l^{i\to\mathcal{J}_i} = \Lambda_{l+1}$ is the probability that a class-i VN has l connections to CNs of classes in $\mathcal{J}_i$ and in (a) we have made the change of variables $l \to l-1$. The corresponding edge-perspective degree distributions are,

$$\begin{aligned}\lambda^{i\to i}(x) &= 1,\\ \lambda^{i\to\mathcal{J}_i}(x) &= \frac{\left(\Lambda^{i\to\mathcal{J}_i}\right)'(x)}{\left(\Lambda^{i\to\mathcal{J}_i}\right)'(1)} = \sum_l \lambda_l^{i\to\mathcal{J}_i} x^{l-2},\end{aligned} \tag{3.6}$$

with $\lambda_l^{i\to\mathcal{J}_i} = \Lambda_l(l-1)/\sum_l \Lambda_l(l-1)$.

On the other hand, a class-i CN is connected to r_1 class-i VNs and to r_2 VNs of classes in the range

$$\mathcal{K}_i \triangleq \begin{cases}[1, i-1] & \text{for} \quad 1 \le i < n\\ [i-n+1, i-1] & \text{for} \quad i \ge n\end{cases}. \tag{3.7}$$

Correspondingly, we define the degree distributions

$$\mathrm{P}^{i\to i}(x) = \sum_{r_1} \mathrm{P}_{r_1}^{i\to i}\, x^{r_1} \quad \text{and} \quad \mathrm{P}^{i\to\mathcal{K}_i}(x) = \sum_{r_2} \mathrm{P}_{r_2}^{i\to\mathcal{K}_i}\, x^{r_2}, \tag{3.8}$$

where $\mathrm{P}_{r_1}^{i\to i}$ is the probability that a class-i CN has r_1 edges incident to class-i VNs, and $\mathrm{P}_{r_2}^{i\to\mathcal{K}_i}$ is the probability that a class-i CN has r_2 connections to VNs of classes in $\mathcal{K}_i$. The corresponding edge-perspective degree distributions are

$$\rho^{i\to i}(x) = \frac{\left(\mathrm{P}^{i\to i}\right)'(x)}{\left(\mathrm{P}^{i\to i}\right)'(1)} = \sum_{r_1} \rho_{r_1}^{i\to i} x^{r_1-1} \tag{3.9}$$

and

$$\rho^{i\to\mathcal{K}_i}(x) = \frac{\left(\mathrm{P}^{i\to\mathcal{K}_i}\right)'(x)}{\left(\mathrm{P}^{i\to\mathcal{K}_i}\right)'(1)} = \sum_{r_2} \rho_{r_2}^{i\to\mathcal{K}_i} x^{r_2-1}. \tag{3.10}$$

Proposition 1. *The class-i CN degree distributions for FA-CSA-F are given by*

$$\mathrm{P}^{i\to i}(x) = \rho^{i\to i}(x) = \exp(-g(1-x)) \tag{3.11}$$

and

$$\mathrm{P}^{i\to\mathcal{K}_i}(x)=\rho^{i\to\mathcal{K}_i}(x)=\exp\left(-\frac{\delta_i(\Lambda'(1)-1)}{n-1}(1-x)\right), \tag{3.12}$$

with

$$\delta_i=\begin{cases}\min(i-1,n-1)g & \textit{for} \quad \text{FA-CSA-FB}\\ (n-1)g & \textit{for} \quad \text{FA-CSA-FNB}\end{cases}. \tag{3.13}$$

Proof: Denote by R_1 the RV representing the number of edges connecting a class-i CN to class-i VNs. Clearly, $R_1\sim\mathsf{Po}(g)$, because each class-i VN is connected through a single edge to the class-i CN. Therefore,

$$\mathrm{P}_{r_1}^{i\to i}=\Pr(R_1=r_1)=\exp(-g)\frac{g^{r_1}}{r_1!}. \tag{3.14}$$

Now $\mathrm{P}^{i\to i}(x)$ is given by

$$\mathrm{P}^{i\to i}(x)=\sum_{r_1}\mathrm{P}_{r_1}^{i\to i}\,x^{r_1}=\sum_{r_1=0}^{\infty}\exp(-g)\frac{g^{r_1}}{r_1!}x^{r_1}\overset{(a)}{=}\exp\left(-g(1-x)\right), \tag{3.15}$$

where in (a) we used that $\sum_{n=0}^{\infty}\frac{x^n}{n!}=\exp(x)$. Furthermore,

$$\rho^{i\to i}(x)=\frac{\left(\mathrm{P}^{i\to i}\right)'(x)}{\left(\mathrm{P}^{i\to i}\right)'(1)}=\frac{g\exp(-g(1-x))}{g\exp(0)}=\exp(-g(1-x)). \tag{3.16}$$

We now denote by $R_{2,i}$ the number of edges connecting a class-i CN to VNs of classes in the range $\mathcal{K}_i$, as given by (3.7). The number of VNs that joins the system in $\mathcal{K}_i$ is a Poisson RV, denoted by K_i, with mean δ_i given in (3.13). Each VN in $\mathcal{K}_i$ connects to the class-i CN with probability

$$p=\frac{\Lambda'(1)-1}{n-1}. \tag{3.17}$$

Applying the law of total probability this gives,

$$\begin{aligned}\mathrm{P}_{r_2}^{i\to\mathcal{K}_i}&=\sum_{k=r_2}^{\infty}\Pr(R_{2,i}=r_2\mid K_i=k)\Pr(K_i=k)\\&=\sum_{k=r_2}^{\infty}\binom{k}{r_2}p^{r_2}(1-p)^{k-r_2}\exp(-\delta_i)\frac{\delta_i^k}{k!}\\&=\exp(-\delta_i)\left(\frac{p}{1-p}\right)^{r_2}\sum_{k=r_2}^{\infty}\frac{k!}{r_2!(k-r_2)!}(1-p)^k\frac{\delta_i^k}{k!}\\&=\frac{\exp(-\delta_i)}{r_2!}\left(\frac{p}{1-p}\right)^{r_2}\sum_{k=r_2}^{\infty}\frac{((1-p)\delta_i)^k}{(k-r_2)!}\\&\overset{(a)}{=}\frac{\exp(-\delta_i)}{r_2!}(p\delta_i)^{r_2}\sum_{k=0}^{\infty}\frac{((1-p)\delta_i)^k}{k!}\\&\overset{(b)}{=}\exp(-p\delta_i)\frac{(p\delta_i)^{r_2}}{r_2!},\end{aligned} \tag{3.18}$$

where in (a) we used $k' = k - r_2$ and $k' \leftarrow k$ and in (b) we used that $\sum_{n=0}^{\infty} \frac{x^n}{n!} = \exp(x)$.

Following similar steps as in (3.15) and (3.16) gives

$$\mathrm{P}^{i\to\mathcal{K}_i}(x) = \rho^{i\to\mathcal{K}_i}(x) = \exp(-p\delta_i(1-x)) = \exp\left(-\frac{\delta_i(\Lambda'(1)-1)}{n-1}(1-x)\right), \quad (3.19)$$

where p is given in (3.17) ■

3.1.2 Frame Asynchronous CSA with Uniform Slot Selection

For FA-CSA-U, we need to consider only one degree distribution per CN class as defined in (3.1) and denote by $\mathrm{P}_i(x)$ and $\rho_i(x)$ the node-perspective and edge-perspective class-i CN degree distributions respectively.

Proposition 2. *The class-i CN degree distribution for FA-CSA-U is given by*

$$\mathrm{P}_i(x) = \rho_i(x) = \exp\left(-\frac{\mu_i}{n}\Lambda'(1)(1-x)\right), \quad (3.20)$$

where for FA-CSA-UB μ_i is given by (2.3) *and for FA-CSA-UNB μ_i is given by* (2.4).

Proof: A class-i CN can be connected with any of the M_i VNs at position i. The probability that each of the M_i VNs connects to the class-i CN is $\Lambda'(1)/n$. This setup is similar to the setup for the derivation of $\mathrm{P}^{i\to\mathcal{K}_i}(x)$ in Proposition 1. Taking similar steps, it directly follows that

$$\mathrm{P}_i(x) = \rho_i(x) = \exp\left(-\frac{\mu_i}{n}\Lambda'(1)(1-x)\right). \quad (3.21)$$

■

The CN degree distribution for FS-CSA, found in [9] holds also when a Poisson user model is used, although the assumption that $n \to \infty$ is not necessary. In fact, the CN degree distributions for FA-CSA-UNB and FS-CSA are the same, which is not unexpected since FA-CSA-U is similar to FS-CSA in that the edges of an arbitrary VN are connected to its local frame the same way as a VN of FS-CSA connects edges to a global frame.

3.2 Density Evolution Analysis

In this section we derive the DE equations for FA-CSA with boundary effect that will predict the asymptotic performance and allows us to find the decoding threshold $g^\star$ for an FA-CSA system. The derivation is similar to that for standard CSA given in Section 1.2.

3.2.1 Frame Asynchronous CSA with First Slot Fixed

Because a class-i VN is always connected to a class-i CN, all edges of FA-CSA-F are not equivalent, see Fig. 3.1. Therefore, we must differentiate between *edge types*, and thus update $p_{i\to i}$, $p_{i\to j}$, $q_{i\to i}$, and $q_{i\to j}$ separately in the DE, where $p_{i\to i}$ denotes the probability of an erasure message from a class-i VN to a class-i CN, $p_{i\to j}$ denotes the probability of an erasure message from a class-i VN to a class-j CN with $j \neq i$, $q_{i\to i}$ denotes the probability of an erasure message from a class-i CN to a class-i VN and $q_{i\to j}$ denotes the probability of an erasure message from a class-i CN o to a class-j-VN with $j \neq i$.

A message from a class-i VN is in erasure if all incoming messages on other edges are in erasure, i.e.,

$$p_{i\to i} = \sum_l \Lambda_l \tilde{q}_i^{\,l-1} = \Lambda^{i\to\mathcal{J}_i}(\tilde{q}_i), \tag{3.22}$$

$$p_{i\to j} = q_{i\to i} \sum_l \lambda_l^{i\to\mathcal{J}_i} \tilde{q}_i^{\,l-2} = q_{i\to i}\lambda^{i\to\mathcal{J}_i}(\tilde{q}_i), \tag{3.23}$$

for $j \in \mathcal{J}_i$, where $\mathcal{J}_i$ is given in (3.4), and

$$\tilde{q}_i = \frac{1}{n-1}\sum_{j\in\mathcal{J}_i} q_{j\to i}, \tag{3.24}$$

is the average erasure probability of incoming messages from CNs in $\mathcal{J}_i$.

