Link Layer Protocol for Molecular Communication Networks

Yaman Sangar
sangar@wisc.edu

Bhuvana Krishnaswamy
bhuvana@ece.wisc.edu

ABSTRACT

Molecular communication networks (MC) consisting of biosensors and bioprocessors have unique characteristics that distinguish them from traditional electromagnetic networks. MC networks are *super-slow* owing to high processing and propagation delays, need algorithms that are *ultra-simple* due to the limited computational capabilities of biological circuits, and are *highly asymmetric* as the receiver/gateway have access to electronic components and other resources. These characteristics demand a re-thinking in the design of communication algorithms for MC networks. In this work, *LMN*, a link layer protocol specifically focusing on the above three characteristics of an MC network is proposed. *LMN* proposes a time interval modulation that reduces the overall communication delay and an embedded medium access control that is simple to implement and offloads computational complexity to the receiver.

KEYWORDS

Link layer, Molecular communication networks, Modulation, Medium Access Control

1 INTRODUCTION

Naturally existing and synthetically engineered biosensors have evolved and developed to sense and respond to biological phenomenon, chemicals, and pathogens, among others. Bacteriophage based sensors to detect pesticides in the environment [1, 2], whole-cell based biosensors that combine living cells and sensors in biomedicine [3] and bio luminescent bacteria to measure toxicity of heavy metals [4] are some examples of biosensors providing access to domains that are otherwise unreachable. The natural sensing mechanisms combined with advancements in synthetic biology to alter/engineer DNA have made live biosensors possible. In addition to sensors, advancements in synthetic biology have also led to the development of computing and processing units using biological entities. A number of synthetically engineered circuits have been developed for oscillators [5, 6], switches [7], sensors [8], and storage [9, 10]. Even

though inter-operability of the individual components is a challenging task, the successful operations and availability of biological circuits and components for data processing makes bacteria a strong candidate to use as computing machines. Genetically engineered bacteria for example, can be used to sense and perform computations on the data sensed [11]. Biosensors, along with bio-processors provide the opportunity to build sensors and processors without electronic components. Currently, biosensors are used primarily for sensing and rely on off-line data collection and post-processing in the laboratory. Though biosensors give access to new domains, data collection in real-time is challenging.

Molecular communication (MC) networks that connect biosensors, allowing them to communicate with each other and/or to a gateway receiver, present a solution for real-time monitoring [12]. However, existing communication networks and protocols are not designed to meet the challenges, constraints, and opportunities that arise from having biosensors as their components. For example, while energy efficiency is an import aspect in the design of algorithms for an EM network, the abundance and availability of biosensors that can utilize natural sources of energy from their ecosystem provides opportunities for the designer to shift the focus on other aspects of the problem. Therefore, a fundamental approach to MC algorithms is required.

The focus of this work is to identify the unique characteristics of a MC network and develop a unified link layer protocol that can realize a multi-sender, multi-receiver implementation. Traditionally, link layer encompasses the physical layer challenges such as modulation, channel or medium geometry, achievable data-rate as well as medium access control and addressing. A number of research works focusing on channel modeling [13, 14], capacity analysis [15], modulation [16], medium access control [17], bit error detection and correction [18], and data compression and storage [19] have been reported in the literature. In order to realize a practical MC network, it is crucial to identify and develop protocols that are compatible with the each other's goals. Specifically, in this work, the unique characteristics of molecular networks are identified as, (1) super-slow (2) ultra-simple (3) highly asymmetric. A link layer protocol for molecular network (**LMN**) that accounts for the above characteristics is proposed in this work. *LMN* proposes a modified time interval modulation technique and an embedded medium access control (MAC) to reduce the complexity of implementation and execution, to reduce the overall delay in the system, by leveraging the asymmetry in the resources available at the sender and the receiver. *LMN* is evaluated in a custom-built python simulator that implements a bacterial communication link model [13].

