This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/LCOMM.2023.3303091,

MCFormer: A Transformer-Based Detector for Molecular Communication with Accelerated Particle-Based Solution

Xiwen Lu, *Student Member, IEEE*, Chenyao Bai, *Member, IEEE*, Aoji Zhu, *Student Member, IEEE*, Yunlong Zhu, Kezhi Wang, *Senior Member, IEEE*

***Abstract*—Molecular communication (MC) enables communication at the nanoscale where traditional electromagnetic waves are ineffective, and accurate signal detection is essential for practical implementation. However, due to the lack of accurate mathematical models, statistical-based signal detection methods are not applicable, and existing deep learning-based models exhibit relative simplicity in design. This paper integrates ideas from natural language processing into MC and proposes the MCFormer, a detector based on the classical Transformer model. Additionally, we propose an accelerated particle-based simulation algorithm using matrix operations for rapid generation of high-quality training data with a lower complexity than traditional methods. The experimental results demonstrate that the MCFormer achieves nearly optimal accuracy in a noise-free environment, surpassing the performance of the Deep Neural Network (DNN). Moreover, MCFormer can show optimal performance in environments with significant levels of unknown noise. All the codes can be found at https://github.com/Xiwen-Lu/MCFormer.**

***Index Terms*—Molecular Communication, Detector Design, Signal Detection, Simulation, Transformer.**

I. Introduction

MOLECULAR communication (MC) is a field of communication at the nanoscale, where information is transmitted using molecules [1]. It holds significant potential in various fields, such as targeted drug delivery [2]. The accurate recovery of the original transmitted from the received signal is crucial for the practical use of MC, which is similar to radio communication systems. Initial attempts employed decoding approach using a single list of thresholds [3], however, this method proved impractical as it requires comprehensive communication model information. Subsequently, a Bayesian approach based on the MAP method was applied [4], and this method can work under the condition that only the molecular arrival probability function is known.

Since precise channel models are often unavailable in MC, researchers have started utilizing deep learning methods to learn patterns from data and achieve signal detection [5]–[7]. Authors in [5] employed a neural network for signal detection, while in [6], a vanilla recurrent neural network is employed to detect and decode the signals transmitted through the MC channel. In [7], a modified temporal convolutional network is proposed for signal detection for a special MC communication system.

Existing analytical and numerical approaches have struggled to effectively tackle the issue of inter-symbol interference (ISI) in MC systems. This challenge has motivated the adoption of natural language processing (NLP) tools in our work, as these techniques have demonstrated the ability to selectively focus on relevant information and mitigate the effects of ISI. By leveraging the power of NLP [8], we recognize that ISI in MC bears striking similarity to inter-word interference in NLP, where adjacent characters in a sentence typically exhibit a semantic relationship. This observation makes it both reasonable and necessary to explore the direct application of NLP approaches in the context of MC. Consequently, our work represents the first attempt to introduce NLP methods, particularly transformer techniques [9], into the field of MC.

The success of deep learning relies heavily on large quantities of high-quality training data. In this paper, we aim to generate training data by directly simulating the Brownian motion process of particles. Although some previous works, such as [10], [11], pursue similar concepts, using a plain multi-layer cyclic approach can be excessively time-consuming for large data sizes. To address this issue, we propose an accelerated particle-based data simulation algorithm that leverages matrix operations to generate training data more efficiently.

The primary contributions of this letter are as follows:

1) We are the first to operate MC signal detection from a NLP perspective and propose a novel detector based on the Transformer model, denoted as MCFormer, which mitigates the effect of ISI in MC.
2) We introduce an accelerated particle-based data simulation method that uses matrix operations to substantially expedite the data generation process, which can greatly reduce the complexity of deployment of this method in MC system..

The remainder of this letter is organized as follows: Section II presents the MC system model that we specifically analyze. Section III introduces the proposed MCFormer receiver, which is based on the Transformer model. Section IV outlines the accelerated data simulation process. Section V presents the conducted detector experiments and results. Finally, Section VI summarizes our work and discusses potential directions for future research.