A class-i CN is resolved whenever at most one of its edges carries an erasure message. Equivalently, a message from a CN is not an erasure if none of the incoming $r_1 + r_2 - 1$ messages on other edges are in erasure. Therefore,

$$\begin{aligned} q_{i\to i} &= 1 - \left(\sum_{r_1=1}^{\infty} \rho_{r_1}^{i\to i}(1-p_{i\to i})^{r_1-1}\right)\left(\sum_{r_2=0}^{\infty} \mathrm{P}_{r_2}^{i\to\mathcal{K}_i}(1-\tilde{p}_i)^{r_2}\right) \\ &\overset{(a)}{=} 1 - \left(\sum_{r_1=0}^{\infty} \frac{e^{-g}g^{r_1}}{r_1!}(1-p_{i\to i})^{r_1}\right)\mathrm{P}^{i\to\mathcal{K}_i}(1-\tilde{p}_i) \\ &= 1 - \mathrm{P}^{i\to i}(1-p_{i\to i})\,\mathrm{P}^{i\to\mathcal{K}_i}(1-\tilde{p}_i) \\ &= 1 - \exp\left(-gp_{i\to i}\right)\exp\left(-\frac{\delta_i(\Lambda'(1)-1)}{n-1}\tilde{p}_i\right) \end{aligned} \tag{3.25}$$

where in (a) we used $\rho_{r_1}^{i\to i} = \mathrm{P}_{r_1}^{i\to i}$, $r_1' = r_1 - 1$ and $r_1' \leftarrow r_1$, and where

$$\tilde{p}_i = \begin{cases} 0 & \text{for} \quad i = 1 \\ \sum_{k\in\mathcal{K}_i} p_{k\to i}/(i-1) & \text{for} \quad 1 < i < n \\ \sum_{k\in\mathcal{K}_i} p_{k\to i}/(n-1) & \text{for} \quad i \geq n \end{cases}. \tag{3.26}$$

is the average erasure probability of incoming messages from VNs in $\mathcal{K}_i$ and $\mathcal{K}_i$ is given by (3.7). Similarly,

$$\begin{aligned} q_{i\to j} &= 1 - \left(\sum_{r_1=0}^{\infty} \mathrm{P}_{r_1}^{i\to i}(1-p_{i\to i})^{r_1}\right)\left(\sum_{r_2=1}^{\infty} \rho_{r_2}^{i\to\mathcal{K}_i}(1-\tilde{p}_i)^{r_2-1}\right) \\ &= 1 - \exp\left(-gp_{i\to i}\right)\exp\left(-\frac{\delta_i(\Lambda'(1)-1)}{n-1}\tilde{p}_i\right) \end{aligned} \tag{3.27}$$

for $j \in \mathcal{K}_i$. Note that $q_{i\to i} = q_{i\to j}$, which follows from the fact that $\mathrm{P}^{i\to i}(x) = \rho^{i\to i}(x)$ and $\mathrm{P}^{i\to\mathcal{K}_i}(x) = \rho^{i\to\mathcal{K}_i}(x)$ which, in turn, follows from the properties of the Poisson distribution (see Proposition 1). For a general user model, however, $q_{i\to i} \neq q_{i\to j}$.

DE is now performed by iteratively updating (3.22)-(3.27), with $p_{i\to i}$, $p_{i\to j}$, $q_{i\to i}$ and $q_{i\to j}$ initialized to 1. The PLR at position i can be computed as $\bar{p}_i = \Lambda(\tilde{q}_i)q_{i\to i}/\tilde{q}_i$ and the threshold $g^\star$ is found by searching for the largest value of g for which $\bar{p}_i$ converges to 0 for all positions. For a system without boundary effect (3.22)-(3.27) are only updated for $i > n$

We remark that exact DE requires $n \to \infty$. This would require to keep track of an infinite number of node classes, which is unfeasible in practice. Therefore, the thresholds computed in Chapter 4 must be seen as *approximate* DE thresholds. However, we have found that it is sufficient to set $n \approx 100$ and run DE over a chain of $20n$ positions in order to obtain $g^\star$ with good precision. Considering larger values of n does not change the obtained thresholds.

3.2.2 Frame Asynchronous CSA with Uniform Slot Selection

For FA-CSA-U all edges are equivalent and therefore we do not need to consider different edge-types. We denote by p_i the probability that an erasure message is passed from a class-i VN , and by q_i the probability that an erasure message is passed from the class-i CN. It follows that,

$$p_i = \sum_l \lambda_l \tilde{q}_i^{l-1} = \lambda(\tilde{q}_i), \tag{3.28}$$

where

$$\tilde{q}_i = \frac{1}{n}\sum_{j=i}^{i+n-1} q_i, \tag{3.29}$$

is the average erasure probability of the incoming messages to a class-i VN. Furthermore,

$$q_i = 1 - \sum_r \rho_{i,r}(1-\tilde{p}_i)^{r-1} = 1 - \rho_i(1-\tilde{p}_i) = 1 - \exp\left(-\frac{\mu_i}{n}\Lambda'(1)\tilde{p}_i\right). \tag{3.30}$$

where

$$\tilde{p}_i = \begin{cases} \sum_{j=1}^{i} p_j/i & \text{for} \quad 1 \le i < n \\ \sum_{j=i-n+1}^{i} p_j/n & \text{for} \quad i \ge n \end{cases}. \tag{3.31}$$

is the average erasure probability of the incoming messages to the class-i CN.

DE is performed similarly to FA-CSA-F, by iteratively updating (3.28)-(3.31), with p_i and q_i initialized to 1. The PLR at position i can be computed as $\bar{p}_i = \Lambda(\tilde{q}_i)$.

3.3 Finite Frame Length Analysis

In the finite frame length regime, CSA exhibits an error floor in its PLR performance for low to medium loads g. The EF is due to stopping sets, i.e., graph structures which make the iterative decoder fail.

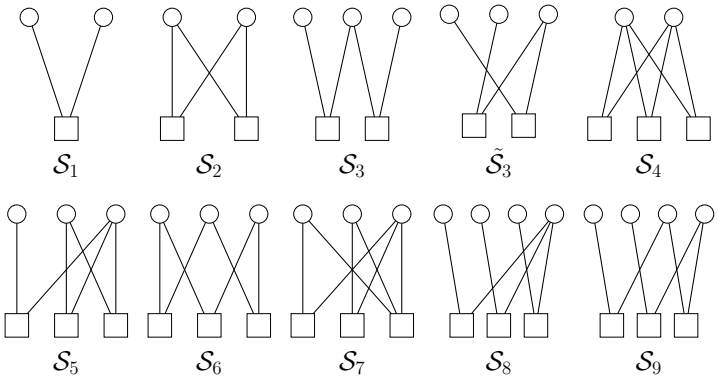


Figure 3.2: All minimal stopping sets with $\mu(\mathcal{S}) \leq 3$, also including two isomorphic graphs $\mathcal{S}_3$ and $\tilde{\mathcal{S}}_3$.

Definition 3. *A stopping set $\mathcal{S}$ is a connected bipartite subgraph with all CNs of degree strictly larger than 1.*

The EF of CSA is dominated by *minimal stopping sets*.

Definition 4. *A minimal stopping set is a stopping set that does not contain a nonempty stopping set of smaller size.*

In this section, we find estimates of the EF by approximating the probability of occurrence of minimal stopping sets.

We first introduce some useful notation for a stopping set $\mathcal{S}$. Let $\mu(\mathcal{S})$ denote the number of CNs, $\nu(\mathcal{S})$ the number of VNs, and $v_l(\mathcal{S})$ the number of degree-l VNs in $\mathcal{S}$. Moreover, we define the degree profile of a stopping set as the vector $\boldsymbol{v}(\mathcal{S}) = [v_0(\mathcal{S}), v_1(\mathcal{S}), \ldots, v_{\mu(\mathcal{S})}(\mathcal{S})]$, and denote by $c(\mathcal{S})$ the number of graph isomorphisms of $\mathcal{S}$ [23, p.4]. Unfortunately, there is no straightforward analytical expression for $c(\mathcal{S})$. However, $c(\mathcal{S})$ is tabulated in Appendix A, along with $\boldsymbol{v}(\mathcal{S})$, $\nu(\mathcal{S})$, $\mu(\mathcal{S})$ and $c(\mathcal{S})$ for all 142 minimal stopping sets of FA-CSA and FS-CSA with $\mu(\mathcal{S}) \leq 5$. Furthermore, we remark that similar finite frame length analysis for regular VN-degree distributions has previously been presented in [19, 20] for the fully asynchronous CSA and FS-CSA respectively. In Fig. 3.2 we depict all nine minimal stopping sets for FA-CSA with $\mu(\mathcal{S}) \leq 3$, including two isomorphic stopping sets, $\mathcal{S}_3$ and $\tilde{\mathcal{S}}_3$.

If we allow infinitely many decoding iterations and let $n_{\mathrm{RX}} \to \infty$ for FA-CSA, all packet losses in CSA are caused by stopping sets. The PLR, see Definition 1, is then equivalent to the probability that an arbitrary VN is part of a stopping set.

We denote by $\mathcal{A}$ the set of all stopping sets, and by $\mathcal{A}^\star \subset \mathcal{A}$ a smaller set of minimal stopping sets that dominate the PLR in the EF region. Furthermore, let u denote an arbitrary VN in a CSA system. The PLR can be approximated as follows,

$$\begin{aligned} \bar{p} &= \Pr\left(\bigcup_{\mathcal{S}\in\mathcal{A}} u \in \mathcal{S}\right) \overset{(a)}{\leq} \sum_{\mathcal{S}\in\mathcal{A}} \Pr(u \in \mathcal{S}) \overset{(b)}{\approx} \sum_{\mathcal{S}\in\mathcal{A}^\star} \Pr(u \in \mathcal{S}) \\ &\overset{(c)}{=} \sum_{\mathcal{S}\in\mathcal{A}^\star} \sum_{m=0}^{\infty} \Pr(u \in \mathcal{S}|m) \Pr(M = m). \end{aligned} \tag{3.32}$$

In (a) the probability is upper bounded using the union bound. In (b) we consider a summation over the subset $\mathcal{A}^\star$, turning the upper bound into an approximation.

Lastly, in (c) we condition the probability of u being part of a stopping set $\mathcal{S}$ on the RV M, representing the number of VNs that can create the stopping set $\mathcal{S}$ with u, and average over all possible values of M.

Using (3.32) as a starting point, we derive EF approximations for FA-CSA-FNB and FA-CSA-UNB. We do not consider boundary effects as this simplifies the analysis. Furthermore, we remark that a boundary has negligible impact on the EF of a system that runs for a longer time.