The rest of the paper is organized as follows: section 2.2 provides a primer on modulation techniques and MAC protocols in molecular networks and section 3.5 details the proposed link layer protocol and the integrated operation of the modulation, addressing and MAC, and the decoder design. In section 4.5 the performance of
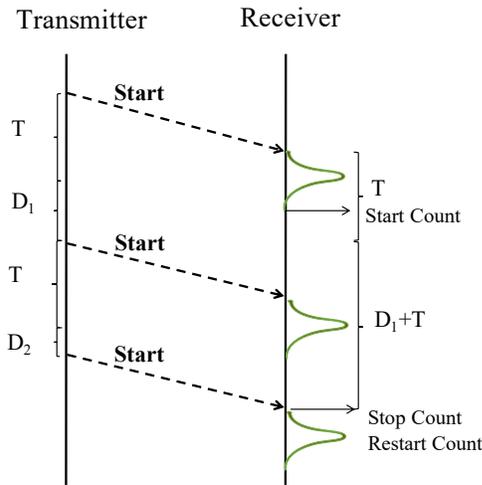
Figure 1: Illustration of Time Elapse Communication

LMN is presented and section 5 concludes the paper following discussions on future work.

2 BACKGROUND AND MOTIVATION

2.1 Modulating time intervals

On-Off Keying (OOK) is widely used in MC network research. OOK transmits a rectangular signal of a given amplitude and duration to transmit bit 1 and no signal for a given duration to transmit bit 0. OOK addresses one of the three characteristic of MC network viz., *ultra-simple*. However, OOK is not suitable for *super-slow* networks since the total time to convey information increases with the number of bits. Majority of the modulation schemes designed for MC networks such as Concentration Shift Keying(CSK) [16], Molecule Shift Keying(MoSK) [20] rely on the energy of the signal to convey information. Therefore, the number of bits transmitted is proportional to the number of signals transmitted in the channel. Higher the number of bits to be transmitted, higher the number of signals transmitted, higher the delay, making it unsuitable for *super-slow* networks.

Time Elapse Communication(TEC) [21] is a modulation technique that encodes information in the time interval between symbols and focuses on minimizing the overall communication delay. TEC reduces the number of signals transmitted by encoding information in time interval. A data value *d* to be transmitted is conveyed by sending exactly two signals viz., *start* and a *stop*. *d* modulates the number of clock cycles between the *start* and *stop* signal. Since information is encoded in the time interval, it is sufficient for the sender and the receiver to have the same clock rates and need not be synchronized. Fig 1 shows an illustration of TEC, where the *start* signal triggers clock counting at the receiver and the next *start* signal triggers the receiver to stop counting and restart counting the next data. The number of clock cycles elapsed between the trigger conveys the information. In *super-slow* networks, processing delays are much higher and therefore, clock rates can be higher than transmission rates. For e.g., in a bacterial communication system, processing delays at the receiver is in the order of hours while bacterial clocks of the order of 1 Hz have been built [22]. Therefore,

by offloading communication delays to clocks, overall delay us reduced and throughput improved. The number of signals required is a constant (2 - start and stop) per message, irrespective of the number of bits per message. TEC improves the throughput by over an order of magnitude for a clock rate of 1 Hz compared to OOK.

However, TEC does not have a mechanism to distinguish between the *start* and the *start* signals. TEC assumes that the sender are always backlogged i.e., it always has data to send. Therefore, the start of the next data D_2 also indicates the stop of the current data D_1. In real world applications, biological sensors are used to periodically monitor a variable of interest and/or triggered by events of interest. Typical sensing applications do not turn the sensor **ON** all the time. In addition to that, since the start and the stop signals are simply bit 1 s, implementing TEC in a multi-sender network heavily relies on an addressing and MAC algorithm to uniquely identify the sender. In the following section, we analyze addressing and MAC solutions for the MC network to integrate with TEC as a modulation technique.

2.2 Embedded addressing and multiple access

Addressing in traditional communication system is achieved by using addressing fields in packet header. Addressing fields can be implemented in two ways using TEC. 1) The address of the sender is encoded in the *start* signal. For example, a sender with address 1100 conveys 1100 using energy based modulation such as OOK or CSK and then starts counting data value D_i before sending a *stop* signal, also represented by 1100. Such an addressing scheme will increase the number of symbols transmitted by sender (address fields). As the number of senders in the network increases, the number of bits required to address each sender increases further increasing the probability of address bit collisions. Such an addressing bit based approach will require a MAC protocol to coordinate the transmission of multiple senders to reduce this probability of collisions. An addressing mechanism that inherently solves MAC is desirable to gain the benefits of TEC as the underlying modulation technique. Code Division Multiple Access (CDMA) is an example of such an addressing mechanism that allows multiple senders to share the channel simultaneously, without being affected by collisions. However, CDMA is a spread spectrum technique and increases the number of signals transmitted in the channel, which goes against the motivation of TEC to reduce the number of signals in the channel.

