STOCHASTIC DIFFUSION MODEL OF HETEROGENEOUS POPULATIONS

by

Armen Zakharyan

M.S., UCD, 2006

A thesis submitted to the
University of Colorado at Denver
and Health Sciences Center
in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
Applied Mathematics
2008
This thesis for the Doctor of Philosophy degree by Armen Zakharyan has been approved by

Karen Kafadar

Stanley Payne

Arnold Levinson

Barbara Bailey

Lisa Johansen

Kathe Bjork

1 May 2009
Date
Zakharyan, Armen (Ph.D., Applied Mathematics)
Stochastic diffusion model of heterogeneous populations
Thesis directed by Professor Karen Kafadar

ABSTRACT

Kolmogorov's first and second equations in partial differential equations can be used to find solutions of transition probabilities in simple systems; while they could be solved for simple cases, it is virtually not possible to solve them for complex processes that involve more than 3 or 4 population types. In this work we propose a framework which allows exact simulation of such complex processes. This framework will aid in our understanding in many processes in various areas of science. In this work we develop a generic approach to simulation of interactions between populations of particles in an arbitrary shaped closed volume. The work describes direct (mesoscopic) simulation of particles, where each particle is represented as a computational unit in the program, and stochastic simulation of particles, where particle behavior is modeled using probabilistic and statistical techniques. We simulate directly and stochastically the following interactions: random flight, collision, chemoattraction/repulsion and aging/death. By showing that the stochastic simulation algorithm (SSA) and the direct simulation algorithm (DSA) are equivalent, the SSA can be applied to simulation of large scale populations. The complexity of the SSA is deter-
mined by the size of the finite-volume partitioning, and the complexity of DSA is determined by the size of populations, which is, in many simulation tasks, computationally intractable. Therefore, the aim of this work is to show that SSA can be applied to simulation of large-scale populations and to provide an illustrative example using a model of disease progression. The disease used in the illustration is multiple sclerosis, a paralytic disease of the central nervous system initiated by myelin specific autoaggressive T cells. Many processes in this disease are well suited for simulating with SSA. Mechanisms triggering relapses in multiple sclerosis are not well understood; they are the result of complex interactions between immune and nervous systems containing large populations of cells, proteins, etc. We apply the developed approach to simulation of some of the processes in a closed volume that contains several populations interacting with each other in the vascular, interstitial and endothelial compartments.

This work presents a new (faster) algorithm for simulations of interacting populations in space, a new implementation for statistical properties of collided particles, and a new implementation for calculated simple systems using Kolmogorov’s first and second equations in generating functions. As a part of this work we developed a software that is based on the developed algorithm.

This work was motivated by complexity of processes that take place in the phenomena of multiple sclerosis. The complexity of processes that occur in different spaces in different times makes it computationally impossible to find a solution in a closed form. Moreover, because the number of particles and the number of reactions that are taking place in such problems are very large, it is also very difficult to develop an algorithm that simulates all reactions in a
relatively short time.

The SSA model is based on branching processes and processes with particles interaction. We consider a community of different cell types that meets the following conditions. The entire community is divided into several populations. Each type of cell in the community is described by the set of parameters, which can be changed with the age of the cell. These parameters are the same for one species of a population and different for populations of different species. During their lifetimes, cells can divide, interact with other cells and be exposed to the influence of external factors. The division property of a cell, and also the intensity, with which it interacts with other types of cells, depend on cell parameters. Cells could die either as a result of interactions with other cells or because they have reached the end of their lifetimes.

This thesis describes the model for the case when the community consists of a single isolated population and then the more generalize case involving an arbitrary number of populations. The model assumes that 1) the population is subject to certain continuous external shocks, the intensity of which at various specimens is not the same and is proportional to the weight, and 2) the interaction between the populations of individuals could be one of \( m \) different types. Following the model specifications is a description of the modeling algorithm in the SSA model. The DSA model employs Monte-Carlo simulation where each individual particle is simulated, in contrast to the SSA model, where simulation is executed at the level of finite-volume partitions. Deterministic solutions for simple systems will be compared with the results when run under DSA.
This abstract accurately represents the content of the candidate's thesis. I recommend its publication.

Signed
Karen Kafadar
DEDICATION

This work is dedicated to my uncle Michael.
ACKNOWLEDGMENT

This thesis would not have been possible without the generous support of my adviser, Professor Karen Kafadar.
CONTENTS

1. Introduction ......................................................... 1
2. Compare solutions of different methods .......................... 8
4. Direct Simulation Algorithm ...................................... 23
   4.1 The Algorithm .................................................. 23
   4.2 Random walk/diffusion. ........................................ 28
   4.3 Chemo-attraction. ............................................... 33
   4.4 Reaction upon collision, birth and death of objects .......... 33
   4.5 Tau Leaping Algorithm ......................................... 49
   4.6 Model of Infiltration Demyelination .......................... 53
   4.7 Identification .................................................. 55
5. Elements of theory and research tasks ........................... 58
   5.1 Markov branching processes with a finite number of types of particles 58
   5.2 Markov process on the $N^n$. Kolmogorov equations .......... 58
   5.3 Branching process with $n$ types of particles ............... 60
   5.4 Multidimensional generating functions ........................ 61
   5.5 Equations of branching processes ............................ 63
   5.6 Probabilities of extinction .................................... 65
   5.7 Equations for generating functions ........................... 65
   5.7.1 Equations for expectations ................................. 66
6. Transformations of the form $A \rightarrow \gamma_1 A + \gamma_2 B; B \rightarrow 0$ .......................... 68
6.1 Transformation $A \rightarrow A + B, 2A + B; B \rightarrow 0$ .......................... 68
6.2 Transformation: $A \rightarrow 0, B, A + B, 2A; B \rightarrow 0$ .......................... 71
6.2.1 System of non-linear equations ..................................................... 71
6.3 Transformation $A \rightarrow 0, A + B, 2A + B, 2A; B \rightarrow 0$ .......................... 75
6.4 Transformation $A \rightarrow B, 2A + B; B \rightarrow 0$ .......................... 79
7. Transformations of the form $A \rightarrow \gamma_1 A + \gamma_2 B; B \rightarrow 2B$ ..................... 83
7.1 Transformation $A \rightarrow 2A + B; B \rightarrow 2B$ .......................... 83
7.2 Transformation $A \rightarrow 0, B, A + B, 2A; B \rightarrow 2B$ .......................... 86
7.3 Transformation $A \rightarrow B, 2A + B; B \rightarrow 2B$ .......................... 90
8. Other forms of transformations ............................................................. 93
8.1 Transformation $A \rightarrow B; B \rightarrow A$ .......................... 93
8.2 Transformation $A \rightarrow A + B; B \rightarrow A$ .......................... 96
8.3 Transformation $A \rightarrow A + B; B \rightarrow A + B$ .......................... 99
8.4 Transformation $A \rightarrow 2A; B \rightarrow A + 2B$ .......................... 101
9. Conclusion .................................................................................. 106

