DOI:10.34918/85111

Jarosław GIL[1,*], Krzysztof SZYMICZEK[1], Andrzej POLAŃSKI[1]

Chapter 11. IMPROVING GILLESPIE SIMULATION ALGORITHM FOR FITNESS IN CLONAL EVOLUTION

11.1. Introduction

Neoplastic transformations in human tissues are consequences of accumulating somatic, clonal mutations. In the ongoing research on cancers, observations on occurrences of somatic mutations are collected and then their roles in neoplasm are explained in biological and physiological terms. A strong impulse in studies on somatic mutations in cancers are provided by large, experimental projects of DNA and RNA sequencing of cancer tissues leading to creation of large databases of cancer mutations, such as TCGA project, TCGA database [1] and COSMIC database [2]. These researches led to significant advance in understanding cancer development as well as improving tools for diagnosis and therapy.

Large volumes of data and its detailedness encourages elaborating mathematical models, which would correspond to scenarios of cancer initiation and development. Mathematical modelling of tumor growth is based on probabilistic description of events seen in the neoplastic processes, cellular replications and deaths and occurrences of somatic mutations. Mathematical models most often used are Markov birth - death processes, branching processes or multitype branching processes [3–5].

Variety of possible, potentially complicated laws for probability distributions of events in mathematical models of neoplastic transformations motivate for developing stochastic simulations algorithms. Numerous papers devoted to simulation systems of evolution of cancer cell populations already appeared in the literature, e.g., [6–8]. The basic approach for simulation of events occurring in neoplastic processes is by using

[1] Silesian University of Technology, Poland.

[*] Corresponding author: jaroslaw.gil@polsl.pl.

Gillespie algorithm [9], i.e., the algorithm of successive updates of processes state vector and reactions / events propensities on the basis of simulated times of reactions / events. Application of Gillespie's algorithm to clonal evolution of populations was described e.g., in [4] and [10]. In [4] a very simple version of Gillespie's algorithm for simulating clonal evolution of cancer cells was used, with the state vector containing two components. However, it enabled simulating evolutionary effects of mildly deleterious passenger mutations leading to shrinking population size versus strongly advantageous driver mutations causing selective sweeps. In [10] a high-resolution simulation tool was presented with high dimensional state vector and mutations with different effects on fitness was presented. It enabled observing various scenarios of cancer clonal evolution but required substantial amount of computational power.

In this study we present Gillespie simulation algorithm for generating evolutionary events in developing cancer cells populations, with ability of simulating growth of cancer cells population detailed enough for observing moving wave of fitness of cancer cells. Mutations in our model have equal, mildly advantageous fitness effects. We pay attention to efficiency of simulation, aiming to achieving populations of sizes of millions of cells on desktop computer. We compare efficiency of two algorithms, one with the state vector containing components corresponding to each of the cells of the population, and another with smaller state vector built of bins corresponding to groups of cells with equal number of mutations.

11.2. Model description

The idea of the Gillespie algorithm provides methodology to simulate population evolution analyzing it cell by cell. For small population sizes that algorithm is very good but for large cell number it causes long simulation time. Improvement can make simulation faster with keeping results of calculations the same or similar.

11.2.1. Possible events in clonal evolution

Genetic forces behind clonal evolution are replications, mutations, selection and genetic drift. Mainly can be highlighted cell death, division and mutation. In Fig. 1 were presented phenomena simulated by Gillespie algorithm.

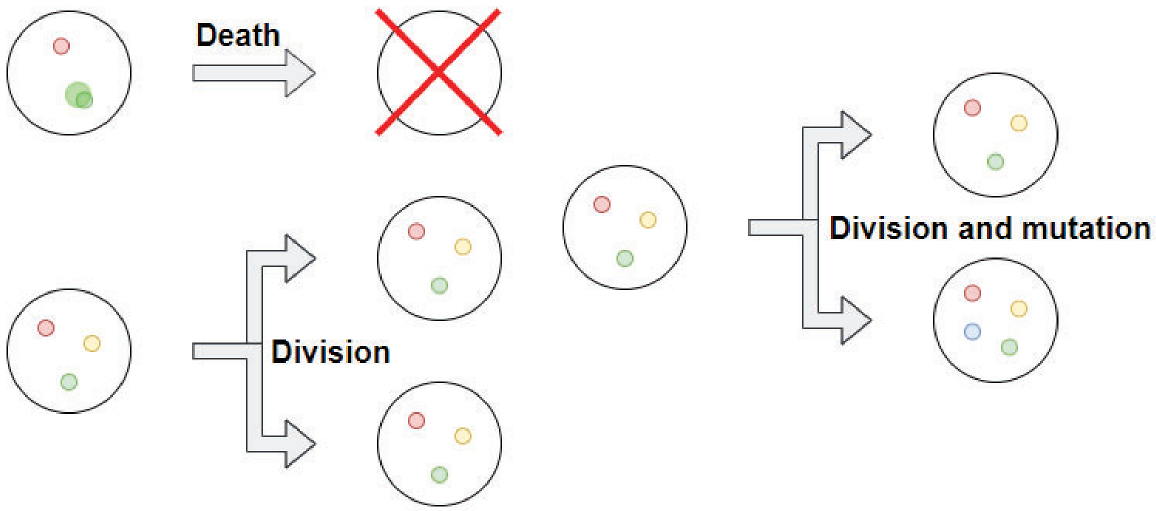


Fig. 1. Events analyzed in model
Rys. 1. Zdarzenia analizowane w modelu

Cell death is dependent on population capacity. If number of cells is greater than population capacity death probability coefficient should be larger. Equation for intensity of cell death process has the following form:

$$\mu_D = \frac{N}{K} \tag{1}$$

where N denotes population size and K stands for environment capacity.

