# Difusi Bebas 1D dan 2D dengan Monte Carlo: Perbandingan Distribusi Bilangan Random Normal dan Seragam dengan Box-Müller

Fairusy Fitria Haryani (1,a)\*, Freddy Haryanto (2,b) dan Sparisoma Viridi (2,c)

(1) Prodi Pendidikan Fisika, Universitas Sebelas Maret, Surakarta, Indonesia, 57126 (2) Jurusan Fisika, Institut Teknologi Bandung, Bandung, Indonesia, 40134 Email: (a\*)fairusy.fita@staff.uns.ac.id, (b) dudung@gmail.com, (c) freddy@fi.itb.ac.id

Diterima (28 Juli 2020), Direvisi (30 Januari 2021)

Abstract. Many biological processes in the human body are based on the diffusion system. Various studies of diffusion have been done both experimentally and computationally. Because the particle interaction is stochastic, the Monte Carlo (MC) method is used in performing particle simulations. The main of MC method is the use of random numbers. Many software have provided uniform random number generators. But based on the analytic results, the solution is normal distribution. Therefore, Box-Müller can be used as a transformation of particle distribution. The software used, MATLAB, has a normal random generator. Hence, the aim of this study is to compare particle distribution of these two different random number generators with MATLAB and showing the impact of timestep parameter to these random number generators. The results can be used as based for the modelling of more complex biological systems.

**Keywords**: Diffusion, monte carlo, random number generator.

Abstrak. Banyak proses biologis dalam tubuh manusia didasari oleh sistem difusi. Berbagai kajian dalam proses difusi telah banyak dilakukan baik secara eksperimen maupun komputasi. Karena interaksi partikel yang terjadi dalam proses difusi bersifat stokastik, maka metode Monte Carlo (MC) digunakan dalam menjalankan simulasi difusi partikel. Dasar utama dalam metode MC adalah penggunaan bilangan random. Banyak software telah menyediakan pembangkit bilangan random berdistribusi seragam. Namun berdasarkan hasil analitik, solusi persamaan difusi memiliki distribusi normal. Oleh karena itu, metode Box-Müller dapat digunakan sebagai transformasi distribusi partikel tersebut. Software yang digunakan untuk menjalankan simulasi, MATLAB, memiliki pembangkit bilangan random berdistribusi normal. Oleh karena itu, penelitian ini bertujuan untuk membandingkan hasil distribusi partikel dari dua pembangkit bilangan random yang berbeda dengan menggunakan software MATLAB dan menunjukkan pengaruh parameter timestep terhadap kedua algoritma tersebut. Hasil proses difusi tersebut dapat digunakan sebagai landasan dalam pemodelan struktur sistem biologis yang lebih kompleks.

Kata kunci: Difusi, monte carlo, pembangkit bilangan random.

#### **PENDAHULUAN**

Manusia merupakan makhluk hidup yang memiliki struktur dan sistem fungsional biologis yang cukup kompleks. Berbagai metode telah dikembangkan dalam mengkaji segala proses biologis yang terjadi dalam tubuh. Pada dasarnya, banyak aktivitas biologis dalam tubuh manusia melibatkan sistem difusi dan proses reaksi [1-4]. Salah satu bentuk peristiwa difusi yang terjadi dalam tubuh adalah proses transpor oksigen menuju sel darah merah [5-7]. Saat alveolus mengandung banyak oksigen, pipa kapiler dekat alveolus yang mengandung sel darah merah akan mendapatkan asupan oksigen akibat proses difusi oksigen dari alveolus. Dengan demikian, arah gradien konsentrasi adalah dari yang lebih tinggi menuju ke yang rendah. Berdasarkan penjelasan tersebut, difusi dapat didefinisikan sebagai proses bergeraknya molekul dari daerah yang memiliki konsentrasi lebih tinggi menuju daerah yang konsentrasinya lebih rendah [8].

