ADMA: Amplitude-Division Multiple Access for Bacterial Communication Networks

Bhuvana Krishnaswamy, Yubing Jian, Caitlin M. Austin, Jorge E. Perdomo, Sagar C. Patel,
Brian K. Hammer, Craig R. Forest, and Raghupathy Sivakumar

***Abstract*—In a communication network with extremely high processing delays, an efficient addressing and multiple-access control mechanism to improve the throughput performance of the system is a necessity. This paper focuses on source addressing in multiple-source single-receiver bacterial communication networks. We propose amplitude-division multiple access (ADMA), a method that assigns the amplitude of the transmitted signal as the address of the source. We demonstrate using genetically engineered *Escherichia coli* bacteria in a microfluidic device that using amplitude for addressing is feasible. We analyze the performance of the network with several addressing mechanisms and propose an amplitude sequence and a low-complexity receiver design that minimizes error in resolving the source addresses in the presence of collisions. Finally, we demonstrate that ADMA implicitly solves the problem of multiple-access control.**

***Index Terms*—Molecular communication, embedded addressing, medium access control.**

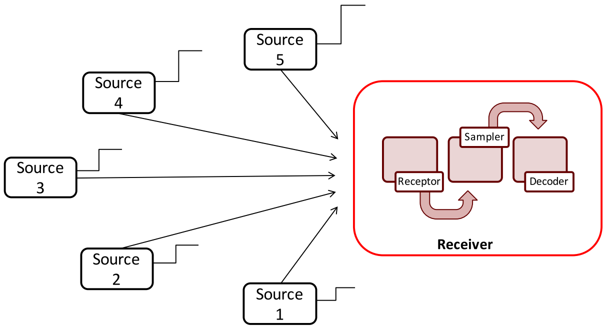


Fig. 1. Network Setup.

I. Introduction

NANO-SCALE communication can be categorized into two broad domains: *electromagnetic communication (EM)* at the nanoscale, which involves the extension of EM-based communication techniques for use in nonbiological applications [1], [2]; and *molecular communication (MC)*, which involves strategies for use in biological applications [3]–[5]. Advancements in synthetic biology and nanotechnology now make MC possible and practical for applications like toxicology, environmental monitoring [6], and drug delivery [7].

The ability to genetically engineer bacteria to introduce or delete DNA for specific traits (e.g., bioluminescence, motility, adhesion), and their natural properties for communication [8], [9] have made bacteria emerge as candidates for nanomachines [10]. Bacterial nano-machines hold promise for use in biological applications, including toxicology, biofueling, and biosensing. For example, receiver bacteria are used as biosensors to detect the presence of metals [10], arsenic pollution [11], environmental monitoring [6], and drug delivery [7].

The context for this work is molecular communication between bacterial populations. Specifically, we consider bacterial populations as transceivers connected through microfluidic pathways that act as conduits for molecular signals. Prior research in MC have focused on channel and system modeling [12], [13], capacity derivation [14]–[16], modulation techniques [17]–[19] and analysis of channel and inter symbol interference [16], [20], [21]. These studies focus on a single link and do not consider the challenges in implementing the algorithms in a real-life environment. Our work focuses on a star topology with multiple sources and a single receiver, as shown in Figure 1. This topology is most commonly seen in sensor networks, where multiple sensors communicate with a single receiver/sink. The sensors broadcast information and do not require a destination address. The receiver, on the other hand, receives a cumulative signal from multiple sources, making it necessary to have an efficient addressing mechanism that uniquely identifies the sources.

In this paper, we make the following five contributions:

1) We propose an addressing mechanism, *Amplitude-Division Multiple Access/Addressing (ADMA)*, for a topology with multiple sources and a single receiver. ADMA embeds the address of each source in the amplitude of its corresponding transmitted signal by assigning a unique amplitude for each source. We prove that a

Manuscript received May 16, 2017; revised October 16, 2017 and December 21, 2017; accepted January 3, 2018. Date of publication January 9, 2018; date of current version June 5, 2018. The associate editor coordinating the review of this paper and approving it for publication was S. M. Moser. *(Corresponding author: Bhuvana Krishnaswamy.)*

B. Krishnaswamy, Y. Jian, and R. Sivakumar are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: bkrishnaswamy3@gatech.edu).

C. M. Austin, J. E. Perdomo, S. C. Patel, and C. R. Forest are with the George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA.

B. K. Hammer is with the School of Biology, Georgia Institute of Technology, Atlanta, GA 30332 USA.

Digital Object Identifier 10.1109/TMBMC.2018.2791448

careful assignment of the source amplitudes can implicitly solve the multiple-access control problem, and allow the receiver to uniquely identify the sources.

2) We demonstrate the feasibility of ADMA using *Escherichia coli (E. coli)* bacteria genetically engineered to exhibit fluorescence on the receipt of a specific signal molecule (N-(3-Oxyhexanoyl)-L-homoserine lactone, or C6-HSL). We show that the response of a receiver bacterial population is significantly different for distinct input amplitudes. We also demodulate the receiver response obtained from the experiments and evaluate the performance of *ADMA* in a practical system.
3) We propose an optimum, low-complexity receiver design that maximizes the network throughput in a multiple access channel. We then derive a theoretical upper bound for the throughput performance of *amplitude source addressing* under the proposed receiver designs.
4) We propose a heuristic algorithm that selects an address sequence for minimizing the collision resolution error for a given number of sources in the network.
5) We implement the proposed receiver design and amplitude assignment algorithm in *nanoNS3*, a bacterial communication simulator built on top of NS3. nanoNS3 simulates the response of receiver bacteria as modeled in [22] and introduces channel and receiver errors to represent a practical system.

The remainder of the paper is organized as follows: Section II presents the advantages and disadvantages of different source addressing mechanisms and provides experimental validation of the amplitude source addressing. Section III details our proposed optimal source addressing mechanism. Section IV describes two receiver designs, and provides an upper bound on the maximum number of successful bits. Section V describes our proposed address assignment algorithm that minimizes interference in a dense network. Section VI presents the simulation results of the performance of amplitude sequences proposed. Section VII concludes the paper by discussing some of the challenges and future work.

II. The Addressing Problem

A. Application Scenario

We discuss the communication modules required for a practical bacterial communication network with the help of a specific application: pathogen-detection using bacterial sensors. *Transmitter bacteria* transmit information sensed by *sensor bacteria*, that are genetically engineered to detect specific pathogens to a receiver (sink). The transmitter needs a modulation scheme to communicate the information to the receiver through a channel or medium. Multiple sensors are deployed to detect different pathogens, and communicate with the receiver via point-to-point channels. Therefore, an addressing scheme that will uniquely identify each source and a multiple access control mechanism that will allow the sources to share the single receiver are also required.

Modulation techniques specifically targeting MC have been developed by a number of researchers [16]–[18]. Concentration shift keying [17] and molecule shift keying [16], [18] are two well-known methods that encode information in the concentration (amplitude) of the signal and the molecule type respectively. A majority of the existing research on MC has focused on channel and system modeling, and algorithms for a single link [14], [15], [19]. With the growth of network sizes, there is a need for algorithms that perform addressing, medium access control (MAC), routing and reliability. In [23], a distance-based addressing that estimates the distance between transmitter and receiver using beacons, propagation delay, and path loss is proposed. Reference [23] establishes a coordinate system from the distances measured and the molecules move to the desired location. This addressing mechanism assumes that the transmitter can identify its location in the channel and guide molecules to a particular direction. Developing such a transmitter using biological circuits is highly challenging. Extending this scheme to more than one link will be challenging as it requires accurate channel estimation. Reference [21] proposes a MAC using *molecule type*, wherein each transmitter communicates with a distinct molecule, and thus do not interfere with other transmitters, analogous to frequency division multiplexing. We explore this approach in detail in the following section.

In this paper, we focus on *source addressing*, an addressing mechanism that can distinguish multiple sources communicating with a single receiver in a star topology as shown in Figure 1. We consider a system where sources transmit chemical molecules that propagate through a microfluidic channel and trigger the receiver to generate Green Fluorescent Protein (GFP), which is then sampled, demodulated and decoded. Each source uses On-Off-Keying (OOK), a simple modulation technique that transmits a rectangular pulse $m(t)$, as shown in Equation (1), with concentration A for T seconds (bit period) to transmit bit 1 and an absence of signal for T seconds to transmit bit 0.

$$m(t) = \begin{cases} A & 0 \leq t \leq T \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

B. Types of Addressing

1) Address Fields: In traditional communication systems, it is common to allocate a fixed number of bits in the packet header for addressing, e.g., the IP address and MAC address fields. In bacterial communication networks, on the other hand, very high processing delays at the transceiver nodes result in extremely low data rates (order of 10^{-5} bits per second [24]). Overheads in the form of address fields can result in additional per-frame delays as well as a decreased per-user throughput.

With the use of modulation techniques such as concentration shift keying [17], the number of symbols (signals) required to transmit an address can be reduced. However, a significant reduction in the network throughput cannot be avoided without fully eliminating the use of address fields. We propose *Embedded Addressing*, an addressing mechanism that eliminates the need for address fields by embedding the source address in the transmitted signals. It uses the unique characteristics of molecular signals to identify the senders (sources).

TABLE I
SOURCE ADDRESSING MECHANISMS

Addressing Mechanism	Strength	Weakness
Address Fields	Global Address	Increased Delays, Reduced Throughput
Molecule Type	Increased Throughput, Fair	Not Scalable, Complex Receiver Circuits
Pulse Duration	Simple Transceiver Circuits	Reduced Throughput, Introduces Unfairness
Amplitude	Simple Transceiver Circuits, Scalable, Fair	Network Size Limited by Receiver

This is a local addressing mechanism, i.e., the address embedded is local to the network and not the sender's unique global address. We assume that each source has already been assigned a unique global address.[1] The receiver uses only the received signal and its knowledge of the characteristics of the source signals to identify them locally. It is worth mentioning here that code division multiple access (CDMA) is also a technique that uses embedded addressing by assigning a unique pseudo-random spreading code to each source. However, it does not provide a solution for slow networks (that are common in MC) because of its use of spreading codes which further decreases the data rate of the network.

In MC, we identify the following key characteristics of a molecular signal that allow us to embed the source addresses in the transmitted signals without compromising the network throughput: *molecule type*, *signal duration*, and *signal amplitude*. We next elaborate on each of these properties.

2) Molecule Type: The address is embedded in the signaling *molecule type* with the same amplitude and duration for all sources. Each source is assigned a unique molecule, and the receiver must be capable of receiving all the molecules, allowing all the sources equal access to the receiver without contention. Thus, source addressing with molecule type also solves the MAC problem.

The other two characteristics, pulse amplitude and duration, allow all the sources to transmit the same molecule. The receiver accepts only one type of molecule, simplifying the receiver and transmitter designs.

3) Pulse Duration: The amplitude of the signal and the molecule type are fixed across sources, with the address embedded in the signal duration. Each source is assigned a unique pulse duration. When a source has bit 1 to transmit, it transmits a signal with a given amplitude and the duration assigned to it as the address. The sources are assigned distinct durations, which leads to increased latency. The per-frame delay of each source is different from the others, leading to unfair throughput and increased network delays.

4) Signal Amplitude: In electromagnetic communication, amplitudes have been used for modulating the signal. Here, we consider using amplitudes as the address. Each source is assigned a unique amplitude, and transmits its signal with the assigned amplitude for a duration fixed for the network. The receiver maps the received amplitude to the respective source. When multiple sources transmit at the same time, the receiver receives the sum of amplitudes. To identify each source, the receiver must determine the individual amplitudes for all received sums.

Table I summarizes the strengths and weaknesses of the above mechanisms, and we conclude that embedding the address in the amplitude of the signal is the most efficient among the four source addressing mechanisms.

