Parallel Simulation Framework for Nanonetworking

Kerim Gökarslan and Erhan Çağırıcı

June 1, 2017

1 Introduction / Motivation

Nano Networking is one of the newest fields of computer networks. According to IEEE's definition of Nano Networking, a network is considered as a nano network if any component of that network has an element such that at least one of its dimension is less than or equal to 100 nm[1]. On the other hand, a communication model called as Molecular Communication if it uses selected types of molecules to digitally encode messages, and since most of the molecules are in the nano dimension, most of the molecular communication models are kinds of nano networking.

One of the fundamental principles of molecular communication is Brownian Motion or Random Walk. That is, all of the molecules have some kinetic energy due to their non-zero temperature and they continuously move with or without any bias. This movement is modelled with Brownian Motion. Moreover, some external bias, a flux for instance, might be seen sometimes in that communication.

Most scientist working on molecular communication needs computer simulation in order to analyze their communication models. There are a number of molecular communication simulators however most of them are developed for specific purposes rather than general molecular communication simulation. We, however, want to design and implement a general purpose molecular communication simulator so that one could be simulate their model with only editing configuration files.

Our aim in this project is not only generating a general purpose molecular communication simulator but also a simulator that works on GPU rather than CPU. Since most of simulation scenarios include a lot of molecules, and parallel computing mechanism very suitable for that. The fact that modern GPUs could have more than thousand cores in them, using a GPU will be very fast than using a eight core CPU. We are designing our project with markable scientists in molecular communication and the lead of Prof. Tugcu, who is a member of 1906.1-2015 - IEEE Recommended Practice for Nanoscale and Molecular Communication Framework group.

We will also want to distribute our project as open source, so that if one wants to change a simulation technique or implement a specific situation, he or she is able to implement. Our aim is to attract new researchers into the nano-communications area and, encourage and support the ones who are already working in this area by providing an useful tool on their research.

2 State of the art

One of the proposed models is the molecular communication approach. It uses molecules as the information carrier. "Inspired by the communication methods used by biological systems (eukaryotic and prokaryotic cells), a variety of communication systems are proposed in the literature"[2]. For short to medium range communications communication via diffusion (CvD) is proposed as well as calcium signaling, microtubules, pheromone signaling, and bacterium-based communication[3].

Among these communication methods communication via diffusion is currently being studied by our advisor Prof. Tugcu and his research group. This approach tries to encode information on specific messenger molecules and receive the diffused molecules at the receiver somehow in the sense that real biological cells communicate.

For this purpose we often collaborate with bio-physicians on understanding molecule behavior inside body, how various molecules interact with each other and what mechanisms does biological organisms have in terms of cell communications that might be used as a reference when building a communication model. This might help deciding a messenger molecule and receivers.

Molecular communication is a fairly new area, but extensive studies on molecular behavior and movement were done in the past. And with the rapid increase in computation power shifted this studies into computer simulations. Molecular systems are highly complex, therefore reaching an analytical solution for the complete system is not always possible. Therefore, simulating according to known small movements helped reaching a solution for such systems. These approaches are investigated under Molecular Dynamics (MD) . MD approach considers physical phenomena such as interatomic forces, force fields etc. This model is widely adopted in chemistry applications. Interested reader can further be informed from cited paper at https://udel.edu/~arthij/MD.pdf [4]

Due to long computation times and high parallelization capabilities of molecular simulation, GPUs have gained popularity in this area.

3 Methods

Our methodology follows basic molecular simulation workflow.

- Get environment properties from user.
- Determine medium characteristics. Calculate step size of a molecule.
- Place receiver(s) and transmitter(s).
- Generate molecules from transmitter release points.
- Move all molecules
- Check collisions, boundaries, and whether the molecules are received. Fix movements if needed.
- Collect statistics

For the sake of simplicity and scope of the study, molecular interactions and collisions are ignored. In the context of molecular communication, the probability of collusion of two particles moving according to Brownian motion in 3D space is shown to be negligible. Therefore intermolecular interactions are not taken into account.

To gain improvement from GPU parallelization, we try to express the operations on the molecules as matrix operations. This is a main design constraint for our simulation tool to work efficiently.

