# *When Bacteria Talk:*
# Time Elapse Communication for Super-Slow Networks

Bhuvana Krishnaswamy [a], Caitlin M. Henegar [b], J. Patrick Bardill [c], Daniel Russakow [b], Gregory L. Holst [b]
Brian K. Hammer [c], Craig R. Forest [b], Raghupathy Sivakumar [a]
[a] School of Electrical and Computer Engineering, [b] George W. Woodruff School of Mechanical Engineering, [c] School of Biology,
Georgia Institute of Technology, Atlanta, GA, USA

***Abstract*—In this work we consider nano-scale communication using bacterial populations as transceivers. We demonstrate using a microfluidic test-bed and a population of genetically engineered *Escherichia coli* bacteria serving as the communication receiver that a simple modulation like on-off keying ($OOK$) is indeed achievable, but suffers from very poor data-rates. We explore an alternative communication strategy called time elapse communication ($TEC$) that uses the time period between signals to encode information. We identify the severe limitations of $TEC$ under practical non-zero error conditions in the target environment, and propose an advanced communication strategy called smart time elapse communication ($TEC$-$SMART$) that achieves over a 10x improvement in data-rate over OOK.**

***Index Terms*—Molecular communication, On-Off Keying, Time Elapse Communication**

## I. INTRODUCTION

Nano-scale communication strategies can be categorized into two broad domains depending upon their target environment: *electromagnetic communication (EM)* at the nano-scale involves the extension of traditional EM based communication techniques for use in inorganic or non-biological applications [1], [2]; and *molecular communication* involves strategies (typically bio-inspired) for use in biological applications [3]–[5]. In recent years, bacteria have emerged as promising candidates for nano-machines in biological applications. Bacteria are prokaryotic microorganisms, typically about 1 $\mu$m in size, that are well-studied and understood in terms of morphology, structure, behavior, and genetics. Genetic engineering of bacteria to introduce or delete DNA for specific traits (e.g., bioluminescence, motility, adhesion, etc.) has enabled recent advancements in synthetic biology. This has rendered bacteria as candidates to build nano-machines that may be used, with appropriate manipulation, in biological applications such as toxicology, biofouling, biosensing, etc. Furthermore, bacterial cells naturally behave as transceivers that interact with one another by transmitting and receiving nanometer-scale chemical signal molecules [6], [7]

*The context for this work is thus molecular communication between bacterial populations.* Specifically, we consider a system in which bacterial populations are used as transceivers connected through pathways for molecular signals. Figure 1 shows a high level illustration of the system considered[1]. The focus of this work is to study the communication performance between the transceivers and develop strategies to improve the same. To this end, we make three major contributions:

First, we use *Escherichia coli* bacteria genetically engineered to exhibit fluorescence upon the receipt of a specific signal molecule (N-(3-Oxyhexanoyl)-L-homoserine lactone, or C6-HSL). A microfluidic experimental system houses bacterial populations within micrometer sized chambers fed by channels that provide both nutrients and controllable levels of C6-HSL, to demonstrate that a chemical signal at the sender can be reproduced as a fluorescence signal at the receiver reliably. Specifically, we demonstrate that it is indeed feasible to realize a simple modulation technique such as *On-Off Keying (OOK)* for communication between the bacterial populations, but the consequent data rates achievable is as low as $10^{-5}$ bps. We term such environments where the transmission rates are very low as *super-slow networks* ($SSN$).

Second, we introduce a new communication strategy called *time elapse communication* ($TEC$) for $SSN$ that relies on the time interval between two signals to encode information. Thus offloading some of the communication burden to the sender and receiver (in the form of measuring time periods), we show that *TEC* under idealized conditions can deliver data-rate improvements of an order of magnitude in the target environment. We also evaluate $TEC$ under realistic conditions that involve non-zero error and show that the performance of $TEC$ reduces to being marginally better than $OOK$.

Finally, we propose an improved communication strategy called *smart time elapse communication* ($TEC$-$SMART$) that uses a combination of mechanisms that improve how information is represented and how error is interpreted. Using simulations driven by experimental data, we show that $TEC$-$SMART$ approaches the original promise of $TEC$ even under realistic conditions involving non-zero error.