C. Experimental Validation of Amplitude Differentiation

Due to the challenges in the design of state-of-the-art microfluidic system [25], the proof-of-concept presented in this paper extrapolates results from a system with one source and one receiver. We extrapolate experimental validation in the following steps. Using the experimental setup, we verify that a receiver can distinguish between amplitudes from the GFP response of the receiver bacteria. Reference [22] proved experimentally that the response of the receiver bacteria is distinct for distinct input amplitudes. We use the model developed in [22] that was validated with experimental results to model the response of receiver bacteria to a molecular signal.

In this work, we implement the numerical inverse of the above model as a demodulator. In the *inverse model*, the receiver response is the input and an estimate of the molecular signal that triggered this response is the output. We use a model that was experimentally validated to generate the response of receiver and an inverse module that demodulates the signal from the response of the receiver. We assume N parallel point-to-point channels from N sources to 1 receiver as shown in Figure 1. Therefore, the molecular signal reaching the receiver is the sum of amplitudes of the signals transmitted by each source. The cumulative signal from the individual sources becomes the input to the receiver model. The response of the receiver to the cumulative signal simulates a multiple-source-single-receiver topology. The modulation and channel error of each channel is handled individually. In this work, we consider a system in which genetically-engineered bacterial populations are receivers connected through microfluidic pathways. Microfluidic pathways allow for dynamic changes in media composition. The constant stream of media keeps the bacteria in ideal growth conditions, eliminating growth phase dependent variables from the experiments. The bacteria density does not change during the experiment. The bacteria are first seeded into the trapping chambers and grow until the chamber is filled. The chamber is in direct fluidic contact with the main channel, which has constant flow providing nutrients to the bacteria and removing excess bacteria. This means the density of the chamber contents remains constant.

The bacteria used in all experiments were a genetically-engineered strain of DH5α *E. coli*. Methods and functionality of the bacteria and the microfluidic device fabrication and specifications can be seen in previous works [22], [24], but

[1] Assignment of global addresses is out of the scope of this work.

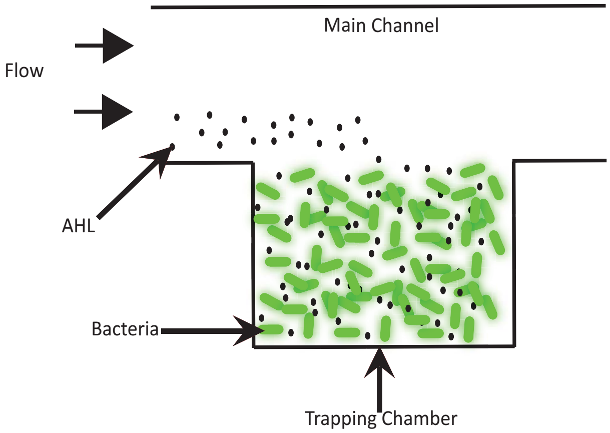


Fig. 2. The experimental setup consists of microfluidic channels in direct fluidic contact with trapping chambers housing bacteria. The main channel provides nutrients and AHL to the bacteria.

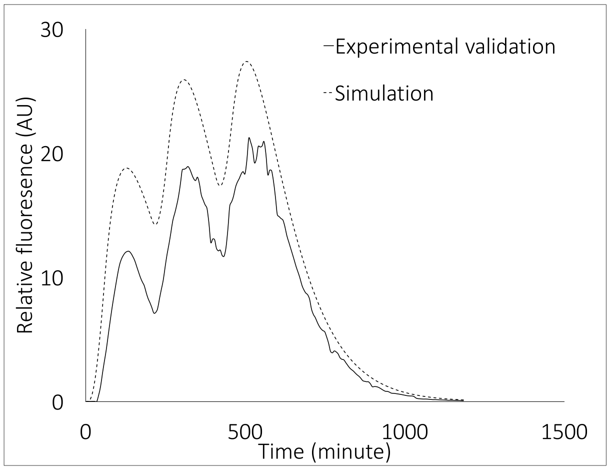


Fig. 3. Bacterial Receiver Response.

will be briefly mentioned here. In this system, the chemical stimulus is autoinducer N-acyl homoserine lactone (AHL), and the bacterial response is the expression of GFP. To fabricate the microfluidic devices, we utilize standard soft lithography [26] resulting in a polydimethylsiloxane (PDMS) device bonded to a glass coverslip. The design consists of the main channel which has direct fluidic contact with adjacent chambers that house the bacteria for the duration of the experiment, as seen in Figure 2. Figure 2 shows the receiver bacteria. The "Main Channel" is the channel through which molecules carry information. After initial loading of the chip, the bacteria were allowed to populate the chambers for 24 hours to reach capacity. During this time the bacteria were supplied with a constant flow of 2xYT lysogeny broth (LB) at 100 $\mu l/hr$. Once the bacteria had filled the chamber, flow rate was increased to 360 $\mu l/hr$. One syringe was used for LB (at 350 $\mu l/hr$), while the second (10 $\mu l/hr$) was used to deliver varying concentrations of AHL. An "AHL pulse" is the duration and concentration over which this AHL was delivered to the bacteria chamber. Pulse durations of 50 minutes that we optimized [24] were used for "on" or "bit 1" state; while 50 minutes of no AHL equated to "off" or "bit 0" state. Unless otherwise mentioned, we use the above pulse concentration and duration throughout the paper. Fluorescent images were captured once every 10 minutes and post-processed using MATLAB. The intensity of the pixels within the bacteria chamber was averaged and the background fluorescence was subtracted, yielding relative fluorescence (arbitrary units, or AU). Four dynamically programmed signals of *on* and *off* states were delivered to the receiver bacteria populations while their fluorescent outputs were recorded. The following input bit patterns were tested: 1010101010, 1000100010, 1010000010, and 0000000000. The bit patterns are chosen to represent different probabilities of bit 1. Utilizing the receiver response model developed in [22], we compared the experimental results to the predicted response and found that the model captures the dynamics well. Figure 3 shows the response of the receiver bacteria from experiments and simulations to an input of 1000100010 in one trial.

To further test the model's capabilities to demodulate the received signal, we used the experimental GFP results to decode the original AHL input signal by using the inverse of the model proposed in [22]. The decoding efficiencies of the model are, respectively: 90%, 100%, 80%, 100%. Each bit pattern was repeated four times, and the average decoding efficiency we present here is the average over the four repetitions. These experiments are time intensive and challenging due to the nature of bacteria. There are numerous factors that affect the ability of bacteria to process the AHL input and produce the GFP output. We have made great strides in controlling several of these factors, such as temperature, population size, and nutrients, by utilizing a microfluidic device, but several remain out of our control. These uncontrollable factors can fluctuate and cause small variations in the bacterial response, which can lead to lower decoding efficiencies. This is most likely the cause for the 80% decoding efficiency.

III. Amplitude-Division Multiple Access

A. Problem Definition

As summarized in Table I, relying on *Address Fields*, *Pulse Duration*, and *Molecule Type* as addressing mechanisms can result in increased network delays, decreased throughput, complex receiver circuits, and throughput unfairness. On the other hand, embedding address in the amplitude of the signal is both fair and throughput friendly, but the maximum number of users is limited by the number of amplitude levels the receiver can distinguish. In the arguments above, we show that embedding the address in the amplitude is best suited for source addressing in a super-slow network, such as a bacterial communication network. Amplitude source addressing requires simple source and receiver bacterial circuits, which will be described in Section V-A. Reference [22] demonstrated that the response of receiver bacteria is distinct for each distinct amplitude, making the use of amplitude address practical. However, the receiver response model developed by [22] does not consider the distance between the source and the receiver as the microfluidic system in [22] uses flow channels, i.e., the nutrients and signals are carried to the receiver at a given flow rate. In this case, the distance between the source and

TABLE II
CONFIGURATIONS

b_{u1}	b_{u2}	b_{u3}	Configuration(C_i)	Mag(C_i)	$\Pr(C_i)$	Received Amplitude
0	0	0	0,0,0	0	$(1-p_t)^3$	0
0	0	1	0,0,1	3	$(1-p_t)^2 \cdot (p_t)$	3
0	1	0	0,1,0	2	$(1-p_t)^2 \cdot (p_t)$	2
0	1	1	0,1,1	5	$(1-p_t) \cdot (p_t)^2$	3
1	0	0	1,0,0	1	$(1-p_t)^2 \cdot (p_t)$	1
1	0	1	1,0,1	4	$(1-p_t) \cdot (p_t)^2$	3
1	1	0	1,1,0	3	$(1-p_t) \cdot (p_t)^2$	3
1	1	1	1,1,1	6	$(p_t)^3$	3

the receiver [24] affects only the propagation delay and has no impact on the performance of ADMA.

To this end, we propose *ADMA*, a source addressing mechanism that embeds the source address in the amplitude of the signal transmitted. In this section, we present the goals and challenges of using *ADMA* in a practical system. When multiple sources access the channel at the same time, their signals collide, and as a result, amplitudes are summed up in the channel. In the following section, we present an optimal amplitude assignment that can achieve zero address resolution error. Goals of *ADMA* are,

- to resolve the bits transmitted by sources, given a received amplitude, with minimum error, and
- to maximize the number of users that can be addressed.

B. Optimal Amplitude Addressing

Amplitude source addressing can be 100% accurate if the receiver can resolve every source given a received amplitude. Addressing and MAC has two main goals: 1) reliable packet delivery 2) throughput fairness. We define the Collision Resolution Error (CoRE) as a metric to measure reliability. CoRE is the ratio of the number of sources identified incorrectly to the total number of sources. An optimal amplitude assignment will achieve zero CoRE. CoRE is determined by the choice of amplitudes and the receiver design. We derive the conditions to achieve zero CoRE and propose an optimal amplitude assignment that achieves zero CoRE. Let b_i be the bit transmitted by i^{th} source with amplitude a_i in a given time slot.

$$b_i = \begin{cases} 1 & \text{if } i^{th} \text{ source transmits bit 1} \\ 0 & \text{if } i^{th} \text{ source transmits bit 0} \end{cases}$$

When molecular signals collide in the channel, the receiver obtains the sum of amplitudes transmitted $y = \sum_{i=1}^{N} b_i a_i$, where N is the number of sources in the network. If the number of partitions of the received amplitude y is one, i.e., the number of ways in which different a_i can be added to reach a sum y is one, then CoRE will be zero. If the number of partitions of y is greater than one, then CoRE is strictly greater than zero. Table II shows a sample network with three sources, each transmitting $b_i \in \{0, 1\}$. The amplitudes assigned to sources are $\{1, 2, 3\}$, creating 2^3 possible combinations of bits. We refer to each bit combination as a configuration, as shown in Column 4. The sum of amplitudes corresponding to each configuration is defined as its magnitude, which depends on both the configuration and the amplitudes assigned. Note that the configurations $\{0, 0, 1\}$ and $\{1, 1, 0\}$ have a magnitude of 3, which implies that the number of partitions for *3* in this setup, is two. On receiving an amplitude 3, the receiver must choose from the partitions of 3.

To achieve zero CoRE, the number of partitions of every received amplitude must be less than or equal to one. In other words, the magnitude of each configuration must be unique. This problem is studied in Number Theory as "Distinct subset sum (SSD)". A set is defined an SSD if and only if the sum of every subset of the sequence is unique [27]. An example of an SSD is the binary set, the set of powers of 2, $S = \{1, 2, 4, 8\}$. Each subset has a unique sum as every configuration has a unique magnitude. The majority of research on SSD focuses on finding the limit of the maximum value in a subset sum sequence [28]. In bacterial communication, the range of amplitudes $[R_{min}, R_{\max}]$ that the receiver can distinguish is determined by the receiver circuit design. The receiver circuit thus determines the following parameters of the network.