Gerak partikel difusi dalam proses biologis bersifat stokastik. Hal dikarenakan jangkauan konsentrasi partikel adalah nanomolar hingga mikromolar. Sehingga gerak partikel tersebut dapat digambarkan melalui metode Monte Carlo (MC). Karena prinsip simulasinya yang memanfaatkan serangkaian bilangan random, maka metode MC telah banyak digunakan dalam membantu simulasi kajian sistem biologis. Salah satunya adalah kajian perhitungan interaksi pada proses difusi partikel larutan DNA terkonsentrasi dan molekul, protein dalam skala vang dilakukan oleh Dwyer dan Bloomfield [9]. Metode MC telah berhasil menghitung partikel probabilitas interaksi yang berdifusi tersebut. Selain itu, Gomer [10]. telah berhasil mensimulasikan komunikasi dan perilaku sel dengan menggunakan metode MC.

Berbagai mekanisme difusi partikel yang terjadi dapat dikelompokkan menjadi beberapa proses difusi berdasarkan syaratsyarat dan sifat-sifat yang digunakan. Salah satunya yaitu proses difusi bebas, yang dapat digambarkan sebagai proses difusi dalam suatu ruang membran transparan atau tidak terdapat batasan ruang partikel untuk berdifusi. Dalam penelitian ini, proses difusi partikel digambarkan dalam bentuk distribusi penyebaran partikel pada posisi dan waktu tertentu dalam 1-dimensi (1D) dan 2dimensi (2D). Hal ini bertujuan untuk mengetahui probabilitas partikel yang berdifusi dengan menggunakan MC.

Salah satu penentu penting dalam metode MC adalah pembangkit bilangan random atau Random Number Generator (RNG). Kebanyakan software memiliki dasar RNG vang berdistribusi seragam. seperti Matlab, Excel, Phyton, dan lain-lain. Bahkan saat ini, banyak yang telah menyediakan package RNG dalam berbagai pilihan distribusi. Dengan demikian, bagaimana pola distribusi partikel sangat ditentukan oleh RNG. Oleh karena itu, dalam penelitian ini, akan dilakukan perbandingan dua teknik pembangkit bilangan random yang berbeda, yaitu melalui pembangkit bilangan random normal dan transformasi Box-Müller untuk bilangan random seragam.

Dalam metode komputasi, optimasi parameter sangat penting untuk dilakukan, salah satunya *timestep*. *Timestep* atau *dt* merupakan ukuran waktu dalam simulasi. Besar kecilnya parameter yang digunakan sangat berpengaruh dalam waktu komputasi dan hasil simulasi. Selain membandingkan dua algoritma yang berbeda, penelitian ini juga akan menunjukkan pengaruh *timestep* dan optimasinya untuk kedua metode pembangkit bilangan random.

Berdasarkan penjelasan tersebut, penelitian ini bertujuan untuk: membandingkan distribusi partikel yang berdifusi bebas dalam 1D dan 2D dengan pembangkit bilangan random normal dan Box Müller dan menunjukkan pengaruh timestep untuk kedua pembangkit bilangan random.

## **METODE PENELITIAN**

Salah satu variabel penentu dalam proses difusi adalah konsentrasi. Finucane [11] mendefinisikan konsentrasi sebagai jumlah kuantitas zat (mol) tiap satuan volume. Oleh karena itu, kuantitas konsentrasi menitikberatkan pada jumlah partikel dalam ruang. Dengan kata lain, konsentrasi juga dapat digambarkan sebagai

fungsi dari ruang (r) dan waktu (t) yang dilambangkan dengan c(r,t).

Dalam peristiwa difusi, hukum Fick menjadi salah satu dasar munculnya persamaan diferensial difusi linier. Berdasarkan analisis hukum I Fick dan persaman kontinuitas, dapat dirumuskan persamaan diferensial difusi linier yang juga disebut sebagai hukum II Fick, sebagaimana dituliskan **Persamaan 1**.

$$\frac{\partial c(r,t)}{\partial r} = D\nabla^2 c(r,t) \tag{1}$$

Faktor kesebandingan D menggambarkan koefisien difusi. Karena konsentrasi menitikberatkan pada jumlah partikel dalam ruang, maka konsentrasi juga menggambarkan distribusi partikel pada posisi dan waktu tertentu. Apabila probabilitas distribusi partikel pada posisi (r) dan waktu (t) tertentu dengan posisi awal  $r_0$  dilambangkan dengan  $p(r,t|r_0)$ , maka konsentrasi dapat dituliskan seperti pada **Persamaan (2)**.