Cell mutation can occur while cell is dividing. Cell division is dependent on fitness coefficient which increase while cell is mutating. Mutation can cast positive, neutral or negative effect on fitness coefficient. In clonal evolution process cell fitness should be mostly increasing to provide fast population growth. For simulation purposes mutation effect is assumed as positive value. Intensity of cellular divisions/births is different for each cell and is given by

$$\mu_B(l) = (1+s)^l \tag{2}$$

where s is the fitness effect of a single mutation and l is the number of mutations accumulated in the cell. Probability of occurrence of a mutation is denoted by p so the intensity of mutations process is

$$\mu_p(l) = p(1+s)^l. \tag{3}$$

11.2.2. Gillespie algorithm

The first approach involves simulating evolution process cell by cell treating cell division or death as two reactions and using first reaction or next reaction version of Gillespie's algorithm [11]. After analysis, the time value is subtracted from all other cells and for updated cell new death or division time is generated as random variable with exponential distribution. Event kind depends on which time variable is the smallest. For death cell is simply erased from population but while division cell mutation probability is checked. If cell mutates clone with one more mutation is added to population. Fig. 2 contains block diagram of that algorithm.

Finding the smallest time variable in population needs to compare all cells to each other. In worst case computational complexity of algorithm is equal to O((2n)!). In every loop one variable is compared to every other time variable in whole population composed of n cells. For small population this complexity is not a problem but for large initial size simulation time will be very long.

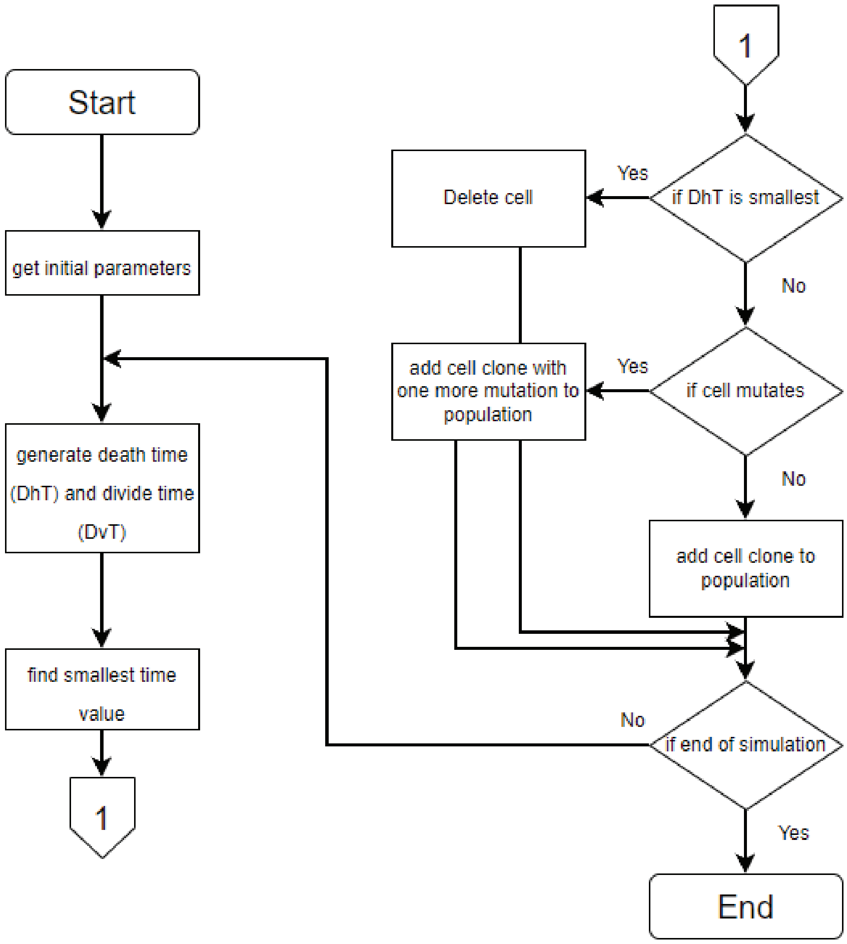


Fig. 2. Original Gillespie algorithm block diagram
Rys. 2. Schemat blokowy podstawowego algorytmu Gillespiego

11.2.3. Tau leap algorithm

To simplify algorithm complexity in one simulation cycle can be analyzed more cells. Specifying tau value – time step, limits number of iterations to population size. Fig. 3 contains block schema for tau loop algorithm. After time generation for all cells two comparisons are made – if death time or divide time is smaller than tau. First phenomena kind is determined by smaller number for one cell. Algorithm idea is the same as that of the original approach.

To update all cells analyzed in one cycle it is needed to compare both times to tau. For one cell two comparisons are done so the algorithm complexity can be described as O(2n). That mean simulation time is linear dependent on population size and one cycle time is smaller than in original approach.