Appendix
A. The pseudocode .......................................................... 108
A.0.1 Description of Vixdum ................................................. 108
A.0.2 Pseudocode .......................................................... 119

Figures .................................................................................. 153

References .............................................................................. 158
1. Introduction

Kolmogorov’s first and second equations in partial differential equations can be used to solve for the transition probabilities in simple systems; while they can be solved for simple cases, it is virtually not possible to solve them for complex processes that involve more than 3 or 4 population types. In this work we propose a framework that allows exact simulation of such complex processes. Such a framework will aid in understanding in many processes in various areas of science. For example, processes that lead to infiltration of T-Cells from the vascular space to the interstitial space are not well understood. Various types of objects such as cells, and proteins, peptides interact together to transform into new states, produce proteins, bind with each other, etc. It is believed that this complex interplay among populations of these objects defines the location of infiltration of activated T-Cells into the brain, which follows by attack of infiltrated activated T-Cells against the myelin, a protective layer around axons of the neural cells [82]. Numerous pathways of interactions among populations of the objects have been described. Since it is their cooperative interaction that results in a blood brain barrier breakdown, development of numerical methods for simulation of these processes is of significant importance. We will attempt to simulate some of the known processes in this simulation framework. Discrete stochastic algorithms for simulating reactions among populations are more computationally demanding than deterministic algorithms employing ordinary differential equations due to the randomness that must be taken into account
in stochastic models of reactions. However, discrete stochastic algorithms offer more realistic representations of the interactions as they are better suited for modeling fluctuations resulting from stochasticity of processes and the randomness of processes.

Algorithms for simulation of reactions among populations of objects in homogeneous media have been developed earlier. Deterministic algorithms solve ordinary differential equations; while stochastic algorithms address simulation from probabilistic random standpoint. Gillespie's algorithm [35], and the consequent algorithms based on his algorithm, calculate the probability of every reaction using reaction rate constants and population sizes, then based on the probabilities random numbers are generated to determine the length of the step and the reaction. The StochSim [49] algorithm employs a constant time step; however, the time complexity of StochSim is comparable to Gillespie's algorithm. Hybrid approaches were also implemented: fast reactions are simulated with deterministic or Langevin equations, and the slow reactions are simulated with Gillespie [35]. The basic problem with these algorithms is their assumption of homogeneity and inability to represent 3D models and geometry. This problem was resolved later; Stundzia in 1996 [71] offered a finite volume approach to a stochastic simulation algorithm. Cellular reproduction and death have been implemented later. We adopted the ideas from the above authors to develop finite volume based simulation algorithms for modeling reactions among populations, object reproduction, death, chemo-atraction, and random flight movement. The current challenge of the algorithms in this field is to take into account fluid dynamics [83], in our case, plasma dynamics.
Objectives of this proposed activity are summarized as follows:

- Implement a discrete particle dynamics algorithm for simulation of plasma in the vascular space of a blood vessel.

- Integrate the implemented discrete particle dynamics algorithm with a prototyped stochastic population simulation algorithm of a discrete time step.

- Apply the integrated algorithm to simulation of some known reactions and phenomena that causes lesions in the brain during the autoimmune disorders such as multiple sclerosis.

- Identify the scope of complexity to create a model with mixed mesoscopic-probabilistic representation of objects in a space partitioned into finite volumes.

- Use solution of simple models with differential equations and compare the results with discrete particle dynamics algorithm’s results.

Numerous models of blood flow and plasma dynamics use computational fluid dynamics [84], molecular dynamics [85], lattice Boltzmann model [86], etc. While each approach has its advantages and disadvantages, application of a discrete particle dynamics technique seems to be attractive as it fits in the context of a simulation algorithm. The main advantage is that both models use discrete time steps, and both models represent the model at the mesoscopic level. We have developed a prototype of a stochastic algorithm for simulation of interactions between populations of moving objects in 3D space. The prototyped
Figure 1.1: A sample screen showing objects with random walk that collide in an invisible collection of finite volume, undergo proliferation, death and reactions with other population objects to transform and produce other objects.

version is lacking an accurate model of plasma dynamics. However, the prototyped model implements a stochastic algorithm with the discrete time step
interactions between dynamic populations of cells, proliferation and apoptosis
of cells, chemo-attraction of one population of cells by another and the dynam-ics of plasma, where the population are located. To address this multifaceted
problem we started with a basic model with a weak assumption that the ob-
jects move with a random flight in the space. There is an established theory
behind the random walk movement [6], and it can be extended easily to the
finite-volume simulation; therefore, random walk has been selected as a starting
point. The prototyped simulation algorithm [unpublished data] is similar to
the StochSim algorithm. The algorithm is extended to 3D space by the finite
volume technique, which partitions space into tiny volumes such that hetero-
geneous media can be considered as a collection of volumes with homogeneous
concentrations of population of objects. In the prototyped model, the objects
move as a random walk of constant step and constant velocity. Therefore, all ob-
jects of a population move with constant velocity and step size that are assigned
to the population at the model design stage. New objects, which are created
at a collision triggering a reaction, move with a predefined velocity assigned to
the population. In contrast to collision-reaction, population reproduction and
death occur as a programmed process on the population level. At reproduction
and death the energy conservation is preserved by corresponding changes in the
velocity of the objects that are involved in the reproduction and death and the
objects in the finite-volume, that are located at the time of reproduction and
death. In this proposal we would like to make the next step, which is to replace
the random walk motion of the objects with a model that correctly represents
the plasma fluid. The task is to implement movement of the object in this finite-
volume space in such a way that the fluid dynamics in a vessel are accurately represented and simulated together with simulation of other phenomena such as reaction, production, transformation, destruction, proliferation, apoptosis, and chemo-attraction processes. We are considering implementation of a well described discrete particle method, particularly, a dissipative particle dynamics introduced by [87], to simulate plasma dynamics in a vessel.