This work was supported by China Postdoctoral Science Foundation (Grant No. 2021M690701). (*Corresponding authors: Chenyao Bai, Yunlong Zhu*)

Xiwen Lu, Chenyao Bai, Aoji Zhu and Yunlong Zhu are with the Academy for Engineering and Technology, Fudan University, Shanghai 200433, China. (e-mail: xwlu21@m.fudan.edu.cn;baichenyao@fudan.edu.cn; ajzhu21@m.fudan.edu.cn; zyl@fudan.edu.cn)

Kezhi Wang is with Department of Computer Science, Brunel University London, Uxbridge, Middlesex, UB8 3PH. (email: kezhi.wang@brunel.ac.uk)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/LCOMM.2023.3303091,

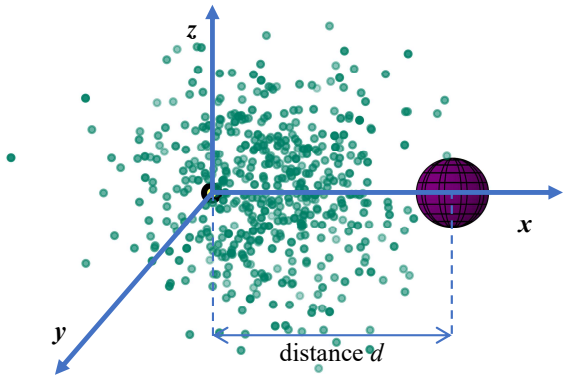

Fig. 1. MC system in 3-D space with a transmitter TN (yellow mass), a receiver RN (purple sphere) and released molecules (green point).

II. MC System Model

Fig. 1 illustrates the overall structure of our MC model, with a point transmitter, a diffusion channel with flow, and a spherical penetrating receiver.

A. Transmitter Model

In this letter, a common pulse-transmitted system with *On-Off Keying* (OOK) modulation is studied. For each release interval T, the transmitter TN releases M molecules when it transmits bit '1' at the ith time slot. s_i represents the signal transmitted at the ith time slot.

B. Channel Model

Without loss of generality, this letter considers a diffusion channel under uniform flow, where the motion of the particles is affected by a uniform flow velocity v with Brownian motion and the direction of flow is along the x-axis. The real-time position of each particle (x_t, y_t, z_t) can be calculated by accumulating the incremental displacements per unit time Δt:

$$(x_t, y_t, z_t) = (x_{t-\Delta t}, y_{t-\Delta t}, z_{t-\Delta t}) + (\Delta x, \Delta y, \Delta z). \quad (1)$$

Here, the incremental displacement $(\Delta x, \Delta y, \Delta z)$ satisfies:

$$\begin{aligned} \Delta x &\sim \mathcal{N}(v\Delta t, 2\mathrm{D}\Delta t), \\ \Delta y, \Delta z &\sim \mathcal{N}(0, 2\mathrm{D}\Delta t), \end{aligned} \quad (2)$$

where D represents the diffusion coefficient.

C. Receiver Model

In the receiver section, we choose a passive receiver which counted particles to study, such as [12]. Specifically, RN is a sphere of radius r. The number of particles being observed in the entire sphere region at the end of ith time slot n_i is then:

$$n_i = |S_{\mathbf{P}i} \cap \mathbf{S}_{\mathrm{RN}}|. \quad (3)$$

Here, $S_{\mathbf{P}i}$ is the set consisting of all particles at the ith time slot, and $S_{\mathbf{P}i} \cap S_{\mathrm{RN}}$ is the intersection of $S_{\mathbf{P}i}$ with the receiver sphere space S_{RN}. $|\cdot|$ denotes the set cardinality, indicating the number of elements in the set.

As we can see from Fig. 2a, for the signal detection task, with the decoding function $\mathcal{F}$ for the received molecules n_i, we can obtain the detected signal $\mathcal{F}(n_i)$ corresponding to the transmitted signal s_i, which can be denoted as $\hat{s}_i$. Based on this, the decoding bit error rate (BER) for the whole signal detection task can be obtained, as shown in Eq. 4, where L is the length of the whole transmitted signal sequence $S = [s_1, s_2, \cdots, s_i, \cdots]$.

$$BER = 1 - \frac{\sum_{i=1}^{L} \delta(\hat{s}_i, s_i)}{L} \quad (4)$$

It is noted that $\delta(i, j)$ is defined to be 1 if $i = j$ and 0 if $i \neq j$.