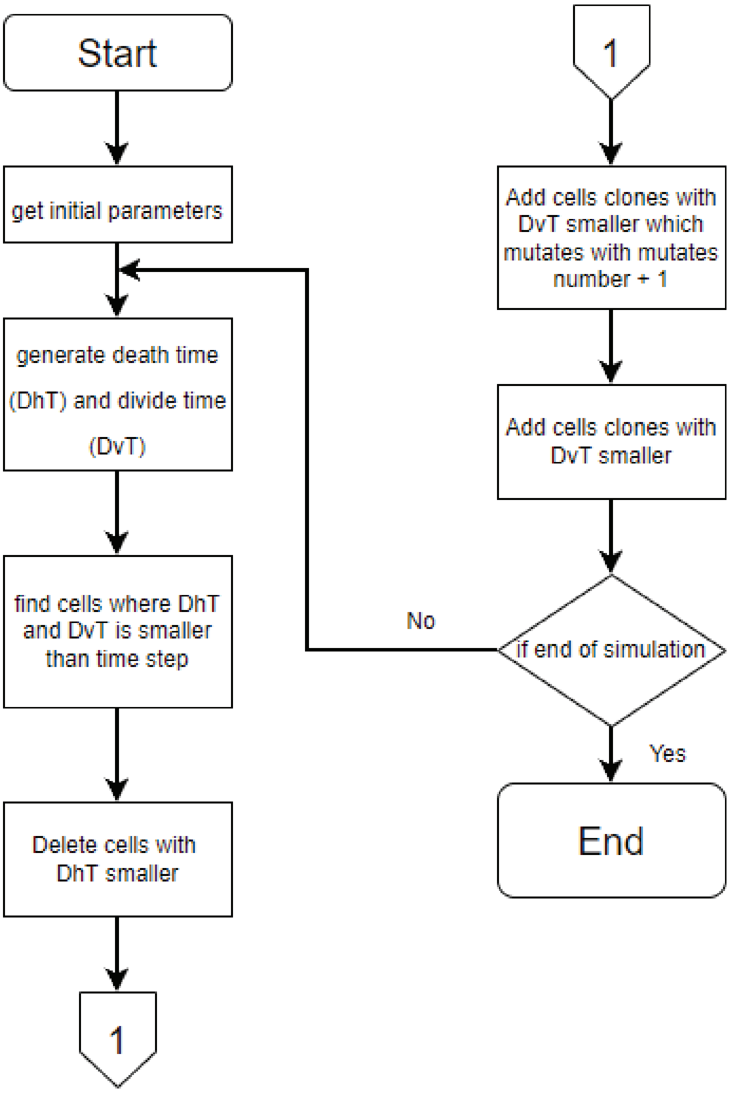

Start
get initial parameters
generate death time (DhT) and divide time (DvT)
find cells where DhT and DvT is smaller than time step
Delete cells with DhT smaller
1
1
Add cells clones with DvT smaller which mutates with mutates number + 1
Add cells clones with DvT smaller
if end of simulation
No
Yes
End

Fig. 3. “Tau loop” algorithm block diagram
Rys. 3. Schemat blokowy algorytmu „tau loop”

11.2.4. Binned Gillespie algorithm

Still lowering original Gillespie algorithm cause loss of data accuracy. More assumptions are needed to take which can provide false simulation data. To simplify Gillespie algorithm, we propose its binned version. For large initial population basic algorithm is very slow cause of iteration through whole population. Every cell can be characterized by mutation number which can provide method for cell grouping. In one cycle than would be much less groups than cells in population.

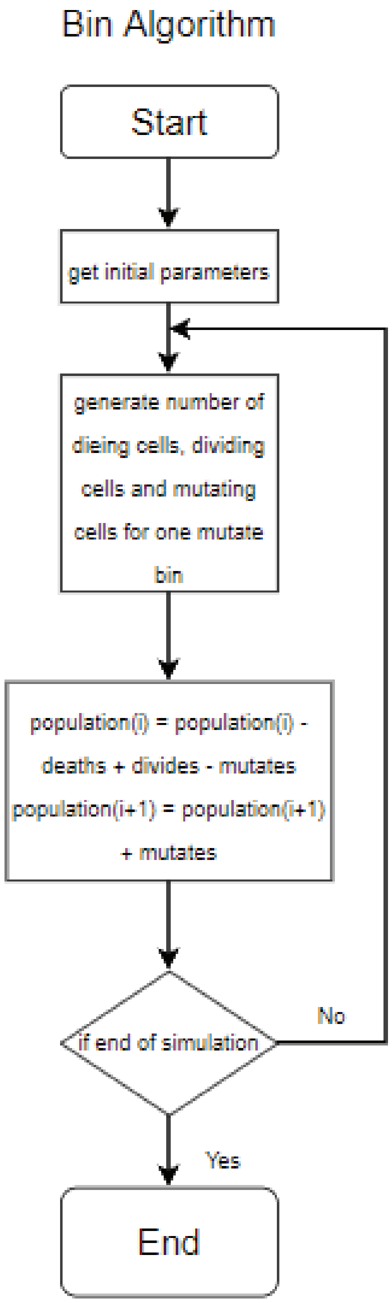

Bin Algorithm
Start
get initial parameters
generate number of dieing cells, dividing cells and mutating cells for one mutate bin
population(i) = population(i) - deaths + divides - mutates population(i+1) = population(i+1) + mutates
if end of simulation
No
Yes
End