1) The minimum decodable amplitude at the receiver $R_{\min}$.
2) The maximum receivable amplitude $R_{\max}$ beyond which the receiver saturates. If an amplitude greater than $R_{\max}$ is transmitted, it is received as $R_{\max}$.
3) The step size of the levels of amplitudes that the receiver can distinguish. $R_{\max}$ and step size δ of the amplitudes determines the number of amplitude levels that can be distinguished, i.e., $N = \dfrac{R_{\max} - R_{\min}}{\delta}$ levels. By factoring the step size out and subtracting $R_{\min}$, the amplitudes that can be assigned are integer values $1, 2, 3, \ldots, N$. Therefore, in *ADMA*, integer amplitudes are used to analyze the performance of the network. The addresses proposed here can be used in a network with a step size greater or less than "1" by multiplying the proposed addresses with δ.

Theorem 1: For a given maximum sum, the set of powers of two (binary set) is an optimum set of amplitudes that render zero CoRE.

Proof: A set **S** with n elements has up to $2^n - 1$ non-empty subsets, hence $2^n - 1$ non-zero sums. To achieve zero CoRE, these $2^n - 1$ sums must be distinct. Each sum is different from another by at least one, so, the sum of all elements is at least $2^n - 1$. A binary set, consisting of powers of 2, satisfies the above condition. A binary set with all powers of 2 is an optimum set of addresses that minimizes CoRE. The maximum number of sources that can be supported by the binary set is related to $R_{\max}$. ■

A binary set can accommodate up to $N_{\log}$ sources, where $N_{\log} = \lfloor \log_2(R_{\max} + 1) \rfloor$. By assigning addresses from the binary set to sources, we also solve the multiple access problem. When signals from multiple sources with amplitudes from the binary set collide at the receiver, their sum is mapped to a unique configuration, allowing the receiver to decode with zero CoRE. As a corollary of Theorem 1, when $N > \lfloor \log_2(R_{\max} + 1) \rfloor$, CoRE is strictly greater than zero.

Though the binary set achieves CoRE, it limits the number of sources to $N_{\log}$. If $N > N_{\log}$, a set of addresses that can accommodate all the sources and minimizes CoRE is required. We propose a heuristic algorithm to select a set of amplitudes that approaches minimum CoRE, given N and $R_{\max}$. The two major challenges in designing an *ADMA* system that minimizes CoRE are,

- designing a sequence of amplitudes,
- designing a scalable and low complexity decoder.

C. Components of the ADMA Architecture

In this section, we define and describe each component of the receiver architecture of ADMA. The three components of the receiver architecture are sampler, demodulator, and decoder. The response of the receiver bacteria, shown in Figure 3, is input to the *sampler*, that samples it to discrete received amplitudes. Sampling and demodulation utilize the inverse of the bacterial receiver response model derived in [22]. The output of the sampler (the inverse model) is the time sequence of the molecular signal samples received. Thus, the inverse model determines the sampler's accuracy. The model developed in [22] is deterministic and does not account for the stochastic nature of receiver bacteria. In order to realize the stochastic nature of receiver response to input chemical signal, we introduce random noise to the "k parameters" of the inverse model. The "k parameters" are the different rate constants of the receiver bacteria (example, GFP expression rate) that define the state of the receiver bacteria. By varying these parameters randomly within a specified range, we implement sampling and demodulation errors in the receiver response.

The samples are then input to the *decoder* to resolve the addresses and bits. We present decoder designs in detail in Section IV. The *decoder* outputs a vector of bits, the estimate of bits transmitted by all sources that sum to $y[i]$, the received sample. Thus, the receiver design determines address resolution on receiving a signal, which in turn, determines CoRE. Here, we propose two receiver designs based on the principles of Maximum a posteriori detection: 1) a probabilistic receiver, and 2) a deterministic receiver. The receiver decodes the samples, which are then used to decode the bits. The optimality of the receiver design and the time complexity of the decoder are analyzed at the sample level. The number of samples per bit is pre-determined based on the application and system constraints. Thus, there is no need for coordination or time synchronization between sources, and the sources do not require additional processing to synchronize transmission, i.e., each source can transmit data as and when it has information to transmit. Assuming that all sources always have data to transmit, the receiver is continuously receiving samples. The most recurring value of samples is then used to determine bits. A receiver that maximizes the probability of success of each sample, in turn, maximizes the probability of success of the bit decoded from these samples.

IV. ADMA Receiver Designs

In the previous section, we presented an optimal amplitude addressing that achieves zero CoRE. But, the maximum number of sources N is limited by $\lfloor \log_2(R_{\max} + 1) \rfloor$. As derived in Theorem 1, when N is $> \lfloor \log_2(R_{\max} + 1) \rfloor$, average CoRE will be strictly greater than zero. In a network with $N > \lfloor \log_2(R_{\max} + 1) \rfloor$, at least one amplitude has more than one partition and therefore the receiver design also contributes to CoRE. In this section, we propose two receiver designs and derive an upper bound on the expected number of success in resolving address with each receiver design.

We make the following assumptions about the network in the design and evaluation of the receiver designs. $R_{\max}$ is the maximum amplitude that the receiver can uniquely identify. Each source is assigned a distinct amplitude, and hence up to $R_{\max}$ sources can be accommodated, i.e., $N \leq R_{\max}$. If $N > R_{\max}$, the network is divided into subnets. $S = \{a_1, a_2, \ldots a_N\}$ is the set of amplitudes assigned to sources $u_1, u_2, \ldots u_N$ respectively. The sources always have data to transmit; at a given time, a source is either transmitting bit 1 or transmitting bit 0 and these N sources can be transmitting in one of the 2^N configurations. Consider the example in Table II. Columns 1 to 3 in Table II are the bits transmitted by sources 1 to 3. Column 5 is the magnitude of the configuration, equal to the sum of amplitudes corresponding to each configuration and Column 7 is the amplitude received when the corresponding configuration is transmitted. Multiple configurations can add up to the same sum. All configurations with the same magnitude are called the *partitions* of that magnitude. Since $R_{\max} = 3$, any amplitude ≥ 3 is received as 3. Here, $\{0, 0, 1\}, \{0, 1, 1\}, \{1, 0, 1\}, \{1, 1, 0\}, \{1, 1, 1\}$ are the partitions of 3, as all these configurations are received as 3 by the receiver. p_t and $1 - p_t$ are the probabilities of a source transmitting bit 1 and bit 0 respectively. The probability of a configuration occurring in the channel depends on p_t and the addresses. For example, the probability of C_0 is $(1 - p_t)^N$ as all N sources transmit bit 0, each with probability $1 - p_t$. The probability of each configuration is shown in Column 6 of Table II.

We propose *Probabilistic Receiver* to minimize the error in decoding a configuration and *Deterministic Receiver* to decrease the bit error for individual sources. We also derive an upper bound on the expected number of successful address resolutions for the above two receivers, which is then used to calculate a bound on the throughput performance of ADMA with each receiver.

A. Probabilistic Receiver (PR)

The Probabilistic Receiver is designed to minimize the number of errors in resolving the transmitted configuration. On receiving an amplitude, PR chooses the configuration that minimizes the probability of error in decoding that amplitude, in turn, maximizing the probability of success in decoding the received amplitude. PR follows the Maximum a Posteriori (MAP) detection rule [29] to minimize the bit error in decoding a configuration, given the received amplitude.

1) Observations on Optimality and Practicality of PR: The received amplitude is the sum of the magnitude of the transmitted configuration and the amplitude errors due to channel noise. The receiver observes this noisy signal and estimates the transmitted configuration using a priori estimates of the channel and the transmitter distributions. We use the probability of error in decoding the configuration as the metric to evaluate the performance of the receiver. We follow the logic of MAP detection rule that minimizes the expected symbol decoding error to define an optimum receiver [29].

An optimum receiver is a receiver which minimizes the probability of error in resolving the transmitted configuration given the received amplitude.

Minimizing the probability of error on receiving a configuration $\hat{C}_i$ in turn maximizes the probability of success.

$$\Pr\left(\hat{C}_i \neq C_i\right) = 1 - \Pr\left(\hat{C}_i = C_i\right) \tag{2}$$

where C_i is the configuration transmitted. The receiver estimates $\hat{C}_i$ on receiving y. The receiver decision is considered a success if the estimated configuration was the actual transmitted configuration. On receiving an amplitude y, the probability of success in decoding the transmitted configuration is,

$$\Pr\left(\hat{C}_i = C_i, y\right) = \Pr\left(\hat{C}_i = C_i \mid y\right) \cdot \Pr(y) \tag{3}$$

Since $\Pr(y)$, the probability of receiving amplitude y, is a constant for a known source distribution and channel model, the optimum receiver will choose $\hat{C}_i$ such that it maximizes the conditional probability $\Pr(\hat{C}_i = C_i \mid y)$. Thus, the probability of success in choosing a configuration $\hat{C}_i$ on receiving y is,

$$\Pr\left(\hat{C}_i \mid y\right) = \frac{\Pr\left(y \mid \hat{C}_i\right) \cdot \Pr\left(\hat{C}_i\right)}{\Pr(y)} \tag{4}$$

where $\Pr(\hat{C}_i)$, the probability of $\hat{C}_i$ being transmitted, is obtained from a priori estimates of the source distribution, $\Pr(y)$ is known for given amplitudes, and $\Pr(y \mid \hat{C}_i)$ from the a priori estimates of the channel transition probabilities. Thus, for every received amplitude y, the optimum receiver chooses a configuration that maximizes the probability of success with accurate a priori estimates of the source and the channel.

We design PR using the above MAP detection rule and iterate through all possible configurations and choose the most probable configuration which maximizes the overall probability of success given by,

$$\sum_{y=0}^{R_{\max}} \Pr\left(\hat{C}_i, y\right) = \sum_{y=0}^{R_{\max}} \Pr\left(\hat{C}_i \mid y\right) \cdot \Pr(y) \tag{5}$$

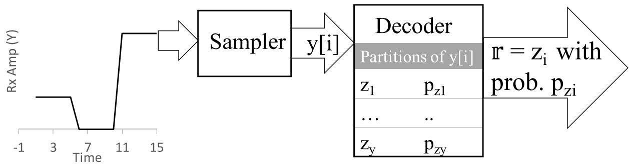


Fig. 4. Probabilistic Receiver Illustration.

PR is therefore an optimum receiver that maximizes the probability of success in decoding the transmitted configuration for a received amplitude, with an accurate a priori estimate of the source distribution and the channel transition probabilities.

Though PR maximizes the probability of success in estimating the transmitted configuration, it is an idealized receiver that assumes accurate a priori estimates of the source and channel distributions. It is computationally complex to obtain an accurate estimate of these distributions. The computational complexity of the receiver to iterate through all possible configurations for each received amplitude increases exponentially with the number sources as the number of configurations increases exponentially with the number of sources.

To overcome these challenges, we propose Deterministic Receiver (DR), a practical, low-complexity, heuristic receiver design later in this section. PR maximizes the probability of success in decoding the configuration while DR focuses on individual bits, i.e., PR minimizes the symbol error rate while DR focuses on reducing the bit error rate of individual sources.

We derive an upper bound on the expected number of successes using PR, which is then used to develop DR and the amplitude assignment algorithm in Section V.

a) Probabilistic receiver architecture: Figure 4 is an illustration of the PR architecture. The *sampler* in this architecture is the *inverse model* derived from [22]. The sampler takes as input the response of bacteria and generates a time sequence of amplitude samples as received by the receiver bacteria. The *decoder* of PR takes a sampled signal $y[i]$ as input and generates a table of partitions of $y[i]$. z_1 to z_j are the partitions of $y[i]$. The decoder chooses a partition that maximizes the probability of success. The a priori channel transition probabilities are considered in generating the table of partitions for each received amplitude. If more than one partition can maximize the probability of success, one of them is chosen randomly as it will not affect the performance statistically.