According to [13], the theoretical received number of molecules n_i can be derived obeying a poisson distribution $\pi(M \times p(t))$, where,

$$p(t) = \frac{V_{\mathrm{RN}}}{(4\pi \mathrm{D}t)^{3/2}} \cdot \exp\left(-\frac{\|d - tv\|^2}{4\mathrm{D}t}\right), \quad (5)$$

Here, V_{RN} is the volume of the receiver RN and d represents the distance between TN and RN.

III. MCFormer: Detector in MC based on Transformer

In this section, we introduce MCFormer to detect the original transmit signal from n_i whose backbone structure is mostly based on Transformer, as shown in Fig. 2c. MCFormer aims to reduce the effect of ISI in MC by leveraging ideas from NLP and deep learning based signal detection methods.

A. Overall Framework

MCFormer, a model designed for processing MC data, has been simplified from the original Transformer model to better suit the differences between MC and natural language data. The key adaptations include:

- Removing the input padding mask component, as simulation data can be set to a fixed length before being input into the network.
- Limiting the encoding vocabulary vector length and reducing network parameters, since the input data of molecular detectors has a finite numerical upper limit.
- Using simpler self-attention instead of multi-attention, as MC is primarily affected by ISI rather than semantic interference in natural language.

Despite these changes, MCFormer retains the standard NLP operations like word embedding, self attention, and feed forward network components, with the aim of adapting to the MC domain. Further modification details on these components can be found below.

B. Word Embedding

In the context of MCFormer, we apply word embedding to convert the number of received molecules n_i into a dense vector representation. This process allows the input sequence $[n_1, n_2, \cdots, n_i, \cdots]$ to be transformed into a dense matrix, which serves as the input for the whole receiver system. The mathematical representation of word embedding is $A \cdot V$, where matrix A represents the one-hot form of the input sequence vector. Vocabulary V encompasses the complete set of dense

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/LCOMM.2023.3303091,

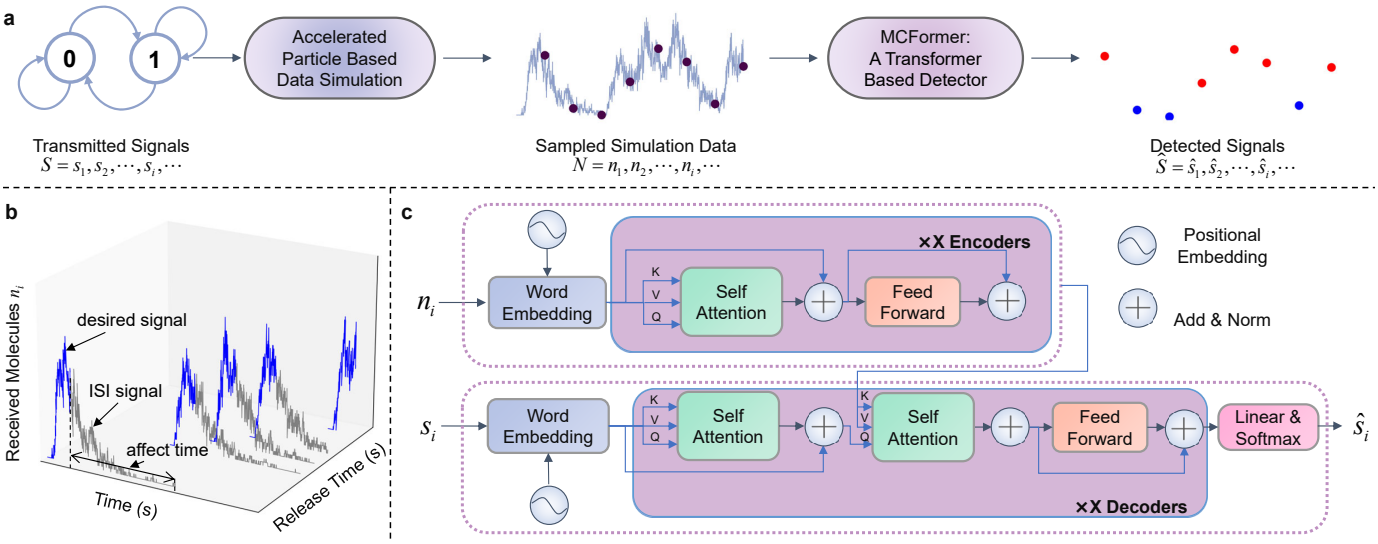

Fig. 2. (a) The pipeline of our solution for the signal detection task in MC. (b) The accelerated particle-based simulation process. (c) The structure of MCFormer.

vectors for all distinct words, with each vector containing d_m values.