There are variety of different models that can benefit from the software, such as applications of down-regulatory phenomena, central nervous system endothelial cell-polymorphonuclear cell interactions, T and B-cell interactions in autoimmune syndromes, changing and testing cytokines and chemokines influence in autoimmune diseases, renal tubular epithelial and T-cell interactions in autoimmune renal disease, T-cell vaccination in Experimental Autoimmune Encephalomyelitis. Other applications beyond this one in molecular biology include population dynamics and epidemiological processes, though such applications may require modification of the ideas presented here. This thesis discusses the use of this algorithm only for applications in molecular biology.

Like all software this software has some limitations. First, it handles only three pre-defined compartments in the space: vascular, endothelial and interstitial. This limitation is relatively easy to address: the number of compartments could be 10 or more, depending on the computer power and the nature of the problem. The number of compartments that can be created is a function of the number of cells and the number of types of interactions in each compartment. Another limitation in this program is that it does not take advantage of multi-tasking or parallel processing. To extend the program to multi-tasking
or parallel processing the source code will need to be changed substantially. Adding multi-tasking or parallel processing would add a lot of additional power and speed, and would help in increasing the scale of the problem by calculating interactions in different compartments simultaneously, also, by increasing the number of compartments, number of cells in each compartments and number of interactions.
2. Compare solutions of different methods

We compare solutions of different methods by solving deterministically simple models with differential equations and compare the results with the discrete particle dynamics algorithm. Suppose, there are three types of particles - A, B, C. Transitions in these particles take place according to the following schema $(N = 0, 1, 2, ...)$:

$$\{A \rightarrow \beta_1^1 A + \beta_2^1 B + \beta_3^1 C; B \rightarrow \beta_1^2 B + \beta_2^2 C\}$$

(2.1)

$\beta_1^1, \beta_2^1, \beta_3^1, \beta_2^2, \beta_3^2 \in N$, which means that in the considered process the particle A can transform to group of particles consisting of $\beta_1^1$ particles of type A, $\beta_2^1$ particles of type B, $\beta_3^1$ particles of type C; the particle B can transform to $\beta_2^2$ particles of type B and $\beta_3^2$ particles of type C. The particles of type C are final and cannot transform further. From the mathematical standpoint this model is considered to be Markov process $\xi(t) = (\xi_1(t), \xi_2(t), \xi_3(t))$, with continuous time $t \in [0; \infty)$ and two types of interactions: $\varepsilon^1 = (1, 0, 0)$; $\varepsilon^2 = (0, 1, 0)$; $N^3 = \{(\alpha_1, \alpha_2, \alpha_3), \alpha_1, \alpha_2, \alpha_3 = 0, 1, 2, ...\}$ is a set of vectors with non-negative components, where $\alpha_1 = \text{number of particles of type A}$, $\alpha_2 = \text{number of particles of type B}$ and $\alpha_3 = \text{number of particles of type C}$. The intensity of transformations are $\lambda$ and $\mu$. The process $\xi(t)$ with transition probabilities:

$$P^{(N^3)}_{(N^0^3, N^0_3)} = P\{\xi(t) = (\beta_1, \beta_2, \beta_3) | \xi(0) = (\alpha_1, \alpha_2, \alpha_3)\}$$

is a special class of Markov processes.
If the Markov process is in the initial state \( \alpha = (\alpha_1, \alpha_2, \alpha_3) \), then after a random time \( \tau^1_{(\alpha_1,\alpha_2,\alpha_3)} \), \( \mathbb{P}\left\{\tau^1_{(\alpha_1,\alpha_2,\alpha_3)} < t\right\} = 1 - e^{-\lambda \alpha_1 t} \) the particles of type A transform to a group of particles with probability distributions \( p^1_{(\beta_1,\beta_2,\beta_3)} \), which means that the process is moving to a condition with parameters, which corresponds to the vector: \( (\alpha_1 + \beta_1 - 1; \alpha_2 + \beta_2; \alpha_3 + \beta_3) \). After a random time \( \tau^2_{(\alpha_1,\alpha_2,\alpha_3)} \), \( \mathbb{P}\left\{\tau^2_{(\alpha_1,\alpha_2,\alpha_3)} < t\right\} = 1 - e^{-\lambda \alpha_2 t} \) particles of type B transform to a group of particles with probability distributions: \( p^2_{(\beta_1,\beta_2,\beta_3)} \) and the process transforms to condition: \( (\alpha_1 + \beta_1; \alpha_2 + \beta_2 - 1; \alpha_3 + \beta_3) \), i.e., \( \tau^1 \) and \( \tau^2 \) are exponentially distributed with mean \( \lambda \alpha_1 \) and \( \lambda \alpha_2 \), respectively. In the state \( (\alpha_1, \alpha_2, \alpha_3) \) the process spends a random time:

\[
\tau_{(\alpha_1,\alpha_2,\alpha_3)} = \min\left(\tau^1_{(\alpha_1,\alpha_2,\alpha_3)}, \tau^2_{(\alpha_1,\alpha_2,\alpha_3)}\right)
\] (2.2)