Consider the example in Table II. Each configuration has a magnitude and a received amplitude. For simplicity, we do not present channel transition probabilities in Table II. In the presence of channel errors, the received amplitude for each configuration will be a range of amplitudes with a probability associated to each amplitude, unlike the single received amplitude shown here. Thus, in an ideal channel, on receiving an amplitude 3, PR chooses configuration {1, 1, 0} with a probability $\frac{(1-p_t)(p_t)^2}{\Pr(y=3)}$.

It can be noted that the conditional probability is affected by two types of errors,

- channel error - channel induced errors alter the amplitudes received,

TABLE III
PROBABILISTIC RX EXAMPLE

Rx Amplitude	Partitions	Conditional Prob.
0	{0,0,0}	1
1	{1,0,0}	1
2	{0,1,0}	1
3	{0,0,1},{0,1,1},{1,0,1},{1,1,0},{1,1,1}	0.2

- collision error - when multiple sources collide in the channel, receiver receives the sum of amplitudes.

PR decodes C_i on receiving an amplitude y based on its probability of occurrence, i.e., $\Pr(C_i \mid y)$ is equal to $\frac{\Pr(C_i)}{\Pr(y)}$, if the channel is noise free, as shown in Table III. With the knowledge of the transition probabilities of the channel, PR chooses a partition C_i as shown in Equation (4).

In the following section, we derive an upper bound on the expected number of successful address resolutions using PR. The number of successful address resolutions can be determined accurately for a known amplitude assignment. We derive the upper bound to compare different amplitude assignment mechanisms. There are $\binom{N}{R_{\max}}$ ways of assigning $R_{\max}$ amplitudes to N sources. The upper bound is then used as a measure to evaluate the data-rate performance of different sets of addresses. In the derivation of the upper bound, we consider only the decoding error from collisions. The channel noise is a function of the amplitudes assigned, and therefore it becomes impossible to derive an upper bound in the presence of channel noise, without the knowledge of the amplitudes. In the implementation and evaluation of ADMA in nanoNS3, we introduce channel noise. As shown in Figure 4, PR maintains the list of partitions for each received amplitude. For a given p_t, the receiver determines the a priori probabilities of each partition. The decoder chooses one of the partitions with a probability equal to its probability of occurrence. An example of PR decoder table for $R_{\max} = 3, N = 3, p_t = 0.5$ is shown in Table III. Column 3 shows the conditional probabilities of each configuration given the received amplitude. On receiving amplitude 3, all five partitions have the equal probability of being chosen, and hence the conditional probability of choosing a partition on receiving amplitude 3 is 0.2. The probability of decoding a signal is proportional to that of its occurrence. A partition with a high probability of transmission is received with high probability. Let Y be the random variable representing the received amplitude and m_k be the bit sample transmitted by source u_k at a given time and $\hat{m}_k$ be the bit sample the receiver decodes. Decoding is successful for source u_k if $m_k = \hat{m}_k$. The expected number of success, where success is the event of accurate address resolution, is given by

$$\mathbb{E}(\text{No. of succ}) = \sum_{y=1}^{R_{\max}} \mathbb{E}(\text{No. of succ.} \mid Y = y) \cdot \Pr(Y = y)$$

$$\mathbb{E}(\text{No. of succ.} \mid Y = y) = \sum_{k=1}^{N} \Pr\big(\hat{m}_k = m_k \mid Y = y\big) \tag{6}$$

where $\Pr(\hat{m}_k = m_k \mid Y = y)$ is the probability of success for source u_k on receiving y. We derive the probability of success for source u_k using PR below. As the sources are always backlogged, the message bit is either 1 or 0 and hence $m_k \in \{0, 1\}$. We further condition on $m_k = 1$ and $m_k = 0$. Let $p_y = \Pr(Y = y)$ be the probability of receiving amplitude y. Using Bayes's rule,

$$\begin{aligned}
&\Pr\big(\hat{m}_k = m_k, m_k = 1 \mid Y = y\big)\\
&\qquad = \frac{\Pr\big(\hat{m}_k = m_k, m_k = 1, Y = y\big)}{\Pr(Y = y)}\\
&\qquad = \frac{\Pr\big(\hat{m}_k = m_k \mid m_k = 1, Y = y\big) \cdot p_{yk^1}}{p_y}
\end{aligned}$$

where $p_{yk^1} = \Pr(m_k = 1, Y = y)$ is the probability of source u_k transmitting bit 1 and $Y = y$. A source always has a bit to transmit. Therefore, $p_{yk^1} + p_{yk^0} = p_y$

$$\begin{aligned}
&\Pr\big(\hat{m}_k = m_k, m_k = 0 \mid Y = y\big)\\
&\qquad = \frac{\Pr\big(\hat{m}_k = m_k \mid m_k = 0, Y = y\big) \cdot p_{yk^0}}{p_y}\\
&\qquad = \frac{\Pr\big(\hat{m}_k = m_k \mid m_k = 0, Y = y\big) \cdot \big(p_y - p_{yk^0}\big)}{p_y}
\end{aligned}$$

Since PR chooses a partition with a probability equal to its conditional probability, on receiving an amplitude y given $m_k = 1$, u_k is successfully received, if the receiver chooses any one of the partitions such that $m_k = 1$.

$$\Pr\big(\hat{m}_k = m_k \mid m_k = 1, Y = y\big) = \frac{p_{yk^1}}{p_y} \tag{7}$$

$$\Pr\big(\hat{m}_k = m_k \mid m_k = 0, Y = y\big) = \frac{p_{yk^0}}{p_y} = \frac{p_y - p_{yk^1}}{p_y}$$

$$\Pr\big(\hat{m}_k = m_k, m_k = 1 \mid Y = y\big) = \frac{p_{yk^1} \cdot p_{yk^1}}{p_y \cdot p_y} \tag{8}$$

$$\Pr\big(\hat{m}_k = m_k, m_k = 0 \mid Y = y\big) = \frac{p_{yk^0} \cdot p_{yk^0}}{p_y \cdot p_y}$$

Substituting above probabilities, we derive,

$$\Pr\big(\hat{m}_k = m_k \mid Y = y\big) = \left(\frac{p_{yk^1}}{p_y}\right)^2 + \left(\frac{p_{yk^0}}{p_y}\right)^2 \tag{9}$$

The expected number of success depends on the probability of success of all sources and hence an upper bound on the probability of success per source will provide an upper bound on the expected number of success. Substituting Equation (9) in Equation (6),

$$\mathbb{E}(\text{No. of succ.} \mid Y = y) = \sum_{k=1}^{N} \left(\frac{p_{yk^1}}{p_y}\right)^2 + \left(\frac{p_{yk^0}}{p_y}\right)^2 \tag{10}$$

$$\begin{aligned}
\mathbb{E}(\text{No. of succ.}) &= \sum_{y=1}^{R_{\max}} \sum_{k=1}^{N} \Pr\big(\hat{m}_k = m_k \mid Y = y\big) \cdot p_y\\
&\leq \sum_{y=1}^{R_{\max}} N \cdot \left(\frac{\big(p_{yk^1}\big)^2}{p_y} + \frac{\big(p_y - p_{yk^1}\big)^2}{p_y}\right)
\end{aligned} \tag{11}$$

From Equation (11), note that the expected number of success is maximized when $p_{yk^1} = p_y$. This supports the theorem

Algorithm 1 Deterministic Rx Implementation

```
$N \leftarrow$ Number of sources, $R(y) \leftarrow \{\}$
for $y := 0$ to $R_{\max}$ do
    for $k := 1$ to $N$ do
        $p_{yk^1} \leftarrow \Pr\{\text{Source } k \text{ transmitting bit } 1 \mid \text{y}\}$
        $p_{yk^0} \leftarrow \Pr\{\text{source } k \text{ transmitting bit } 0\}$
        if $p_{yk^0} \geq p_{yk^1}$ then
            $R(y) \leftarrow R(y), \{0\}$
        else
            $R(y) \leftarrow R(y), \{1\}$
        end if
    end for
end for
```

that the number of successes is N only when the number of partitions is 1. Equation (11) can be further simplified as

$$\mathbb{E}(\text{No. of succ.}) \leq N \sum_{y=1}^{R_{\max}} \left(p_y - \frac{p_y - p_{yk^1}}{p_y} 2p_{yk^1} \right) \tag{12}$$

$p_{yk^1} \leq p_y$. When the equality does not hold we can write $p_{yk^1} = r \cdot p_y$ where $r < 1$

$$\mathbb{E}(\text{No. of succ.}) \leq N \sum_{y=1}^{R_{\max}} \left\{ p_y \cdot \left(1 - 2r + 2r^2 \right) \right\} \tag{13}$$

Average network throughput $= \frac{\text{Expected number of successes}}{\text{Pulse duration}}$. For each signal received, PR chooses one of the partitions with its probability of occurrence. For a set of integer amplitudes, the number of partitions increases exponentially [30] for large values of N. On receiving a signal, the receiver goes through the partitions of integers contributing $\mathcal{O}(e^N)$ to the time complexity; where N is the number of sources. This is repeated for each signal received and hence the overall time complexity is $\mathcal{O}(e^N|Z|)$, where $|Z|$ is the number of received signals. As the number of partitions increases exponentially, the space required to store all the partitions is given by $\mathcal{O}(e^N)$. The exponential time and space complexity limit the practical implementation of PR. In the following section, we propose a simpler receiver with reduced time and space complexity and an improved network throughput.

B. Deterministic Receiver (DR)

PR maximizes the probability of success with accurate estimates of source distribution and channel noise model. Also, the decoder complexity is exponential to the number of sources. To reduce the complexity of the receiver, we propose Deterministic Receiver (DR), a heuristic receiver design which chooses a pre-determined configuration on receiving an amplitude to reduce the bit error of each source independently.

Deterministic Receiver Architecture: For each received amplitude y, DR chooses a bit sample for individual sources independently. DR goes through all the partitions of the amplitude y and chooses the most probable bit for each source. This is pre-determined during receiver setup. In this work, we focus on minimizing decoding errors due to collisions in DR analysis. Knowledge of channel transition probabilities is incorporated by including channel noise in the estimation of the most probable bit. DR thus differs from PR in building and updating the decoder table. With the help of Algorithm 1, we explain the operation of DR.

TABLE IV
DR: UPPER BOUND ESTIMATION

Max No. of successes	Case 1	Case 2
N	1	0
$N-1$	$N-1$	1
$N-2$	$\leq \binom{N}{2}$	$\leq \binom{N}{2}$
$N-3$	$\leq \binom{N}{3}$	$\leq \binom{N}{3}$
..	..	..
1	$\leq \binom{N}{N-1}$	$\leq \binom{N}{N-1}$
0	0	1

Line 2 loops over all receivable amplitudes. For each amplitude received, DR compares p_{yk^1} and p_{yk^0} for each source, i.e., the probability of the source transmitting bit 1 and bit 0 given that amplitude y was received, is compared. In line 6, if $p_{yk^0} \geq p_{yk^1}$ for source u_k, the receiver updates the received signal $R(y) = 0$ for source u_k. $R(y)$ is the vector of bits decoded by the receiver on receiving y. $R(y)$ is calculated once, and the receiver uses a constant-time lookup to decode. DR updates the vector of bits decoded for all amplitudes that can be received. DR maximizes the conditional probability of source u_k transmitting a bit given the received signal y. The maximization is performed a priori, and hence the time complexity is $\mathcal{O}(1)$. The bit vector calculated independently for each source may not be one of the partitions or configurations of the amplitude received. But this vector maximizes the number of successes for each source independently. We study the average performance of DR for different states of the receiver. To study the performance of DR, we use the same parameters as that of PR. The average number of successes using DR can be described as,

$$\begin{aligned} \mathbb{E}(\text{No. of succ.} \mid Y = y) &\leq N \cdot \max_{1 \leq k \leq N} \max\left(p_{yk^1}, p_{yk^0}\right) \\ &\leq N \cdot \max_{1 \leq k \leq N} p_y \cdot \max(r, 1 - r) \end{aligned}$$

It can be observed that the bound on the probability of success reaches its maximum value of 1 at $r = 0$ or $r = 1$. We also proved in Theorem 1 that a practical system can achieve a probability of success of 1 only when $N \leq \lfloor \log_2(R_{\max} + 1) \rfloor$. We derive a practical upper bound on the expected number of successes with DR, analytically.