We express $\Pr(u \in \mathcal{S}|m)$ in (3.32) in terms of factors that are simpler to derive,

$$\Pr(u \in \mathcal{S}|m) = \frac{a(\mathcal{S},m)b(\mathcal{S})c(\mathcal{S})}{d(\mathcal{S})} \cdot \frac{\nu(\mathcal{S})}{m}, \tag{3.33}$$

where $a(\mathcal{S},m)$ is the expected number of ways to select $\nu(\mathcal{S})$ VNs with the degree profile $\boldsymbol{v}(\mathcal{S})$ from a set of m VNs with degree distribution $\Lambda(x)$, $b(\mathcal{S})$ is the number of ways to select the CNs of $\mathcal{S}$ such that $u \in \mathcal{S}$, $c(\mathcal{S})$ is the number of graph-isomorphisms of $\mathcal{S}$, and $d(\mathcal{S})$ is the total number of ways in which $\nu(\mathcal{S})$ VNs (including u) with degree profile $\boldsymbol{v}(\mathcal{S})$ can connect edges to CNs in their local frames. The fraction $\frac{\nu(\mathcal{S})}{m}$ represents the probability that VN u is one of the $\nu(\mathcal{S})$ VNs in $\mathcal{S}$.

We give first the factor $a(\mathcal{S})$, because it is the same for FA-CSA-FNB and FA-CSA-UNB,

$$a(\mathcal{S},m) = \binom{m}{\nu(\mathcal{S})} \nu(\mathcal{S})! \prod_l \frac{\Lambda_l^{v_l(\mathcal{S})}}{v_l(\mathcal{S})!}, \tag{3.34}$$

which stems from the multinomial distribution and was derived in [14]. In the following, we derive expressions for the factors $b(\mathcal{S})$ and $d(\mathcal{S})$.

3.3.1 Frame Asynchronous CSA with First Slot Fixed

Let u represent a VN active in the range $[i, i+n-1]$. Furthermore, to simplify the derivation we make the assumption that $\mathcal{S}$ spans at most n slots. Without loss of generality, we consider the range $[i, i+n-1]$.

Since we are considering stopping sets constrained to the slots in the range $[i, i+n-1]$ that contain u, the first slot of the stopping set must be i. According to our assumption, the remaining $\mu(\mathcal{S})-1$ slots of $\mathcal{S}$ are chosen with equal probability from the subsequent $n-1$ slots. This gives,

$$b_{\mathrm{FA-F}}(\mathcal{S}) \approx \binom{n-1}{\mu(\mathcal{S})-1}. \tag{3.35}$$

We now consider $d_{\mathrm{FA-F}}(\mathcal{S})$. An arbitrary user in slot $i+n-1$ has n equiprobable slots for its first replica, i.e., the slots in $[i, i+n-1]$. However, the first replica of user u is fixed to slot i. For each placement of a degree-l user's first replica, there are $\binom{n-1}{l-1}$ possible placements for its remaining replicas. Furthermore, each user places its replicas independently of other users. Thus,

$$d_{\mathrm{FA-F}}(\mathcal{S}) = n^{-1} \prod_l \left(n \binom{n-1}{l-1} \right)^{v_l(\mathcal{S})}. \tag{3.36}$$

An EF-approximation for FA-CSA-F is now given by evaluating (3.32), using (3.33)-(3.36) and $\Pr(M=m)=e^{-ng}(ng)^m/m!$,

$$\bar{p}_{\mathrm{FA-F}} \approx \sum_{\mathcal{S}\in\mathcal{A}^\star}\sum_{m=0}^{\infty} \frac{a(\mathcal{S},m)b_{\mathrm{FA-F}}(\mathcal{S})c(\mathcal{S})}{d_{\mathrm{FA-F}}(\mathcal{S})}\frac{\nu\left(\mathcal{S}\right)}{m}\frac{e^{-ng}(ng)^m}{m!}. \tag{3.37}$$

3.3.2 Frame Asynchronous CSA with Uniform Slot Selection

We denote by u an arbitrary VN in an FA-CSA-U system. Without loss of generality we assume that if a VN $u\in\mathcal{S}$, then u is the highest degree VN of $\mathcal{S}$. We make a simplifying assumption that all the VNs in $\mathcal{S}$ must be active in the entire range $[k_{\mathrm{f}},k_{\mathrm{l}}]$ where k_{f} and k_{l} are the positions of the first and last CNs that u is connected to respectively and we let $q(\mathcal{S})$ denote the degree of u.

If we denote by D the RV representing the distance $k_{\mathrm{l}}-k_{\mathrm{f}}$, then its probability mass function (pmf) is given by,

$$\Pr(D=d)=(n-d)\frac{\binom{d-1}{q(\mathcal{S})-2}}{\binom{n}{q(\mathcal{S})}} \tag{3.38}$$

for $d\in[q(\mathcal{S})-1,n-1]$. According to our assumption, the number of VNs from which the VNs of $\mathcal{S}$ can be selected is Poisson distributed with mean $g(n-D)$. We let $M\sim\mathsf{Po}(g(n-D))$ be the RV representing this number, then m in (3.32) is a realization of M, such that,

$$\Pr(M=m)=\sum_{d=q(\mathcal{S})-1}^{n-1}\frac{e^{-g(n-d)}(g(n-d))^m}{m!}(n-d)\frac{\binom{d-1}{q(\mathcal{S})-2}}{\binom{n}{q(\mathcal{S})}}, \tag{3.39}$$

obtained by averaging over D.

The CNs for $\mathcal{S}$ are, according to the assumption, selected randomly from a set of n CNs corresponding to the local frame of u, thus,

$$b_{\mathrm{FA-U}}(\mathcal{S})\approx\binom{n}{\mu\left(\mathcal{S}\right)}. \tag{3.40}$$

A degree-l VN can connect its edges in $\binom{n}{l}$ ways to its local frame, hence,

$$d_{\mathrm{FA-U}}(\mathcal{S})=\prod_{l}\binom{n}{l}^{v_l(\mathcal{S})}. \tag{3.41}$$

Now, evaluating (3.32), using (3.34), (3.40)-(3.41), and (3.39) gives,

$$\bar{p}_{\mathrm{FA-U}}\approx\sum_{\mathcal{S}\in\mathcal{A}^\star}\frac{b_{\mathrm{FA-U}}(\mathcal{S})c(\mathcal{S})}{d_{\mathrm{FA-U}}(\mathcal{S})}\sum_{\substack{d=\\ q(\mathcal{S})-1}}^{n-1}\Pr(D=d)\sum_{m=0}^{\infty}a(\mathcal{S},m)\frac{\nu\left(\mathcal{S}\right)}{m}\frac{e^{-g(n-d)}(g(n-d))^m}{m!}, \tag{3.42}$$

3.3.3 Frame Synchronous CSA

The probability $\Pr(u \in \mathcal{S}|m)$ in (3.33) for an FS system with constant number of users per frame m, has previously been derived in [13] and [14]. For completeness, we give the corresponding expressions of the factors in (3.33), because the formulation that we use is slightly different and also includes the Poisson user model.

The CNs for $\mathcal{S}$ are selected randomly and uniformly from a set of n CNs (corresponding to the n slots of the frame) and thus,

$$b_{\mathrm{FS}}(\mathcal{S}) = \binom{n}{\mu(\mathcal{S})}. \tag{3.43}$$

A degree-l VN can connect its edges in $\binom{n}{l}$ ways to the frame, hence

$$d_{\mathrm{FS}}(\mathcal{S}) = \prod_l \binom{n}{l}^{v_l(\mathcal{S})}. \tag{3.44}$$

Now, evaluating (3.32), using (3.34), (3.43)-(3.44), and $\Pr(M=m) = e^{-ng}(ng)^m/m!$ gives,

$$\bar{p}_{\mathrm{FS}} \approx \sum_{\mathcal{S}\in\mathcal{A}^\star} \sum_{m=0}^{\infty} \frac{a(\mathcal{S},m) b_{\mathrm{FS}}(\mathcal{S}) c(\mathcal{S})}{d_{\mathrm{FS}}(\mathcal{S})} \frac{\nu(\mathcal{S})}{m} \frac{e^{-ng}(ng)^m}{m!}. \tag{3.45}$$

3.3.4 Numerical Evaluation of Error Floor Approximations

We give an easy-to-use formula to evaluate (3.37),(3.42) and (3.45),

$$\bar{p} \approx \sum_{\mathcal{S}\in\mathcal{A}^\star} \phi(\mathcal{S}) \nu(\mathcal{S})\, c(\mathcal{S}) \binom{n}{\mu(\mathcal{S})} \prod_l \frac{\Lambda_l^{\nu_l(\mathcal{S})}}{\nu_l(\mathcal{S})!} \binom{n}{l}^{-\nu_l(\mathcal{S})}, \tag{3.46}$$

where $\phi(\mathcal{S})$ is given by

$$\phi(\mathcal{S}) = \begin{cases} \mu(\mathcal{S}) \prod_d d^{-\nu_d(\mathcal{S})} \sum_{k=0}^{\nu(\mathcal{S})-1} (-1)^{\nu(\mathcal{S})-1+k} \frac{(\nu(\mathcal{S})-1)!}{k!} (ng)^k & \text{for FA-F} \\ \sum_{k=0}^{\nu(\mathcal{S})-1} \sum_{d=q(\mathcal{S})-1}^{n-q(\mathcal{S})} (-1)^{\nu(\mathcal{S})-1+k} \frac{(\nu(\mathcal{S})-1)!}{k!} ((n-d)g)^k (n-d) \frac{\binom{d-1}{q(\mathcal{S})-2}}{\binom{n}{q(\mathcal{S})}} & \text{for FA-U} \\ \sum_{k=0}^{\nu(\mathcal{S})-1} (-1)^{\nu(\mathcal{S})-1+k} \frac{(\nu(\mathcal{S})-1)!}{k!} (ng)^k & \text{for FS.} \end{cases} \tag{3.47}$$

In (3.47), it has been used that

$$\sum_{m=0}^{\infty} \frac{e^{-ng}(ng)^m}{m(m-\nu(\mathcal{S}))!} = \sum_{k=0}^{\nu(\mathcal{S})-1} (-1)^{\nu(\mathcal{S})-1+k} \frac{(\nu(\mathcal{S})-1)!}{k!} (ng)^k \tag{3.48}$$

thus, the infinite sum over m in (3.37), (3.42) and (3.45) can be replaced by a finite sum, which makes the evaluation simpler.