An *embedded addressing* scheme that solves addressing and MAC for bacterial communication networks has been proposed in [23, 24]
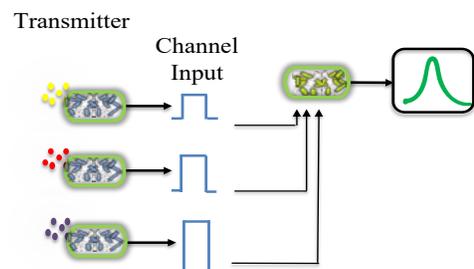

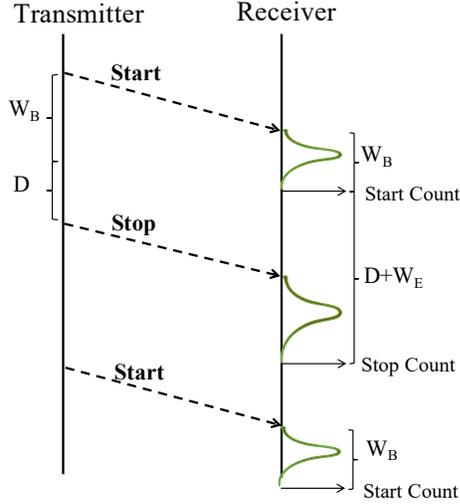**Figure 2: Illustration of Amplitude Division Multiple Access**

Figure 3: Timing interval modulation in *LMN*

as Amplitude Division Multiple Access(ADMA). ADMA proposes to embed address in the amplitude of the signal transmitted. Fig 2 illustrates ADMA in a multi-sender single-receiver network. ADMA assigns a unique amplitude to each sender, which is then used to transmit a rectangular signal of the assigned amplitude for a duration W to transmit bit 1 and no signal for a duration W to transmit bit 0. When multiple senders transmit simultaneously, the receiver receives the sum of amplitudes. ADMA proposes an amplitude assignment algorithm to reduce the probability of error in decoding. The receiver implements a maximum aposteriosri (MAP) decoder, with the knowledge of the amplitudes assigned and the channel model. ADMA is easy to implement and require very little processing, thus targeting *ultra-simple* challenge of MC networks. By eliminating the need for addressing bits, ADMA reduces the delay and addresses the *super-slow* characteristic of MC network. However, since ADMA uses OOK as the underlying modulation, ADMA as a link layer protocol does not reduce the total delay. [1]

3 LINK LAYER PROTOCOL

The focus of this work is to develop a link layer protocol for an MC network that is practical and simple to implement.

3.1 Modulation

Based on the discussions on modulations, *LMN* identifies and addresses the challenges in extending time interval modulation (TIM) to a multi-sender network using amplitude division multiple access. As discussed in the section above, TEC does not answer the following questions that arise in a multi-sender network.

- How to distinguish *start* and *stop* signals?
- How to distinguish senders?

TEC assumes the sender to be always backlogged, which is not valid in monitoring applications. *LMN* proposes to use signal duration W, a characteristic of the start(or stop) signal to differentiate them

_____
[1]Embedded addressing can only be used as a local address. A global address must be assigned one-time during setup and further communication use embedded addressing.

as below,

$$start(t) = \begin{cases} A, 0 \leq t \leq W_B \\ 0, \text{otherwise} \end{cases} \quad stop(t) = \begin{cases} A, 0 \leq t \leq W_E \\ 0, \text{otherwise} \end{cases}$$

In other words, when the transmitter has a data value D to send, it transmits a molecular signal of duration W_B, and waits for D slots and transmits a molecular signal of duration W_E as shown in Fig 3. On decoding a molecular signal, the receiver deciphers the amplitude and the duration of the signal. If the decoded duration is W_B, the receiver starts counting and stops counting if it is W_E. Since *LMN* uses two signals to represent a message D, the data rate of *LMN* is lower than that of TEC as shown in Fig 8. This is attributed to the increase in total time required to transmit an extra signal (Stop) to communicate a value D. On an average, a TEC transmitter sends one signal per message, while a *LMN* transmitter sends two signals per message (Fig 3). This improvement on TEC is practical and easy to implement, addressing both *super-slow* and *ultra-simple* characteristics of MC networks.