$$c(r,t) = p(r,t \mid r_0) \tag{2}$$

Untuk menyelesaikan persamaan diferensial difusi tersebut digunakan syarat awal, yang menyatakan konsentrasi partikel saat t = 0, yaitu memenuhi fungsi Delta Dirac. Oleh karena itu kondisi awal konsentrasi yakni saat t = 0 dapat dituliskan seperti **Persamaan (3)**.

$$p(r,0 | r_0) = \mathcal{O}(r - r_0)$$
 (3)

Melalui penerapan syarat awal, jika posisi dinyatakan dalam sumbu-x, maka solusi **Persamaan** (1) untuk 1D didapatkan seperti pada **Persamaan** (4).

$$p(x,t|x_0) = \frac{1}{4\pi Dt} e^{\frac{(x-x_0)^2}{4Dt}}$$
 (4)

Solusi **Persamaan** (4) juga dapat disebut sebagai fungsi Green. Sedangkan untuk peristiwa difusi yang digambarkan dalam 2D, fungsi Green antar kedua arah bersifat independen satu sama lain. Dengan demikian, melalui teknik separasi variabel, solusi persamaan difusi 2D dapat dituliskan seperti **Persamaan** (5).

$$p(x, y, t | x_0, y_0) = p(x, t | x_0) p(y, t | y_0)$$

$$p(x, y, t | x_0, y_0) = \frac{1}{4\pi Dt} e^{-\frac{(x - x_0)^2 + (y - y_0)^2}{4Dt}}$$
(5)

Salah satu teknik dalam simulasi Monte Carlo adalah random walk. Random walk dapat digambarkan ketika partikel bergerak dalam 1D (horisontal), maka ia hanya memiliki kemungkinan gerak ke kanan atau ke kiri. Dengan demikian probabilitas gerak partikel baik ke kanan ataupun ke kiri adalah sama, yaitu ½. Hal ini memunculkan asumsi bahwa gerak partikel yang dihitung adalah rerata dan diasumsikan tidak terjadi interaksi antarpartikel. Oleh karena itu, penyebaran partikel ditentukan melalui Root Mean Square Distance (RMSD). Melalui hasil analisis mikroskopik dan hukum I Fick, didapatkan definisi RMSD yaitu seperti pada Persamaan (6).

$$\langle x^2(t)\rangle^{1/2} = \sqrt{2Dt} \tag{6}$$

Berbagai *software* telah menyediakan pembangkit bilangan random berdistribusi seragam, salah satunya MATLAB. Bilangan random seragam yang dibangkitkan merupakan bilangan desimal dari nol (0) hingga satu (1), yang memiliki notasi perintah rand (). Tidak hanya itu, MATLAB juga memiliki pembangkit bilangan random yang berdistribusi normal,

dengan perintah randn(), yang sesuai dengan **Persamaan (4)**. Tetapi, karena banyak pembangkit bilangan random dalam *software* berdistribusi seragam, maka metode Box-Müller diperlukan dalam transformasi distribusi seragam menjadi normal [11].

Selain menunjukkan gerak partikel dalam arah x dan y tidak saling dipengaruhi satu sama lain, **Persamaan** (5) juga menunjukkan bahwa visualisasi dalam 2D berbentuk radial simetris. Untuk itu, variabel random dapat didefinisikan dalam koordinat polar yaitu  $(R, \theta)$ , sebagaimana **Persamaan** (7) dan (8).

$$x = R\cos q \tag{7}$$

$$y = R\cos q \tag{8}$$

R merupakan vektor radius pada interval  $[0, \infty]$ . Sedangkan distribusi  $\theta$ bersifat seragam pada interval  $[0, 2\pi]$ . Dengan subtitusi asumsi tersebut ke dalam Persamaan didapatkan bentuk **(5)** transformasi variabel, seperti Persamaan (9) dan (10).

$$x = x_0 + \sqrt{-4Dt \ln U_1} \cos(2\rho U_2)$$
 (9)

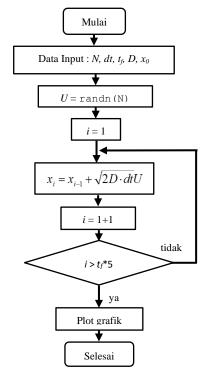
$$y = y_0 + \sqrt{-4Dt \ln U_1} \cos(2\rho U_2)$$
 (10)

 $U_1$  dan  $U_2$  merupakan bilangan random distribusi seragam (bilangan desimal) yang dihasilkan dari komputer pada interval [0,1].