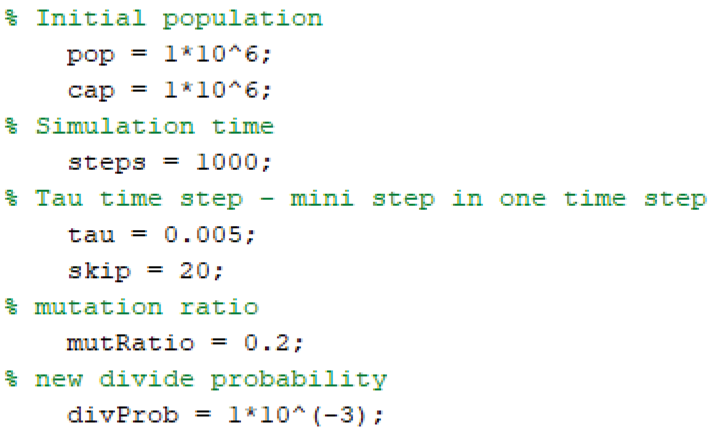
Fig. 4. Binned algorithm block diagram
Rys. 4. Schemat blokowy grupowanego algorytmu

Fig. 4 contains block schema for binned Gillespie algorithm. In each simulation cycle all mutation groups are updated by generating random probabilities based on population parameters. Death probability depends on population size and population capacity, division probability depends on fitness factor of each group. For cell mutation constant probability is assumed – part of dividing cells is mutating.

The complexity of that approach is equal to O(n) where n is interpreted as number of bins. For large population size number of mutation groups is much smaller then number of cells so simulation time is also smaller than cell-based algorithm.

11.2.5. Simulation parameters

To properly simulate population evolution few parameters are taken in consideration. Fig. 5 contains model initial parameters which are base for calculating events probability.



```
% Initial population
    pop = 1*10^6;
    cap = 1*10^6;
% Simulation time
    steps = 1000;
% Tau time step - mini step in one time step
    tau = 0.005;
    skip = 20;
% mutation ratio
    mutRatio = 0.2;
% new divide probability
    divProb = 1*10^(-3);
```

Fig. 5. Model initial parameters
Rys. 5. Początkowe parametry modelu

Death probability is calculated based on population size and population capacity. Divide probability depends on bin fitness. Mutation probability is assumed as constant value describing mutation-division ratio. In each simulation cycle only, events for cells with death/division time smaller than tau are occurring. In large population tau value can be interpreted as probability that event for cell will occurs. In simulation every calculated parameter is multiplied by tau. That describes how many cells should be updated.

11.3. Results and discussion

Introduced algorithm modification reduce algorithm complexity and simulation time. To analyze impact on data accuracy few experiments were made.

11.3.1. Algorithms comparison

To prove that binned version of Gillespie algorithm result accuracy does not differ from original and tau loop version comparison of results after 10 cycles is presented. Experiment was performed multiple times, for all attempts results were as described. Fig. 6 shows result of "tau loop" algorithm and binned algorithm. Simulation parameters assumed in that experiment are shown on Fig. 5.

Simulation time for one cycle is eight times greater for the "tau loop" approach. The complexity of binned version is better. Distribution of mutated cells are very similar. Can be assumed the differences between both results are neglectable and could occurs because of randomness in coefficient generation.

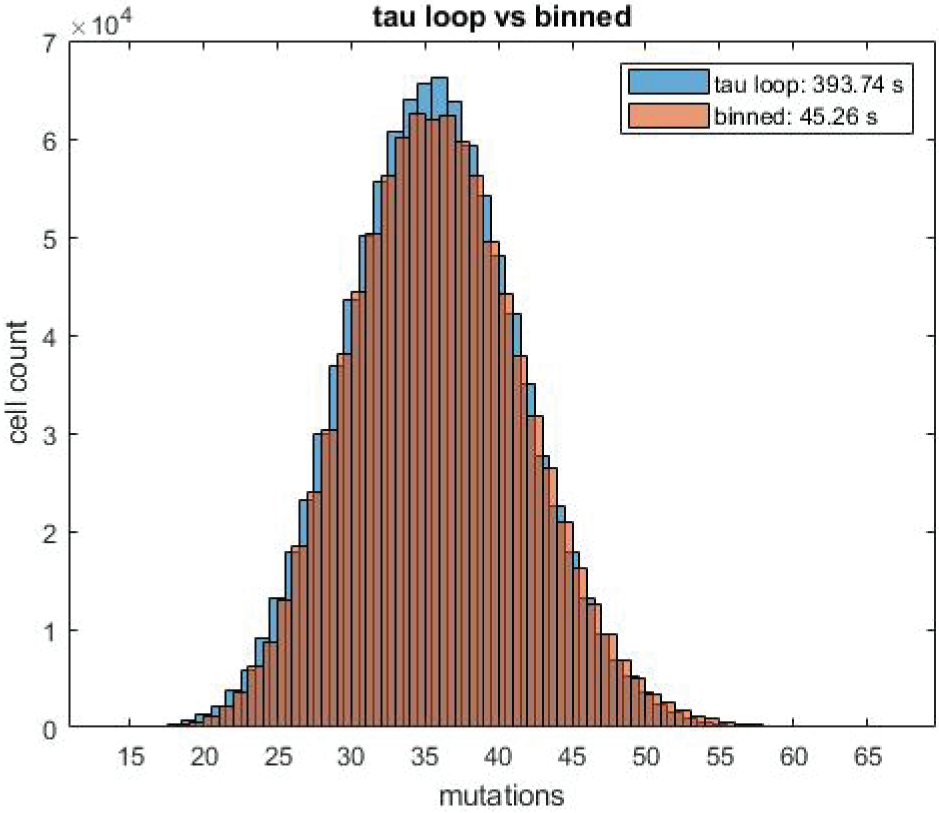


Fig. 6. "Tau loop" and binned algorithm comparison after 10 cycles of simulation
Rys. 6. Porównanie wyników algorytmów "tau loop" i zbinowanego po 10 cyklach symulacji

11.3.2. Mutation wave

Accumulation of somatic mutations in cancer clone can be seen a traveling wave of increasing numbers of mutations in cancer cells [12]. The mutation wave obtained in our simulations is presented on Fig. 6 for “tau loop” algorithm and binned algorithm. Clonal evolution can be described as chaotic process with causes very fast population growth and cell mutations. To analyze that phenomena experiments with simulation parameters were made. Fig. 7, Fig. 9 and Fig. 11 shows mutation wave speed while capacity, mutation/division ratio and fitness modifier were change.