3.1.1 Error Detection. The proposed *LMN* modulation enhances the ability to detect bit errors at the receiver. Loss of the start or stop signal is detected by maintaining the status of the previous message i.e., a start followed by a start indicates an error/loss in the stop signal and vice versa for stop signal. By maintaining the status of the previous signal, a *LMN* receiver is able to detect an error in the start or stop signal. On detecting a start followed by a start, the receiver ignores the previous start and the corresponding time; it begins counting from the latest start signal and stores the previous message as corrupted. TEC assumes a closed system and does not account for signal loss.

LMN includes an unbounded timing error using apriori knowledge of the network. During system setup, *LMN* setup d_{max}, the maximum message length (duration) of the link. In other words, the maximum value of the message is known during initialization. The receiver uses the apriori knowledge of d_{max} to detect propagation delay error. Analogous to timeout in wireless systems, the receiver sets a timeout TO equal to the sum of d_{max} and $maxd_{prop} + 2d_{tsm}$. d_{prop} and d_{tsm} are random variables that indicate the propagation time and the transmission time of the start (and stop) signals. When the receiver counter goes beyond TO, the receiver detects an error and resets the counter and waits for the next start signal. Such a timeout-based error detection is especially relevant in applications where sensing events occur rarely. For example, bacterial sensor detecting pathogen concentrations in an environment reports only when the concentration is above a threshold. Timeout based error detection detects an error much before waiting for the next start signal. Such an error detection mechanism is achieved at the cost of reduced throughput (Fig 8). The data rate of TIM in *LMN* is,

$$\text{Data-rate} = \frac{d}{f(d) + W_B + W_E} \quad (1)$$

where f(d) is the number of clock cycles corresponding to d and W_B and W_E is the time to transmit a start and a stop signal respectively.

3.2 Medium Access Control

The second challenge identified in using TEC in a multi-sender network is MAC protocol. TEC proposes a modulation technique for a
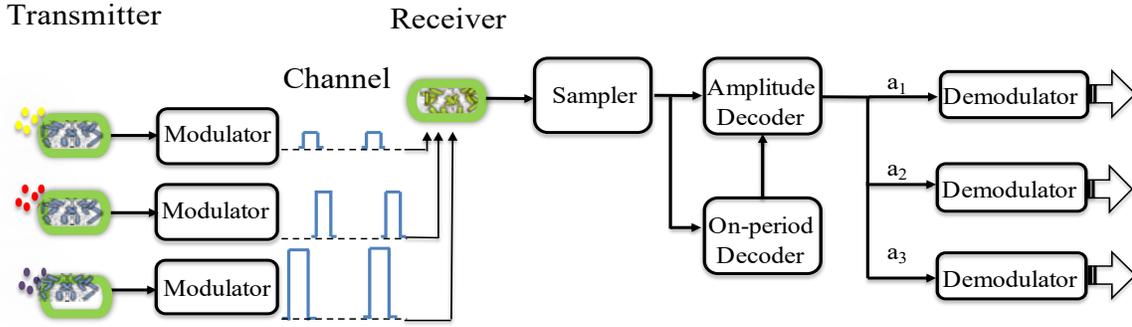
Figure 4: Integrated link layer solution for molecular communication network

single link and relies on upper layer protocols to handle addressing and MAC. As discussed in section 2.2, additional address bits will increase the number of signals transmitted and hence the probability of collisions. To reduce the number of symbols and improve network throughput, *LMN* uses ADMA, an embedded addressing and MAC protocol proposed in [23, 24], that encodes address in the amplitude of the signal transmitted. An illustration in Fig 2 provides an overview of ADMA, where each sender s_i is assigned a unique amplitude a_i. The sender s_i transmits bit 1 by sending a rectangular signal of amplitude a_i for a predetermined duration W and bit 0 by not sending a signal for W. ADMA sender experiences collisions only when transmitting bit 1 since bit 0 is communicated by an absence of a signal. Additionally, ADMA also assume that the senders always have data to send. ADMA does not distinguish between bit 0 and no data.