Consider monomolecular type of reaction \( A \rightarrow B \). Let \( X(t) \) represent a Poisson process which denotes a random number of particles in the system at time \( t \), \( t > 0 \). The equation of total balance of the reaction is:

\[
P_x(t + \Delta t) = k (x + 1) \Delta t P_{x+1}(t) + (1 - k x \Delta t) P_x(t) + o(\Delta t)
\] (2.3)

where \( P_x(t) = \mathbb{P}(X(t) = x) \) and \( k \) is a local coefficient of the current Poisson process, i.e., the conditional probability of the event in the interval of \( (t, t+\Delta t) \) = \( k(\Delta t) + o(\Delta t) \), and the probability of more than one event is \( o(\Delta t) \). Thus, \( X(t) \) itself is a random number of particles in the system at time \( t \) is not a Poisson process. Taking the limit \( \Delta t \to 0 \) leads to Kolmogorov’s differential equation:

\[
\frac{\partial P_x}{\partial t} = k (x + 1) P_{x+1}(t) - k x P_x(t)
\] (2.4)
Define the generating function $F(s, t)$ as follows:

$$F(s, t) = \sum_{x=0}^{\infty} P_x(t) s^x \quad (2.5)$$

where $|s| \leq 1$ or the series will diverge. By itself $s$ is not interpretable, except to show when the series will converge. When $s = 1$ the expectation and the variance can be found by differentiating (2.5). Then the equation (2.3) can be presented in the form of a differential equation:

$$\frac{\partial F}{\partial t} = k(1 - s) \frac{\partial F}{\partial s} \quad (2.6)$$

With initial condition $F(s, 0) = s^{x_0}$ equation (2.6) has a solution

$$F(s, t) = \left[1 + (s - 1)e^{-kt}\right]^{x_0} \quad (2.7)$$

Applying expressions for mean and variance:

$$M\{X(t)\} = \left. \frac{\partial F}{\partial s} \right|_{s=1},$$
$$D\{X(t)\} = \left. \frac{\partial^2 F}{\partial s^2} \right|_{s=1} + \left. \frac{\partial F}{\partial s} \right|_{s=1} - \left. \left( \frac{\partial F}{\partial s} \right)^2 \right|_{s=1} \quad (2.8)$$

and hence:

$$M\{X(t)\} = x_0 e^{-kt},$$
$$D\{X(t)\} = x_0 e^{-kt} (1 - e^{-kt}) \quad (2.9)$$

The deterministic solutions of the equations of this type obtained in the work of McQuarrie [89] where the monomolecular process is described by differential equation $dx/dt = -kx$, and solution has a form: $x(t) = x_0 \exp(-kt)$, which fully coincide with the expectations derived in this work.

In computational problems of system biology, it is required to compute the probability of occurrence of certain interactions between populations on a mesoscopic level in a solution given population characteristics such as concentration and volume. Moreover, the modern approach to solving computational problems of system biology increasingly requires implementation of stochastic models describing the occurrence of an interaction based on the probabilistic nature of the involved processes. Combinatorial methods are required to complete the least number of simulations possible. For that reason it is critical to establish a theoretical platform that supports the development of various probabilistic models operating with characteristics of populations and reactions among them.

Since the number of cells in humans and other mammals is very large, we adopt the concept of finite volumes. This concept has already been successfully applied in system biology to study various processes. Finite volumes represent small volumes in which the distribution of populations across the volume can be assumed to be uniform. In assuming a uniform distribution in finite volumes we rely on the well-known work of Gillespie [35], with the same assumptions. In particular, a population of objects is assumed to be randomly distributed in the space of each finite volume, and the number of objects within each volume can be very large.

First, determine the probabilistic picture of the population objects distributed in the volume—that is, the portion of the volume that is taken by the
population objects and its variance. It is easy to see that in such computational problems the issue becomes the placement of $r$ cells in $n$ volumes in such a way that each volume can contain up to $r$ cells. The probability of a cell entering into each finite volume is $n^{-r}$. Therefore, we apply combinatorial methods relevant to the problem, where $r$ is the number of cells, $n$ is the number of volumes, and $m$ is the number of empty volumes after applying $r$ cells into $n$ volumes.

The probability $P_m(r, n)$ of the number of empty volumes is expressed by the recurrent formula

$$P_m(r + 1, n) = \frac{n - m}{n} P_m(r, n) + \frac{m + 1}{n} P_{m+1}(r, n) \quad (3.1)$$

which can be simply proved by the formula of total probability. First, calculate the recurrent formula for the expectation:

$$E(r, n) = \sum_{m=1}^{n} mP_m(r, n)$$

where $E(r, n)$ does not depend on whether $r < n$ or $r > n$. For the expectation we have

$$E(r, n) = \sum_{m=1}^{n} mP_m(r, n) \quad (3.2)$$

Multiplying both parts of $(3.1)$ by $m$ and taking the sum from 1 to $n$ will lead to:

$$\sum_{m=1}^{n} mP_m(r + 1, n) = \sum_{m=1}^{n} \frac{m(n - m)}{n} P_m(r, n) + \sum_{m=1}^{n} \frac{m(m + 1)}{n} P_{m+1}(r, n) \quad (3.3)$$
Let us examine the second part by substituting: \( k = m + 1 \)

\[
\sum_{m=1}^{n} \frac{m(m+1)}{n} P_{m+1}(r,n) = \sum_{k=2}^{n+1} \frac{k(k-1)}{n} P_{k}(r,n)
\]

\[
= \sum_{k=1}^{n} \frac{k(k-1)}{n} P_{k}(r,n)
\]

\[
= \sum_{m=1}^{n} \frac{m(m-1)}{n} P_{m}(r,n)
\]

Substituting the last expression in (3.3) will lead to:

\[
\sum_{m=1}^{n} m P_{m}(r+1,n) =
\]

\[
= \sum_{m=1}^{n} \frac{m(m-m)}{n} P_{m}(r,n) + \sum_{m=1}^{n} \frac{m(m-1)}{n} P_{m}(r,n)
\]

\[
= \sum_{m=1}^{n} \left[ \frac{m(m-m)}{n} - \frac{m(m-1)}{n} \right] P_{m}(r,n)
\]

\[
= \left(1 - \frac{1}{n}\right) \sum_{m=1}^{n} m P_{m}(r,n)
\]

