Molecular Communications

Ömer Faruk Ünal

January 10, 2024

Abstract

This paper explores molecular communication with a focus on error correction mechanisms, the document investigates challenges unique to nanoscale environments, where limited energy resources and precise timing pose significant constraints. It reviews existing models, particularly the application of Hamming codes for error correction. The proposed solution introduces an approach of sequentially releasing particles A and B in an alternating pattern (ABABABABAB) to convey information, aiming to address error correction and energy efficiency challenges.

1 Introduction

Molecular communication, a rapidly growing field at the intersection of nanotechnology and communication systems, explores the transmission of information through the use of molecules as carriers. Understanding the dynamics of molecule diffusion and absorption within a three-dimensional space is crucial in designing efficient communication systems based on this paradigm.

Computer communication, which relies on the transmission of data through electromagnetic signals, has its limitations, especially when applied to nanoscale environments. In this context, molecular communication has gathered attention as an innovative communication paradigm that leverages the use of biochemical entities, such as molecules, to transmit information between nanoscale devices.

One critical aspect in the communication systems is the occurrence of errors during information transmission. Bit errors in computer networks and molecular communication systems pose unique challenges in their respective domains. In computer networks, errors often result from external factors such as noise, interference, and attenuation. The electromagnetic signals that transmit data can be affected by various environmental conditions, leading to distortions and corruptions. To address these issues, error detection mechanisms such as checksums and cyclic redundancy checks (CRC) are employed. Additionally, error correction techniques, including forward error correction (FEC) and Automatic Repeat reQuest (ARQ), play a crucial role in ensuring data integrity. However, these mechanisms come with trade-offs, as they can reduce throughput and introduce latency, particularly in real-time applications.

In molecular communication systems, operating at the nanoscale introduces its own set of challenges. The limited energy budget, characteristic of nodes scavenging energy from the surrounding environment, imposes strict constraints on power availability. Bit errors in this context may arise from the Brownian motion and diffusion of molecules, causing deviations from their intended paths. Precise timing is also critical, and synchronization issues between the release and reception of molecules can lead to errors. To overcome these challenges, forward error correction techniques are used, adding redundancy to enable error correction. Moreover, statistical approaches are adopted to account for the probability distributions of molecular arrivals, facilitating the development of adaptive communication strategies.

Energy efficiency becomes an important consideration in molecular communication systems due to the constrained energy resources at the nanoscale. Coding schemes must operate within these limitations, ensuring optimal use of available energy. Adaptive transmission rates, varying the release rate of molecules based on environmental conditions, contribute to enhanced energy efficiency. As researchers explore solutions for bit errors in molecular communication, striking a balance between data rate and reliability remains a central

focus to advance the effectiveness of communication in nanoscale environments.

This document aims to explore the error correction mechanism in molecule diffusion towards a spherical absorber, detailing the computational steps involved in simulating the absorption of molecules over discrete time intervals and the numerical modeling of molecule movement within a 3D space.

2 Existing System Model

The paper "Forward error correction for molecular communications" by Mark S. Leeson and Matthew D. Higgins focuses on addressing challenges related to the diffusion of messenger molecules, which may lead to deviations from their intended time slots or potential non-arrival. To enhance nanonetwork performance, the researchers propose the use of error-correcting codes, specifically a simple block code. The chosen code, Hamming codes, demonstrates a coding gain of approximately 1.7 dB at transmission distances of 1 µm. However, it is acknowledged that energy is required for coding and decoding processes.

The paper emphasizes the trade-off between the benefits of error correction and the associated energy consumption. The energy considerations involve the use of adenosine triphosphate (ATP) as a measure, given its role in energy transfer within living organisms. The authors estimate the energy cost of encoding and decoding operations using Hamming codes, taking into account the ATP consumption associated with logical operations.

Hamming codes, defined for an integer $m \geq 2$, involve adding parity check bits to correct errors in blocks of data bits. Despite their simplicity, Hamming codes are considered viable for energy-budgeted nanoscale communication devices. To assess the performance of Hamming coded transmission, the Bit Error Rate (BER) must be determined. The paper presents an approximate expression for the BER of block codes, considering the probability of error for the uncoded case. This involves accounting for different error probabilities due to Intersymbol Interference (ISI) in molecular communications. The results indicate a coding benefit when the BER surpasses specific thresholds for different values of m (3, 4, and 5).