In the realm of NLP, there exist commonly used vocabularies V [14]. However, given that there are no established word vectors in the MC domain, we utilize a randomly initialized vocabulary which will be integrated into the network and trained concurrently. This approach allows the system to learn the most effective representations for the MC data.

C. Self Attention

Attention serves as the most vital part, as it enables MCFormer to learn and understand the global relationships between different words within the context of MC. The overall attention mechanism is outlined as [9]:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^{\mathrm{T}}}{\sqrt{d_m}}\right)V \tag{6}$$

where the weight of the value field (V) is obtained by dotting the query field (Q) with the keyword field (K), and $(\cdot)^{\mathrm{T}}$ represents the transpose of the matrix. In MCFormer, Q, K and V are all set as the processed input matrix, which is derived after applying the word embedding process. The weight assigned to V, denoted as $\text{softmax}\left(QK^{\mathrm{T}}{d_m}^{-1/2}\right)$, plays a crucial role in establishing connections between the number of molecules received at various time intervals n_i. This relationship corresponds to the ISI phenomenon in MC.

As a result of this attention mechanism, each data point being processed can effectively incorporate information from other points based on their similarities. This allows the model to develop a more comprehensive understanding of the input data, leading to a better overall representation of the MC process. This contributes significantly to the MCFormer's performance in processing and interpreting MC data.

D. Feed Forward Network

The Feed Forward Network (FFN) is a fully connected layer with one hidden layer. Both the input and output dimensions of the FFN are d_m while the hidden layer dimension, which can be adjusted to optimize performance, is represented as d_h. Additionally, when data is processed through the FFN after the attention mechanism, a residual connection is incorporated. This connection helps preserve the input information by adding it to the output of the FFN simultaneously. This enhancement leads to better overall performance and stability in the training process.

By combining the attention mechanism, word embedding, and the FFN with residual connections, MCFormer effectively adapts standard NLP operations for the MC domain.

IV. Accelerated Particle Based Data Simulation

Typically, particle-based solutions keep account of individual particles; in this manner, the complete trajectory of all particles can be obtained, albeit at a slower rate. Especially for detection tasks, it is not necessary to know the complete trajectory of all particles. We propose a simulation algorithm that operates as a matrix operation and includes an effect phase in order to generate a dataset for the detection task rapidly.

A. Acceleration Algorithm with Effect Step

The complete algorithm is depicted in Algorithm 1, which iterates through batches of particles as they are released. For each batch of particles, we maintain a matrix that stores the coordinates of each moment of the batch in a manner that accumulates the motion's displacements in a single step of time, in a manner that resembles the simulation of the wiener process. For moments where the particle's coordinates fall within the receiver's range, a "1" is recorded to denote that the particle was observed at that time. Eventually, the number of received particles N is accumulated and sampled, yielding the sequence of received particle numbers $[n_1, n_2, \cdots, n_i, \cdots]$.

Notably, as the total time increases, so does the memory overhead associated with storing the matrix. In order for the simulation algorithm to generate data more robustly for arbitrary amounts of time, we propose the variable of "effect

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/LCOMM.2023.3303091,

step", denoted K_e, which incorporates the decreasing character of ISI in MC with increasing time. Specifically, K_e examines the situation in which the number of particles observed at time i is only affected by the particles released at time $[i, i-T, i-2T, ..., i-(K_e-1)T]$. As we can see from Fig. 2b, the particles observed in the current time slot are in blue, the ISI part is represented in grey, the simulation algorithm only records the current time slot and time slots with a certain effect step K_e for the ISI part. The effect time marked on the Fig. 2b is the region of interest (ROI) which we will caculate in our algorithm, denoted as N_{ROI}.