1) Case 1 (Chosen Configuration Is a Partition): If the chosen configuration is one of the partitions, then the expected number of successes when it is transmitted is N. When the chosen configuration is transmitted, it will be decoded without error. All the partitions that differ from the chosen configuration by one bit will be received with $N - 1$ successes and so on.

Algorithm 2 Deterministic Receiver Upper bound

```
1: $R \leftarrow R_{\max}$
2: $N \leftarrow$ Number of sources
3: $\mathbb{E} \leftarrow 0,$ Expected number of successes
4: for $i := 1$ to $R$ do
5:     $\mathbb{E} \leftarrow \mathbb{E} + C[0 : i] \cdot N, \hat{C} \leftarrow C[i : \text{end}]$
6:     $\mathbb{E} \leftarrow \mathbb{E} + \hat{C}[0 : (R-i)] \cdot (N-1)$
7:     $\hat{C} \leftarrow C[(R-i) : \text{end}]$
8:     $\mathbb{E} \leftarrow \mathbb{E} + C[0 : \binom{N}{1}] \cdot (N-1)$
9:     $\hat{C} \leftarrow C[R + \binom{N}{1} : \text{end}]$
10:    $\mathbb{E} \leftarrow \mathbb{E} + \hat{C}[0 : \binom{N}{2}] \cdot (N-2)$
11:    $\hat{C} \leftarrow C[R + \binom{N}{1} + \binom{N}{2} : \text{end}]$
12:    $\mathbb{E} \leftarrow \mathbb{E} + \hat{C}[0 : (R-i) \cdot \binom{N}{2}] \cdot (N-2)$
13:    $\hat{C} \leftarrow C[R + \binom{N}{1} + \binom{N}{2} + (R-i) \cdot \binom{N}{2} : \text{end}]$
14:    $i_{start} \leftarrow R + \binom{N}{1} + \binom{N}{2} + (R-i) * \binom{N}{2}$
15:    $i_{\text{end}} \leftarrow i_{start} + \binom{N}{3} + \binom{N}{2} + (R-i) \cdot \binom{N}{3}$
16:    for $j = 3$ to N do
17:        $\mathbb{E} \leftarrow \mathbb{E} + C[i_{start} : i_{\text{end}}], i_{start} \leftarrow i_{\text{end}}$
18:        $i_{\text{end}} \leftarrow i_{start} + i \cdot \binom{N}{j} + (R-i) \cdot \binom{N}{j}$
19:    end for
20: end for
```

All configurations that sum to y will be different from the chosen one by at least one bit. No two configurations that differ by one bit can add to the same integer. For example, a configuration 1101 and 1100 differs by one bit. Let y_1 be the sum of configuration 110. $y_1 + a_4$ cannot be equal to y_1. Thus, if the chosen configuration is one of the partitions, no partition of that sum will have $N-1$ success. Configurations that map to $R_{\max}$ is an exception. Since all configurations that sum to $R_{\max}$ and greater are mapped to $R_{\max}$, two configurations that differ by one bit can add to $R_{\max}$. For example, configurations 001 and 011 are mapped to 3 in Table IV. Therefore, at most $N-1$ configurations that differ by one bit from chosen configuration and mapped to $R_{\max}$ can have $N-1$ successes.

Following the same argument, no partition of a sum can differ from the chosen one by two bits if the chosen configuration is a partition. For example, 1101 and 1110. Let sum of amplitude 11 be y_2. $y_2 + a_4$ cannot be equal to $y_2 + a_3$ as all amplitudes are distinct. Therefore, if the chosen one is one of the partitions, no configuration that sum to X will have $\leq N-3$ successes. Configurations summing to $R_{\max}$ are an exception. Up to $\binom{N}{2}$ configurations that differ from the chosen configuration that add to $R_{\max}$ can have $N-2$ successes. There can be up to $\binom{N}{3}$ partitions with a Hamming distance of 3 between themselves and the chosen configuration and up to $\binom{N}{4}$ partitions with a Hamming distance of 4 from the chosen partition and so on. Column 1 in Table IV shows the maximum number of successes and Column 2 defines the maximum number of configurations that can achieve these successes in Case 1.

2) Case 2 (Chosen Configuration Is Not a Partition): If the chosen configuration is not one of the partitions, none of the partitions of that integer can have N success. At most, one partition can have $N-1$ success. At most, one partition has a Hamming distance of one from the decoded configuration. Let us assume there are two partitions with a Hamming distance of one between them and the decoded configuration. Then the Hamming distance between these partitions is at most 2. Since two partitions of an integer differ by at least 3 bits, this is not possible. Thus, if the chosen configuration is not one of the partitions, up to one configuration can have $N-1$ success. There can be up to $\binom{N}{2}$ partitions that have a Hamming distance of 2 from the chosen configuration. Similarly, there can be up to $\binom{N}{3}$ partitions that are 3 Hamming distance away from the chosen configuration. Columns 2 and 3 in Table IV summarizes Case 1 and 2 respectively. There are $R_{\max}$ distinct integers receiver can receive. For some amplitudes, the receiver can choose one of its partitions; and for others, the receiver chooses a configuration that is not a partition. We use Algorithm 2 to find an upper bound using DR. C is the sorted array of the probability of each configuration being transmitted in descending order and $\mathbb{E}$ is the expected number of success on receiving an amplitude sample. Given N and p_t, there are $\binom{N}{k}$ configurations with k ones and $N-k$ zeros, where k varies from 0 to N. Lines 4 loops over i where i varies from 1 to R. i represents the number of received amplitudes whose decoded configuration is one of the partitions. $R-i$ amplitudes have decoded configurations that are not their partition. From the above table, we know that only those decoded configurations that are partitions of the amplitude can have N success. Thus, in the *for loop* in Algorithm 2, up to i configurations can have N success. To calculate an upper bound, we assume that the highest probable i configurations can have N success. Also, one configuration per received amplitude can have $N-1$ success, if the decoded configuration is not its partition. Thus, up to $R-i$ configurations can have $N-1$ success. All amplitudes greater than $R_{\max}$ is received as $R_{\max}$, and hence the highest receivable amplitude can have up to N partitions with $N-1$ success and similarly up to $\binom{N}{2}$ configurations with $N-2$ successes. Lines 16 to 19 loop through remaining configurations assigning $N-j$ successes to $\binom{N}{j} \cdot N$ configurations. Maximum E over all i gives the upper bound on the expected number of success.

V. Amplitude Assignment Algorithm

As discussed in Section II, the two factors that impact Collision Resolution Error (CoRE) are the receiver design and the source addresses. In this section, we present a heuristic algorithm that chooses an amplitude sequence that minimizes CoRE based on the insights we gained from the derivation of the probability of success in Section IV. Following Theorem 1, the maximum number of sources that can be accommodated with zero CoRE is $\lfloor \log_2(R_{\max}+1) \rfloor$. Since the addresses are distinct, the maximum number of sources is limited by $R_{\max}$. As the number of sources increases, the required $R_{\max}$ to achieve zero CoRE increases exponentially. In this section, we propose an algorithm to choose a set of amplitudes that minimizes CoRE, for a fixed N and $R_{\max}$. From the probability of success derivation for receiver design in Section IV, we obtain the following insights, which is then used to propose four amplitude sequences.

Insight 1: The higher the number of partitions, the lower is the probability of success on receiving the sum.

Insight 2: At a low probability of transmitting a *bit* 1, the number of colliding signals is ≤ 2 with high probability.

When the number of collisions is less than or equal to two, an address sequence that can recover from two collisions will have a high probability of success. We define *shifted (natural) sequence* as a sequence of integers with the first element shifted by $\frac{R_{\max}-1}{2}$, $S = \{\frac{R_{\max}-1}{2}, \frac{R_{\max}-1}{2}+1, \ldots, R_{\max}-1\}$. A maximum of $\frac{R_{\max}}{2}$ sources can be supported using this sequence. No element in the shifted sequence can be written as a sum of any two elements. $a_i+a_j \neq a_k$ where $a_i, a_j, a_k \in S$ and $i \neq j \neq k$. Therefore, when one source transmits a bit 1, it will be received without error. When two sources transmit bit 1 and collide, the number of partitions of received amplitude is reduced, and therefore the error is smaller compared to other address sequences since an error in one address will not affect others, making it suitable for scenarios with a low probability of bit 1 collisions. A shifted natural sequence allows $\frac{R_{\max}}{2}$ sources. If $N \geq \frac{R_{\max}}{2}$, we extrapolate the shifted sequence as $S = \{R_{\max}-N_u-1, R_{\max}-N_u-2, \ldots R_{\max}-1, R_{\max}\}$. The extrapolated sequence does not hold the property of a shifted sequence. The number of elements that can be written as sum of two other elements is small in this sequence since its elements are decreasing integers from $R_{\max}$.

Insight 3: At a high probability of transmitting a bit 1, the number of colliding signals is $\geq N_{\max}$ with high probability.

Let $N_{\max}$ be the number of sources such that sum of amplitudes up to $N_{\max}$ is less than equal to $R_{\max}$. When more than $N_{\max}$ sources collide, a *natural sequence*, which is the sequence of integers beginning from 1 till N, will have a high probability of success. When the number of sources transmitting bit 1 at a given time increases, the probability of receiving configurations that sum to $\geq R_{\max}$ increases. All sums greater than $R_{\max}$ are received as $R_{\max}$. On receiving a sum $R_{\max}$, the receiver chooses one of the configurations with sum $\geq R_{\max}$. The probability of choosing a particular configuration is $\frac{1}{N(x \geq R_{\max})}$, where $N(x \geq R_{\max})$ is the number of configurations that sum up to $R_{\max}$ and more. The fewer the number of configurations with sum $\geq R_{\max}$, the higher the probability of success. Amplitudes that can minimize $N(x \geq R_{\max})$ will minimize CoRE. A natural sequence has the maximum number of configurations $< R_{\max}$, and hence minimum $N(x \geq R_{\max})$.

Insight 4: At an intermediate probability of transmitting a bit 1, the number of colliding signals is $\leq N_{\log}$ with high probability.

At intermediate values of p_t, the per-source probability of transmitting a bit 1, up to $N_{\log}$ signals collide. A sequence that has the maximum number of sums with unique configurations with up to N elements will have a high probability of success. *Extrapolated binary sequence* $S = \{2^0, 2^1, 2^2, \ldots N_{\log}, R_{\max} - N - N_{\log}, R_{\max} - N - N_{\log} - 1, \ldots, R_{\max}\}$ combines a binary sequence and a shifted sequence. $N_{\log} = \lfloor \log_2(R_{\max} + 1 \rfloor)$ is the number of sources a binary sequence can support. A binary sequence is an optimum solution to achieve zero CoRE. An extrapolated binary sequence utilizes the maximum number of distinct sums when using a binary sequence. These distinct sums are obtained from the binary elements. The rest of the elements are integers decreasing from $R_{\max}$, which reduces the number of overlapping sums. As the number of collisions approaches $N_{\log}$, the distinct sums contributed by the binary elements will improve the probability of success. Algorithm 3 summarizes the insights gained from the sequences observed and the probability of success derivation.