4 Experiments - Verification

4.1 Probability of hitting a spherical absorber

To verify our simulation model, we implemented the case with an one point-like transmitter and one spherical receiver, which is analytically explained in [3]. We are interested in the number of molecules hitting the receiver across time. The expected number of received molecules after t time from releasing is:

$$\frac{r_r}{r_0}\,\mathrm{erf}[\frac{r_0 - r_r}{\sqrt{4Dt}}] \tag{1}$$

Where D is diffusion coefficient of the environment, erf is the error function, r_r is the distance between the transmitter and the center of the receiver and r_0 is the radius of the receiver. Henceforth, we selected 6 different configurations for the verification. For each case 1000 simulation is run. Figure 1 depicts this configurations, their analytical solutions along with the simulation results.

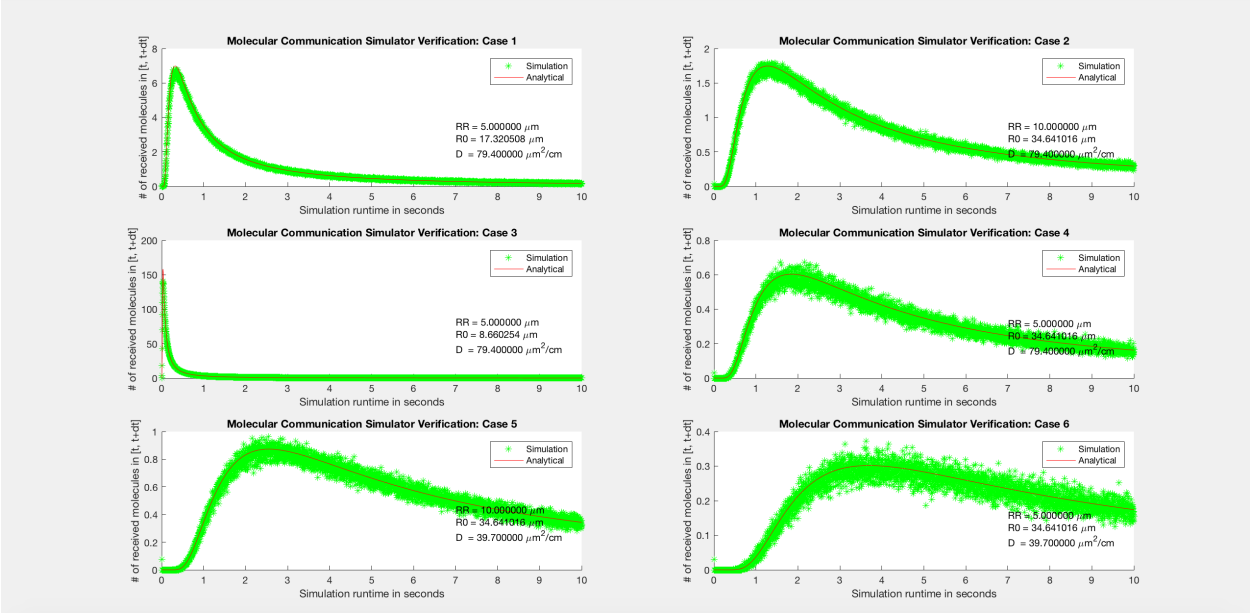

Figure 1: The single-shot communication between a point transmitter and a spherical receiver in an unbounded environment

As it can be seen from the figures, our simulation model is highly consistent with analytical results, and the more close the receiver and the transmitter, the more molecules are received, and the model is more consistent, as in the Case 3. The worst case, Case 6, on the other hand, is a setting where almost no communication can be done because the possibility of a single molecule to be received is much less than 1%

4.2 Spherical Transmitter to Spherical Receiver

The second verification model, is a spherical transmitter with a single-point transmission and spherical receiver where the transmitter have a reflecting surface. The simulation results of this setting is to be expected in two different cases:

- If the transmission point of transmitter is directly facing with receiver, the expected number of received molecules will be higher than the a point transmitter and a spherical receiver case
- If the transmission point of transmitter is not directly facing with receiver, i.e. the transmitter's volume is between the transmission point and the receiver, the expected number of received molecules will be less than the a point transmitter and a spherical receiver case.