Algorithm 1 Accelerated Particle Based Data Simulation

Require: signals sequence $S = [s_1, s_2, \cdots, s_i, \cdots]$; release time interval T; flow velocity v; RN center point coordinate P_s; signals length L; num of molecules per release M; precision control parameter ϵ $(0 < \epsilon < 1)$

Initialization N_{total} $\quad zeros\,(size = (LT))$
Calculate $K_e = \arg\max_{k \in \mathbb{N}^+} p(kT)$, s.t. $M \cdot p(kT) < \epsilon$
for $i = 0$ **to** $L-1$ **if** $s_i \neq 0$ **do**
  N_{ROI} $\quad zeros\,(size = (M, K_eT))$
  d_x $\quad \underset{axis=0}{cumsum}\left(\mathcal{N}\left(v\Delta t, \sqrt{2\mathrm{D}\Delta t}, size = (M, K_eT)\right)\right)$
  d_y, d_z $\quad \underset{axis=0}{cumsum}\left(\mathcal{N}\left(0, \sqrt{2\mathrm{D}\Delta t}, size = (M, K_eT)\right)\right)$
  $distance$ $\quad sqrt\left(\sum_{axis=0}\left([d_x, d_y, d_z] - P_s\right)^2\right)$
  $N_{ROI}\,[distance < r] = 1$
  $N_{total}\,[iT : (i + K_e)\,T] += \underset{axis=1}{sum}\, N_{ROI}$
end for
//sample N_{total} with matrix slicing, i.e.[start:end:step]
N $\quad N_{total}\,[T-1 :: T]$
return sampled number of received molecules N.

B. Performance Analysis

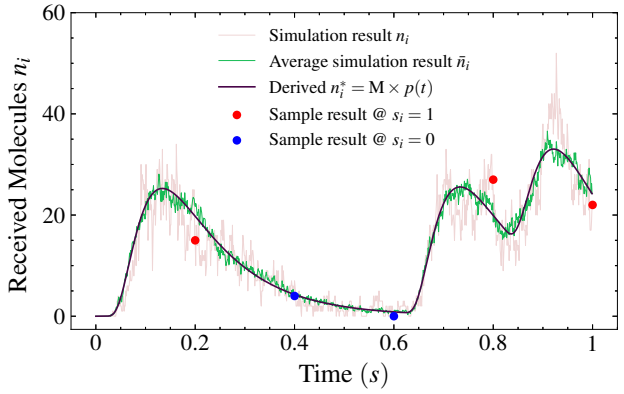

Fig. 3. Simulation results of n_i by Eq. 3 when transmit signal suquence $s = [1, 0, 0, 1, 1]$.

The complexity of the cumulative sum function is $\mathcal{O}(MK_eT)$, so the complexity of Algorithm 1 is $\mathcal{O}(LMK_eT)$, which is more effective than the complexity of $\mathcal{O}(LT{\cdot}LM)$ for the previous method [10] of tracking all particle trajectories. Furthermore, we compare the simulation results with the theoretical results from Eq. 5 to test the accuracy of the proposed simulation method. As shown in Fig. 3, the mean results curve over 30 simulation iterations closely matches the theoretical curve, demonstrating the accuracy of Algorithm 1. The parameters used to generate Fig. 3 are outlined in Table I.

TABLE I
THE PARAMETERS OF MOLECULAR COMMUNICATION SYSTEM

Parameter	Symbol	Value
diffucision coefficient	D	$79.4\mu m^2/s$ [13]
flow velocity	v	$30\mu m/s$
distance between TN and RN	d	$10\mu m$
radius of RN	r	$1.5\mu m$
release rime interval	T	$0.2s$
released molecules per release time	M	4000
precision control parameter	ϵ	10^{-2}

By reducing computational complexity and memory usage, this algorithm enables efficient generation of high-quality training data, thereby facilitating the development of more effective MC models.

V. Experiments and Results

To achieve the goal of analyzing the effectiveness of MCFormer in reducing ISI, we choose different drift velocity v to compare the detect performance of MCFormer, a Deep Neural Network (DNN), and an MAP decoder. Finally, the BER performance of the detector was further assessed under unknown noisy conditions.

A. Data Generation

Before performing performance tests on MCFormer, we generate 9000 random signals for training and validation (7:2) and 5000 signals for testing. The parameter settings during data generation are the same as those shown in Table I, except for the drift velocity v. We limit the input data length to 100 bits, i.e., each time we use $[n_1, n_2, \cdots, n_{100}]$ of the n_i sequence as input to the network. Fig. 4a shows the boxplot of some testing data.