Salah satu parameter penting dalam metode MC adalah *timestep* (*dt*). *Timestep* dapat didefinisikan sebagai interval waktu gerak partikel. Dengan demikian semakin kecil besar *dt*, maka gerak partikel akan semakin detail diketahui. Tetapi, hal tersebut mengakibatkan waktu komputasi yang akan lebih lama. Untuk itu, optimasi *dt* sangat penting untuk dilakukan. Selain itu, dengan munculnya faktor *dt* akan

mengakibatkan perhitungan posisi partikel didasarkan pada histori posisi partikel sebelumnya. Untuk menunjukkan pentingnya penggunaan dt, maka dapat disusun algoritma yang menggunakan histori dan tidak. Algoritma yang tidak menggunakan histori juga memanfaatkan histori, tetapi hanya untuk membandingkan waktu komputasi. Pemanfaatan dt untuk algoritma tanpa histori tidak berpengaruh hasil simulasi, banyak pada karena berapapun nilainya, hasil distribusi partikel tidak berubah. Algoritma tanpa histori hanya didasarkan pada posisi partikel saat t = 0 dan waktu partikel.

Dalam penelitian ini, algoritma Matlab R2016a akan disusun berdasarkan perintah rand () dengan metode Box-Muller dan randn () serta didasarkan ada atau tidaknya histori partikel dengan dt. melakukan optimasi Gambar mengungkapkan algoritma untuk difusi bebas partikel 1D dengan randn() dengan histori.

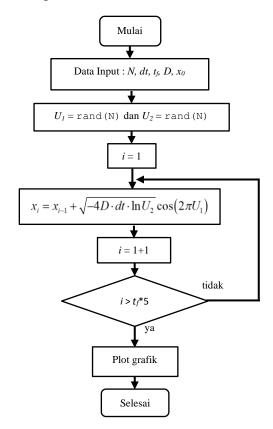


**Gambar 1.** Algoritma difusi bebas partikel 1D dengan randn () dan histori.

Sedangkan **Gambar 2** menggambarkan algoritma dengan histori untuk difusi bebas 1D dengan rand () dan Box-Müller.

Untuk algoritma tanpa histori baik untuk randn() dan rand() dengan Box-Müller, karena hanya didasarkan pada posisi awal partikel saat t=0, maka besarnya posisi partikel sebelumnya yang diungkapkan oleh indeks  $x_{i-1}$  diubah menjadi  $x_0$  yang nilainya selalu tetap.

Pada kasus difusi bebas 2D, karena fungsi distribusi *x* dan *y* bersifat independen satu sama lain, maka algoritmanya dapat dilakukan dengan teknik separasi variabel biasa, yakni dengan menambahkan variabel *y* selama proses iterasi.

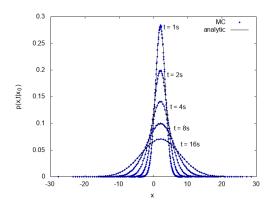


**Gambar 2.** Algoritma difusi bebas partikel 1D dengan rand () dan histori.

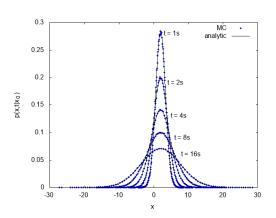
#### HASIL DAN PEMBAHASAN

Parameter tetap dalam simulasi yang dijalankan adalah jumlah partikel (N), posisi awal partikel  $(x_0)$ , koefisien difusi (D), dan waktu akhir distribusi partikel  $(t_f)$ . Berdasarkan simulasi MC yang dilakukan Plante dan Cucinotta [12], pada **Persamaan** (9) dan (10) diketahui bahwa untuk jumlah partikel sebanyak 1.000.000 telah memenuhi konvergensi. Sedangkan nilai parameter  $x_0$ ,  $y_0$ , D, dan  $t_f$  secara berturut turut adalah 2, 0, 1 dan 16.

Simulasi tahapan awal yang dilakukan adalah distribusi partikel yang berdifusi baik dengan menggunakan randn() dan rand() tanpa histori untuk dt = 1 s. Pemilihan dt dimulai dari nilai yang maksimum untuk meminimalkan waktu komputasi. Hasil distribusi kedua algoritma tersebut secara berturut-turut digambarkan pada **Gambar 3** dan **4**.