That is,

\[
E(r+1,n) = \left(1 - \frac{1}{n}\right) E(r,n).
\]

Applying the equation \( r - 1 \) times yields

\[
E(r,n) = \left(1 - \frac{1}{n}\right)^{r-1} E(1,n).
\]

Note that

\[
E(1,n) = n - 1
\]

because, with only one cell and \( n \) volumes, the number of empty volumes is naturally equal to \( n - 1 \). Therefore

\[
E(r,n) = (n - 1) \left(1 - \frac{1}{n}\right)^{r-1} = n \left(1 - \frac{1}{n}\right)^{r}
\]

(3.4)
We could derive the same results by letting $M(R, N)$ be the number of empty volumes with $R$ cells and $N$ volumes. Conditional on $R = r$ cells observed in $N = n$ volumes with probability $P_m(r, n)$. Then $E(M(r, n)) = n(1 - 1/r)^r$, so finally $E(M_r,n) = n(1 - 1/n)^r$.

To calculate the variance it is convenient to designate the expected number of empty volumes through $m(r, n)$ (a number of empty volumes after placement of $r$ cells in $n$ volumes); i.e.,

$$E(r, n) = M(m(r, n)) = n \left(1 - \frac{1}{n}\right)^r$$

First, calculate the expectation

$$K(r, n) = M(m(r, n))(m(r, n) - 1)) = \sum_{m=1}^{n} m(m - 1)P_m(r, m)$$

Note that the first element in $m = 1$ is zero, so

$$K(r, n) = M(m(r, n))(m(r, n) - 1)) = \sum_{m=2}^{n} m(m - 1)P_m(r, m)$$

Using (3.1) by $m(m - 1)$:

\[
K(r + 1, n) = \sum_{m=2}^{n} m(m - 1)P_m(r + 1, n) \\
= \sum_{m=2}^{n} m(m - 1)\frac{n-m}{n}P_m(r, n) + \sum_{m=2}^{n} \frac{m(m-1)(m+1)}{n}P_{m+1}(r, n) \\
= \sum_{m=2}^{n} \frac{m(m-1)(n-m)}{n}P_m(r, n) + \sum_{m=2}^{n} \frac{m(m-1)(m-2)}{n}P_m(r, n) \\
= \sum_{m=2}^{n} \left[ \frac{m(m-1)(n-m)}{n} + \frac{m(m-1)(m-2)}{n} \right]P_m(r, n) \\
= (1 - \frac{2}{n}) \sum_{m=2}^{n} m(m - 1)P_m(r, n) \\
= (1 - \frac{2}{n})K(r, n).
\]

Carrying on the iteration, we arrive at:

$$K(r + 1, n) = \left(1 - \frac{2}{n}\right)K(1, n);$$
i.e.,

\[ K(r, n) = \left( 1 - \frac{2}{n} \right)^{r-1} K(1, n) \]

Note that

\[ K(1, n) = M(m^2(1, n)) - M(m(1, n)) = (n - 1)^2 - (n - 1) = (n - 1)(n - 2) = n(n - 1) \left( 1 - \frac{2}{n} \right) \]

Whence

\[ K(r, n) = n(n - 1) \left( 1 - \frac{2}{n} \right)^r \quad (3.5) \]

Finally, we use (3.4) and (3.5) to calculate the variance as:

\[ D(m(r, n)) = M(m^2(r, n)) - (M(m(r, n)))^2 = M(m(r, n)(m(r, n) - 1)) + M(m(r, n)) - (M(m(r, n)))^2 = K(r, n) + E(r, n) - E^2(r, n) = n(n - 1) \left( 1 - \frac{2}{n} \right)^r + n \left( 1 - \frac{1}{n} \right)^r - n^2 \left( 1 - \frac{1}{n} \right)^{2r} \quad (3.6) \]

It should be noted that the equations (3.4) and (3.6) are exact, and that they can determine the exact value of expectations and variances for any \( n \) and \( r \).

However, their calculation for large \( n \) and \( r \) (\( r > 1000 \)) is considerably more difficult. We use the assumption of a constant proportion of cells in volumes; i.e., \( r/n \equiv \text{const} \). In reality, the cell concentration is not perfectly constant, and we here consider \( C = r/n \) to have a probability distribution. We consider
such a problem as future enhancement. For here, assume that the concentration $C = r/n$ is constant.

For large $n$ the expectation is equal to:

$$\tilde{E}(r, n) = n \exp \left( -\frac{r}{n} \right)$$

(3.7)

and the variance is equal to:

$$\tilde{D}(n, r) = n \exp \left( -\frac{r}{n} \right) \left( 1 - \left( 1 + \frac{r}{n} \right) \exp \left( -\frac{r}{n} \right) \right)$$

(3.8)

Analyzing formulas (3.7) and (3.8) we should pay special attention to the fact that the expectation and variance are factors of $n$ and the concentration only.

Now, by substituting concentration $C = r/n$ will get to:

$$\tilde{E}(r, n) = ne^{-C}$$

$$\tilde{D}(r, n) = ne^{-C} [1 - (1 + C)e^{-C}]$$

Note that if there are $pr$ cells in $pn$ volumes, then the entire volume could be $pn$ divided into $p$ smaller volumes, and in each $p$ volume there will be $S_p$ cells, so $\sum_{i=1}^{p} S_i = pr$. As the cells distribute themselves evenly, we can expect ultimately that

$$\frac{S_i}{S_j} \to 1 \quad (i = 1, 2, \ldots p, \ j = 1, 2, \ldots p, \ i \neq j)$$

and, according to limit theorems, $S_i \to r$ when $n \to \infty$. Then, the expectation and variance of $M(r, n) \equiv M(C) =$ number of empty cells in volume $S_i$ shall be $\tilde{E}(r, n)$ and $\tilde{D}(r, n)$, and hence the expectation and variance of the number

16
of empty cells in the total $pn$ volume satisfy the assumptions for CLT to apply. That is, $p\tilde{E}(r, n)$ and $p\tilde{D}(r, n)$. It is not difficult to prove that if $m, n \to \infty$, such that $\tilde{E}(r, n) \to \infty$ and $n - \tilde{E}(r, n) \to \infty$, the limiting distribution of the random variable $\frac{m(r, n) - E(r, n)}{\sqrt{\sigma m(r, n)}}$ is standard normal.