## II. BACKGROUND AND MOTIVATION

### A. Methods

The illustration in Figure 1 presents the set-up we consider in this work. The sender has access to a ”sender” bacterial population that acts as a transceiver, and which when provided the appropriate stimulation by the sender releases molecular signals. The signals propagate along a predefined pathway toward the bacterial population that acts as the transceiver at the receiver. The ”receiver” bacterial population, on receipt of the molecular signals, responds with fluorescence, which then is detected and interpreted by the receiver. Thus, the sender is able to convey information to the receiver using the bacterial

---

[1] The sender and receiver themselves are nano-machines that can be constructed either using inorganic or biological components, but the implications thereof are outside the scope of this work.

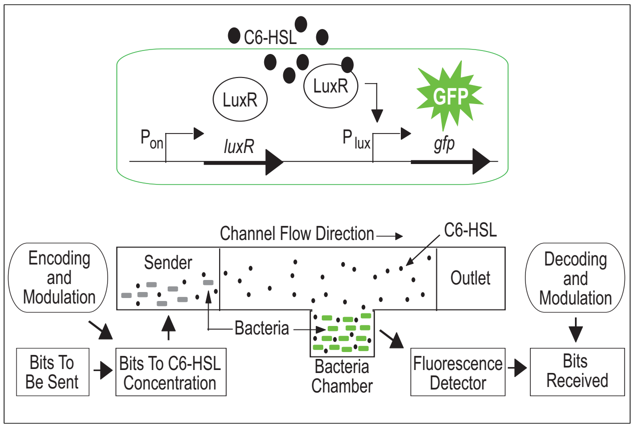


Fig. 1. An illustration of a molecular communication system with bacterial populations as nano-machines. The top figure describes the production of GFP for an input C6-HSL.

populations as the transceivers and the pathway as the channel in-between.

In this work we focus exclusively on the communication between the bacterial populations without regard to the specific nature of the sender and receiver. We now briefly describe the experimental set-up used for the motivation results.

*1) Genetically Modified E. coli Bacteria:* The bacterium *E. coli* was engineered to serve as a receiver in the test bed. *E. coli* is an extensively-studied, widely used model organism with genetic tools available to modify its DNA and resulting characteristics. *E. coli* does not carry the molecular communication described here. However, we modified it to serve as a receiver of signal molecules naturally transmitted and received by the marine bacterium *Vibrio fischeri*. *V. fischeri* cells generates light (bioluminescence) when in large populations through a phenomenon called quorum sensing, as natural *V. fischeri* cells have DNA allowing them to serve as transceivers of C6-HSL signal molecules [6]. We engineered *E. coli* to carry *V. fischeri* DNA with a promoter ($\mathrm{P_{on}}$) that is always on and drives the gene encoding the C6-HSL receptor protein, LuxR; thus these receiver cells can respond to, but do not transmit, C6-HSL. In our simple model here, rather than utilizing transmitter cells, purified C6-HSL is injected into the channel and flows to the receiver cell population in the chamber (described below). When C6-HSL enters the cells and binds LuxR, there is activation of the $\mathrm{P_{lux}}$ promoter to produce green fluorescent protein (GFP), allowing detection by fluorescence microscopy (Figure 1).

*2) Microfluidic System:* We use an approximation of the set-up shown in Figure 1 to build the experimental microfluidic system. The microfluidic system enables the measurement of the bacterial response to C6-HSL while keeping the bacterial population alive and stable within a chamber. In operation, $\approx 10^5$ bacteria are first grown to a stable population within the chamber (100 $\mu$m x 150 $\mu$m x 5 $\mu$m) for a duration of 1 day by delivering Luria-Bertani (LB) medium containing ampicillin at 10 $\mu$g/ml at a flow rate of 350 $\mu$l/hr using a syringe pump (Harvard Apparatus) connected by tygon tubing to a flow channel on the microfluidic chip. The flow channel (250 $\mu$m x 10 $\mu$m) is in direct fluidic contact with the chamber as depicted in Figure 1. The media flow delivers necessary nutrients for bacterial growth while the ampicillin maintains the plasmid carrying the *V. fischeri* DNA. The flow rate ensures that bacteria growth beyond the chamber dimensions is washed away. Following this growth phase, C6-HSL can be applied to the bacteria for a desired duration (e.g., 30 min) by similarly flowing media containing C6-HSL at 0.01 mM at 10 $\mu$l/hr in LB medium. In response to C6-HSL, the bacteria express GFP. The microfluidic system is mounted on a microscope (Nikon TE 2000) such that these proteins can be optically excited and their emitted fluorescence can be imaged every 15 min and analyzed using MATLAB.