As we do not finish our verification for this case yet, our first observations are shown in the below figure.

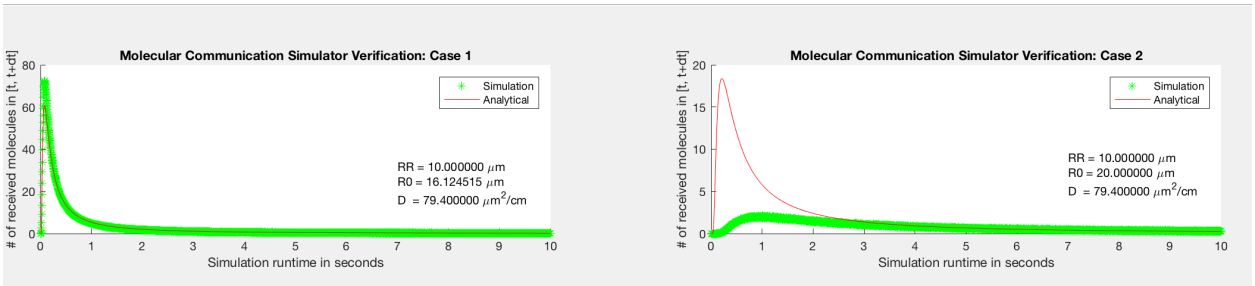

Figure 2: The single-shot communication between a spherical transmitter with a single point transmission and a spherical receiver in an unbounded environment

In both cases transmitter and receiver have the radius of 10 µm, transmitter centered at the point (30, 30, 30) and the molecules are released from the point (40, 30, 30). In the first case the receiver is centered at (42, 46, 30) and as our previous prediction the received number of molecules are higher than the case of only a point transmitter, thanks to reflecting surface of the transmitter. In the second case however, the receiver is centered at (18, 14, 30) and hence the volume of the transmitter are in between the release point and the receiver, and since it has a reflecting surface it causes the decrease in the number of received molecules. However, more analysis on this matter is required and currently we are running our simulation with different cases.

5 Results

As we concluded in the CmpE 491 the fact that running the same simulation algorithm on GPU is much more time efficient than running on CPU, which is shown in the figures below.

Config.	Diff. Coeff.	Delta Time	Run time	Symbol Size	Sym. Duration	Molecule Radius	Receiver Radius	Transmitter Radius
C1	79.4	0.005	100	100	0.5	0.0025	10	10
C2	79.4	0.005	100	100	1	0.0025	10	10
C3	79.4	0.005	100	250	1	0.0025	10	10
C4	79.4	0.0005	50	100	1	0.0025	10	10
C5	79.4	0.005	10	100	1	0.0025	10	10

Figure 3: Different simulation configurations to test performances of GPU and CPU