Algorithm 3 Address Allocation

```
$R_{\max} \leftarrow$ Maximum Receivable amplitude
$N_s \leftarrow$ Number of simultaneous sources
$N_u \leftarrow$ Total number of sources
$S \leftarrow$ Set of addresses
**if** $\Pr(N_s \leq 2) > \Pr(N_s > 2)$ and $N_u \leq \frac{R_{\max}}{2}$ **then**
    $S \leftarrow \{\frac{R_{\max}-1}{2}, \frac{R_{\max}-1}{2}+1, \ldots, R_{\max}-1\}$
**else if** $\Pr(N_s \leq 2) > \Pr(N_s > 2)$ and $N_u > \frac{R_{\max}}{2}$ **then**
    $S \leftarrow \{R_{\max}-N_u, R_{\max}-N_u-1, \ldots, R_{\max}-1\}$
**else if** $\Pr(N_s \leq N_{\log}) > \Pr(N_s > N_{\log})$ **then**
    $S \leftarrow \{2^0, 2^1, \ldots, 2^{N_{\log}}, R_{\max}, R_{\max}-1, \ldots\}$
**else if** $\Pr(N_s \geq N_{\max}) > \Pr(N_s < N_{\max})$ **then**
    $S \leftarrow \{1, 2, 3, \ldots, N_u\}$
**else** $S \leftarrow \{2^0, 2^1, \ldots, 2^{N_{\log}}, R_{\max}, R_{\max}-1, \ldots\}$
**end if**
```

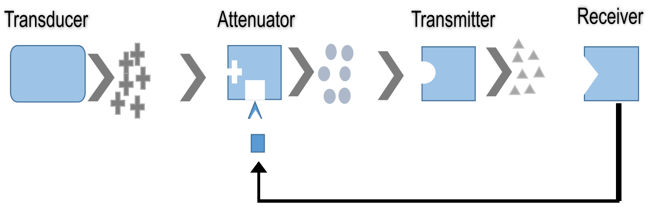


Fig. 5. Practical Implementation: Illustration.

A. Amplitude Assignment: Practical Implementation

We presented an optimum receiver design and an algorithm to choose an amplitude sequence that minimizes CoRE. In this section, we discuss network architecture and a practical bacterial communication system that considers practical amplitude assignment. We describe how a receiver can assign amplitudes to the sources based on its $R_{\max}$. To the best of our knowledge, no existing work provides a practical transmitter design that can transmit different amplitudes using only components built from bacterial populations. In Figure 5, we present a circuit design to implement a bacterial transmitter for ADMA. As shown in Figure 5, each source consists of three major components viz., a transducer, an attenuator and a transmitter. Each *source* in Figure 1 is composed of the above three components. In a sensing network, the transducer is genetically engineered, based on the application, to sense a specific signal and convert to another signal, as it is now possible to design networks that utilize multiple signal molecules with little cross-talk [31]. The transducer emits one quorum sensing signal (AIP for example, crosses) at a concentration proportional to the inducer (IPTG) concentration. The AIP thus generated is attenuated by the amplifier/attenuator, which produces a second signal (AI-2 for example, circles). The amount of attenuation can

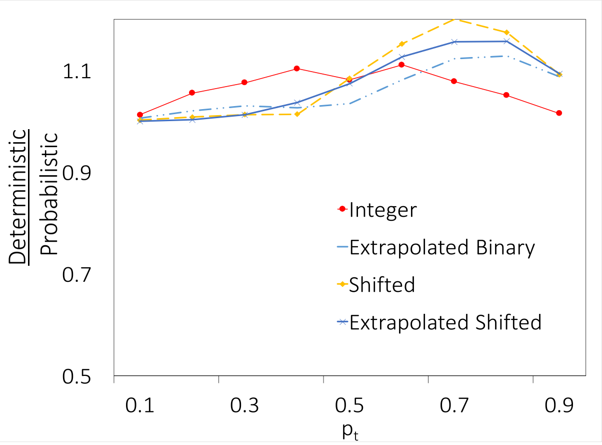


Fig. 6. Data-rate of Deterministic vs. Probabilistic Receiver $N = 5, R_{\max} = 15$.

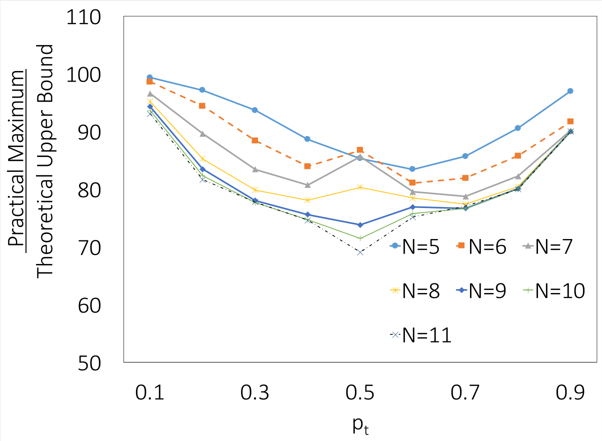


Fig. 7. Theoretical vs. Practical Upper Bound.

be controlled externally using feedback. This attenuated AI-2 signal is the input to the transmitter which in turn emits a distinct signal (C6-HSL for example, triangles) to be transmitted to the receiver through a microfluidic channel. The attenuator modifies the transmitted signal concentration, and therefore is the address allocator in the architecture. Assuming a feedback path from the receiver to the attenuator, the receiver assigns the amplitude to each source of bacterial population by modifying the attenuator. The attenuator can be a bacterial population that emits AI-2 in a manner so that the concentration of AI-2 emitted is controlled by an external trigger (squares) to the attenuator population or additional signal (such as AIP itself) from the receiver. Such a trigger is provided by the receiver (or a centralized server) to each source based on the amplitude assigned to the source. Thus, the receiver (or the server) controls the amplitude/concentration of the signal transmitted to the receiver which in turn is used to identify the source on receiving a signal. It can be noted that the concentration of molecules generated by all the sources is the same. Thus, ADMA is also an energy fair mechanism.

VI. Performance Evaluation

We evaluate *ADMA* in progressive steps.

1) We evaluate the performance of amplitude assignment algorithm and the receiver designs, using *BCS* (Bacterial Communication Simulator), a custom-built Python-based simulator in idealized channel conditions, i.e., assuming zero sampling and demodulation error at the receiver.
2) We introduce channel errors in *BCS* and analyze the performance of ADMA with channel errors.
3) We implement the response of a receiver bacteria located in a microfluidic chip derived in [22] in nanoNS3 [32], the bacterial molecular communication simulator developed on top of NS3 simulates source error, channel error, and sampling and demodulation error at the receiver. We implement *amplitude assignment* algorithm and *deterministic receiver (DR)* in nanoNS3 and evaluate the performance with errors. We also show using experimental results and nanoNS3, that by using the inverse of the receiver model, receiver response can be demodulated.

A. Idealized Network Conditions

Unless otherwise mentioned, we make the following assumptions in the implementation and evaluation of ADMA in *BCS*. Each source transmits bit 1 with a probability p_t and has an uninterrupted supply of data to transmit. Each data point in the results presented is averaged over 100 simulations. Before evaluating ADMA, we present the tightness of upper bound derived in Section IV.

1) Upper Bound Tightness: In Algorithm 2, we derive an upper bound on the expected number of success, assuming that the highest probability configuration achieves the highest number of success without considering the amplitudes assigned. In practice, configurations are not independent of the amplitudes assigned and the probability of success depends on the choice of amplitudes and the receiver. We evaluate the theoretical upper bound tightness by performing an exhaustive search on all possible address assignment and determine the practical upper bound. Given $R_{\max}, N$, there are $\binom{R_{\max}}{N}$ possible address sequences; N addresses chosen from $[1, 2, 3, \ldots, R_{\max}]$. The average expected number of success is calculated for each address sequence possible and compared to the theoretical upper bound. We limit our exhaustive search to a maximum of $R_{\max} = 15$ as $\binom{R_{\max}}{N}$ increases as the factorial of $R_{\max}$ increases and an exhaustive search is computationally not feasible. Figure 7 shows the ratio of practical bound to that of theoretical bound in y-axis as a function of p_t, for $R_{\max} = 15$ and number of sources N ranging from 5 to 11. At smaller values of N, the probability of fewer "bit 1" collisions is higher, i.e., the probability of collision of $\frac{N}{2}$ or fewer sources is higher than the collision of more than $\frac{N}{2}$ sources and the probability of receiver saturation is small. As N increases, the probability of collision of $\frac{N}{2}$ or more sources increases, further increasing the probability of receiver saturation, which in turn decreases the effective throughput. For $R_{\max} = 15, N = 5$, the practical bound is 99.3% of the theoretical bound at $p_t = 0.1$ and 91.2% on an average. Theoretical upper bound, on an average is 95.8% at $p_t = 0.1$, 78.9% at $p_t = 0.5$ and 91.3% at

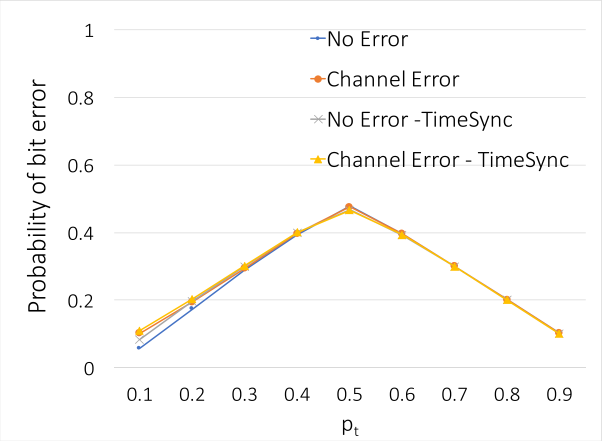


Fig. 8. Bit error rate performance of ADMA, load aware DR,$R_{\max} = 30$, $N = 14$.

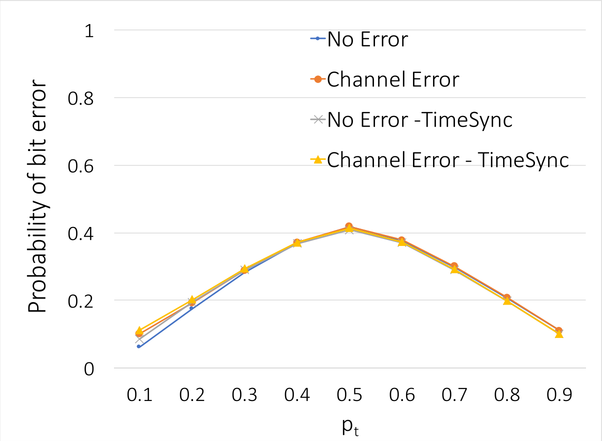


Fig. 9. Bit error rate performance of ADMA, load aware DR,$R_{\max} = 70$, $N = 15$.

$p_t = 0.9$ for $R_{\max} = 15$ and N varying from 5 to 11. At very low and very high p_t, the theoretical bound is close to 90% of the practical upper bound.