TABLE I
BIT PERIOD

| Sender(minutes) | Receiver(minutes) | Plateau (%) |
|---|---|---|
| 300 | 607.5 | 62.8 |
| 100 | 577.5 | 51.9 |
| 50 | 457.5 | 35.8 |
| 30 | 217.5 | 6.2 |

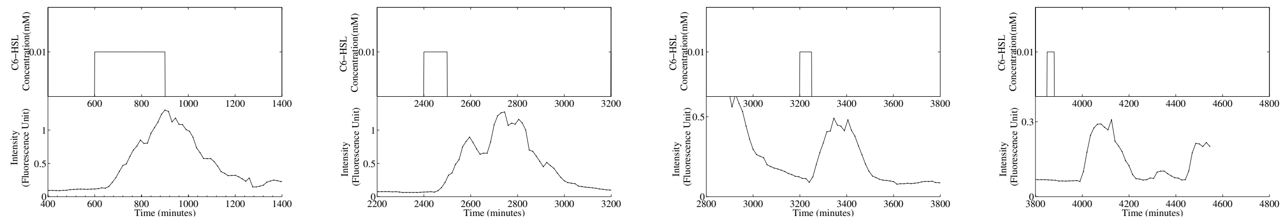
### B. On-Off Keying (OOK)

We used the microfluidic system to demonstrate that $OOK$ is (a) achievable in the target environment; and (b) has a data-rate performance that is quite low. $OOK$ is the simplest form of amplitude shift keying wherein the presence of a signal (ON) represents a "1", and the absence (OFF) represents a "0". In the experimental system, arbitrary ON-OFF patterns are configured with specified time periods at the sender that in turn triggers the transmission of molecular signals through the syringe pump. Using fluorescence microscopy images at the receiver, the intensity of fluorescence at different time periods is captured as the received signal. Figures 2(a)-2(d) illustrate the performance of the bacteria within the microfluidic system in response to C6-HSL signals of pulse durations 300, 100, 50 and 30 min respectively.

*It can be seen that the receive signals clearly follow the ON-OFF patterns at the sending side, albeit offset by the propagation delay in the environment*. While the above results demonstrate that $OOK$ can indeed be relied upon for conveying information from the sender to the receiver, we now proceed to derive the achievable data-rates using $OOK$ based on parameters extracted from the experiments. The key parameter of interest in determining the achievable data-rate is the *bit period* possible in the target environment. Table I presents the corresponding bit periods (for ON) at the receiving side for different bit periods at the sending side. The bit period at the receiver is greater than the bit period at the sending side due to the biological processing at the receiver bacteria. We define the maximum of the two bit periods as the effective bit period $t_b$. We restrict the study of those signals for which the *plateau* region (the region where output fluorescence is atleast 5x noise) is greater than 10% of $t_b$. It can be observed from Table I that the achievable bit period at receiver based on the above restriction is 457.5 min. The data-rate of $OOK$ is thus $\frac{1}{t_b}$, which for a $t_b$ of 457.5 min is $3.6*10^{-5}$ bps. In the rest of the paper, we introduce and describe strategies that are aimed toward improving the achievable data-rates in $SSN$.

## III. TIME ELAPSE COMMUNICATION

The data-rate performance of $OOK$ in bacterial communication is low due to the inordinately large bit period involved. Hence, in this paper we explore a communication strategy



(a) Sender bit Period = 300 min (b) Sender bit Period = 100 min (c) Sender bit Period = 50 min (d) Sender bit Period = 30 min

Fig. 2. Performance of $OOK$ as measured with genetically engineered *E. coli* within the microfluidic system. The chemical stimulus of C6-HSL is delivered for a pulse duration (top row) while the bacterial response is measured fluorescently (bottom row).

called *time elapse communication* ($TEC$), wherein information is encoded in the time period between two consecutive signals. The number of molecular signals generated always remains at two (the start and the stop respectively) independent of the actual information. $TEC$ requires the clock rates at the sender and receiver respectively to be the same, although no clock synchronization is required. Intuitively, $TEC$ improves the data-rate over $OOK$ by reducing the number of communication signals that need to be conveyed per unit of information.