Denoting concentration by $C = r/n$ the above equations gets the following form:

$$E(r, n) = ne^{-c}$$

$$D(r, n) = ne^{-c} \left[ 1 - (1 + c)e^{-c} \right]$$

For the derivation of the generating function we first consider a case in which there are cells of two different types. The analysis is rather more complicated.

Suppose that $n_1$ cells of type 1 and $n_2$ cells of type 2 are placed independently in $m$ volumes. There are, altogether, $n_3 = n_1 + n_2$ cells. A cell of type $j$ falls into the $i$th volume with probability $p_i^{(j)}$ $(j = 1, 2; i = 1, ..., m)$ (multinomial model).

We now introduce new notations. Note that $M_1$, $M_2$, and $M_3$ below are quite distinct from the $M$s in the first paragraph of this section.

Let $M_j$ denote the number of containers not holding cells of type $j$, $(j = 1, 2)$ and $M_3$ denote the number of containers not holding any cells. We write

$$P \left( m \mid \left\{ p_i^{(j)} \right\} \right) = \Pr \left[ \bigcap_{i=1}^{3} (M_j = m_j) \mid \left\{ p_i^{(j)} \right\} \right]$$

Then the joint Probability Generating Function (pgf) of $M_1$, $M_2$, and $M_3$ (given $n_1$ and $n_2$) is

$$\sum_{m_1} \sum_{m_2} \sum_{m_3} \left( \prod_{i=1}^{3} x_i^{m_i} \right) P \left( m \mid \left\{ p_i^{(j)} \right\} \right) = G \left( x_1, x_2, x_3; n_1, n_2 \mid \left\{ p_i^{(j)} \right\} \right)$$

The joint Exponential Probability Generating Function (epgf) of $M_1$, $M_2$, and $M_3$ is
\[
G(z_1, z_2; x_1, x_2, x_3 | \{p_i^{(j)}\})
\]
\[
= \sum_{n_1=1}^{\infty} \sum_{n_2=1}^{\infty} \frac{(z_1 m)^{n_1}}{n_1!} \frac{(z_2 m)^{n_2}}{n_2!} G(x_1, x_2, x_3; n_1, n_2 | \{p_i^{(j)}\})
\]
\[
= \prod_{h=1}^{m} \left[ x_1 x_2 x_3 + x_1 \left\{ \exp \left( z_2 p_h^{(2)} m \right) - 1 \right\} \\
+ x_2 \left\{ \exp \left( z_1 p_h^{(1)} m \right) - 1 \right\} \\
+ \left\{ \exp \left( z_1 p_h^{(1)} m \right) - 1 \right\} \left\{ \exp \left( z_2 p_h^{(2)} m \right) - 1 \right\} \right]
\] (3.9)

where \(z_1\) and \(z_2\) are the arguments of the probability generating function, without specific meaning, but they will help us on defining further values of \(x\) later.

Proof.

First divide the \(m\) volumes into two groups, one group containing a single volume (say, the \(h\)th), and the other group containing the remaining \((m-1)\) volumes. Let \(n'_g\) be the number of cells of type \(g\) in the single volume and \(n''_g = n_g - n'_g\) the number of cells in the remaining \((m-1)\) volumes. Using the equation of total probability:

\[
P(m | \{p_i^{(j)}\}) = \sum_{n'_g + n''_g = n_g \ (g=1,2)} \frac{2}{n'_g n''_g} \prod_{g=1}^{2} \left[ \binom{n_g}{n'_g} \left( p_h^{(g)} \right)^{n'_g} \left( 1 - p_h^{(g)} \right)^{n''_g} \right] \\
\times \sum_{m'_y + m''_y = m_y \ (g=1,2)} \delta \left( m'_1, m'_2, n'_1, n'_2 \right) \frac{1}{m_y!} \left\{ p_i^{(j)} \left( 1 - p_h^{(j)} \right)^{-1} \right\}, \ i \neq h \) (3.10)

where

\[
\delta \left( m'_1, m'_2, n'_1, n'_2 \right) = \begin{cases} 
1 & \text{if } m'_1 n'_1 = m'_2 n'_2 = 0 \\
0 & \text{otherwise}
\end{cases} \) (3.11)
\(m_0\) - total number of empty volumes in first and second groups, i.e. \(m_0 = m'_g + m''_g\) and \(m'_g, m''_g\) are the numbers of volumes in the two groups that do not contain any cells of type \(g\). Note that \(m'_g\) must be 0 or 1, since there is only one volume in the first group. (The indicator function \(\delta(\cdot)\) is introduced because if \(n'_g > 0\) then \(m'_g > 0\), while if \(n'_g = 0\) then \(m'_g = 1\).)

Multiplying both sides of (3.10) by

\[
\frac{(m_{z1})^{n_1} (m_{z2})^{n_2}}{n_1! \; n_2!} x_1^{m_1} x_2^{m_2} x_3^{m_3}
\]

and summing over \(n_1, n_2, m_1, m_2, m_3\), the left-hand side will have the form,

\[
G(z_1, z_2; x_1, x_2, x_3|\{p_i^{(j)}\})
\]