Additionally, the concept of coding gain is introduced, measured in dB at a relevant BER. The coding gain represents the ratio of molecules needed for uncoded and coded systems, assuming effective error-free operation at a specified BER of 10^{-9} and likely bit rates around 100 bit/s. The calculated coding gains for m values of 3, 4, and 5 are approximately 0.5 dB, 1.4 dB, and 1.7 dB, respectively.

The critical distance, defined as the point where the energy gain of the code overcomes its operational energy cost, is introduced. The critical distance is identified as being beneficial for nanonetworks operating over distances greater than 30 µm.

In conclusion, the research highlights the potential of error correction codes, specifically Hamming codes, in improving the performance of nanonetworks. The findings underscore the importance of energy-efficient coding strategies, considering the constrained energy budgets in nanoscale communication devices. The critical distance concept provides insights into when the energy gain from coding becomes advantageous, contributing to the understanding of nanoscale communication systems.

3 Proposed Additions

My proposed solution suggests releasing two particles, labeled A and B, sequentially as a representation of bit one after the other. Instead of sending continuous bits of the same type (e.g., AAAAAAA), the approach involves alternating between A and B particles (ABABABABAB) to convey information. The aim is to utilize this alternating pattern for communication, potentially addressing challenges such as error correction and energy efficiency in nanoscale communication systems. The receiver implements a strategy of waiting for different molecules in distinct time slots, mitigating the impact of particles from the previous slot on the subsequent one. This separation aims to further optimize communication performance.