More precisely, if the clock rate at the sender and receiver is $f_c$, information $v$ is represented by the sender as $|v|/f_c$ time units separating a *start* signal and a *stop* signal, where $|v|$ is the magnitude of $v$. If the communication involves conveying a series of such values, the *stop* signal of a particular value is used as the *start* signal of the next, and hence the number of communication signals per unit of information is amortized to just one. In $OOK$, an information value $|v|$ would be represented using approximately $\log_2 |v|$ bits. However, in $TEC$, the information value $|v|$ is represented using $|v|$ clock cycles, and hence the clock rate has to be exponentially larger than the underlying $OOK$ data-rate in order for $TEC$ to exhibit superior performance. Revisiting the set-up in Section II, for an $OOK$ data-rate of $3.6 * 10^{-5}$ bps and a clock rate of 1 Hz, under idealized channel conditions, $TEC$ will provide an average data-rate of $4.1 * 10^{-4}$ bps, a 11.3x improvement over $OOK$. In general, consider a decimal data $i$ being sent, the total delay required to communicate this data using $TEC$ is the sum of one bit period using molecular signaling and the information delay (say $t_{wait} = \frac{i}{f_c}$) corresponding to the wait time for the data. Thus, it takes $TEC$ a maximum of $t_b + \frac{2^n - 1}{f_c}$ time to transmit a $n$ bit data. The data-rate of $TEC$ is thus given by the following:

$$R_{TEC} = \frac{n}{t_b + \frac{i}{f_c}} : i \in \{0, 1....2^n - 1\}. \quad (1)$$

The notion of encoding information in time periods is not new to this work. *Timing channels* rely on such a notion to achieve covert information transfer [8], while *Pulse-Position Modulation (PPM)* relies on conveying information through the relative position of pulses in environments where there is little or no error conditions. We discuss a few other related works later in the paper, but the key difference between such techniques and this work is significant: the domain of interest - bacterial communication - raises unique and considerable challenges in how a technique like $TEC$ can be realized in the target environment, and hence the solutions we propose to adapt $TEC$ are in turn unique and fundamentally tailored to the domain.

### A. $TEC$ - The Promise

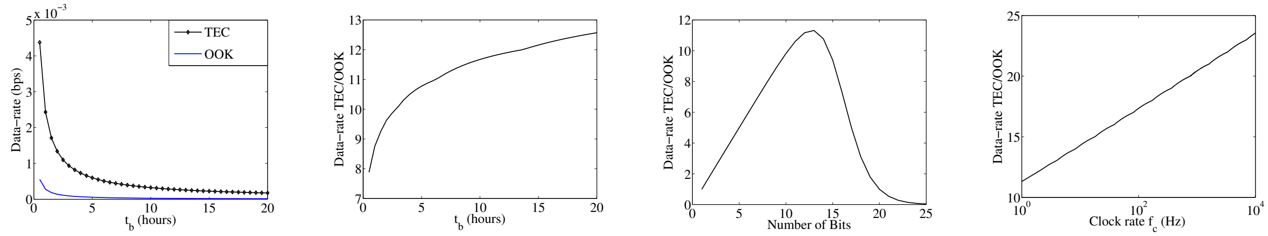
We now use numerical analysis of the data-rate equations of $OOK$ and $TEC$ to study the promise of $TEC$ under variations of different parameters. Unless otherwise specified, we use a molecular signaling bit period $t_b$ of $457.5$ min based on the experimental results presented in Section II, and a clock rate of $1$ Hz. The data-rates of $OOK$ and $TEC$ as a function of the bit period $t_b$ is shown in Figure 3(a), while Figure 3(b) presents the relative performance improvement of $TEC$ with respect to $OOK$. With an increasing $t_b$, $TEC$'s improvement over $OOK$ increases since the dependency of $TEC$'s performance on the parameter is relatively smaller. Figure 3(c) presents the relative performance improvement of $TEC$ with respect to the number of bits. It can be observed that the relative performance of $TEC$ initially improves as the numerator grows in Equation (1), but eventually the waiting term in the denominator begins dominating the performance, leading to a reduction in the relative performance. Thus, for a given set of $t_b$ and $f_c$, there is an optimal value of $n$ that should be used in $TEC$. Finally, if the clock rate is higher, the waiting time between signals corresponding to the data value will be smaller. It can be observed from Figure 3(d) that $TEC$'s relative performance with respect to $OOK$ improves with higher $f_c$. Note that while a higher $f_c$ is always better under idealized zero error conditions, any skew in clock rates between the sender and the receiver will be exacerbated under realistic non-zero error conditions. We discuss this later in the paper motivating a balanced approach to the selection of $f_c$.