B. BER testing with different drift velocity

For the BER performance of MCFormer under different parameter v values to be explored, we use grid search to select the optimal hyperparameters d_m and the number of encoder-decoders X in the network which is shown in Fig. 2c. The best final performance is obtained when d_m is set to 128 and three encoders and three decoders are used. This hyperparameter was fixed to compare the BER.

We also implement DNN [5] to compare with MCFormer and the length of the input layer length of the DNN is set to 100 as well for a fair comparison. Additionally, the number of layers in DNN is determined using grid search, as is the number of neurons in each hidden layer. The single hidden layer structure with 128 neurons has the lowest BER and is ultimately used for comparison with MCFormer.

An optimal decoder MAP [7] with group length 10 is implemented to be compared too, in which considers the ISI between K_e numbers and this MAP decoder can be seen as theoretically optimal detector . Fig. 4b shows that MAP achieves lowest BER on almost all data, which is in line with usual perception as it knows all channel information. MCFormer achieves a close performance to MAP demonstrating

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/LCOMM.2023.3303091,

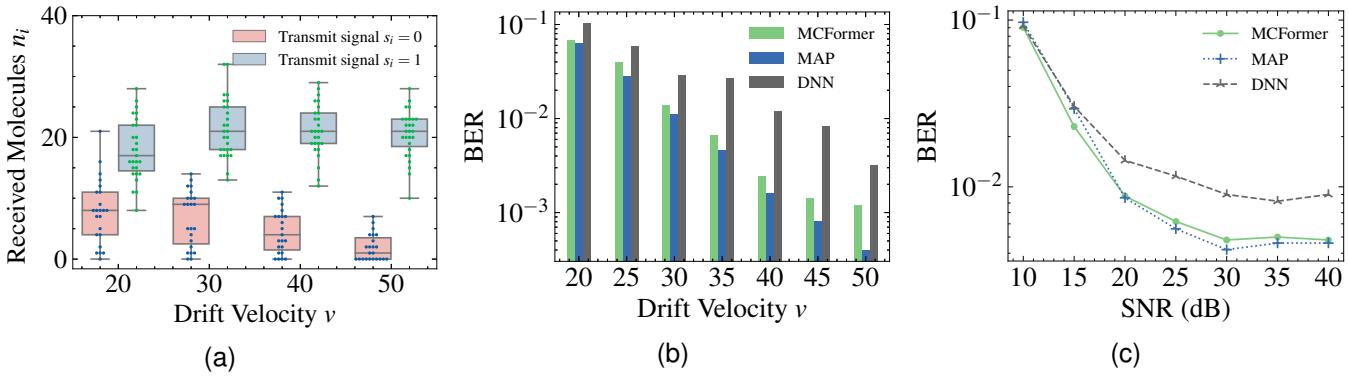

Fig. 4. Experiment results. (a) The boxplot at data with different drift velocity. (b) The BER performance comparison of MCFormer, DNN and MAP detector with different drift velocity. (c) The BER performance comparison of MCFormer, DNN and MAP under unknown channel noise.

that Transformer-based methods with attention structure can be well suit to attenuate the effects of ISI and thus can achieve near-optimal decoding accuracy without any knowledge of the channel information.

C. BER testing under unknown channel noise

To further test the performance of MCFormer under unknown channel noise, Additive White Gaussian Noise was introduced to the test sequence $N = [n_1, n_2, \cdots, n_L]$. Then the new input sequence to networks is $N' = [n'_1, n'_2, \cdots, n'_i, \cdots, n'_L]$, where $n'_i = n_i + (n_{noise})_i$ and $n_{noise} \sim \mathcal{N}(0, \sigma^2_{noise})$, σ^2_{noise} is variation of the added noise.

$$\text{SNR}_{\text{dB}} = 10\log_{10}(\frac{P_{signal}}{P_{noise}}) = 10\log_{10}(\frac{\frac{\sum_{i=1}^{L} n_i^2}{L}}{\sigma^2_{noise}}) \quad (7)$$

The Signal-to-Noise Ratio (SNR) is defined as the ratio of the average power of the signal P_{signal} to the average power of noise P_{noise} in Eq. 7, where L represents the length of N.