Since only bit 1s experience collision, a higher probability of a sender transmitting bit 1 leads to an increase in collision rate. Also, with the increase in the number of senders N, ADMA suffers from high collision and low probability of success in decoding the sender address. It has been shown in [23] that the probability of success is close to 90% only at very low probability of bit 1 being transmitted. ADMA offloads the burden of achieving very low probability of bit 1 to coding techniques. Complex operations like matrix multiplication are required to implement coding techniques and are challenging to implement using a biological sender. Such a coding technique that minimizes bit 1 in the channel could also decrease the effective throughput. *LMN* proposes to address the collision challenge in ADMA by integrating ADMA with time interval modulation, a coded modulation that minimizes the number of signals (and hence the number of bit 1s) transmitted in the channel. Therefore, by integrating TIM with ADMA *LMN* can reduce the probability of collisions, while also improving the link and network throughput.

3.3 Integration Operation

Integrating TIM with ADMA, LMN *proposes a link layer protocol for MC networks to address its unique characteristics viz., super-slow and ultra-simple by leveraging the high asymmetry in resources. LMN* assigns a unique amplitudes a_i to each sender, when is then used to transmit the *start* and the *stop* signals, with different durations W_B and W_E respectively as illustrated in Fig 4. Therefore, the start (and stop) signals of each sender can be uniquely identified using their

amplitude and the start On receiving an amplitude a_i, the receiver decodes the duration the signal. A received pulse of amplitude a_i for a duration W_B triggers counter for sender s_i and a pulse of amplitude a_i for a duration of W_E indicates end of message for sender s_i. Thus, the amplitude, duration, and the time interval between symbols are combined to build the link layer protocol of a molecular communication network. The modified signals are,

$$start_i(t) = \begin{cases} a_i, 0 \le t \le W_B \\ 0, \text{otherwise} \end{cases} \quad stop_i(t) = \begin{cases} a_i, 0 \le t \le W_E \\ 0, \text{otherwise} \end{cases}$$

where $i \in [1, N]$ is the index of the sender. With the inherent reduction in the number of signals per message transmitted (TIM) and the reduced number of address bits (ADMA), *LMN* addresses the unique characteristics of MC networks.

3.4 Amplitude Assignment

In [23], ADMA proposes a heuristic algorithm to assign amplitudes to individual senders with the knowledge of the probability of bit 1. It was shown that at low probabilities of bit 1, the probability of collisions is small and therefore an integer amplitude assignment would perform the best. Let $S = z_1, z_2, z_3, ...z_R$ be the set of the amplitudes that can be uniquely identified by the receiver where $z_j - z_i = \delta$ for all $i \ne j$ is the minimum step-size of the decodable amplitudes. Then S can be represented as the set of integers with a common divisor δ (except z_1). In the rest of the analysis, the amplitudes will be treated as integers without loss of generalization. *LMN* assigns amplitudes from the range of available amplitudes in an increasing order of magnitude. *LMN* proposes to choose amplitudes such that the minimum difference between adjacent amplitudes is larger than the amplitude error in the channel.

3.5 Decoder Design

As shown in Fig 4, the receiver consists of a sampler, amplitude decoder, period decoder, and the demodulator. For a predetermined sampling duration, the sampler output discrete samples of the continuous received amplitude, which is then fed to the amplitude and the on-period decoder. The *on-period decoder* filters for the duration during which an amplitude remains constant. If this duration is W_B, the received signal is a start signal and if it is W_E, it is a stop signal. The samples are also fed to the *amplitude decoder*, which identifies the transmitted amplitudes from the received sum. The
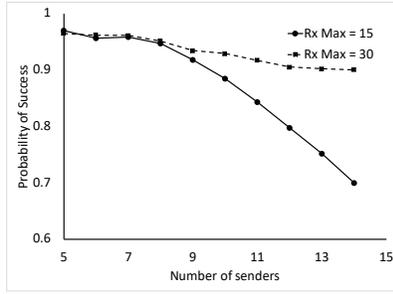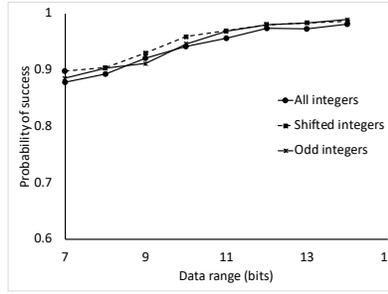
Figure 5: Vs network size
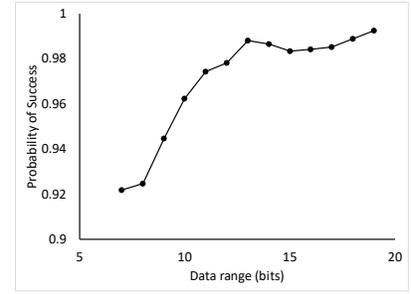