### B. $TEC$ - The Challenge

Thus far, we have explored the performance of $TEC$ under idealized zero error conditions. In reality, the responses of biological systems will vary across time. To the best of our knowledge, there has not been any work that models the statistical distribution of the response of bacteria to molecular signals. Hence, we consider a simple uniform distribution *U* ($t_b$-$\epsilon$,$t_b$+$\epsilon$) to model the real response time of receiver bacteria[2]. On an average, one bit period is $t_b$ with a bounded error that is uniformly distributed *U(-*$\epsilon$*,+*$\epsilon$*)*. Any deviation from the average is termed as error. The net error $\epsilon$ is the sum of all errors from the introduction of molecules into the medium to the detection of fluorescence output. Given bounded error, it is possible for the receiver to decode with zero error by the simple technique of increasing the minimum

---

[2] We leave the consideration of other distributions for future work.



(a) Absolute TEC vs. Bit period (b) TEC vs. Bit Period (c) TEC vs. Number of bits (d) TEC vs. Clock Rate

Fig. 3. Performance of $TEC$ under ideal zero error conditions

distance between messages. A message is defined by both the start and the stop signals, and both these signals can be subject to an error of $\pm\ \epsilon$. If the minimum distance between adjacent messages is at least $4\epsilon$, the receiver can decode messages correctly in spite of any errors. We refer to $TEC$ with simple error correction as $TEC$-$SIMPLE$. The data-rate performance of $TEC$-$SIMPLE$ under non-zero error conditions is shown in Figure 5. The results are discussed in detail in Section V. We observe that the relative data-rate performance of $TEC$-$SIMPLE$ in a realistic system has reduced to approximately 1.8x $OOK$ (for an error of 10% in $t_b$). Thus, the introduction of error in the system has brought down the performance of $TEC$ considerably. We introduce several mechanisms in Section IV that in tandem improve the performance of $TEC$ multifold even under non-zero error conditions.

## IV. SMART TIME ELAPSE COMMUNICATION

In this section we propose multiple techniques that in tandem improve the performance of $TEC$ under non-zero error conditions. Specifically, we present (i) an error curtailment/differentiation strategy that reduces the impact of error on $TEC's$ performance; (ii) a differential coding strategy that is uniquely targeted towards amortizing the cost of $t_b$ across multiple pieces of information; and (iii) an optimization to the differential coding strategy that reduces overheads. We refer to a communication strategy that uses $TEC$ along with the aforementioned mechanisms as *smart time elapse communication* ($TEC$-$SMART$).

### A. Error Curtailment/Differentiation

The uniformly distributed error $U(-\epsilon,+\epsilon)$ is actually the sum of multiple error components: propagation-time error $e_d$, rise-time error $e_r$, and fall-time error $e_f$ corresponding to the propagation of molecules through the medium, the ramp-up of fluorescence, and the ramp-down of fluorescence respectively. Instead of handling the composite error in its entirety, we propose handling the error in two independent stages by introducing redundancy in the *bit period* to handle $e_r$ and $e_f$, and by introducing redundancy in the *information delay* to handle $e_d$.

*1) Fall-Time Error Correction:* The time period between the end of the $i^{th}$ signal and the start of the $i+1^{th}$ signal at the receiver represents the $i^{th}$ message. Any deviation from the estimated fall-time alters the stop of the current message, in-turn changing the absolute value of the data. Such an error in fall-time can be corrected by a proper choice of the sampling point. Assuming all other processes to be without error, it is sufficient to start measuring the time period in the rise phase of the receiver response and stop measuring upon the onset of the next rise phase. On subtracting the bit period from the total measured time, the actual message is retrieved. The fall-time error is thus absorbed in the time measurement phase. Such a correction can potentially lead to inter-symbol interference (ISI). To overcome ISI, the bit period is increased from $t_b$ to $t_b + e_f$. A pictorial representation is shown in Figure 4(a).

*2) Rise-Time Error Correction:* The fall-time error correction was based on the assumption that all other timing components are error-free. An accurate ramp-up phase is thus essential in correcting fall-time error. If the propagation delay is error-free, the time at which the leading edge of signal reaches the receiver is error-free. Thus, assuming that the propagation delay is error-free, the response of the receiver is extrapolated to identify the time at which leading edge of signal reached the receiver. The receiver adds (or subtracts) the difference between the actual and estimated times of arrival to its time measure. Again, in order to ensure that two adjacent signals do not interfere, the bit period is further increased from $t_b + e_f$ to $t_b + e_f + e_r$. Thus, both rise-time and fall-time errors are corrected by simply increasing the bit period. The information to be transmitted remains unaltered.

*3) Propagation Error Correction:* The propagation delay determines the time at which the leading edge of a signal reaches the receiver, which in turn conveys the start of a message. Therefore, error in the propagation time is corrected by introducing redundancy in the message as in the simple error correction scheme with the minimum distance between messages being controlled to be $4e_d$.