B. Load Aware Receiver

From Equation (1), we find that the parameters influencing CoRE are, 1) receiver design, 2) per-user probability of transmitting bit 1, p_t, 3) maximum receivable amplitude $R_{\max}$, and 4) number of sources N. We evaluate the performance of ADMA by varying p_t, N and $R_{\max}$. In Figure 6, the throughput performance of PR is compared against that of DR for $N = 5, R_{\max} = 15$ under idealized conditions. The throughput achieved using DR outperforms that of PR for different sequences, which is attributed to the objective of each receiver design. The goal of DR is to reduce per-user bit error and increase the average per-user throughput, whereas PR maximizes the joint throughput of the network. We evaluate the performance of ADMA using DR in the rest of the section due to its low time complexity in implementation. We use bit error rate as the metric to evaluate ADMA for different sets of $R_{\max}, N$ and p_t, which is then used to calculate per-user throughput. We calculate the expected number of successful bits received per user from bit error and average over total time taken to calculate the average throughput. Figures 8 and 9 shows the bit error rate performance of ADMA for increasing values of p_t when receiver is aware of p_t. Four curves in each graph plots the bit error rate performance of ADMA in the presence and absence of channel error, and presence and absence of time synchronization. As discussed in Section IV, the receivers are designed to decode samples and do not require any time synchronization. Channel error is introduced by adding up to 20% amplitude error to 10% of the bits transmitted by each source. Figures 8 and 9 shows that ADMA is robust to asynchronous transmissions even in the presence of channel error. When integer amplitudes are assigned, DR chooses a configuration with the most probable bit for each source. The configuration chosen by DR for received amplitude y differs only by few sources for amplitudes close to y, making ADMA robust to channel error. For $R_{\max} = 30, N = 14$, the throughput performance of ADMA is 92.5% of the upper bound at $p_t = 0.1$ and on an average 80.5% of the upper bound. For $R_{\max} = 70, N = 15$, the throughput performance of ADMA is 92.4% of the upper bound at $p_t = 0.1$ and on an average 82.3% of the upper bound. As shown in Figure 7, the practical bound on an average is close to 90% of the upper bound. Extrapolating this result to the performance at $R_{\max} = 70, N = 15$ and $R_{\max} = 30, N = 14$, the average performance of ADMA is close to 90% of the absolute maximum.

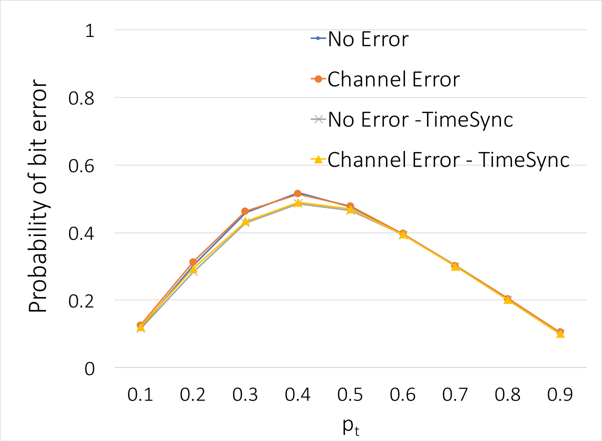


Fig. 10. Bit error rate performance of ADMA, load unaware DR, $R_{\max} = 30$, $N = 14$.

C. Load Unaware Receiver

In the above results, we assume that the receiver is aware of the load distribution (p_t). It may not be possible in a practical system to estimate p_t accurately. We propose a load unaware receiver that updates the decoding table for $p_t = 0.5$, when p_t is unknown. In a load unaware scenario, ADMA always chooses the integer address, i.e., $\{1, 2, 3, \ldots\}$. Figures 10 and 11 shows that the load unaware receiver does not affect the bit error rate performance of ADMA. For $R_{\max} = 30, N = 14$, the throughput performance of ADMA is 88.5% of the upper bound at $p_t = 0.1$ and 76.1% of the upper bound on an average. For $R_{\max} = 70, N = 15$, the throughput performance of ADMA is 91% of the upper bound at $p_t = 0.1$

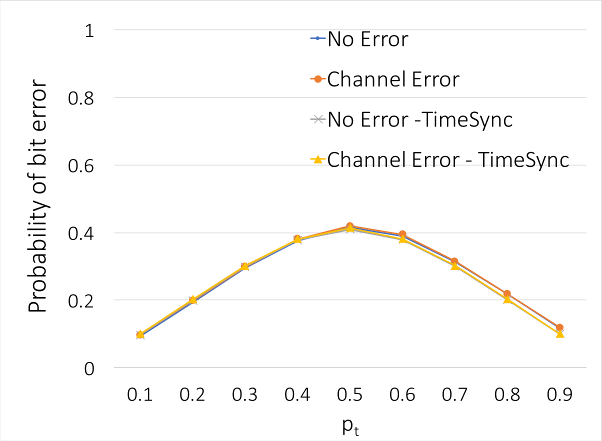


Fig. 11. Bit error rate performance of ADMA, load unaware DR, $R_{\max} = 70$, $N = 15$.

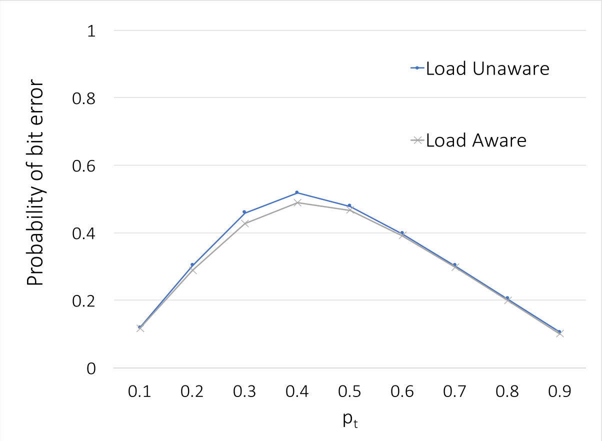


Fig. 12. Bit error rate performance of ADMA in nanoNS3, $R_{\max} = 30$, $N = 14$.

and 81.4% of the upper bound on an average. We observed that the integer sequence has the best performance on an average when the receiver decodes using $p_t = 0.5$, simplifying amplitude assignment algorithm. On average, the performance of load unaware DR using integer sequence remains within 95% of the load aware DR. Also, we note that the performance of integer sequence is within 99% of the algorithm performance on average.

D. ADMA With Bacteria Receiver in Microfluidic Channel

So far, we evaluated the performance of ADMA assuming zero sampling and demodulation error. Here, we implement and evaluate ADMA in nanoNS3 to simulate a practical bacterial communication system with source errors, channel errors and sampling and demodulation errors. We also show using experimental results and the receiver response model of [22] in nanoNS3 that the inverse of the receiver model can be used to demodulate the received signal with high accuracy. We implement the bacterial receiver response derived in [22] that models the performance of receiver bacteria in a microfluidic chip. The receiver model outputs the GFP response of the bacteria in the chamber for a given input. nanoNS3 generates a train of rectangular pulses for different sources and sums the signals in the channel in time-domain. A channel attenuation model that attenuates the amplitude of the signal in the channel introduces a 20% amplitude attenuation per source. The cumulative signal (from multiple sources) is input to the receiver model, which generates the response of the receiver for the corresponding input. nanoNS3 builds an *inverse* of the model numerically, taking GFP response as input and outputs an estimate of the molecular signal. The output of *inverse model* is a time sequence of an estimated amplitude sample. Sampling and demodulation error is introduced at the receiver by varying "k parameters" of the inverse model which define different rate constants in the receiver response and varying them introduces random errors at the receiver. Each source is assigned a bit duration of 50 min and sampled every 10 min generating five samples. The samples from the *inverse model* are divided into blocks of 5 (5 samples per bit) and input to the decoder that outputs bit 1 if the average amplitude of a block of samples is greater than the average, and bit 0 if below. For example, a time sequence output of DR block for user u_1 with amplitude $a_1, a_1, a_1, a_1, 0, a_1$, is decoded as bit 1 and $0, 0, 0, a_1, a_1$ is decoded as bit 0.

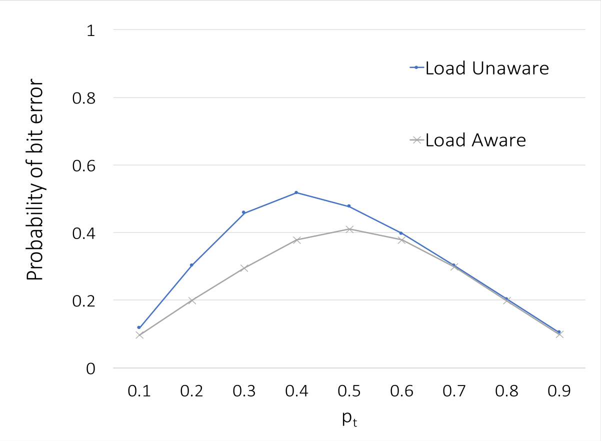


Fig. 13. Bit error rate performance of ADMA in nanoNS3, $R_{\max} = 70$, $N = 15$.

nanoNS3 thus uses *forward response* model to generate GFP response of receiver bacteria and the *inverse model* to demodulate. The results from nanoNS3 thus represents a practical system by considering non-ideal conditions at the source, channel, demodulator and the receiver. Figures 12 and 13 show the bit error rate performance of ADMA in nanoNS3 for $R_{\max} = 30, N = 14$ and $R_{\max} = 70, N = 15$. The load unaware and load aware results are very close to each other, as actual input distribution $p_t \geq 0.5$. This is due to receiver saturation at $R_{\max}$. All amplitudes greater than $R_{\max}$ are received as $R_{\max}$. For $p_t \geq 0.5$, the probability of receiving $R_{\max}$ or higher is high, i.e., the receiver observes $R_{\max}$ with high probability and the number of partitions mapped to $R_{\max}$ is much higher than other received amplitudes. Therefore, even with $p_t \geq 0.5$, the probability of bit 1 and bit 0 is close to 0.5 on receiving $R_{\max}$. For $R_{\max} = 30, N = 14$, the throughput performance of ADMA in nanoNS3 is 90.2% on average, and 99.9% in best case, of that achieved using idealized simulator BSC; at $R_{\max} = 70, N = 15$, throughput of ADMA is 89.1% on an average and a maximum of 99% of that of BSC.

ADMA also acts as a multiple access control (MAC) mechanism. One of the requirements of a MAC protocol is fairness. Using simulations, we also calculate the fairness of *ADMA*. We calculate the Jain's fairness index [33] for different sets of $R_{\max}$, N and p_t for both load aware and load unaware receivers. for $N = 14, R_{\max} = 30$ and $N = 15, R_{\max} = 70$ respectively. Fairness achieved by ADMA with deterministic receiver, on an average is 0.99 indicating high fairness.

VII. Conclusion and Future Work

The following assumptions were made in deriving the best address sequence for a network. We discuss in detail each of the assumptions below.

A. p_t Is Known

As shown in Figure 8, the performance of an addressing sequence depends on the probability of sources transmitting bit 1. To select the best sequence, we must know the approximate range of p_t. Figure 10 plots the performance of an amplitude assignment algorithm when the receiver is unaware of input load; it assumes $p_t = 0.5$. The *integer sequence* under load unaware conditions performs close to that of a load aware deterministic receiver (within 95%). While the knowledge of p_t can improve the performance of the system, when using an *integer sequence* with a load unaware deterministic receiver, there is not a significant decrease in the throughput performance. Thus, p_t does not affect the performance of *ADMA*.

B. p_t Is Same Across Sources

The insights developed in ADMA assume same p_t for all sources. Deriving the probability of success and the best sequence for different p_t across sources is a challenging problem. We showed that even when the receiver is not aware of p_t, throughput performance is not affected significantly. Following the same argument, if sources transmit bit 1 with a different probability, the performance of the system is not significantly affected.

C. Non Empty Data Queue

We implement OOK modulation, where an absence of a signal indicates bit 0. In practice, it is necessary to differentiate between bit 0 and *no data*. We propose the use of start and end-of-frame sequences. A pre-assigned bit sequence can define the start and end of a frame and an absence of signal outside this start and end of frames is considered as no-data.

We propose *ADMA* and demonstrate that the amplitude of the transmitted signal can be used to address sources in a bacterial communication network. We also verify experimentally using genetically engineered bacteria in the microfluidic chip as a proof-of-concept that it is possible to implement *ADMA* in a practical system. The state-of-the-art microfluidic system accommodates only a single receiver. Due to constraints in the system design, the proof-of-concept presented in this paper extrapolates results from a system with one source and one receiver. A microfluidic system design that can accommodate multiple sources and multiple receivers is a work in progress.