Chapter 4

Numerical Results

In this chapter, we give numerical results on the performance of FA-CSA, both in the asymptotic regime and in the finite frame length regime. We compare the performance of FA-CSA to FS-CSA and SC-CSA in terms of decoding threshold, EF, and delay.

4.1 Iterative Decoding Thresholds

In Table 4.1, we give (asymptotic) iterative decoding thresholds for FA-CSA, for $\Lambda(x) = x^l$ with $l = 3$, 4, 5, 6, 7 and 8, and $\Lambda(x) = \Lambda^\star(x) = 0.86x^3 + 0.14x^8$. $\Lambda^\star(x)$ was obtained in [14] for FS-CSA by a joint optimization of the EF and the threshold.

We observe that for FA-CSA with boundary effect, the decoding threshold improves significantly with respect to the case where there are already active users at time $i = 0$. This is due to a boundary effect (thus its name) caused by the lower degree of the CNs for $i \in [1, n-1]$, which results in a wave-like decoding effect similar to that of spatially coupled LDPC (SC-LDPC) codes. Furthermore, for FA-CSA with boundary effect and with regular VN-degree distribution $\Lambda(x) = x^l$, the decoding threshold improves with increasing VN degree, whereas the opposite occurs for the systems without boundary effect. This behavior is similar to that of regular LDPC codes, where a larger VN degree improves the threshold for SC-LDPC codes but has the opposite effect for uncoupled LDPC codes.

Table 4.1: DE thresholds for FA-CSA and FS-CSA

$\Lambda(x)$	x^3	x^4	x^5	x^6	x^7	x^8	$\Lambda^\star(x)$
$g^\star_{\text{FA-FB}}$	0.917	0.976	0.992	0.997	0.998	0.999	0.963
$g^\star_{\text{FA-UB}}$	0.917	0.976	0.992	0.997	0.998	0.999	0.963
$g^\star_{\text{FA-FNB}}$	0.818	0.772	0.701	0.637	0.581	0.534	0.851
$g^\star_{\text{FA-UNB}}$	0.818	0.772	0.701	0.637	0.581	0.534	0.851
$g^\star_{\text{FS}}$	0.818	0.772	0.701	0.637	0.581	0.534	0.851

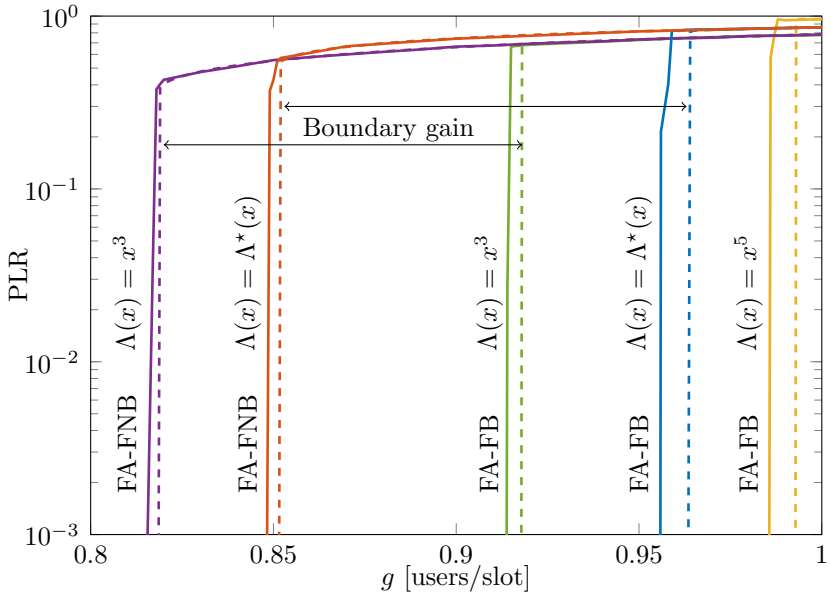


Figure 4.1: DE (dashed lines) and simulation results for $n = 10^5$ (solid lines) of the PLR for FA-CSA-FB and FA-CSA-FNB

We also give in Table 4.1 the corresponding decoding thresholds for FS-CSA, denoted by $g^\star_{\text{FS}}$. FA-CSA with boundary effect yields significantly better thresholds than FS-CSA. Interestingly, the thresholds for FA-CSA-FNB, FA-CSA-UNB and FS-CSA are identical. Indeed, the systems are very similar in that FS-CSA and FA-CSA-UNB have the same CN-degree distribution and CNs of FA-CSA-FNB have the same average degree, but a slightly different node connectivity.

In Fig. 4.1, we plot the PLR of FA-CSA with boundary effect obtained from DE (dashed lines) together with simulation results for $n = 10^5$ (solid lines), for $\Lambda(x) = x^l$ with $l = 3$ and 5, and $\Lambda(x) = \Lambda^\star(x)$. The figure shows that the DE equations are in good agreement with the simulations and make apparent the boundary gain for $\Lambda(x) = x^3$ and $\Lambda^\star(x)$.

4.2 Finite Frame Length Packet Loss Rate and Error Floors

In Fig. 4.2 we plot the simulated PLR as a function of the system load, g, for FA-CSA-FNB, FA-CSA-UNB and FS-CSA, with $\Lambda(x) = \Lambda^\star(x)$ and $n = 200$. The EF predictions, as derived in Section 3.3, are also shown with dotted lines. We observe that both instances of FA-CSA outperforms FS-CSA in the EF and WF regions. Furthermore, FA-CSA-FNB has a lower EF than FA-CSA-UNB, as predicted by the EF approximations. This hierarchy of the EF performance holds in general, i.e., for any n, and $\Lambda(x)$. We remark further that corresponding FA systems with boundary effect does not exhibit a better EF if they are run for a long period.

Even though the two FA-CSA systems whose PLR is presented in Fig. 4.2 do not have a boundary effect, it is apparent that they exhibit much superior WF performance as compared to FS-CSA. Since all three systems have the same asymptotic decoding thresholds $g^\star = 0.851$, this means that FA-CSA systems without boundary

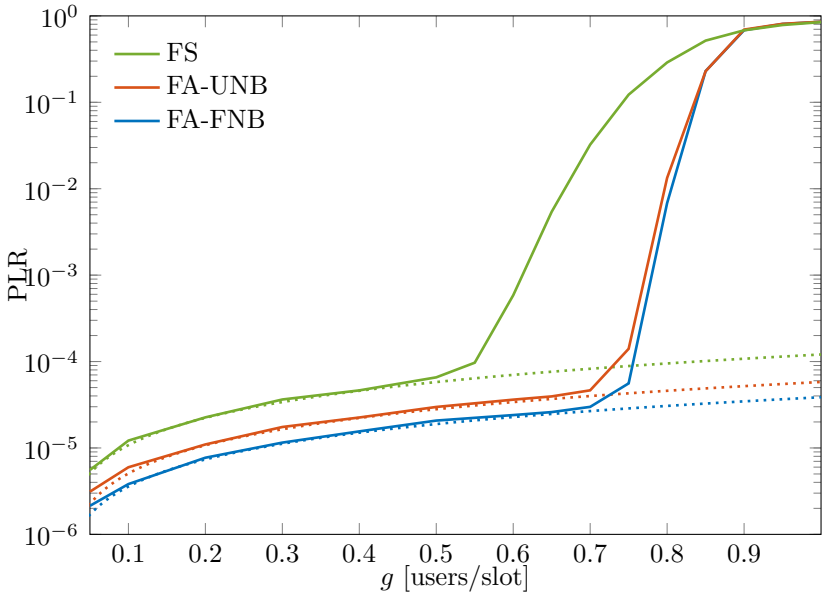


Figure 4.2: Simulated PLR (solid) and EF approximations (dotted) for FA-CSA-FNB, FA-CSA-UNB and FS-CSA with $n = 200$ and $\Lambda(x) = \Lambda^{\star}(x)$.

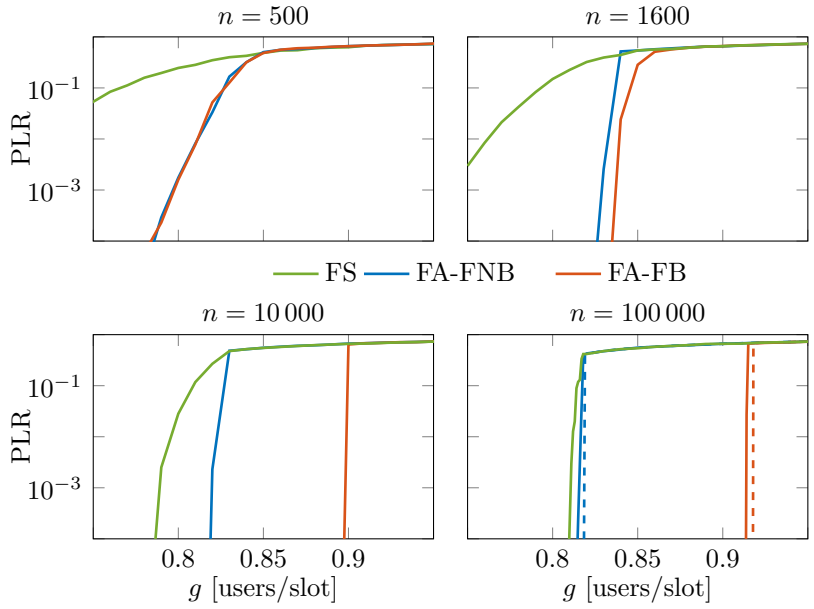


Figure 4.3: Simulated PLR performance in the WF region for FA-CSA-FNB, FA-CSA-FB, and FS-CSA with $\Lambda(x) = x^3$ for increasing frame lengths n.

are punished less by the finite frame length restriction and can therefore support larger system loads while still providing reliable multiple access in the finite frame length regime.

In Fig. 4.3 we compare the PLR performance in the WF region of FA-CSA-FB, FA-CSA-FNB, and FS-CSA for $\Lambda(x) = x^3$ and frame lengths $n = 500$, 1600, 10 000, and 100 000. For short frame lengths ($n = 500$) FA-CSA-FB and FA-CSA-FNB have similar PLR performance, whereas FS-CSA performs worse. When the frame length is increased however, the FA system with boundary effect outperforms the system without boundary effect in the WF region. This is seen already at $n = 1600$ where the performance of FA-CSA-FB is slightly better as compared to FA-CSA-FNB. This *boundary gain* increases with the frame length, as observed for $n = 10\,000$ and $n = 100\,000$. The asymptotic performance for FA-CSA-FB and FA-CSA-FNB given by DE is also plotted as dashed lines with the results for $n = 100\,000$. As the frame length is increased, we also notice how the performance of FS-CSA approaches that of FA-CSA-FNB, as predicted by DE and the results in Table 4.1.