and, on the right-hand side,

\[
\sum_{n_1=1}^{\infty} \sum_{n_2=1}^{\infty} \sum_{m_1=0}^{n_1} \sum_{m_2=0}^{n_2} \sum_{n'_g + n''_g = n_g}^{2} \prod_{g=1}^{n_g} \left[ \frac{(m_{z_g} p_{g}^{(y)})^{n'_g} (m_{z_g} (1-p_{g}^{(y)}))^{n''_g}}{n'_g! n''_g!} \right] \\
\times \sum_{n'_g + n''_g = n_g} \sum_{m_1=0}^{m_1''} \sum_{m_2=0}^{m_2''} \sum_{m_3=0}^{m_3''} \delta(m'_1, m'_2; n'_1, n'_2) x_1^{m_1''} x_2^{m_2''} x_3^{m_3''} \\
\times \sum_{n_1=1}^{\infty} \sum_{n_2=1}^{\infty} \sum_{m_1=0}^{n_1} \sum_{m_2=0}^{n_2} \sum_{n'_g + n''_g = n_g}^{2} \sum_{m_1=0}^{m_1'} \sum_{m_2=0}^{m_2'} \sum_{m_3=0}^{m_3'} \left[ \frac{(m_{z_g} p_{g}^{(y)})^{n'_g} (m_{z_g} (1-p_{g}^{(y)}))^{n''_g}}{n'_g! n''_g!} \right] \\
\times \sum_{m_1=0}^{m_1} \sum_{m_2=0}^{m_2} \sum_{m_3=0}^{m_3} \delta(m'_1, m'_2; n'_1, n'_2) \\
\times \sum_{n_1=1}^{\infty} \sum_{n_2=1}^{\infty} \sum_{m_1=0}^{n_1} \sum_{m_2=0}^{n_2} \sum_{n'_g + n''_g = n_g}^{2} \sum_{m_1=0}^{m_1} \sum_{m_2=0}^{m_2} \sum_{m_3=0}^{m_3} \left[ \frac{(m_{z_g} p_{g}^{(y)})^{n'_g} (m_{z_g} (1-p_{g}^{(y)}))^{n''_g}}{n'_g! n''_g!} \right] \\
\times \sum_{m_1=0}^{m_1} \sum_{m_2=0}^{m_2} \sum_{m_3=0}^{m_3} \delta(m'_1, m'_2; n'_1, n'_2) \\
G(z_1, (1-p_h^{(1)}), z_2, (1-p_h^{(2)}); x_1, x_2, x_3|\{p_i^{(j)} (1-p_i^{(j)})^{-1}\}, i \neq h) \\
XG(z_1 p_h^{(1)}, z_2 p_h^{(2)}; x_1, x_2, x_3|1, 1)
\]

(3.12)
Hence

\[
G \left( z_1, z_2; x_1, x_2, x_3 | p_i^{(j)} \right) \\
= G \left( z_1 \left( 1 - p_h^{(1)} \right), z_2 \left( 1 - p_h^{(2)} \right); x_1, x_2, x_3 | \left\{ p_i^{(j)} \left( 1 - p_h^{(j)} \right)^{-1} \right\}, i \neq h \right) \\
\times G \left( z_1 p_h^{(1)}, z_2 p_h^{(2)}; x_1, x_2, x_3 | 1, 1 \right)
\]

(3.13)

Proceeding inductively, expressing the first \( G (\cdot) \) function on the right-hand side of \( \text{(3.13)} \) as a product of two \( G (\cdot) \) functions, and so on. Eventually, we obtain

\[
\left( z_1, z_2; x_1, x_2, x_3 | \left\{ p_i^{(j)} \right\} \right) = \prod_{h=1}^{m} G \left( z_1 p_h^{(1)}, z_2 p_h^{(2)}; x_1, x_2, x_3 | 1, 1 \right)
\]

(3.14)

Finally, we observe that

\[
P (m_1, m_2, m_3; n_1, n_2 | 1, 1) = \begin{cases} 1 & \text{if } m_1 n_1 = m_2 n_2 = 0 \\ 0 & \text{otherwise} \end{cases}
\]

(3.15)

Hence \( G \left( z_1 p_h^{(1)}, z_2 p_h^{(2)}, x_1, x_2, x_3 | 1, 1 \right) \) is the sum of terms: (i) with \( n_1 > 0, n_2 > 0 : m_1 = 0, m_2 = 0, m_3 = 0 : \)

\[
\left\{ \exp \left( z_1 p_h^{(1)} \right) - 1 \right\} \left\{ \exp \left( z_2 p_h^{(2)} \right) - 1 \right\}
\]

(ii) With \( n_1 > 0, n_2 > 0 : m_1 = 0, m_3 = 0 : x_2 \left\{ \exp \left( z_1 p_h^{(1)} \right) - 1 \right\}
\]

(iii) With \( n_1 = 0, n_2 > 0 : m_1 = 1, m_2 = 0, m_3 = 0 : \)

\[
x_1 \left\{ \exp \left( z_2 p_h^{(1)} \right) - 1 \right\}
\]

(iv) With \( n_1 = n_2 = 0 : m_1 = m_2 = m_3 = 1 : x_1 x_2 x_3 \)
These terms sum to give

\[
G \left( z_1 p_h^{(1)}, z_2 p_h^{(2)}; x_1, x_2, x_3 | 1, 1 \right)
= x_1 x_2 x_3 + x_1 \left\{ \exp \left( z_2 p_h^{(2)} m \right) - 1 \right\} \\
+ x_2 \left\{ \exp \left( z_1 p_h^{(1)} m \right) - 1 \right\} \\
+ \left\{ \exp \left( z_1 p_h^{(1)} m \right) - 1 \right\} \left\{ \exp \left( z_2 p_h^{(2)} m \right) - 1 \right\}
\]

(3.16)

A similar formula has been obtained by Bolotnikov [16]. This formula applies to a situation where the assignment probabilities change at certain points. For the first \( n_1 \) cells, they are \( p_1^{(1)}, \ldots, p_m^{(1)} \); for the next \( n_2 \), they are \( p_1^{(2)}, \ldots, p_m^{(2)} \); and so on, concluding with the last \( n_t \) cells, for which they are \( p_1^{(t)}, \ldots, p_m^{(t)} \).