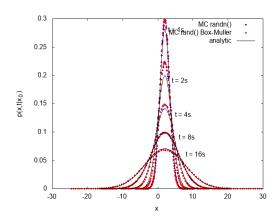
**Gambar 3.** Distribusi partikel difusi bebas 1D dengan randn() tanpa histori saat dt = 1 s.



**Gambar 4.** Distribusi partikel difusi bebas 1D dengan rand() Box-Muller tanpa histori saat dt = 1 s.

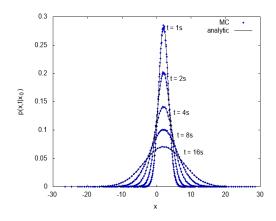
Gambar 3 mendeskripsikan perbandingan antara perhitungan analitik dengan algoritma randn () tanpa histori. Untuk nilai dt yang besar, dt = 1 s, sudah memenuhi perhitungan analitik. halnya dengan Gambar yang menggambarkan perbandingan perhitungan analitik dengan rand() dan Box-Müller tanpa histori dengan nilai dt yang sama. Karena nilai dt maksimum, maka untuk nilai dt yang lebih kecil, hasil grafik sudah memenuhi analitik. Seperti yang telah disebutkan sebelumnya bahwa nilai dt dalam algoritma tanpa histori tidak berpengaruh terhadap hasil distribusi.

Pada proses difusi yang menggunakan histori partikel, seperti pada algoritma Gambar 1 dan 2, hasil distribusi partikel difusi digambarkan pada Gambar 5. Gambar 5 mengungkapkan hasil distribusi partikel dengan menggunakan randn() histori dan rand() dengan Box-Müller dan histori untuk  $dt = 10^{-1}$  s. Berdasarkan Gambar 5, meskipun kurva distribusi telah membentuk kurva distribusi normal, namun pembangkit bilangan kedua random tersebut belum memenuhi hasil perhitungan analitik untuk seluruh waktu difusi. Dengan demikian, dilakukan optimasi dt supaya untuk semua waktu memenuhi hasil analitik.

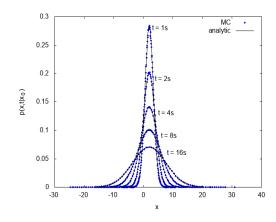


**Gambar 5.** Distribusi partikel difusi bebas 1D dengan randn() dan rand() Box-Müller histori saat  $dt = 10^{-1}$  s.

Optimasi dt yang didapatkan untuk randn () dengan histori adalah 10<sup>-2</sup> s. Seperti halnya rand(), hasil optimasi dt algoritma rand() untuk Box-Müller didapatkan 10<sup>-2</sup> s. Hasil distribusi partikel pembangkit bilangan random tersebut digambarkan pada Gambar 6 dan Gambar 6 mengungkapkan hasil distribusi untuk randn () dengan histori untuk dt yang telah dioptimasi, yaitu 10<sup>-2</sup> s dan Gambar 7 menggambarkan hasil distribusi untuk rand() dengan Box-Müller dan histori saat dt yang juga telah dioptimasi, yaitu 10<sup>-2</sup>s.



**Gambar 6.** Distribusi partikel difusi bebas 1D dengan randn() dan histori saat  $dt = 10^{-2}$  s.



**Gambar 7.** Distribusi partikel difusi bebas 1D dengan rand () Box-Muller dan histori saat  $dt = 10^{-2}$  s.

Hasil distribusi partikel difusi baik dengan randn() dan rand() Box-Müller tanpa histori untuk dt vang besar, yaitu 1 s telah memenuhi hasil perhitungan analitik. Selain itu, berdasarkan waktu simulasi untuk kedua jenis pembangkit bilangan random tersebut, seperti pada Tabel 1. Waktu simulasi algoritma tanpa histori lebih singkat 0.5-0.6 s dibandingkan dengan algoritma yang menggunakan histori. Namun dengan dt yang sama, 0.1 s, waktu simulasi algoritma tanpa histori 10 kali lebih lama dibandingkan dengan histori. Hal ini dikarenakan setiap kelipatan bilangan bulat dari dt, perhitungan harus dimulai dari posisi awal  $x_0$ .