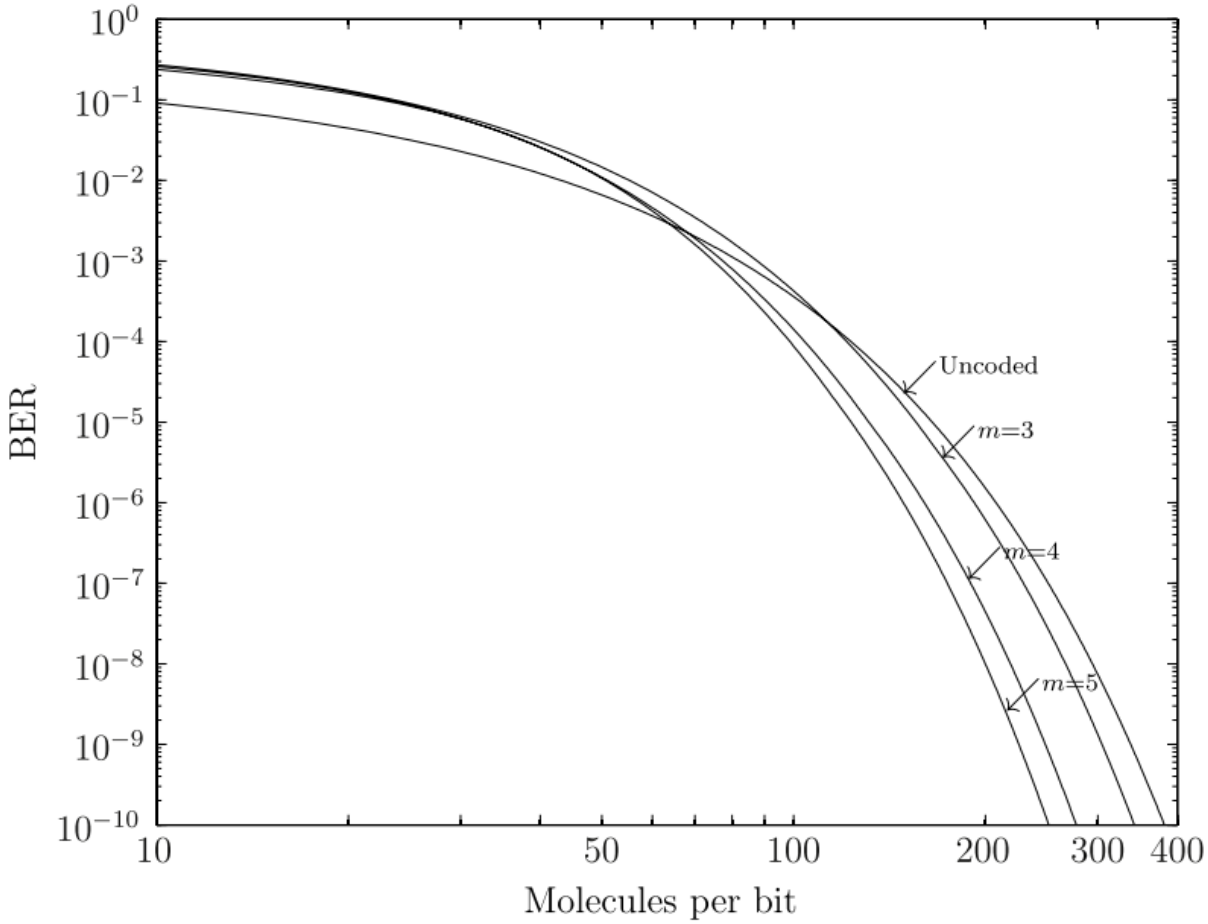


Figure 1: BER for coded and uncoded transmission using Hamming codes having m = 3, 4, 5 with r0 = 6 µm [1]

3.1 Pseudocode

Essentially, the procedure involves releasing particle A, waiting for a specific duration, and subsequently releasing particle B, without removing any particles except those that have been absorbed. In each timeframe, the particles corresponding to that specific timeframe are counted as bits.

For optimization purposes, in the simulation, I removed particles that were moving far from absorbing particles, assuming they would not be absorbed again.

Function `simulate`
    total_absorbed = 0;
    type = 0; `initialize_particles()`;
    for *each time step* **do**
        `initialize_particles(`*type*`)`;
        `move_particles(`*type*`)`;
        `absorb_particles(`*type*`)`;
        update timeline;
        type = 1 - type;
    end
    return simulation result;

Function `initialize_particles`
    Calculate sigma;
    Set initial molecule coordinates;
    Set receiver membrane boundary;

Function `move_particles`
    Generate random Gaussian(0, sigma) movement for each molecule;
    Update molecule positions based on movement;

Function `absorb_particles`
    Calculate distance of molecules from receiver;
    Create mask for particles outside the receiver;
    for *each particle outside the membrane* **do**
        if *type = turn* **then**
            Increase total absorbed count;
            Remove particles inside the receiver;
        end
    end

Algorithm 1: Absorbing Sphere Diffusion Simulator in 3D Without Reflecting Line Pseudo Code