Assuming the first signal in a communication to be error-free, it is possible to achieve zero error with a reduced minimum distance of $2e_d$ as every signal is corrected based on the received and decoded messages. The transmission of first signal is restricted to slots of width one bit period ensuring an error-free signal. In the following sections we assume the first signal to be error-free. The data-rate incorporating smart error correction mechanisms is as follows:

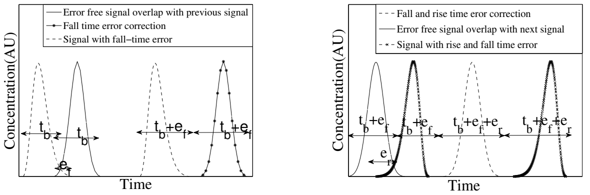
$$R_{TEC} = \frac{n}{t_b + t_{wait}}. \tag{2}$$

Using $TEC$-$SIMPLE$,

$$R_{TEC} = \frac{n}{t_b + \frac{i*(2*(e_d+e_f+e_r)f_c+1)}{f_c}} : i \in \{0,1...2^n-1\}. \tag{3}$$

Employing $TEC$-$SMART$, data-rate is

$$R_{TEC} = \frac{n}{t_b + e_f + e_r + \frac{i*(2e_df_c+1)}{f_c}} : i \in \{0,1...2^n-1\}. \tag{4}$$



(a) Fall time error correction (b) Rise time error correction

Fig. 4. Error correction:Illustrations

### B. *Differential Coding (DC)*

From Equation (3), it is evident that while curtailing the impact of error has a distinct benefit on the performance of $TEC$, the impact of $t_b$ still remains as-is. We thus propose a differential coding (DC) mechanism that leverages correlation between the values of consecutive messages to amortize the impact of $t_b$ across them. The messages at the source are assumed to be independent and identically distributed. Dependence is introduced by taking the differences of pairs of adjacent messages such that every message in the new sequence is smaller in value compared to that of the original. Since the message is encoded in time, the transmitted values cannot be negative. A sequence of *m* messages is hence arranged in increasing order, and a new sequence constituting differences between adjacent values is formed so that each element in the new sequence is positive and smaller than its value in the original sequence.

Consider the following example:
Original sequence : 10, 30, 5, 25, 3
Sequence arranged in increasing order : 3, 5, 10, 25, 30
Sequence of differences : 3, 2, 5, 15, 5

Since the ordering of elements in the original sequence is altered by virtue of the rearrangement, the actual order must be transmitted as a separate message. If a table of different orders is shared by the end systems, where the table has all possible orders for "m" messages (i.e., $m!$ entries), a message of size $\lceil \log_2 m! \rceil$ bits is required to transmit the order.

In the example above, without DC, the largest message is 5 (=$\lceil \log_2 30 \rceil$) bits long. With DC, it is reduced to 4-bits (=$\lceil \log_2 15 \rceil$) and still represents a 5-bit value. It has also reduced the total delay from "73" clock ticks to "30" clock ticks. The number of clock ticks per message is reduced with the use of DC that in turn translates to a higher data-rate. The data-rate equation for DC is thus,

$$R_{DC} = \frac{mn}{(m+1)*(t_b+e_f+e_r)+\frac{(i+j)*(2e_d f_c+1)}{f_c}} \qquad (5)$$
$$: i \in \{0,1..2^n-1\}, j \in \{0,1...m\}.$$

The sum of elements in the new sequence is equal to the largest element in the original sequence and hence the total waiting time is the sum of the waiting time to transmit the largest message in the sequence and the corresponding ordering. The latency involved in DC is higher than that in $TEC$-$SMART$ without coding but is close to that of $OOK$. For an $n$-bit message, $OOK$ takes $nt_b$ time units while DC transmits *mn* bits in a maximum of $mt_b$+ $t_{wait}$ time units. The delay in DC is close to $nt_b$ units if m is close to n (as $t_{wait} \ll t_b$). It has been observed that m is close to n over different values of $t_b$.