We have argued previously that the reason systems with boundary effect asymptotically has a large gain in terms of decoding threshold is because the lower degree CNs at the boundary will induce a wave-like decoding effect. This improvement, of course, will only happen if the wave can propagate through the entirety of the system. Due to the randomness of the Poisson user model the experienced load in a window of n slots will sometimes be above, and sometimes below the expected load g users/slot. Such variations are more distinct for low frame lengths. Therefore, we believe, and our simulations suggest, that for low frame lengths an induced wave will be broken by events where the experienced load is large, causing the wave not to propagate further. Once the wave has been broken the system is equivalent to a system without boundary effect, which explains why an improved WF performance of FA-CSA-FB for $n = 500$ is not observed. For large frame lengths, instead, the variation of the experienced load will be lower and a wave can therefore propagate without being broken and will improve performance in the WF region.

By careful inspection of Fig. 4.3, we notice that the performance at a load $g \approx 0.83$ of FA-CSA-FNB is better for $n = 1600$ compared to $n = 100\,000$. This is counterintuitive at first, but might be explained by the same reasoning as above. For particular frame lengths n, the variations of experienced load could be large enough, such that the low peaks with some chance will induce a decoding wave, similar to the wave of a system with boundary effect. To benefit from such events it would be necessary that the frame length n is not too low, which would cause a decoding wave to break soon after its occurrence. If instead the frame length is large the probability of a sporadic wave's creation would be extremely low due to the low variations of the experienced load. Also this theory has been supported by simulations. For FA-CSA-FNB with $n = 1600$ at a nominal system load $g \approx 0.83$, we have observed a sudden drop of the simulated PLR from a high level down to the level of the EF after a number of slots. After the drop, the PLR remains low for the duration of the simulation, suggesting that a wave is propagating. We remark that this is indeed a sporadic behavior which occurs at different times (with respect to the start of the system) in each round of simulation.

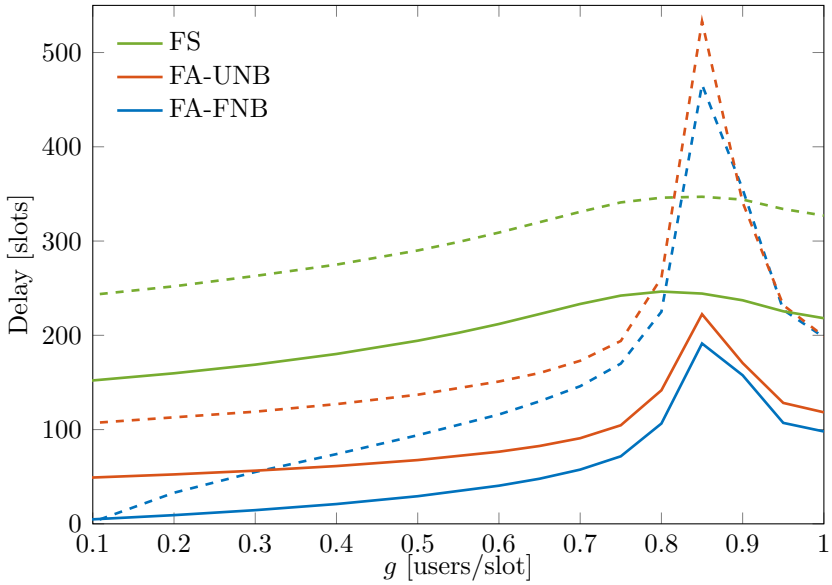


Figure 4.4: Delay performance of FA-CSA-FNB, FA-CSA-UNB and FS-CSA, with $n = 200$ and $\Lambda(x) = \Lambda^{\star}(x)$. Solid lines show the average delay and dashed lines show the 90th percentile of the delay.

4.3 Finite Frame Length Delay Performance

We compare the delay (see Definition 2) performance of FA-CSA-FNB, FA-CSA-UNB and FS-CSA for $n = 200$ and $\Lambda^{\star}(x)$. In Fig. 4.4 the average and 90th percentile of the delay is depicted for $g \in [0.1, 1]$ users/slot. FA-CSA-FNB performs best in terms of average delay and FS-CSA worst. This is expected because a user in FA-CSA-FNB sends its replicas sooner after joining the system than in FA-CSA-UNB and FS-CSA. However, in terms of the 90th percentile, the two FA systems perform worse than FS-CSA for $g \in [0.8, 0.9]$ users/slot. We remark that the delay is only defined for successfully received packets, and for $g \in [0.8, 0.9]$ the PLR is high for all three systems, as seen in Fig. 4.2. Therefore, if reliable communication is desired the systems should not be operated at these loads anyway.

In Fig. 4.5, we plot the delay pmf, i.e., the probability that a user has a certain delay k, for a system load $g = 0.5$ users/slot. The results show again that overall FA-CSA-FNB provides the best delay performance.

Fig. 4.5 also shows that the maximum delay of the FA systems is larger than that of the FS system, however the probability of such large delays is very low. In practice the maximum delay of the FA systems is limited by the frame length and the memory size of the receiver, whereas the maximum delay of an FS system is strictly limited by the frame length.

Indeed, the maximum delay of FS-CSA is $2n - 1$, whereas the maximum delay of FA-CSA is given by $n + n_{\mathrm{RX}}$. In Figs. 4.4 and 4.5, we considered very large n_{RX} in order to not degrade the PLR performance. With a large n_{RX} the maximum possible delay of FA-CSA can be very large. For applications with strict latency requirements this might be unacceptable. In Figs. 4.6 and 4.7, we therefore present a comparison between FS-CSA and FA-CSA-F with a strict delay constraint of $\delta_{\max}$ slots.

Definition 5. *The PLR of a CSA system with a delay constraint δ_{max}, is the average*

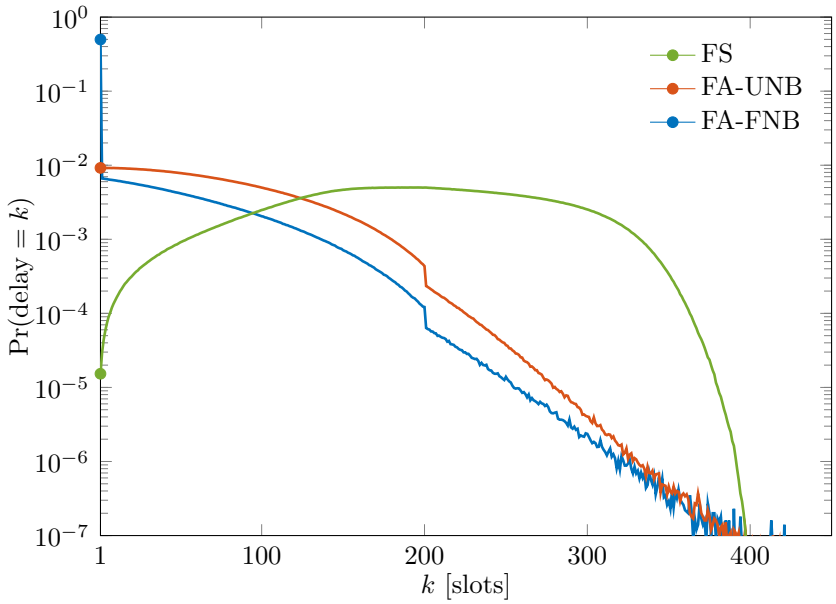


Figure 4.5: The pmf of the delay for FA-CSA-FNB, FA-CSA-UNB and FS-CSA, with $n = 200$ and $\Lambda(x) = \Lambda^\star(x)$ at a system load $g = 0.5$. Dots mark the probability of a 1 slot delay.

probability that an arbitrary user's packet is not resolved within δ_{max} slots of joining the system.

Note that Definition 5 is not a constraint on the memory size n_{RX}. In Fig. 4.6 we depict the delay-constrained PLR (according to Definition 5) of FA-CSA-F with $\delta_{\max} = 200$ and compare it to that of FS-CSA with $n = 100$, using $\Lambda(x) = \Lambda^\star(x)$. We observe that for a given $\delta_{\max}$ it is possible to find a local frame length for FA-CSA-F such that the PLR is strictly better than that of FS-CSA with the same delay constraint. A good choice of n for FA-CSA-F with the maximum delay constraint $\delta_{\max}$ is, in general, half the length of the delay constraint, i.e., $n = \delta_{\max}/2$, as suggested by Fig. 4.6. This choice provides relatively good performance, and outperforms FS-CSA, for all considered system loads.

A large memory can also be costly in practice. Therefore, we make a fair comparison in terms of memory size n_{RX} of FA-CSA-F and FS-CSA. In FS-CSA the only natural choice of memory size is the frame length n, because a decoder gains nothing from capturing more than one frame simultaneously. For FA-CSA, however, a fixed n_{RX} leaves the choice of frame length n open. In Fig. 4.7, we give PLR results for FA-CSA-F with $\Lambda^\star(x)$, $n_{\mathrm{RX}} = 400$ and different frame lengths n. We compare this to the PLR of FS-CSA with $n = 400$. For almost all system loads, it is possible to find an appropriate n for FA-CSA-F, so that it performs better in terms of PLR compared to FS-CSA with the same memory-constraint. The advantage of FA-CSA in terms of memory is that it is more flexible, i.e., for a fixed memory length the frame length can be varied. If memory size is not a constraint, the PLR performance of FA-CSA will be improved by increasing the memory size and adjusting the frame length.

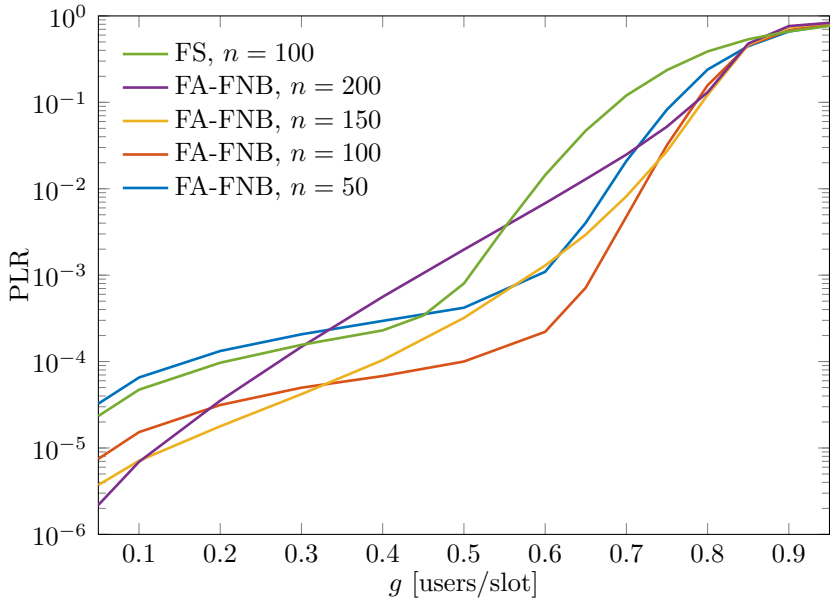


Figure 4.6: PLR performance with a maximum delay constraint $\delta_{\max} = 200$ for FA-CSA-FNB with varying local frame length n and for FS-CSA with $n = 100$, using $\Lambda(x) = \Lambda^{\star}(x)$.