We propose a heuristic algorithm to choose a sequence of amplitudes, to minimize collision resolution error. We also propose an optimum receiver that can minimize collision resolution error and maximizes the number of sources accommodated in the network. We derive a theoretical upper bound on the average number of successes using *ADMA*. Finally, we evaluate the proposed algorithm in nanoNS3, a bacterial communication simulator module built on top of NS3; and through simulations, determine that the optimum receiver design is robust to channel errors and receiver errors.

References

[1] I. F. Akyildiz and J. M. Jornet, "Electromagnetic wireless nanosensor networks," *Nano Commun. Netw.*, vol. 1, no. 1, pp. 3–19, 2010.

[2] G. Deligeorgis *et al.*, "Microwave propagation in graphene," *Appl. Phys. Lett.*, vol. 95, no. 7, 2009, Art. no. 073107.

[3] T. Suda *et al.*, "Exploratory research on molecular communication between nanomachines," in *Proc. Genet. Evol. Comput. Conf. (GECCO)*, vol. 25. Washington, DC, USA, Jun. 2005, p. 29.

[4] I. F. Akyildiz, F. Fekri, R. Sivakumar, C. R. Forest, and B. K. Hammer, "Monaco: Fundamentals of molecular nano-communication networks," *IEEE Wireless Commun. Mag.*, vol. 19, no. 5, pp. 12–18, Oct. 2012.

[5] N. Farsad, A. W. Eckford, S. Hiyama, and Y. Moritani, "On-chip molecular communication: Analysis and design," *IEEE Trans. Nanobiosci.*, vol. 11, no. 3, pp. 304–314, Sep. 2012.

[6] W. Zhang, A. M. Asiri, D. Liu, D. Du, and Y. Lin, "Nanomaterial-based biosensors for environmental and biological monitoring of organophosphorus pesticides and nerve agents," *Trends Anal. Chem.*, vol. 54, pp. 1–10, Feb. 2014.

[7] Q. Liu *et al.*, "Cell-based biosensors and their application in biomedicine," *Chem. Rev.*, vol. 114, no. 12, pp. 6423–6461, 2014.

[8] B. L. Bassler, "How bacteria talk to each other: Regulation of gene expression by quorum sensing," *Current Opin. Microbiol.*, vol. 2, no. 6, pp. 582–587, 1999.

[9] P. Melke, P. Sahlin, A. Levchenko, and H. Jönsson, "A cell-based model for quorum sensing in heterogeneous bacterial colonies," *PLoS Comput. Biol.*, vol. 6, no. 6, 2010, Art. no. e1000819.

[10] T. Charrier *et al.*, "A multi-channel bioluminescent bacterial biosensor for the on-line detection of metals and toxicity. Part II: technical development and proof of concept of the biosensor," *Anal. Bioanal. Chem.*, vol. 400, no. 4, pp. 1061–1070, 2011.

[11] J. Stocker *et al.*, "Development of a set of simple bacterial biosensors for quantitative and rapid measurements of arsenite and arsenate in potable water," *Environ. Sci. Technol.*, vol. 37, no. 20, pp. 4743–4750, 2003.

[12] M. Pierobon and I. F. Akyildiz, "A physical end-to-end model for molecular communication in nanonetworks," *IEEE J. Sel. Areas Commun.*, vol. 28, no. 4, pp. 602–611, May 2010.

[13] B. Atakan and O. B. Akan, "On channel capacity and error compensation in molecular communication," in *Transactions on Computational Systems Biology X*. Heidelberg, Germany: Springer, 2008.

[14] A. Einolghozati, M. Sardari, and F. Fekri, "Design and analysis of wireless communication systems using diffusion-based molecular communication among bacteria," *IEEE Trans. Wireless Commun.*, vol. 12, no. 12, pp. 6096–6105, Dec. 2013.

[15] M. Pierobon and I. Akyildiz, "A statistical–physical model of interference in diffusion-based molecular nanonetworks," *IEEE Trans. Commun.*, vol. 62, no. 6, pp. 2085–2095, Jun. 2014.

[16] N.-R. Kim and C.-B. Chae, "Novel modulation techniques using isomers as messenger molecules for nano communication networks via diffusion," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 12, pp. 847–856, Dec. 2013.

[17] M. S. Kuran, H. B. Yilmaz, T. Tugcu, and I. F. Akyildiz, "Modulation techniques for communication via diffusion in nanonetworks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Kyoto, Japan, 2011, pp. 1–5.

[18] M. U. Mahfuz, D. Makrakis, and H. Mouftah, "Spatiotemporal distribution and modulation schemes for concentration-encoded medium-to-long range molecular communication," in *Proc. 25th Biennial Symp. Commun. (QBSC)*, Kingston, ON, Canada, 2010, pp. 100–105.

[19] C. Rose and I. S. Mian, "A fundamental framework for molecular communication channels: Timing & payload," in *Proc. IEEE ICC*, London, U.K., 2015, pp. 1043–1048.

[20] K. V. Srinivas, A. W. Eckford, and R. S. Adve, "Molecular communication in fluid media: The additive inverse Gaussian noise channel," *IEEE Trans. Inf. Theory*, vol. 58, no. 7, pp. 4678–4692, Jul. 2012.

[21] B. Atakan and O. B. Akan, "Single and multiple-access channel capacity in molecular nanonetworks," in *Proc. Int. Conf. Nano-Netw.*, Lucerne, Switzerland, 2009, pp. 14–23.

[22] C. M. Austin *et al.*, "Modeling and validation of autoinducer-mediated bacterial gene expression in microfluidic environments," *Biomicrofluidics*, vol. 8, no. 3, 2014, Art. no. 034116.

[23] M. J. Moore and T. Nakano, "Addressing by beacon distances using molecular communication," *Nano Commun. Netw.*, vol. 2, nos. 2–3, pp. 161–173, 2011.

[24] B. Krishnaswamy *et al.*, "Time-elapse communication: Bacterial communication on a microfluidic chip," *IEEE Trans. Commun.*, vol. 61, no. 12, pp. 5139–5151, 2013.

[25] F. Su, K. Chakrabarty, and R. B. Fair, "Microfluidics-based biochips: Technology issues, implementation platforms, and design-automation challenges," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 25, no. 2, pp. 211–223, Feb. 2006.

[26] J. McDonald *et al.*, "Fabrication of microfluidic systems in poly (dimethylsiloxane) electrophoresis," vol. 21, no. 1, pp. 27–40, 2000.

[27] T. Bohman, "A sum packing problem of Erdös and the conway-guy sequence," *Proc. Amer. Math. Soc.*, vol. 12, no. 12, pp. 3627–3636, 1996.

[28] R. K. Guy, "Sets of integers whose subsets have distinct sums," *North-Holland Math. Studies*, vol. 60, pp. 141–154, Dec. 1982.

[29] *Signal Processing and Detection*. [Online]. Available: https://web.stanford.edu/group/cioffi/doc/book/chap1.pdf

[30] G. E. Andrews, *The Theory of Partitions*. Cambridge, U.K.: Cambridge Univ. Press, 2003.

[31] R. M. Davis, R. Y. Muller, and K. A. Haynes, "Can the natural diversity of quorum-sensing advance synthetic biology?" *Front. Bioeng. Biotechnol.*, vol. 3, 2015.

[32] Y. Jian *et al.*, "nanoNS3: Simulating bacterial molecular communication based nanonetworks in network simulator 3," in *Proc. ACM NanoCom*, New York, NY, USA, 2016, Art. no. 16.

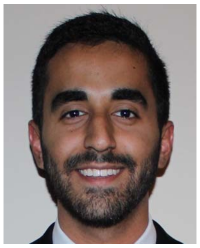
[33] R. Jain, D.-M. Chiu, and W. R. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer system," Eastern Res. Lab., Digit. Equip. Corp., Hudson, MA, USA, DEC Res. Rep. TR-301, 1984.

Bhuvana Krishnaswamy received the B.E. degree in electronics and communication engineering from the College of Engineering, Anna University, Chennai, India, in 2007. She is currently pursuing the Ph.D. degree with the Georgia Institute of Technology, Atlanta, GA, USA. Her research interests include wireless networks, communication systems, and molecular communication.

Yubing Jian received the B.S. (First-Class Hons.) degree with the University of Bristol in 2013. He is currently pursuing the Ph.D. degree in electrical and computer engineering with the Georgia Institute of Technology under the supervision of Prof. R. Sivakumar. His research interests include wireless networks, communication systems, and molecular communication.

Caitlin M. Austin received the bachelor's degree in mechanical engineering and the Ph.D. degree in bioengineering from the Georgia Institute of Technology. Her research interests include microfluidics and bacteria communication. She was a recipient of the National Science Foundation Graduate Fellowship.

Jorge E. Perdomo is currently pursuing the Bachelor of Science degree in biomedical engineering from the Georgia Institute of Technology. His research interests include clinical design, microfluidics, bacterial communication, and artificial intelligence in cyber security. He is a Provost Scholar. He was a recipient of two-time President's Undergraduate Research Award.



Sagar C. Patel received the B.S. degree in biology with a concentration in Business from the Georgia Institute of Technology. He is currently pursuing the Medicine degree with the Philadelphia College of Osteopathic Medicine—GA Campus. His interest in research included microfluidics and biomolecular communication.

Brian K. Hammer received the B.S. degree in biology from Boston College, the M.S. degree in aquatic ecology from the School of Natural Resource and Environment, University of Michigan, and the Ph.D. degree in microbiology from the University of Michigan Medical School. He was a Post-Doctoral Researcher with the Department of Molecular Biology, Princeton University. He is an Assistant Professor with the School of Biology, Georgia Institute of Technology. He was a recipient of the Ruth L. Kirschstein National Research Service Award from the National Institutes of Health in 2003 and the Faculty Early Career Development (CAREER) Award from the National Science Foundation in 2012. His current research interests include microbial genetics, signal transduction, regulatory RNAs, and bacterial quorum sensing.

Craig R. Forest received the B.S. degree in mechanical engineering from Georgia Tech in 2001, and the M.S. and Ph.D. degrees in mechanical engineering from MIT in 2003 and 2007, respectively. He was a Research Fellow in genetics with Harvard Medical School. In 2008, he joined Georgia Tech as an Assistant Professor of mechanical engineering and also holds program faculty positions in bioengineering and biomedical engineering. He conducts research on miniaturized, high-throughput robotic instrumentation to advance neuroscience and genetic science, working at the intersection of bioMEMS, precision machine design, optics, and microfabrication. He is a Co-Founder/Organizer of one of the largest undergraduate invention competitions in the U.S.—The InVenture Prize, and a Founder/Organizer of one of the largest student-run prototyping facilities in the U.S.—The Invention Studio. He was a recipient of the Georgia Tech Institute for BioEngineering and BioSciences Junior Faculty Award in 2010 and was named Engineer of the Year in Education for the state of Georgia in 2013.

Raghupathy Sivakumar received the B.E. degree in computer science from the College of Engineering, Anna University, Chennai, in 1996, and the M.S. and Ph.D. degrees in computer science from the University of Illinois at Urbana-Champaign in 1998 and 2000, respectively. He is the Wayne J. Holman Chair Professor, School of Electrical and Computer Engineering, Georgia Tech. He leads the Georgia Tech Networking and Mobile Computing Research Group, where he and his students do research in the areas of wireless networking, mobile computing, and computer networks. He serves on the editorial board of the IEEE Transactions on Mobile Computing. He served as the Co-Founder, the Chairman, and a CTO for StarMobile, Inc. (currently Powwow), from 2012 to 2016, a Technologist for EMC Corporation from 2011 to 2012 and EG Technology, Inc. (currently Arris), from 2001 to 2004, and the Founder and a CTO of Asankya, Inc. (currently EMC), from 2004 to 2011.