Fig. 4c shows the results at drift velocity $v = 40$, which illustrates the superiority of the MCFormer algorithm over DNN, consistently achieving a lower BER across multiple SNRs. Notably, MAP maintains its optimal performance in low-noise scenarios, particularly when the SNR exceeds 30. In high-noise environments where MAP lacks awareness of the noise, MCFormer demonstrates enhanced performance. These findings highlight the capability of our proposed detector, MCFormer, to perform effectively even in the absence of known channel information.

VI. Conclusion

In this paper, we propose MCFormer, a novel detector design for MC based on the Transformer model. We also introduce an accelerated particle-based simulation method using matrix method to significantly speed up data generation which can be widely used in similar simulation. In our detection experiments, the results demonstrate the efficacy of applying NLP techniques to MC, particularly in mitigating the impact of ISI on signal detection. The proposed MCFormer model achieves superior decoding accuracy compared to traditional methods, and is able to handle large quantities of data in a more efficient manner. The results also suggest that the structure of NLP models can play an important role in achieving better performance in MC tasks and warrants further exploration.

References

[1] S. Hiyama, Y. Moritani, T. Suda, R. Egashira, A. Enomoto, M. Moore, and T. Nakano, "Molecular communication," *Journal-Institute of Electronics Information and Communication Engineers*, vol. 89, no. 2, p. 162, 2006.

[2] U. A. Chude-Okonkwo, R. Malekian, B. T. Maharaj, and A. V. Vasilakos, "Molecular communication and nanonetwork for targeted drug delivery: A survey," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 3046–3096, 2017.

[3] R. Mosayebi, H. Arjmandi, A. Gohari, M. Nasiri-Kenari, and U. Mitra, "Receivers for diffusion-based molecular communication: Exploiting memory and sampling rate," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 12, pp. 2368–2380, 2014.

[4] V. Jamali, A. Ahmadzadeh, C. Jardin, H. Sticht, and R. Schober, "Channel estimation for diffusive molecular communications," *IEEE Transactions on Communications*, vol. 64, no. 10, pp. 4238–4252, Aug. 2016.

[5] A. K. Shrivastava, D. Das, and R. Mahapatra, "Performance evaluation of mobile molecular communication system using neural network detector," *IEEE Wireless Communications Letters*, vol. 10, no. 8, pp. 1776–1779, 2021.

[6] B.-H. Koo, H. J. Kim, J.-Y. Kwon, and C.-B. Chae, "Deep learning-based human implantable nano molecular communications," in *ICC 2020-2020 IEEE International Conference on Communications (ICC)*. IEEE, 2020, pp. 1–7.

[7] C. Bai, A. Zhu, X. Lu, Y. Zhu, and K. Wang, "Temporal convolutional network based signal detection for magnetotactic bacteria communication system," *IEEE Transactions on NanoBioscience*, pp. 1–1, 2023.

[8] L. Floridi and M. Chiriatti, "Gpt-3: Its nature, scope, limits, and consequences," *Minds and Machines*, vol. 30, pp. 681–694, 2020.

[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[10] I. Llatser, D. Demiray, A. Cabellos-Aparicio, D. T. Altilar, and E. Alarcón, "N3sim: Simulation framework for diffusion-based molecular communication nanonetworks," *Simulation Modelling Practice and Theory*, vol. 42, pp. 210–222, 2014.

[11] A. Noel, K. C. Cheung, R. Schober, D. Makrakis, and A. Hafid, "Simulating with accord: Actor-based communication via reaction–diffusion," *Nano Communication Networks*, vol. 11, pp. 44–75, 2017.

[12] W. Wicke, A. Ahmadzadeh, V. Jamali, H. Unterweger, C. Alexiou, and R. Schober, "Magnetic nanoparticle-based molecular communication in microfluidic environments," *IEEE transactions on nanobioscience*, vol. 18, no. 2, pp. 156–169, 2019.

[13] M. S. Leeson and M. D. Higgins, "Error correction coding for molecular communications," in *2012 IEEE International Conference on Communications (ICC)*. IEEE, 2012, pp. 6172–6176.

[14] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 10 2014, pp. 1532–1543.