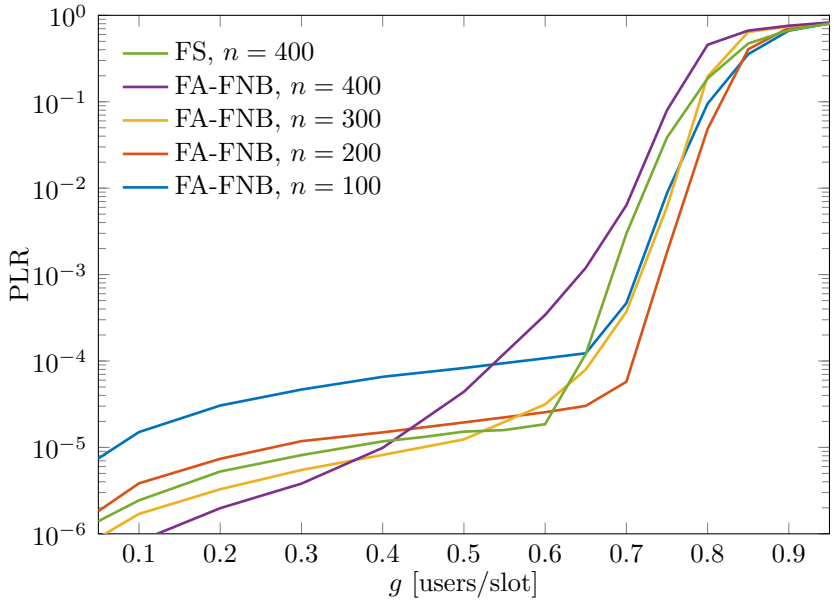


Figure 4.7: PLR performance with a maximum memory constraint $n_{\mathrm{RX}} = 400$ for FA-CSA-FNB with varying local frame length n and for FS-CSA with $n = 400$, using $\Lambda(x) = \Lambda^{\star}(x)$.

4.4 A Comparison With Spatially Coupled Coded Slotted ALOHA

SC-CSA has previously been investigated in [16], where similar improvement of the decoding threshold was observed to that of FA-CSA with boundary effect. SC-CSA is a frame synchronous system, where a degree-l VN connects one edge to a randomly selected CN from each of l consecutive frames. Furthermore, $w + l - 1$ frames are grouped into a *super-frame*. The CNs of the $l - 1$ first and last frames of the super-frame exhibit a lower average degree, creating a boundary effect in both ends of the super-frame.

In [16, Table I] iterative decoding thresholds of SC-CSA are presented for $\Lambda(x) = x^l$ with $l = 2$, 3, 4, 5 and 6. Surprisingly, the thresholds of SC-CSA are identical to the thresholds for FA-CSA-FB and FA-CSA-UB, with the corresponding VN degree distributions, given in Table 4.1. This is remarkable because the systems are quite different, indeed SC-CSA is more structured and enforces the spatially coupled structure, whereas it is inherent to FA-CSA. However, outside of the boundaries, the CN degree distributions of FA-CSA and SC-CSA are identical (again the connectivity of FA-CSA-FB is slightly different). This result suggests that there is a fundamental limit on the graph density for which iterative decoding of CSA can be successfully maintained once a decoding wave has been initiated. It also poses the question of how "small" a boundary can be in order to create a decoding wave that will propagate through the system.

Because of the similarities between SC-CSA and FA-CSA with boundary effect, we present a comparison of FA-CSA-FB, SC-CSA and FS-CSA in the finite frame length regime in terms of PLR and delay.

In order to make a fair comparison of SC-CSA, FA-CSA-FB and FS-CSA, we need to make some modifications to the system model for SC-CSA as it is described in [16]. For our comparison, we consider slot-by-slot Poisson user model, where g is the expected number of users joining per slot. Regular VN-degree distributions are considered, i.e., of the form $\Lambda(x) = x^l$. Furthermore the SC-CSA system is frame-synchronous with frames of length n/l. A user that joins the system in SC-CSA will send one replica in each of the l frames following the frame it joined in. We say that a user is active the whole duration of these l frames. This way, the largest span of a users replicas is equal when comparing SC-CSA, FA-CSA-FB and FS-CSA. Furthermore, we consider a SC-CSA system with boundary effect, i.e., there are no active users in the beginning, meaning that CNs of the first $l - 1$ frames will exhibit lower expected degree than other CNs. We do not terminate the SC-CSA system which would give the CNs of the last $l-1$ frames have lower expected degree too, since this is not done for FA-CSA-FB. Decoding of SC-CSA is performed in the same way as for FA-CSA. Simulation results on the PLR for $n = 120$ and $l = 3$ are depicted in Fig. 4.8 for FA-CSA-FB, SC-CSA and FS-CSA. As expected, the WF performance of FA-CSA-FB and SC-CSA is similar. However, in the EF FA-CSA-FB performs remarkably better than SC-CSA, which has even worse EF than FS-CSA. The reason for this is that in SC-CSA each replica is forced into a smaller frame of size n/l. This makes the probability that two users selects the l same slots for transmission much larger. In fact, the probability can easily be

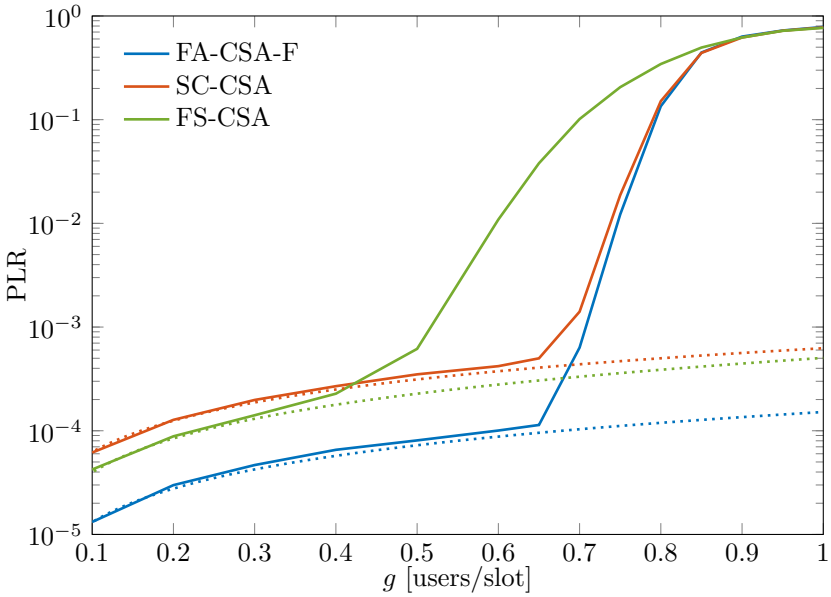


Figure 4.8: Simulated PLR (solid) and EF approximations (dotted) for FA-CSA-FB, FS-CSA and SC-CSA with $n = 120$ and $\Lambda(x) = x^3$.

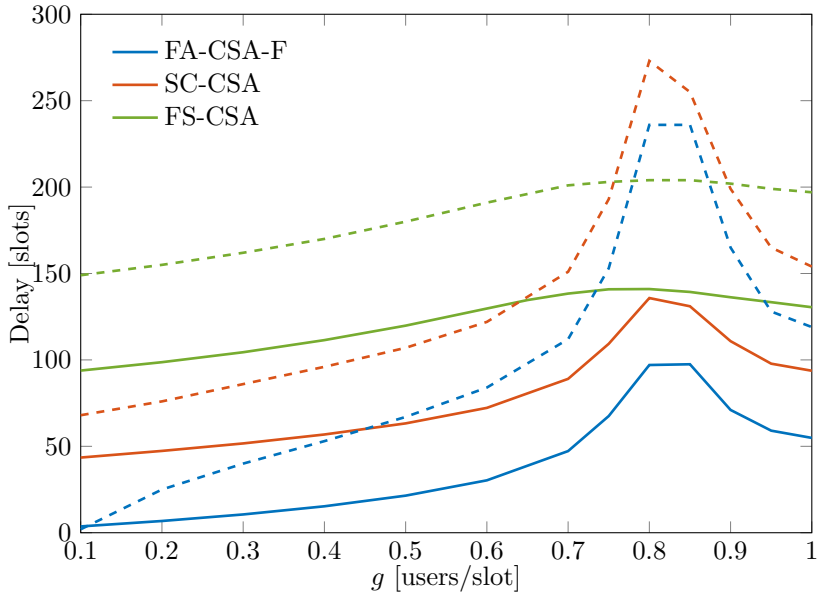


Figure 4.9: Simulated average delay (solid) and 90th percentile delay (dashed) for FA-CSA-FB, FS-CSA and SC-CSA with $n = 120$ and $\Lambda(x) = x^3$.

computed and used as an EF prediction for SC-CSA, as follows,

$$\bar{p}_{\mathrm{SC}} \approx \sum_{m=0}^{\infty} m \left(\frac{l}{n}\right)^{l} \frac{e^{-gn/l}(gn/l)^{m}}{m!} = \left(\frac{l}{n}\right)^{l} \frac{gn}{l}, \tag{4.1}$$

which is plotted with dotted lines in Fig. 4.8.

Additionally, average and 90th percentile curves for the delay are given in Fig. 4.9. For SC-CSA and FA-CSA-FB the delay behavior is similar. Both systems allow a packet to be decoded past the reception of its last replica, which is why the 90th percentile delay is dramatically increased for loads corresponding to the WF-region of the PLR. The delay of FA-CSA-FB is better compared to SC-CSA, because a user that joins the system in SC-CSA needs to wait for the next frame before sending its first replica. Of course, since the time a user in FS-CSA waits after joining before its first replica is sent is even longer, the delay is even worse for FS-CSA.