4 Numerical Results and Plots

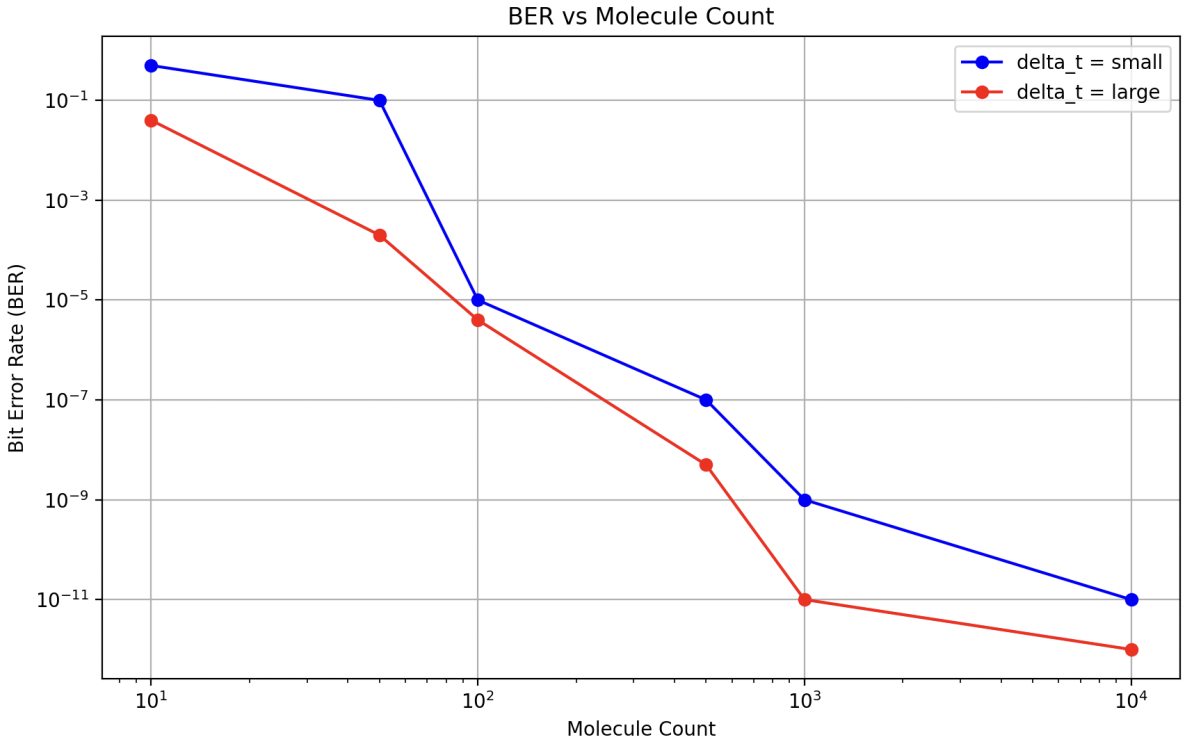
The given algorithms have been implemented, and the results of the stochastic simulations with various parameters are provided. However because it was hard to simulate it with larger parameters, the results are not good enough to provider solid foundation.

4.1 Results

```
Parameter Set Task1-1:
sim_params.rx_center = [0, 0, 0];
sim_params.rx_r_inMicroMeters = 5;
sim_params.rx_tx_distance = 5;
sim_params.tx_emission_pt = [10, 0, 0];
sim_params.D_inMicroMeterSqrPerSecond = 75;
sim_params.remove_condition = 20;
sim_params.tend = 0.4;
sim_params.delta_t = 0.0001;
sim_params.num_molecules = 50000; #changes
sim_params.bit_threshold = 750; #changes

Parameter Set Task1-2:
sim_params.rx_center = [0, 0, 0];
```

```
sim_params.rx_r_inMicroMeters = 5;
sim_params.rx_tx_distance = 5;
sim_params.tx_emission_pt = [10, 0, 0];
sim_params.D_inMicroMeterSqrPerSecond = 75;
sim_params.remove_condition = 20;
sim_params.tend = 0.8;
sim_params.delta_t = 0.0001;
sim_params.num_molecules = 50000; #changes
sim_params.bit_threshold = 750; #changes
```



Based on the outcomes of the proposed solutions, encoding bits with different molecules appears to outperform even the implementation with Hamming codes. Despite the potential complexity of coding in molecular environment for this pattern, the energetic requirements for sending two molecules seem manageable.

However, for a more robust foundation, it is crucial to determine and measure the actual times required to transmit a bit. Increasing the waiting time between two adjacent bits can decrease the bit error rate, but it also reduces the transmission rate. Therefore, before implementing a solution, it is essential to simulate this process with more realistic values on a faster machine for accurate assessments.

References

[1] M. S. Leeson, M. D. Higgins, "Forward error correction for molecular communications," Nano Comm. Networks, vol. 3, pp. 1610167, 2012.

[2] M. S. Kuran, H. B. Yilmaz, T. Tugcu, and B. Ozerman, "Energy model for communication via diffusion in nanonetworks," Nano Comm. Networks, vol. 1, no. 2, pp. 86–95, 2010.

[3] H. B. Yilmaz, A. C. Heren, T. Tugcu, and C.-B. Chae, "Threedimensional channel characteristics for molecular communications with an absorbing receiver," IEEE Commun. Lett., vol. 18, no. 6, pp. 929–932, Jun. 2014.

[4] H. B. Yilmaz, (2023, October). Particle Based Simulations [Lecture slides]. Faculty of Computer Engineering, Bogazici University.