Figure 6: Vs Amp. Assigned

Figure 7: Vs payload length

Probability of success of *LMN*

amplitude decoder can recover from collisions between a start and a stop signal. Since start and stop signals are of different durations, the amplitude decoder looks for amplitude changes in the received sum. If the received amplitude is a_i and the output of on-period decoder is W_B, the receiver triggers the counter (demodulator) corresponding to the sender s_i to start counting and vice-versa for W_E. The decoder proposed in [23] is used as the amplitude decoder and the decoder proposed in [24] is used for on-period decoder.

4 PERFORMANCE EVALUATION

LMN was simulated using a custom-built simulator using Python that implements the experimentally verified model of a receiver bacterial colony developed in [13] as the physical layer module. An additive uniformly distributed noise channel was assumed to account for channel induced amplitude errors. The amplitudes assigned to each sender is spaced out to correct for these errors i.e., in a channel with 10% amplitude error, the minimum difference between adjacent amplitude a_i and a_j is $\geq 0.2^*a_i$. Each data point presented is averaged over 1000 simulations. It is assumed that the receiver saturates beyond a threshold amplitude. The receiver response remains the same for amplitude above this threshold; hence it affects the decoding performance of *LMN*.

4.1 Throughput performance

TIM uses two signals to convey a data value d, while TEC amortizes stop of current signal in the start of next signal. Thus, the time to transmit d for TIM is higher than that of TEC. Fig 8 compares the average throughput improvement of TIM and TEC over OOK. Unlike OOK, the throughput performance of TIM varies with the range of values the message can take (message length). In Fig 8, the x-axis is the number of bits required to represent a message i.e., a data range of 5 implies that the message is uniformly chosen between 0 and $2^5 - 1$. It should be noted that the throughput initially increases with input data range and drops after a maximum point. It has been shown that [25] the maximum throughput is achieved when the data value is comparable to the time to transmit start and stop signals, after which the time to transmit is exponentially higher than $W_B + W_E$. The maximum data range is thus a function of the durations and the channel. *LMN* identifies the data range that maximizes throughput during system setup.
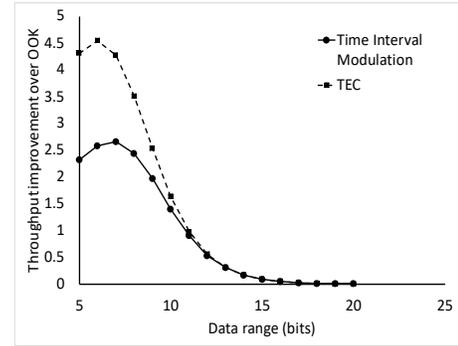

Figure 8: Throughput performance of *LMN*

4.2 Decoding accuracy

The average probability of success in decoding individual senders by *LMN* in a multi-sender network with eight senders and a single receiver is shown in Fig 7, as a function of data range. The larger the data range, higher is the silence period between the start and stop signals. An increase in this silence period results in reduced collisions. It is clear from Fig 7 that the probability of success in decoding transmitted message increases with an increase in data range. However, throughput decreases for larger data ranges. A tradeoff between throughput and data accuracy must be considered is choosing the data range. Comparing this to the bit error rate performance of ADMA, for p_t, the probability of an ADMA sender transmitting bit 1 equal to 0.5, the probability of success is approximately 0.85 in a network of 14 users.