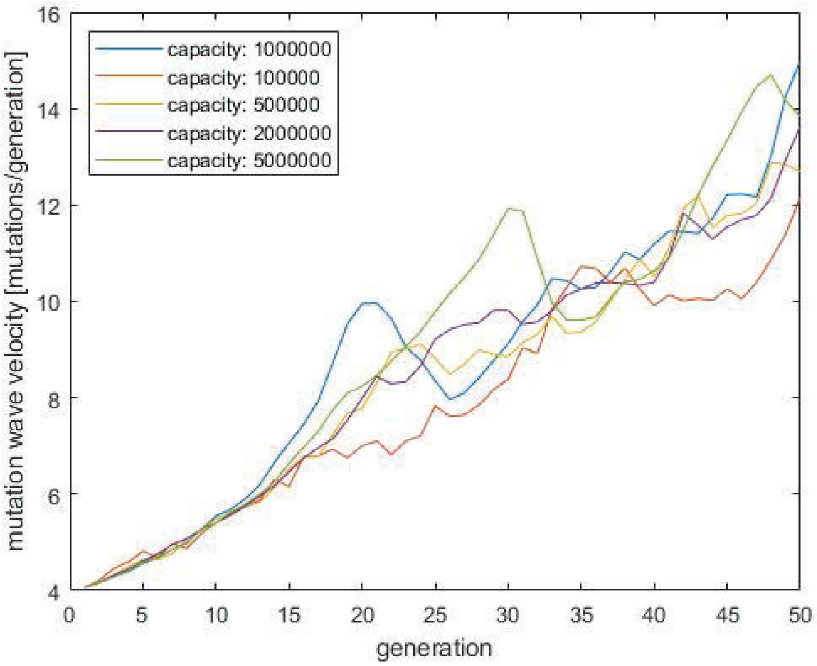


Fig. 7. Mutation wave speed versus population capacity
Rys. 7. Prędkość fali mutacji względem pojemności populacji

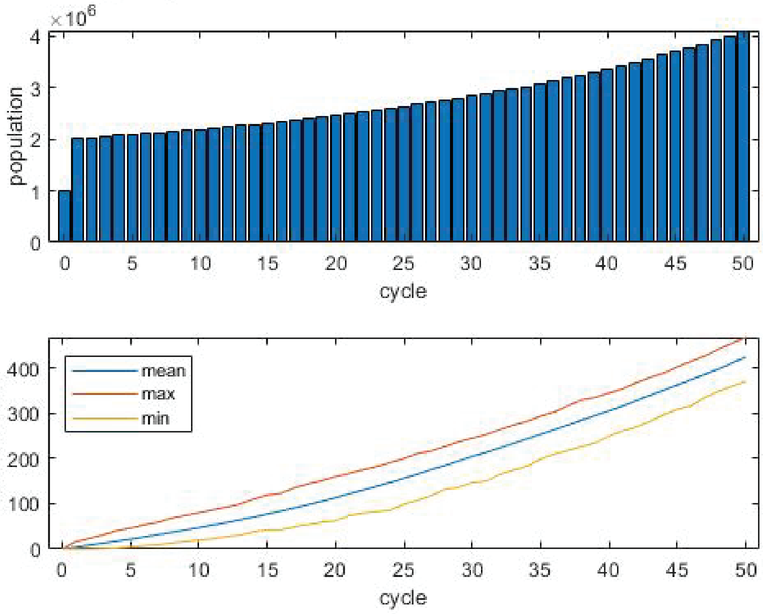


Fig. 8. Population statistics across 50 generations for capacity equal 2000000
Rys. 8. Statystyka populacji przez 50 generacji dla pojemności 2000000

Population capacity seems to have no impact on mutation wave. Independently on its value mutation wave velocity rise with number of generations. When cells population mutates, fitness factor changes in positive way. The cells are mutating and dividing more spontaneously providing wave velocity rise and population growth. Fig. 8 shows population statistics for capacity two times larger than initial population. The number of cells at the very beginning doubles and still rises. Mutation number also rises – at simulation beginning slowly then faster.

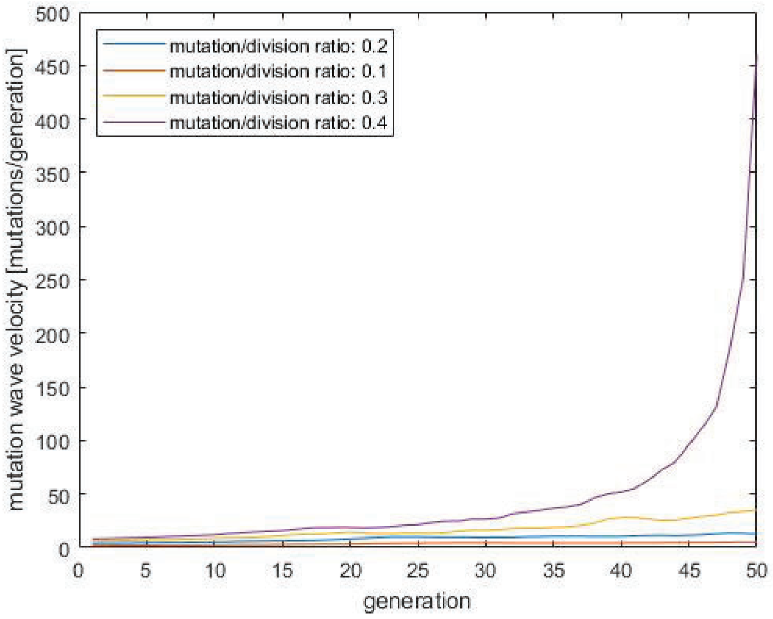


Fig. 9. Mutation wave velocity versus mutation/division ratio
Rys. 9. Prędkość fali mutacji względem współczynnika mutacji