Tabel 1. Contoh format tabel yang terpotong

Jenis Algoritma Difusi		Waktu simulasi (s)
1D  randn () $dt = 1  s$	tanpa histori	4.12
1D randn() $dt = 10^{-1} \text{ s}$	tanpa histori	41.71
$1D$ randn() $10^{-1}$ s	histori $dt =$	3.99
$1D$ randn() $10^{-2}$ s	histori $dt =$	7.06
1D rand()	Box-Müller	4.77

tanpa histori $dt = 1$ s			
1D rand() Box-Müller tanpa histori $dt = 10^{-1}$ s	53.68		
1D rand() Box-Müller histori $dt = 10^{-1}$ s	4.82		
1D rand() Box-Müller histori $dt = 10^{-2}$ s	8.92		
$\frac{2D}{s}$ randn() histori $dt = 10^{-2}$	12.78		
2D rand() Box-Müller histori $dt = 10^{-2}$ s	16.15		

Dengan demikian untuk memulai perhitungan t yang baru, posisi awal partikel adalah saat t = 0. Sedangkan untuk algoritma dengan histori partikel, menyediakan data penyimpanan partikel sebelumnya, sehingga tidak perlu menghitung kembali dari t = 0. Dengan demikian untuk menghitung posisi saat t tertentu, hanya diperlukan data posisi partikel tepat sebelum *t*.

Tetapi, meskipun waktu simulasi dengan algoritma histori jauh lebih cepat, untuk dt yang sama, hasil distribusi partikel belum memenuhi perhitungan analitik karena untuk waktu tertentu distribusi maksimum partikel dengan metode MC berbeda dengan analitik, seperti ditunjukkan pada Gambar 5. Hal ini ditunjukkan dengan kurva distribusi partikel baik untuk randn() dan rand() dengan Box-Muller masih di atas kurva analitik. Tetapi, semakin lama waktu partikel, maka hasil distribusi partikel untuk kedua pembangkit bilangan random telah perhitungan memenuhi hasil analitik. Meskipun demikian, supaya berlaku untuk seluruh waktu, maka perlu dilakukan optimasi parameter dt.

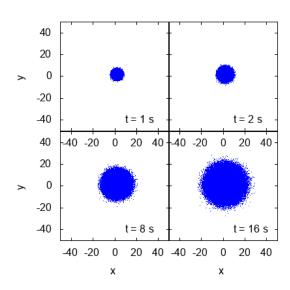
Berdasarkan hasil simulasi, didapatkan bahwa saat  $dt=10^{-2}~{\rm s}$  untuk algoritma randn () dengan menggunakan histori

partikel telah memenuhi perhitungan analitik, seperti pada **Gambar 6**. Karena ketelitian dt lebih kecil, maka waktu komputasi yang dibutuhkan akan lebih lama, mencapai dua kali lipat daripada saat  $dt = 10^{-1}$  s. Demikian halnya dengan menggunakan algoritma rand() Box-Muller dan histori, hasil distribusi partikel memenuhi distribusi difusi secara analitik saat  $dt = 10^{-2}$  s.

Waktu komputasi seluruh simulasi dalam dirangkum **Tabel** yang mengungkapkan bahwa baik pada algoritma dengan dan tanpa histori, waktu simulasi dengan randn() lebih singkat dibandingkan dengan rand() dengan Box-Müller. Semakin kecil nilai dt rentang waktu simulasi keduanya menjadi semakin iauh. Begitu pula untuk 2D. selisih waktu simulasi keduanya mencapai 4.5 s.

distribusi Hasil dengan 1D mengungkapkan bahwa baik untuk algoritma randn() dan rand() Box-Müller distribusi partikel yang terdifusi adalah sama dan optimasi dt juga sama, maka bentuk 2D partikel yang terdifusi pun juga sama. Gambar 8 menggambarkan partikel 2D dengan algoritma rand() Box-Müller menggunakan histori saat dt = $10^{-2}$  s untuk t = 1, 2, 8 dan 16 s. Berdasarkan hasil simulasi, semakin lama waktu partikel berdifusi, maka partikel semakin menyebar. Karena ukuran penyebaran partikel yang semakin besar, maka kerapatan partikel semakin kecil. Dengan demikian konsentrasi zat akan semakin kecil.