Chapter 5

Conclusions and Future Work

In this thesis, we considered uncoordinated multiple access using a frame asynchronous adaptation of CSA called FA-CSA. We showed that FA-CSA is a promising technique for scenarios that demand reliable and low latency multiple access in dynamic networks.

The DE equations that characterize the asymptotic behavior of FA-CSA were derived and we showed that systems with so called boundary effect achieve greatly improved decoding thresholds as compared with systems without boundary effect. This is due to a wave-like decoding effect induced at the boundary and is similar to the effect of spatial coupling for LDPC codes. Moreover, we found that FA-CSA has identical decoding threshold to previously considered spatially coupled version of CSA, SC-CSA, where the spatially coupled structure is enforced.

For the finite frame length regime, we derived EF approximations for FA-CSA, based on the probability of occurrence of the most dominant stopping sets. We further showed that FA-CSA is superior to FS-CSA in terms of WF, EF, and delay. Furthermore, we made comparisons of FA-CSA and FS-CSA with constraints on maximum delay and receiver memory length. Because of the similarities of FA-CSA and SC-CSA in the asymptotic regime, we compared their performance also in the finite frame length regime. We showed that, in general, FA-CSA outperforms SC-CSA in terms of both EF and delay, whereas the performance in the WF region is similar.

In future works it would be interesting to investigate the wave-like decoding effect of FA-CSA further. Perhaps it is possible to induce a decoding wave simply by having a load that varies over time, which in many systems can be a realistic model. It would indeed be interesting to find out how large variations of the load are necessary to create a decoding wave. One could also make a more thorough comparison of FA-CSA and FS-CSA in terms of complexity, by limiting the number of decoding iterations and the receiver memory length. Furthermore, it could be possible in practice to consider the capture effect for FA-CSA, i.e., the possibility to decode packets in slots that are of degree larger than 1 with some probability.

Appendix A

List of Minimal Stopping Sets

We give in this appendix a list of the properties of all minimal stopping sets (see Definition 4) with $\mu(\mathcal{S}) \leq 5$. The stopping sets were found by an exhaustive computer search. The quantities $\boldsymbol{v}(\mathcal{S})$, $\nu(\mathcal{S})$, $\mu(\mathcal{S})$, and $c(\mathcal{S})$ are defined at the beginning of Section 3.3.

Notation	$\boldsymbol{v}(\mathcal{S})$	$\nu(\mathcal{S})$	$\mu(\mathcal{S})$	$c(\mathcal{S})$
$\mathcal{S}_1$	[0, 2, 0, 0, 0, 0]	2	1	1
$\mathcal{S}_2$	[0, 0, 2, 0, 0, 0]	2	2	1
$\mathcal{S}_3$	[0, 2, 1, 0, 0, 0]	3	2	2
$\mathcal{S}_4$	[0, 0, 0, 2, 0, 0]	2	3	1
$\mathcal{S}_5$	[0, 1, 1, 1, 0, 0]	3	3	3
$\mathcal{S}_6$	[0, 0, 3, 0, 0, 0]	3	3	6
$\mathcal{S}_7$	[0, 0, 2, 1, 0, 0]	3	3	6
$\mathcal{S}_8$	[0, 3, 0, 1, 0, 0]	4	3	6
$\mathcal{S}_9$	[0, 2, 2, 0, 0, 0]	4	3	12
$\mathcal{S}_{10}$	[0, 0, 0, 0, 2, 0]	2	4	1
$\mathcal{S}_{11}$	[0, 1, 0, 1, 1, 0]	3	4	4
$\mathcal{S}_{12}$	[0, 0, 2, 0, 1, 0]	3	4	6
$\mathcal{S}_{13}$	[0, 0, 1, 2, 0, 0]	3	4	12
$\mathcal{S}_{14}$	[0, 0, 1, 1, 1, 0]	3	4	12
$\mathcal{S}_{15}$	[0, 0, 0, 3, 0, 0]	3	4	24
$\mathcal{S}_{16}$	[0, 0, 0, 2, 1, 0]	3	4	12
$\mathcal{S}_{17}$	[0, 2, 1, 0, 1, 0]	4	4	12
$\mathcal{S}_{18}$	[0, 2, 0, 2, 0, 0]	4	4	24
$\mathcal{S}_{19}$	[0, 1, 2, 1, 0, 0]	4	4	24
$\mathcal{S}_{20}$	[0, 1, 2, 1, 0, 0]	4	4	24
$\mathcal{S}_{21}$	[0, 0, 4, 0, 0, 0]	4	4	72
$\mathcal{S}_{22}$	[0, 1, 2, 0, 1, 0]	4	4	24
$\mathcal{S}_{23}$	[0, 1, 1, 2, 0, 0]	4	4	48
$\mathcal{S}_{24}$	[0, 0, 3, 1, 0, 0]	4	4	144
$\mathcal{S}_{25}$	[0, 0, 3, 1, 0, 0]	4	4	24
$\mathcal{S}_{26}$	[0, 0, 3, 0, 1, 0]	4	4	24
$\mathcal{S}_{27}$	[0, 0, 2, 2, 0, 0]	4	4	48
$\mathcal{S}_{28}$	[0, 0, 2, 2, 0, 0]	4	4	48
$\mathcal{S}_{29}$	[0, 4, 0, 0, 1, 0]	5	4	24
$\mathcal{S}_{30}$	[0, 3, 1, 1, 0, 0]	5	4	72
$\mathcal{S}_{31}$	[0, 2, 3, 0, 0, 0]	5	4	144
$\mathcal{S}_{32}$	[0, 0, 0, 0, 0, 2]	2	5	1
$\mathcal{S}_{33}$	[0, 1, 0, 0, 1, 1]	3	5	5
$\mathcal{S}_{34}$	[0, 0, 1, 1, 0, 1]	3	5	10

$\mathcal{S}_{35}$	[0, 0, 1, 0, 2, 0]	3	5	20
$\mathcal{S}_{36}$	[0, 0, 0, 2, 1, 0]	3	5	30
$\mathcal{S}_{37}$	[0, 0, 1, 0, 1, 1]	3	5	20
$\mathcal{S}_{38}$	[0, 0, 0, 2, 0, 1]	3	5	30
$\mathcal{S}_{39}$	[0, 0, 0, 1, 2, 0]	3	5	60
$\mathcal{S}_{40}$	[0, 0, 0, 1, 1, 1]	3	5	30
$\mathcal{S}_{41}$	[0, 0, 0, 0, 3, 0]	3	5	60
$\mathcal{S}_{42}$	[0, 0, 0, 0, 2, 1]	3	5	20
$\mathcal{S}_{43}$	[0, 2, 0, 1, 0, 1]	4	5	20
$\mathcal{S}_{44}$	[0, 2, 0, 0, 2, 0]	4	5	40
$\mathcal{S}_{45}$	[0, 1, 2, 0, 0, 1]	4	5	30
$\mathcal{S}_{46}$	[0, 1, 1, 1, 1, 0]	4	5	20
$\mathcal{S}_{47}$	[0, 1, 1, 1, 1, 0]	4	5	30
$\mathcal{S}_{48}$	[0, 1, 1, 1, 1, 0]	4	5	60
$\mathcal{S}_{49}$	[0, 1, 0, 3, 0, 0]	4	5	180
$\mathcal{S}_{50}$	[0, 0, 3, 0, 1, 0]	4	5	180
$\mathcal{S}_{51}$	[0, 0, 2, 2, 0, 0]	4	5	120
$\mathcal{S}_{52}$	[0, 0, 2, 2, 0, 0]	4	5	60
$\mathcal{S}_{53}$	[0, 0, 2, 2, 0, 0]	4	5	120
$\mathcal{S}_{54}$	[0, 1, 1, 1, 0, 1]	4	5	60
$\mathcal{S}_{55}$	[0, 1, 1, 0, 2, 0]	4	5	120
$\mathcal{S}_{56}$	[0, 1, 0, 2, 1, 0]	4	5	120
$\mathcal{S}_{57}$	[0, 1, 0, 2, 1, 0]	4	5	120
$\mathcal{S}_{58}$	[0, 0, 3, 0, 0, 1]	4	5	180
$\mathcal{S}_{59}$	[0, 0, 2, 1, 1, 0]	4	5	120
$\mathcal{S}_{60}$	[0, 0, 2, 1, 1, 0]	4	5	240
$\mathcal{S}_{61}$	[0, 0, 2, 1, 1, 0]	4	5	120
$\mathcal{S}_{62}$	[0, 0, 2, 1, 1, 0]	4	5	120
$\mathcal{S}_{63}$	[0, 0, 2, 1, 1, 0]	4	5	60
$\mathcal{S}_{64}$	[0, 0, 1, 3, 0, 0]	4	5	360
$\mathcal{S}_{65}$	[0, 0, 1, 3, 0, 0]	4	5	360
$\mathcal{S}_{66}$	[0, 0, 1, 3, 0, 0]	4	5	360
$\mathcal{S}_{67}$	[0, 1, 0, 2, 0, 1]	4	5	60
$\mathcal{S}_{68}$	[0, 1, 0, 1, 2, 0]	4	5	120
$\mathcal{S}_{69}$	[0, 0, 2, 1, 0, 1]	4	5	60
$\mathcal{S}_{70}$	[0, 0, 2, 1, 0, 1]	4	5	120
$\mathcal{S}_{71}$	[0, 0, 2, 0, 2, 0]	4	5	120
$\mathcal{S}_{72}$	[0, 0, 2, 0, 2, 0]	4	5	240
$\mathcal{S}_{73}$	[0, 0, 1, 2, 1, 0]	4	5	120
$\mathcal{S}_{74}$	[0, 0, 1, 2, 1, 0]	4	5	120
$\mathcal{S}_{75}$	[0, 0, 1, 2, 1, 0]	4	5	240
$\mathcal{S}_{76}$	[0, 0, 1, 2, 1, 0]	4	5	120
$\mathcal{S}_{77}$	[0, 0, 1, 2, 1, 0]	4	5	120
$\mathcal{S}_{78}$	[0, 0, 1, 2, 1, 0]	4	5	240
$\mathcal{S}_{79}$	[0, 0, 1, 2, 1, 0]	4	5	120
$\mathcal{S}_{80}$	[0, 0, 0, 4, 0, 0]	4	5	240
$\mathcal{S}_{81}$	[0, 0, 0, 4, 0, 0]	4	5	360
$\mathcal{S}_{82}$	[0, 0, 0, 4, 0, 0]	4	5	1440
$\mathcal{S}_{83}$	[0, 0, 1, 2, 0, 1]	4	5	120
$\mathcal{S}_{84}$	[0, 0, 1, 1, 2, 0]	4	5	240
$\mathcal{S}_{85}$	[0, 0, 1, 1, 2, 0]	4	5	120
$\mathcal{S}_{86}$	[0, 0, 0, 3, 1, 0]	4	5	720
$\mathcal{S}_{87}$	[0, 0, 0, 3, 1, 0]	4	5	360
$\mathcal{S}_{88}$	[0, 0, 0, 3, 1, 0]	4	5	120
$\mathcal{S}_{89}$	[0, 0, 0, 3, 0, 1]	4	5	60
$\mathcal{S}_{90}$	[0, 0, 0, 2, 2, 0]	4	5	120