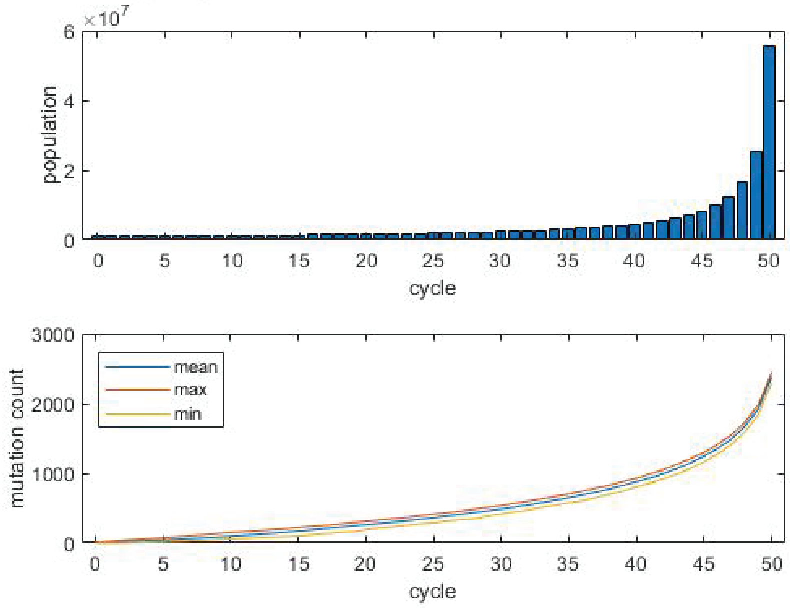


Fig. 10. Population statistics across 50 generations for mutation/division ratio equal 0.4
Rys. 10. Statystyka populacji przez 50 generacji dla współczynnika mutacji 0.4

Mutation-division ratio can be interpreted as chance for cell to mutate while dividing. Its value has the highest impact on mutation probability what is shown on Fig. 9. For small ratio values wave velocity rise slowly. Small rose of ratio causes enormous change in wave speed and population growth. High velocity of mutation wave causes many mutations in cells what is shown on Fig. 10. Also, population size is growing rapidly.

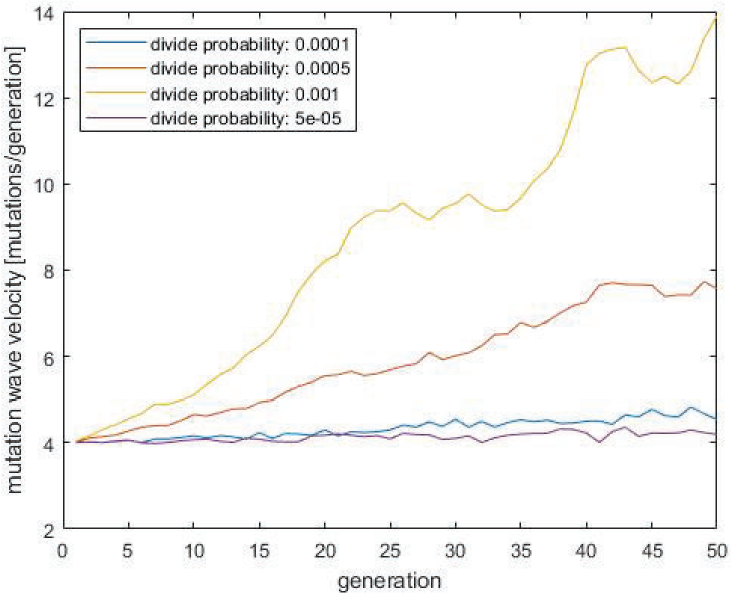


Fig. 11. Mutation wave velocity versus fitness factor modifier
Rys. 11. Prędkość fali mutacji względem modyfikatora współczynnika dopasowania