4.3 Network size

With an increase in the number of senders, the number of signals transmitted in the channel to the receiver increases, affecting the decoding accuracy. Due to its additive nature, molecular communication channel can saturate the receiver with an increase in the number of senders. Fig 5 plots the average probability of success in decoding data for increasing number of senders for different receiver saturation levels. As expected, with increasing number of senders, the probability of collision increases and hence the probability of success decreases. It must be noted the rate of decrease in the probability of success is lower at higher receiver saturation. In a network of 14 senders, with a receiver saturation amplitude as 15,
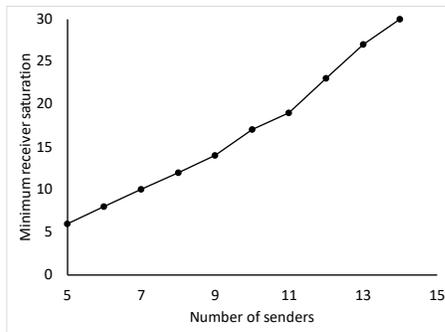
Figure 9: Minimum receiver saturation amplitude required to achieve a bit error rate < 0.1

the receiver saturates often, leading to decoding errors, while that at a receiver saturation of 30 is relatively unaffected by a network size of 14 senders. Therefore, in orcer to accommodate a larger network size, a *LMN* receiver must have a higher saturation amplitude.

4.4 Amplitude assignment

The average probability of success of different amplitude assignment strategies is plotted in Fig 6. *All integers* assigns increasing amplitude from the sequence $S : 1, 2, 3, …R$, *Shifted integers* assigns amplitudes $k, k + 1, k + 2, .., 2k − 1$ if the number of senders=k. This sequence has the property that the sum of two or more entries in this sequence is not an entry of the sequence i.e., if two or more senders collide, it will not be decoded as another sender. Since collisions are less likely in *LMN* network, *Shifted integers* can improve the throughput performance. *Odd integers* assigns odd integers beginning with 1 to each sender. By spacing out the amplitudes, odd integers can further reduce the impact of amplitude errors. It must be noted that the performance of all these amplitude sequences are close to each other. This is due to the fact that time interval modulation reduces the number of signals transmitted and number of collisions. Since amplitude assignment is useful in recovering from multi-sender collisions, their impact in *LMN* is not significant.

4.5 System requirements

As mentioned above, the network size depends on the receiver saturation amplitude. Fig 9 plots the minimum saturation amplitude at the receiver to achieve a minimum 90% probability of success as a function of network size. As the network size increases, the saturation requirements increases. The bottleneck on the network size (number of senders) is the receiver saturation amplitude.

5 CONCLUSIONS

LMN proposes a link layer protocol that maximizes network throughput by offloading communication time to clock counting and embedding address in the amplitude of the signals. Both time interval modulation and embedded addressing reduces the number of signals transmitted in the channel, thus reducing the probability of collisions. *LMN* can therefore achieve a high probability of success in a multi-sender network.