$\mathcal{S}_{91}$	[0, 0, 0, 2, 2, 0]	4	5	240
$\mathcal{S}_{92}$	[0, 3, 1, 0, 0, 1]	5	5	60
$\mathcal{S}_{93}$	[0, 3, 0, 1, 1, 0]	5	5	180
$\mathcal{S}_{94}$	[0, 2, 2, 0, 1, 0]	5	5	240
$\mathcal{S}_{95}$	[0, 2, 2, 0, 1, 0]	5	5	120
$\mathcal{S}_{96}$	[0, 2, 1, 2, 0, 0]	5	5	240
$\mathcal{S}_{97}$	[0, 2, 1, 2, 0, 0]	5	5	120
$\mathcal{S}_{98}$	[0, 2, 1, 2, 0, 0]	5	5	240
$\mathcal{S}_{99}$	[0, 1, 3, 1, 0, 0]	5	5	360
$\mathcal{S}_{100}$	[0, 1, 3, 1, 0, 0]	5	5	360
$\mathcal{S}_{101}$	[0, 1, 3, 1, 0, 0]	5	5	360
$\mathcal{S}_{102}$	[0, 0, 5, 0, 0, 0]	5	5	1440
$\mathcal{S}_{103}$	[0, 2, 2, 0, 0, 1]	5	5	120
$\mathcal{S}_{104}$	[0, 2, 1, 1, 1, 0]	5	5	240
$\mathcal{S}_{105}$	[0, 2, 1, 1, 1, 0]	5	5	120
$\mathcal{S}_{106}$	[0, 2, 0, 3, 0, 0]	5	5	720
$\mathcal{S}_{107}$	[0, 1, 3, 0, 1, 0]	5	5	360
$\mathcal{S}_{108}$	[0, 1, 3, 0, 1, 0]	5	5	720
$\mathcal{S}_{109}$	[0, 1, 3, 0, 1, 0]	5	5	120
$\mathcal{S}_{110}$	[0, 1, 2, 2, 0, 0]	5	5	480
$\mathcal{S}_{111}$	[0, 1, 2, 2, 0, 0]	5	5	480
$\mathcal{S}_{112}$	[0, 1, 2, 2, 0, 0]	5	5	480
$\mathcal{S}_{113}$	[0, 1, 2, 2, 0, 0]	5	5	480
$\mathcal{S}_{114}$	[0, 1, 2, 2, 0, 0]	5	5	240
$\mathcal{S}_{115}$	[0, 0, 4, 1, 0, 0]	5	5	2880
$\mathcal{S}_{116}$	[0, 0, 4, 1, 0, 0]	5	5	1440
$\mathcal{S}_{117}$	[0, 0, 4, 1, 0, 0]	5	5	1440
$\mathcal{S}_{118}$	[0, 1, 3, 0, 0, 1]	5	5	120
$\mathcal{S}_{119}$	[0, 1, 2, 1, 1, 0]	5	5	120
$\mathcal{S}_{120}$	[0, 1, 2, 1, 1, 0]	5	5	240
$\mathcal{S}_{121}$	[0, 1, 2, 1, 1, 0]	5	5	240
$\mathcal{S}_{122}$	[0, 1, 1, 3, 0, 0]	5	5	720
$\mathcal{S}_{123}$	[0, 1, 1, 3, 0, 0]	5	5	720
$\mathcal{S}_{124}$	[0, 0, 4, 0, 1, 0]	5	5	1440
$\mathcal{S}_{125}$	[0, 0, 4, 0, 1, 0]	5	5	120
$\mathcal{S}_{126}$	[0, 0, 3, 2, 0, 0]	5	5	1440
$\mathcal{S}_{127}$	[0, 0, 3, 2, 0, 0]	5	5	1440
$\mathcal{S}_{128}$	[0, 0, 3, 2, 0, 0]	5	5	720
$\mathcal{S}_{129}$	[0, 0, 3, 2, 0, 0]	5	5	720
$\mathcal{S}_{130}$	[0, 0, 3, 2, 0, 0]	5	5	1440
$\mathcal{S}_{131}$	[0, 0, 3, 2, 0, 0]	5	5	720
$\mathcal{S}_{132}$	[0, 0, 4, 0, 0, 1]	5	5	120
$\mathcal{S}_{133}$	[0, 0, 3, 1, 1, 0]	5	5	360
$\mathcal{S}_{134}$	[0, 0, 3, 1, 1, 0]	5	5	360
$\mathcal{S}_{135}$	[0, 0, 2, 3, 0, 0]	5	5	720
$\mathcal{S}_{136}$	[0, 0, 2, 3, 0, 0]	5	5	1440
$\mathcal{S}_{137}$	[0, 5, 0, 0, 0, 1]	6	5	120
$\mathcal{S}_{138}$	[0, 4, 1, 0, 1, 0]	6	5	480
$\mathcal{S}_{139}$	[0, 4, 0, 2, 0, 0]	6	5	720
$\mathcal{S}_{140}$	[0, 3, 2, 1, 0, 0]	6	5	720
$\mathcal{S}_{141}$	[0, 3, 2, 1, 0, 0]	6	5	720
$\mathcal{S}_{142}$	[0, 2, 4, 0, 0, 0]	6	5	2880

Bibliography

[1] A. Goldsmith, *Wireless communications*. Cambridge University Press, 2005, vol. 1, ch. 14, pp. 452–504.

[2] T. S. Rappaport, *Wireless communications: principles and practice*. Prentice Hall, 2002, vol. 2, ch. 14, pp. 447–490.

[3] N. Abramson, "The ALOHA system - Another alternative for computer communications," in *Proc. 1970 Fall Joint Computer Conf.*, vol. 37.

[4] L. G. Roberts, "ALOHA packet system with and without slots and capture," in *ARPANET System Note 8 (NIC11290)*, June 1972.

[5] S. Lin, D. J. Costello, Jr., and M. J. Miller, "Automatic-repeat-request error-control schemes," *IEEE Commun. Mag.*, vol. 22, no. 12, pp. 5 – 17, Dec. 1984.

[6] IEEE Standards Association, "802.11-2012 - IEEE Standard for Information technology- Telecommunications and information exchange between systems Local and metropolitan area networks- Specific requirements: Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," Mar. 2012.

[7] Mikael Fallgren and Bogdan Timus (editors), "Scenarios, requirements and KPIs for 5G mobile and wireless system," METIS deliverable D1.1, 2013.

[8] E. Casini, R. De Gaudenzi, and O. R. Herrero, "Contention resolution diversity slotted ALOHA (CRDSA): an enhanced random access scheme for satellite access packet networks," *IEEE Trans. Wir. Commun.*, vol. 6, no. 4, pp. 1408–1419, Apr. 2007.

[9] G. Liva, "Graph-based analysis and optimization of contention resolution diversity slotted ALOHA," *IEEE Trans. Commun.*, vol. 59, no. 2, pp. 477–487, Feb. 2011.

[10] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, 2008, vol. 1.

[11] W. E. Ryan and S. Lin, *Channel Codes: Classical and Modern*. Cambridge University Press, 2009.

[12] K. R. Narayanan and H. D. Pfister, "Iterative collision resolution for slotted ALOHA: An optimal uncoordinated transmission policy," in *Proc. Int. Symp. on Turbo Codes & Iterative Information Processing (ISTC)*, Gothenburg, Sweden, Aug. 2012.

[13] M. Ivanov, F. Brännström, A. Graell i Amat, and P. Popovski, "Error floor analysis of coded slotted ALOHA over packet erasure channels," *IEEE Commun. Lett.*, vol. 19, no. 3, pp. 419–422, Mar. 2015.

[14] ——, "Broadcast Coded Slotted ALOHA: a finite frame length analysis," May 2016. [Online]. Available: http://arxiv.org/abs/1511.00418

[15] E. Paolini, G. Liva, and M. Chiani, "Coded slotted ALOHA: a graph-based method for uncoordinated multiple access," *IEEE Trans. Inf. Theory*, vol. 61, no. 12, pp. 6815–6832, Dec. 2015.

[16] G. Liva, E. Paolini, M. Lentmaier, and M. Chiani, "Spatially-coupled random access on graphs," in *Proc. IEEE Int. Symp. on Information Theory (ISIT)*, Cambridge, MA, July 2012.

[17] D. J. Costello, Jr., L. Dolecek, T. E. Fuja, J. Kliewer, D. G. M. Mitchell, and R. Smarandache, "Spatially coupled sparse codes on graphs: Theory and practice," *IEEE Commun. Mag.*, vol. 52, no. 7, pp. 168 – 176, Jul. 2014.

[18] A. Meloni, M. Murroni, C. Kissling, and M. Berioli, "Sliding window-based contention resolution diversity slotted ALOHA," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Anaheim, CA, Dec. 2012.

[19] R. De Gaudenzi, O. del Rio Herrero, G. Acar, and E. Garrido Barrabes, "Asynchronous contention resolution diversity ALOHA: Making CRDSA truly asynchronous," *IEEE Trans. Wir. Commun.*, vol. 13, no. 11, pp. 6193 – 6206, July 2014.

[20] O. del Rio Herrero and R. De Gaudenzi, "Generalized analytical framework for the performance assessment of slotted random access protocols," *IEEE Trans. Wir. Commun.*, vol. 13, no. 2, pp. 809 – 821, Jan. 2014.

[21] E. Sandgren, A. Graell i Amat, and F. Brännström, "Asymptotic and finite frame length analysis of frame asynchronous coded slotted ALOHA," in *Proc. Int. Symp. on Turbo Codes & Iterative Information Processing (ISTC)*, Brest, France, Sept. 2016. [Online]. Available: https://arxiv.org/abs/1604.06293

[22] ——, "On frame asynchronous coded slotted ALOHA: Asymptotic, finite length, and delay analysis," June 2016. [Online]. Available: https://arxiv.org/abs/1606.03242

[23] J. A. Bondy and U. S. R. Murty, *Graph theory with applications*. New York: Elsevier, 1976.