### C. *Piggybacked Ordering ($DC_P$)*

Recall that DC adds one extra message per sequence to convey the ordering of messages in the sequence. $DC_P$ is an optimization technique that eliminates the extra message in DC for conveying the ordering of messages. To keep the number of signals equal to the number of messages, the order is conveyed embedded within the message. Thus, one pair of (bit period + delay corresponding to order) is eliminated at the cost of increased waiting time per message. Every message (the difference) is multiplied by a constant $k_1$ and a portion of the ordering information is added. Redundancy in information delay and bit period is then introduced to the resultant message for error correction. The receiver, after performing error correction divides the number by the same constant $k_1$ so that the quotient is the message and the remainder is the portion of ordering. In this fashion, the receiver is able to recreate the ordering message that is embedded in the data messages. The data-rate using $DC_P$ is thus,

$$R_{DC_P} = \frac{mn}{m*(t_b+e_f+e_r)+\frac{(k_1*i+k_2)*(2e_d f_c+1)}{f_c}} \qquad (6)$$
$$: i \in \{0,1....2^n-1\}.$$

The order embedded in each message is $k_2$. The constant $k_1$ is chosen such that $k_1 \geq \frac{\log_2(m!)}{m}$ i.e., the constant should be able to indicate the number of extra bits per message to represent the order. Considering m = 8, the number of bits required to represent 8! is 16 and hence 2 *bits per message* making $k_1 = 4$. $k_1$ cannot be arbitrarily large; the larger the value of $k_1$, the higher the waiting delay per message. An optimization over a range of values must be performed to choose the best possible value of $k_1$, given $t_b$ and $m$. Performance results for DC and its optimization is presented in Figure 5.

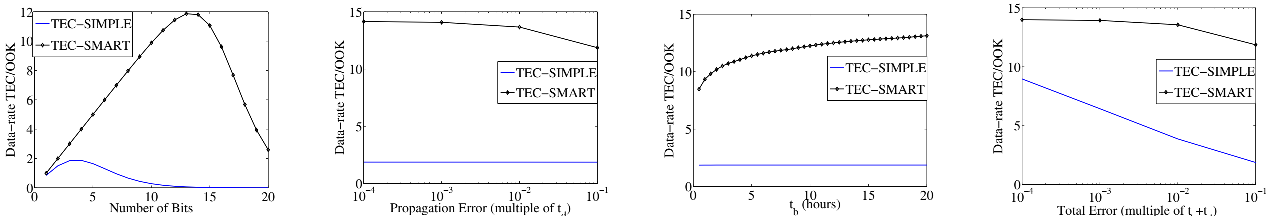
## V. PERFORMANCE EVALUATION

### *Evaluation Methodology*

Results presented in this paper are based on numerical analysis of Equations (1) to (6) using MATLAB. The specific values for the parameters and the ranges for parameters used are driven by the experimental results presented in Section II. Unless otherwise specified we use the following values for the different parameters: $t_b$ =457.5 min, $t_d$ =6 sec, $e_f + e_r = 0.1t_b$, $e_d = 0.1t_d$. Since the performance of $TEC$-$SMART$ is dependent on the message size, the bit period, and the clock rate, we study the sensitivity of its performance to these different parameters. Also, by default, we present only relative performance results for $TEC$ and its variants with respect to $OOK$. Every data point in the result is obtained by taking an average of data-rate corresponding to all messages of frame size $n$.

### A. *Frame Size*

Unlike other modulation techniques, the data-rate of $TEC$ varies with the number of bits per message or the frame size $n$. The total delay involved in a transmission varies with



(a) TEC Vs Number of bits (b) TEC Vs Propagation error (c) TEC Vs Bit Period (d) TEC Vs Total Error

Fig. 5. Performance of $TEC$-$SMART$ and $TEC$-$SIMPLE$ under non-zero error conditions

the absolute value of the message. For small values of *n*, information delay $t_{wait}$ is negligible compared to the bit period $t_b$. Thus, the data-rate increases with increasing *n*. Once $t_{wait}$ is comparable to $t_b$, the data-rate begins to decrease as the $t_{wait}$ starts dominating. The relative data-rate performance of $TEC$-$SIMPLE$ and $TEC$-$SMART$ is presented in Figure 5(a). This motivates the need for an appropriate selection of the frame size given a target environment.

### *B. Error*

Recall from Section III that the performance of $TEC$-$SIMPLE$ reduced to being marginally better than that of $OOK$ under non-zero error conditions. However, $TEC$-$SMART$ is explicitly designed to handle error conditions better by virtue of its error curtailing and differentiation mechanisms. Thus, the increase in *rise-time error* and *fall-time error* has minimal impact on the overall performance of $TEC$-$SMART$. As seen in Figure 5(b), $TEC$-$SMART$ can deliver a data-rate of over 10x even when the the total error is large ($0.1t_b$+$e_d$). Data-rate with respect to varying error components is presented in Figures 5(b)-5(d). Overall, the results demonstrate the better error resiliency exhibited by $TEC$-$SMART$.