REFERENCES

[1] Amit Singh, Somayyeh Poshtiban, and Stephane Evoy. Recent advances in bacteriophage based biosensors for food-borne pathogen detection. *Sensors*, 2013.
[2] Weiying Zhang, Abdullah Mohamed Asiri, Deli Liu, Dan Du, and Yuehe Lin. Nanomaterial-based biosensors for environmental and biological monitoring of organophosphorus pesticides and nerve agents. *TrAC Trends in Analytical Chemistry*, 2014.
[3] Qingjun Liu, Chunsheng Wu, Hua Cai, Ning Hu, Jun Zhou, and Ping Wang. Cell-based biosensors and their application in biomedicine. *Chemical reviews*.
[4] Thomas Charrier, Cyrille Chapeau, Loubna Bendria, 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. *Analytical and Bioanalytical Chemistry*, 2011.
[5] Michael B Elowitz and Stanislas Leibler. A synthetic oscillatory network of transcriptional regulators. *Nature*, 403(6767):335, 2000.
[6] Timothy S Gardner, Charles R Cantor, and James J Collins. Construction of a genetic toggle switch in escherichia coli. *Nature*, 403(6767):339, 2000.
[7] Piro Siuti, John Yazbek, and Timothy K Lu. Synthetic circuits integrating logic and memory in living cells. *Nature biotechnology*, 31(5):448, 2013.
[8] Mirella Di Lorenzo, Alexander R Thomson, Kenneth Schneider, Petra J Cameron, and Ioannis Ieropoulos. A small-scale air-cathode microbial fuel cell for on-line monitoring of water quality. *Biosensors and Bioelectronics*, 2014.
[9] Denise M Wolf, Lisa Fontaine-Bodin, Ilka Bischofs, Gavin Price, Jay Keasling, and Adam P Arkin. Memory in microbes: quantifying history-dependent behavior in a bacterium. *PLOS one*, 3(2):e1700, 2008.
[10] Fahim Farzadfard and Timothy K Lu. Genomically encoded analog memory with precise in vivo dna writing in living cell populations. *Science*, 2014.
[11] Mark Mimee, Phillip Nadeau, Alison Hayward, Sean Carim, Sarah Flanagan, Logan Jerger, Joy Collins, Shane McDonnell, Richard Swartwout, Robert J Citorik, et al. An ingestible bacterial-electronic system to monitor gastrointestinal health. *Science*, 360(6391):915–918, 2018.
[12] Ian F. Akyildiz, Faramarz Fekri, Craig R Forest, Brian K Hammer, and Raghupathy Sivakumar. Monaco: Fundamentals of molecular nano-communication networks (invited paper). *IEEE Wireless Communications Magazine, Special Issue on Wireless Communications at the Nano-Scale*.
[13] Caitlin M Austin, William Stoy, Peter Su, Marie C Harber, J Patrick Bardill, Brian K Hammer, and Craig R Forest. Modeling and validation of autoinducer-mediated bacterial gene expression in microfluidic environments. *Biomicrofluidics*, 2014.
[14] Tal Danino, Mondragon-Palomino, Octavio, Tsimring, et al. A synchronized quorum of genetic clocks. *Nature*, 2010.
[15] Arash Einolghozati, Mohsen Sardari, and Faramarz Fekri. Design and analysis of wireless communication systems using diffusion-based molecular communication among bacteria. *Wireless Communications, IEEE Transactions on*, 2013.
[16] Mehmet S Kuran, Huseyin Birkan Yilmaz, Tuna Tugcu, and Ian F Akyildiz. Modulation techniques for communication via diffusion in nanonetworks. In *2011 IEEE international conference on communications (ICC)*, pages 1–5. IEEE, 2011.
[17] Massimiliano Pierobon and I Akyildiz. A statistical-physical model of interference in diffusion-based molecular nanonetworks. 2014.
[18] Alessio Marcone, Massimiliano Pierobon, and Maurizio Magarini. Parity-check coding based on genetic circuits for engineered molecular communication between biological cells. *IEEE Transactions on Communications*, 2018.
[19] SM Hossein Tabatabaei Yazdi, Yongbo Yuan, Jian Ma, Huimin Zhao, and Olgica Milenkovic. A rewritable, random-access dna-based storage system. *Scientific reports*, 5:14138, 2015.
[20] Na-Rae Kim and Chan-Byoung Chae. Novel modulation techniques using isomers as messenger molecules for nano communication networks via diffusion. *IEEE Journal on Selected Areas in Communications*, 31(12):847–856, 2013.
[21] Bhuvana Krishnaswamy, Caitlin M Austin, J Patrick Bardill, Daniel Russakow, Gregory L Holst, Brian K Hammer, Craig R Forest, and Raghupathy Sivakumar. Time-elapse communication: Bacterial communication on a microfluidic chip. *Communications, IEEE Transactions on*.
[22] Leo Baumgart, William Mather, and Jeff Hasty. Synchronized dna cycling across a bacterial population. *Nature genetics*, 49(8):1282, 2017.
[23] Bhuvana Krishnaswamy, Yubing Jian, Caitlin M Austin, Jorge E Perdomo, Sagar C Patel, Brian K Hammer, Craig R Forest, and Raghupathy Sivakumar. Adma: Amplitude-division multiple access for bacterial communication networks. *IEEE Transactions on Molecular, Biological and Multi-Scale Communications*, 2017.
[24] Bhuvana Krishnaswamy and Raghupathy Sivakumar. Amplitude-width encoding for error correction in bacterial communication networks. In *Proceedings of the 5th ACM International Conference on Nanoscale Computing and Communication*, page 25. ACM, 2018.
[25] Venkat Anantharam and Sergio Verdú. Bits through queues. *IEEE Transactions on Information Theory*, 1996.