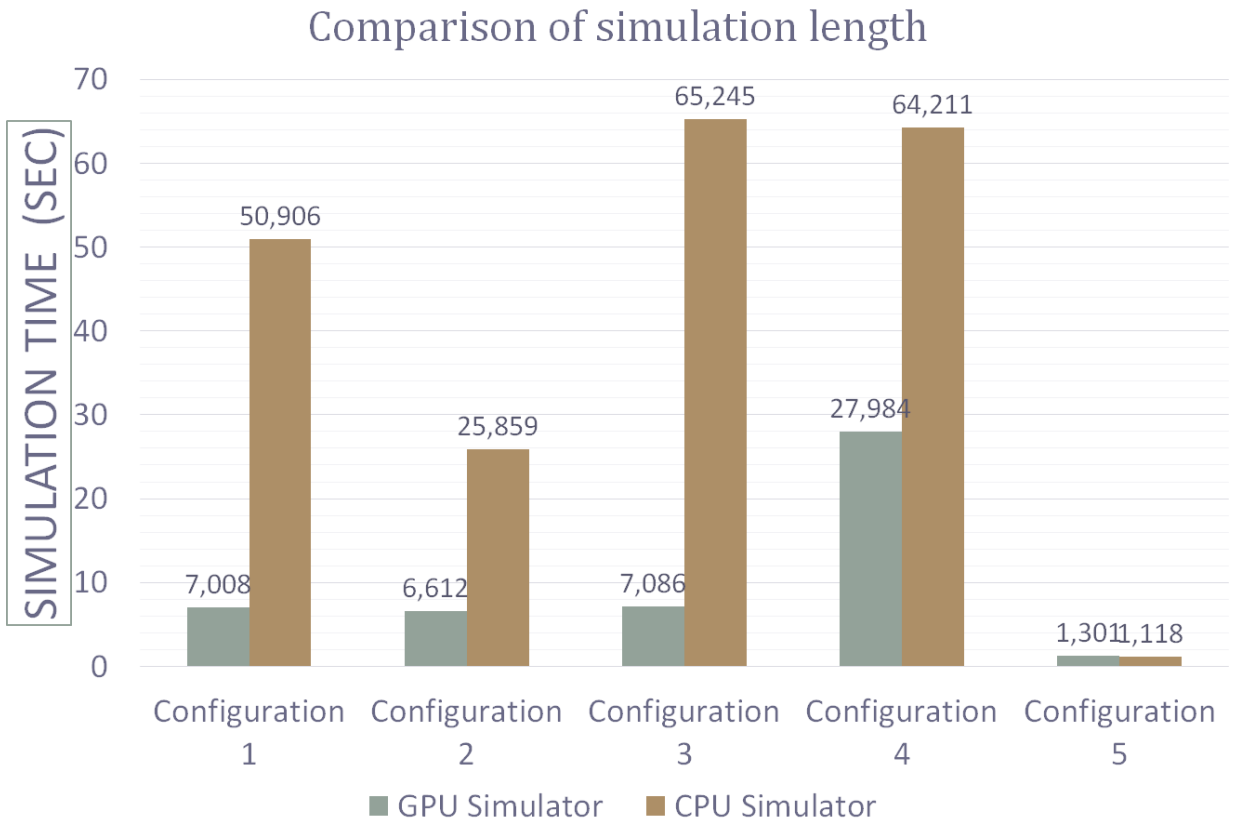

Figure 4: The average simulation runtime in seconds of configurations in Figure 3 (CPU: Intel i5-6600, GPU: NVIDIA GTX 1080)

We are, at this moment, verifying our simulation model with different scenarios and also implementing reflecting objects and boundaries.

6 Conclusion and Discussion

This project aims to implement a fast-working molecular communication simulator. As for now, we have implemented the point and spherical object functionalities of molecular communication simulator that works in GPU. As we mentioned in the results, working on GPU enormously decreases the runtime of simulator when the number of molecules is high.

7 Future Work

This project is aimed to be used in any molecular communication frames. We try to implement project as modular as possible such that anyone could implement a specific test configuration with no changes or minor changes in the source code.

We are currently working on implementing bounded environments, like cylinders or spheres, and also arbitrary combinations of these geometric shapes, including points, spherical objects, cylinders and rectangular shapes.

Depending on the advancement on the current version, we also want to extend our simulation design and capability to a larger and more generic state. The planned enhancements are mostly consist of more complex interactions between entities. For example, handling movements, rotations and collisions of complex irregularly shaped molecules, defining multiple properties that coexist on the same entity (ex. an entity with reflecting on one side, absorbing on the other). Therefore we aim to provide a more generic simulation framework which is more close to real world behavior.

References

[1] "IEEE Recommended Practice for Nanoscale and Molecular Communication Framework.," standard, Institute of Electrical and Electronics Engineers, Jan. 2016.

[2] M. Kuran, H. B. Yilmaz, , T. Tugcu, and B. Ozerman, "Energy Model for Communication via Diffusion in Nanonetworks," *Nano Communication Networks 1*, pp. 86–95, 2010.

[3] H. B. Yilmaz, A. C. Heren, T. Tugcu, and C.-B. Chae, "3-D Channel Characteristics for Molecular Communications with an Absorbing Receiver," *IEEE Communications Letters*, April 2014.

[4] M. Allen, "Introduction to molecular dynamics simulation," *John von Neumann Institute of Computing Series*, pp. 1–28, 2010.