### *C. Bit Period*

Figure 5(c) presents the data-rate performance for $TEC$ and $TEC$-$SMART$ for different bit period. The value of $t_b$ is varied from 1 to 20 hours. It can be observed from the results that while $TEC$ is impacted heavily in its performance by an increase in $t_b$, $TEC$-$SMART$ is considerably more resilient to larger values of $t_b$. This is due to the amortization of the $t_b$ overhead over multiple messages.

### *D. Frequency*

Figure 3(d) shows an increase in the data-rate with increasing clock frequency. With the introduction of error in the system, the clock rate loses its significance. Recall that the transmitter and the receiver measure the number of $e_d$ time units between the start and stop signals. Hence, however high the clock rate is, the time slot is now in terms of error and hence the data-rate performance does not change with frequency once the error correction is introduced.

## VI. RELATED WORK AND CONCLUSIONS

In addition to timing channels and PPM identified in SectionIII, there are few other approaches related to $TEC$. However, these approaches do not cater to the large error-rate or bit periods of the target bacterial communication environment, We discuss some of these approaches below:

**Timing Channels:** In [9] mechanisms to improve the data-rate of timing channel have been proposed. However, the proposed techniques are not targeted for the context of bacterial communication. Specifically, this work involves the use of static and complex coding tables unsuitable for the target environment. More importantly, it does not deal with large error rates and hence will perform similar to $TEC$ without any optimization.

**Communication through Silence:** A solution to use silent periods in sensor networks to communicate information was presented in [10]. The primary goal is to reduce the energy consumption, but non-zero error conditions are not considered and data-rate improvement is not the primary focus.

**Timing modulation in fluid channel:** An information theoretic approach to the timing modulation in molecular communication has been studied in [11]. Absolute timing is relied upon and hence the solution requires strict clock synchronization. Further, data-rate improvement is not a focus of this work.

In this paper, using state-of-art advancements in genetic engineering and microfluidics we have argued with results from an experimental test-bed that a modulation technique like $OOK$ is indeed achievable for communication between bacterial populations relying on molecular signaling. We also have shown that the data-rate performance of $OOK$ is dismally low because of the large bit periods. Finally, we propose a communication strategy called time elapse communication with a set of optimization mechanisms that improves the data-rate over $OOK$ by more than an order of magnitude.

## REFERENCES

[1] I. F. Akyildiz and J. M. Jornet, “Electromagnetic wireless nanosensor networks,” *Nano Communication Networks*, 2010.

[2] G. Deligeorgis, M. Dragoman, D. Neculoiu, D. Dragoman, G. Konstantinidis, A. Cismaru, and R. Plana, “Microwave propagation in graphene,” *Applied Physics Letters*, 2009.

[3] T. Suda, M. Moore, T. Nakano, R. Egashira, and A. Enomoto, “Exploratory research on molecular communication between nanomachines,” in *GECCO 2005*.

[4] A. W. Eckford, “Molecular communication: Physically realistic models and achievable information rates,” *arXiv:0812.1554v1*.

[5] I. F. Akyildiz, F. Fekri, C. R. Forest, B. K. Hammer, and R. Sivakumar, “Monaco: Fundamentals of molecular nano-communication networks (invited paper),” *IEEE Wireless Communications Magazine, Special Issue on Wireless Communications at the Nano-Scale*, 2012.

[6] B. L. Bassler, “How bacteria talk to each other: regulation of gene expression by quorum sensing,” *Current Opinion in Microbiology*, 1999.

[7] P. Melke, P. Sahlin, A. Levchenko, and H. Jönsson, “A cell-based model for quorum sensing in heterogeneous bacterial colonies,” *PLoS Computational Biology*, 2010.

[8] V. Anantharam and S. Verdú, “Bits through queues,” *IEEE Transactions on Information Theory*, 1996.

[9] S. Sellke, C. Wang, S. Bagchi, and N. Shroff, “TCP/IP timing channels: Theory to implementation,” *INFOCOM*, 2009.

[10] Y. Zhu and R. Sivakumar, “Challenges: communication through silence in wireless sensor networks,” in *MobiCom*, 2005.

[11] K. V. Srinivas, A. W. Eckford, and R. S. Adve, “Molecular communication in fluid media: The additive inverse gaussian noise channel,” *IEEE Transactions on Information Theory*, 2012.