Setelah dilakukan beberapa hasil optimasi dt dan iterasi i perhitungan posisi untuk algoritma histori, dapat dirumuskan hubungan keduanya seperti **Persamaan** (11).



**Gambar 8.** Distribusi partikel difusi bebas 2D dengan rand() Box-Müller dengan histori saat  $dt = 10^{-2}$  s.

$$i = \sqrt{\frac{t}{dt}} \tag{11}$$

Melalui **Persamaan** (11) dapat ditentukan urutan iterasi *i* jika dibandingkan dengan *t* waktu difusi partikel secara analitik. Dengan demikian, ukuran *output* simulasi partikel dapat diminimalkan dan juga meminimalkan waktu simulasi.

### KESIMPULAN

Proses difusi bebas yang digambarkan dalam 1D dan 2D baik menggunakan algoritma randn() dan rand() dengan Box-Muller serta dengan dan tanpa histori: 1) telah memenuhi hasil perhitungan analitik ditandai dengan hasil distribusi partikel untuk tiap waktu; 2) besar *timestep* sangat berpengaruh pada hasil simulasi dengan algoritma histori; 3) optimasi *dt* untuk algoritma randn() sama dengan rand() dengan Box-Müller, yaitu  $10^{-2}$  s.

#### DAFTAR PUSTAKA

[1] I. Plante and F. A. Cucinotta, "Model

- of the initiation of signal transduction by ligands in a cell culture: Simulation of molecules near a plane membrane comprising receptors," *Phys. Rev. E Stat. Nonlinear, Soft Matter Phys.*, vol. 84, no. 5, pp. 1–13, 2011.
- [2] P. Mereghetti, D. Kokh, J. A. McCammon, and R. C. Wade, "Diffusion and association processes in biological systems: Theory, computation and experiment," *BMC Biophys.*, vol. 4, no. 2, pp. 1–6, 2011.
- [3] L. Zhou and J. Shen, "Signal Transmission of Biological Reaction-Diffusion System by Using Synchronization," *Front. Comput. Neurosci.*, vol. 11, no. 92, pp. 1–6, 2017.
- [4] V. E. Debets, L. M. C. Janssen, A. Saric, "Characterising the diffusion of biological nanoparticles on fluid and cross-linked membranes," *Soft Matter*, vol. 16, pp. 10628–10639, 2020.
- [5] S. Longeville and L. R. Stingaciu, "Hemoglobin diffusion and the dynamics of oxygen capture by red blood cells," *Sci. Rep.*, vol. 7, no. 1, pp. 1–10, 2017.
- [6] E. Ackerman, *Biophysical science*. Englewood Cliffs, N.J: Prentice-Hall, 1962.
- [7] A. S. Popel, "Theory of oxygen transport to tissue," *Crit. Rev. Biomed. Eng.*, vol. 17, no. 3, pp.

- 257–321, 1989.
- [8] A. S. Scott and E. Fong, *Body Structures and Functions 10th edition*. Florence KY: Delmar Cengage Learning, 2003.
- [9] J. D. Dwyer and V. A. Bloomfield, "Brownian dynamics simulations of probe and self-diffusion in concentrated protein and DNA solutions," *Biophys. J.*, vol. 65, no. 5, 1993.
- [10] R. H. Gomer, "The Use of Diffusion Calculations and Monte Carlo Simulations to Understand the Behaviour of Cells in Dictyostelium Communities," *Comput. Struct. Biotechnol. J.*, vol 17, pp. 684–688, 2019.
- [11] E. W. Finucane, Definitions, Conversions, and Calculations for Occupational Safety and Health Professionals. 2016.
- [12] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, Numerical Recipes in C book set: Numerical Recipes in C: The Art of Scientific Computing, vol. 15, no. 7. 1992.
- [13] I. Plante and F. A., "Monte-Carlo Simulation of Particle Diffusion in Various Geometries and Application to Chemistry and Biology," in *Theory and Applications of Monte Carlo Simulations*, 2013.

Fairusy Fitria Haryani: Difusi Bebas 1D dan 2D dengan Monte Carlo: Perbandingan Distribusi Bilangan Random Normal dan Seragam dengan Box Müller