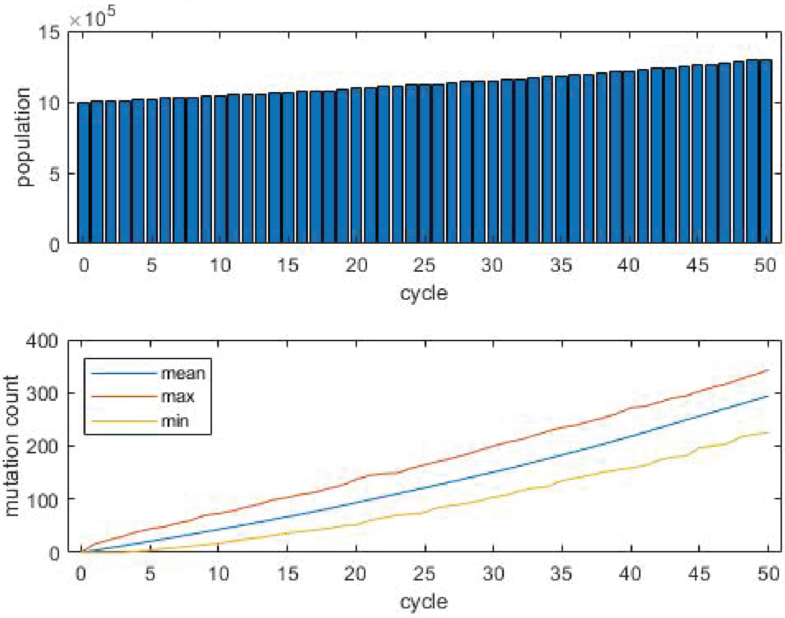


Fig. 12. Population statistics across 50 generations for fitness modifier equals 0.0005
Rys. 12. Statystyka populacji przez 50 generacji dla modyfikatora dopasowania 0.0005

In Fig. 11 is presented mutation wave velocity while changing division probability modifier – fitness factor modifier. Every mutation has impact on cell division probability. In experiment was assumed only positive effect. When its value rise, wave velocity also rises. For small values wave velocity seems to stay constant. Population size, as have been shown on Fig. 12, rises slowly for small modifier value. Mutation number changes nearly linear.

11.3.3. Fitness wave

The moving mutation wave can also be interpreted as the wave of fitness moving in the population of cancer cells. Fitness factor provide information about population adaptation. If its value is higher cell division probability also is higher. Fig. 13, Fig. 14 and Fig. 15 contains fitness waves of few generations dependent on capacity, mutation/division ratio and fitness factor modifier. Data present on these figures complements information from 0.

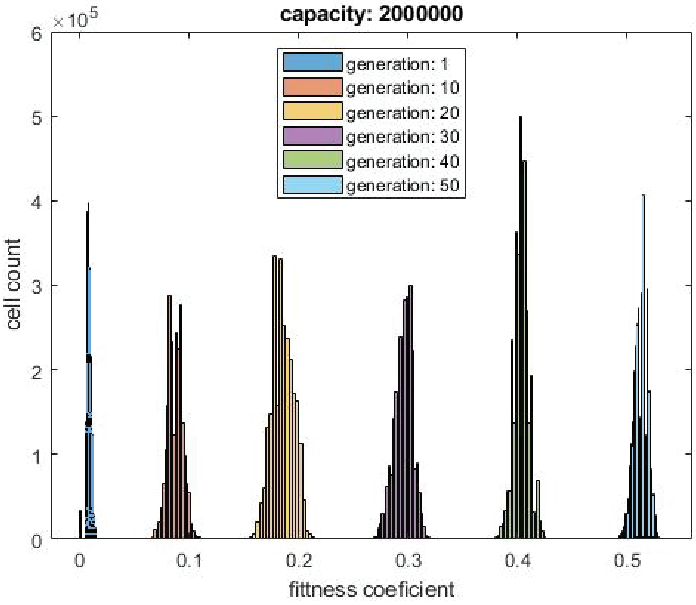


Fig. 13. Fitness wave for population capacity 2000000
Rys. 13. Fala dopasowania dla pojemności populacji 2000000

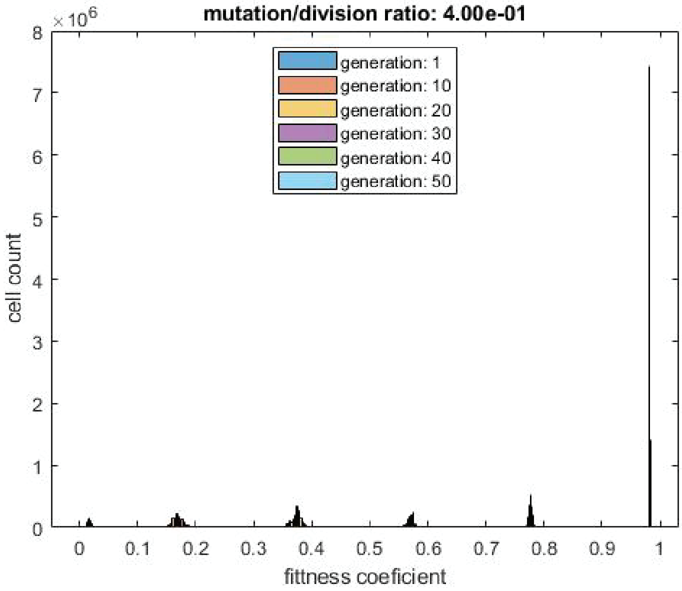


Fig. 14. Fitness wave for mutation/division ratio 0.4
Rys. 14. Fala dopasowania dla współczynnika mutacji 0.4

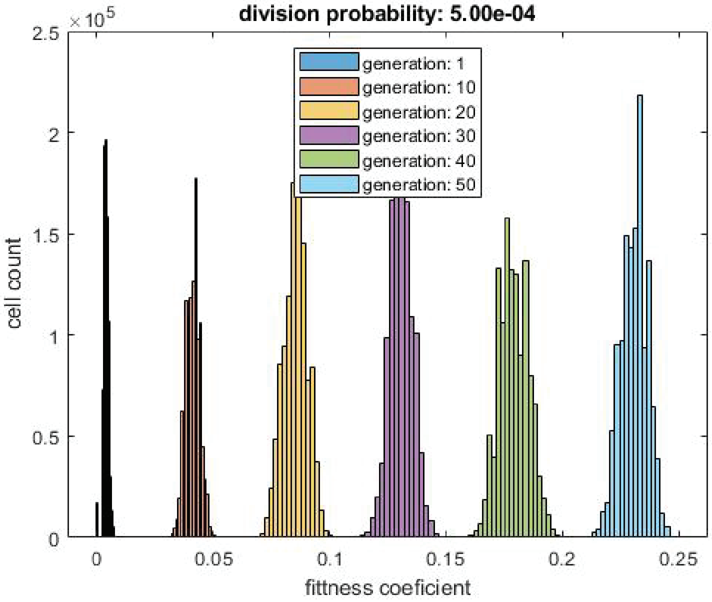


Fig. 15. Fitness wave for fitness modifier 0.0005
Rys. 15. Fala dopasowania dla modyfikatora dopasowania 0.0005

Moving fitness wave provide information about population adaptation so also about population growth. If fitness wave is steady, population evolution should be slow and random. For positive movement of fitness wave can be observed growing cell number and also mutation number.

11.4. Conclusions

Studying clonal evolution of tumor cells reveals changing dynamics of the size of tumor as well as of numbers of somatic mutations in cancer cells. These dynamics is a derivative of acquisition of somatic mutations in the cells and is related with mutations wave moving forward. In principle, the occurring mutations may alter cell fitness/adaptation in different ways: if the point mutation occurs at a gene causing the cell to divide or survive more likely, this mutation gives an advantage for the cell and the underlaying gene. Likewise, the mutation can occur at gene with little to none effect on cell fitness/adaptation or it can cause deterioration of evolutionary fitness of the cancer cell carrying it.

In this study we analyzed simulation scenarios of evolution of cancer clones with each somatic mutation causing small increase of fitness of cancer cells. We have elaborated and implemented two versions of Gillespie's algorithm and we have pursued several computational/simulation experiments.

We have observed phenomena in genomic clonal evolution of cellular populations described in the literature, population growth in response to increasing adaptation of cells and traveling wave of advantageous mutations and fitness in the cancer cells population.

Acknowledgements

This publication was supported by the Department of Graphics, Computer Vision and Digital Systems, under research project for young scientists (Rau6, 2022), Silesian University of Technology (Gliwice, Poland).

Bibliography

1. Liu Jianfang, et al.: An integrated TCGA pan-cancer clinical data resource to drive high-quality survival outcome analytics. Cell 173.2 (2018): 400–416.
2. Bamford Sally, et al.: The COSMIC (Catalogue of Somatic Mutations in Cancer) database and website. British journal of cancer 91.2 (2004): 355–358.
3. Bozic I., et al.: Accumulation of driver and passenger mutations during tumor progression, PNAS, Vol. 107, no. 43, p. 18545–18550
4. McFarland C.D., et al.: Tug-of-war between driver and passenger mutations in cancer and other adaptive processes, PNAS, Vol. 111, no. 42.
5. McDonald Thomas O., Marek Kimmel: Multitype infinite-allele branching processes in continuous time. Journal of Applied Probability, 54.2 (2017): 550–568.
6. Hoban Sean, Giorgio Bertorelle, Oscar E. Gaggiotti: Computer simulations: tools for population and evolutionary genetics. Nature Reviews Genetics, 13.2 (2012): 110–122.
7. Miller Christopher A., et al.: Visualizing tumor evolution with the fishplot package for R., BMC genomics, 17.1 (2016): 1–3.
8. McDonald Thomas O., Franziska Michor: SIApopr: a computational method to simulate evolutionary branching trees for analysis of tumor clonal evolution. Bioinformatics 33.14 (2017): 2221–2223.
9. Gillespie D.: Approximate accelerated stochastic simulation of chemically reacting systems. Journal of Chemical Physics, Volume 115, Number 4, p. 1716–1733.
10. Szymiczek K., et al.: CCES: cancer clonal evolution simulation program, in Advances in Intelligent Systems and Computing vol 659 2194-5357, 5th International Conference on Man-Machine Interactions, ICMMI, 2017, Cracow, Poland, Springer International Publishing, p.: 172–181.

11. Anderson David F.: A modified next reaction method for simulating chemical systems with time dependent propensities and delays. The Journal of chemical physics 127.21 (2007): 214107.
12. Neher Richard A.: Genetic draft, selective interference, and population genetics of rapid adaptation. Annual review of Ecology, evolution, and Systematics, 44 (2013): 195–215.

IMPROVING GILLESPIE SIMULATION ALGORITHM FOR FITNESS IN CLONAL EVOLUTION

Abstract

Haploidal clonal evolution governs adaptation dynamics of many populations to environmental conditions. Importantly, clonal evolution stands behind growth of cancer cells populations in human tumors. Genetical forces behind haploidal evolution are replications, mutations, selection and genetic drift. Due to the lack of recombination in the process of haploidal evolution of a population one observes formation of population clones, i.e., subpopulations of identical/similar genetic profiles.

Important area of studying evolution of clones in haploid evolution is mathematical modelling. Due to nonlinearity, interference of several forces and large scale of models mathematical modelling is often supported by computer simulations. In this study we present a simulation system for modelling clonal evolution of haploidal populations based on Gillespie scenario of generating evolutionary events. Due to large cellular/bacterial population we propose modifications of the algorithm based on binning subgroups of cells with equal number of mutations and generating distributions of times of cellular divisions, deaths and mutations in subgroups. We demonstrate results of simulations and improvements in efficiency of modelling due to introduced mutations.

Keywords: clonal evolution, mutation waves, numerical model, stochastic simulation, Gillespie algorithm