Bolotnikov [16] considered the random variables \( K_j = \) the number of empty volumes after the \((n_1 + n_2 + \cdots + n_j)\)th cell has been assigned \((j = 1, \ldots, t)\) and evaluated the generating function

\[
H \left( z_1, z_2, \ldots, z_t; x_1, x_2, \ldots, x_t; \left\{ p_i^{(j)} \right\} \right) = \sum_{n_1=0}^{\infty} \cdots \sum_{n_t=0}^{\infty} G_n \left( x; \left\{ p_i^{(j)} \right\} \right) \prod_{j=1}^{t} \frac{(m z_j)^{n_j}}{n_j!}
\]

where

\[
G_n \left( x; \left\{ p_i^{(j)} \right\} \right) = \sum_{k_1 \geq k_2 \geq \cdots \geq k_t} \Pr \left[ \bigcap_{j=1}^{t} (K_j = k_j) \right] \prod_{j=1}^{t} x_j^{k_j}.
\]

It is also possible to solve the same problem with a different approach. In the general term the problem is the following: we have \( t \) particles, which should be allocated to \( N \) volumes of types \( n_1, n_2, \ldots, n_k \) with probabilities \( p_1, p_2, \ldots, p_k \) \((\sum_{j=1}^{k} p_j = 1, \sum_{j=1}^{k} n_k = N)\). In essence this is a polynomial generalization, as discussed above, regarding distribution of empty volumes after \( m \) trials. Further, let us assume that in the \( j \)th type of volume there are \( t_j \) particles (it is obvious
that \( \sum_{j=1}^{k} t_j = t \), and according to the polynomial distribution:

\[
\binom{t}{t_1, t_2, \ldots, t_k} p_1^{t_1} p_2^{t_2} \cdots p_k^{t_k}
\]

The probability density of non-empty volumes is \( n_j \) among volumes of type \( j \) and we denote \( \{ f_{t_j}, n_j \} (u_j) \), where \( f_{t,n} \) can be found by the recurrent equation

\[
f_{t+1,n}(u) = u^{t+1} \left( 1 - \frac{1}{N} \right) \cdots \left( 1 - \frac{1}{N} \right) + \cdots + u^r a_r (t+1) \left( 1 + \frac{1}{N} \right) \cdots \left( \frac{1 - (t+1)}{N^{t+1}} \right)
\]

where

\[
a_{t+1} (t+1) = a_1 (t+1) = 1, \quad a_t (t+1) = r a_r (t) + a(t), \quad r = 2, \ldots, t.
\]

Thus the joint probability density for the number of non-empty volumes among volumes of type \( j \) is

\[
f_{t_1,n_1,n_2,\ldots,n_k}(u_1,u_2,\ldots,u_k) = \sum_{t_1 \ldots t_k} \binom{t}{t_1, t_2, \ldots, t_k} p_1^{t_1} p_2^{t_2} \cdots p_k^{t_k} f_{t_1,n_1}(u_1), \ldots, f_{t_k,n_k}(u_k)
\]

In our case, when there are only two types of interactions, this equation takes a much simpler form:

\[
f_{t_1,n_1,n_2}(u_1,u_2) = \sum_{t_1 t_2} \frac{t!}{t_1! (t-t_1)!} p_1^{t_1} \left( 1 - p_1 \right)^{t-t_1} f_{t_1,n_1}(u_1) f_{t-t_1,n_2}(u_2)
\]

The expectation and variance can be easily found:

\[
E[X] = n_2 \left[ 1 - \left( 1 - \frac{p_1}{n_2} \right)^{t-1} \left( 1 - \frac{1}{n_2} \right) \right]
\]

\[
Var[X] = (n_2 - 1) \left[ (n_2 - 2) \left( 1 - \frac{2p_1}{n_2} \right)^{t-1} \right] +
\]

\[
+ \left( 1 - \frac{p}{n_2} \right)^{t-1} - (n_2 - 1) \left( 1 - \frac{p_1}{n_2} \right)^{2t-2}
\]
4. Direct Simulation Algorithm  
4.1 The Algorithm  

There are several methods for modeling interaction of populations using a priori determined rates for reactions and rates of diffusions of the populations. Methods based on the chemical master equation, the chemical Langevin equation and reaction rate equations are based on the known rates, reactions and propensity functions of the occurring reactions. There is a large scope of papers published in this area, with most of the papers using the approaches pioneered by Gillespie [35]. The obvious disadvantage of an approach that is based on the predefined rates is that it requires knowledge of every interaction, which is almost impossible to achieve when designing large systems of interacting populations. The approach that we offer is based on a population’s dynamics, where probabilities of collisions between populations are calculated and the probabilities of interactions are determined. Moreover, the probabilities of diffusion and chemo-attraction are also calculated based on each population’s dynamics. A tau leap algorithm [35] is then applied to model the entire system. Since we divided the volume into finite elements, the fundamental assumption of the model is that every finite volume is homogeneous, that is all objects of all populations in each finite volume are evenly distributed. The smaller the finite volume, the more precise is the result of the simulation.

A set of finite volumes with side \(d\) is used to describe interactions between populations in a large heterogeneous space. To achieve this, the space is divided into \(K\) finite volumes \(V_k = V_1, \ldots, V_K\). Finite volume has its center in
Figure 4.1: Cross-section in XZ plane of the heterogeneous space represented by the finite volumes. Enlarged finite volume shows the number of populations that it represents.

$(x_n^k, y_n^k, z_n^k)$ and the length of the side $h$. Every finite volume $V_k$ contains $S_{1V_k}, ..., S_{nV_k}$ populations, and every population $S_{iV_k}$ contains $N_{SiV_k}$ number of cells that can be greater or equal to zero, such that the total number of objects in the populations at the start of simulation is expressed as follows:

$$N_{S_1} = N_{S_1V_1} + ... + N_{S_1V_k} + ... + N_{S_1V_{k-1}} + N_{S_1V_K}$$
$$\vdots$$
$$N_{S_i} = N_{S_iV_1} + ... + N_{S_iV_k} + ... + N_{S_iV_{k-1}} + N_{S_iV_K}$$
$$\vdots$$
$$N_{S_{n-1}} = N_{S_{n-1}V_1} + ... + N_{S_{n-1}V_k} + ... + N_{S_{n-1}V_{k-1}} + N_{S_{n-1}V_K}$$
$$N_{S_n} = N_{S_nV_1} + ... + N_{S_nV_k} + ... + N_{S_nV_{k-1}} + N_{S_nV_K}$$

The total number of objects in the volume across all populations is
\[ N_S = \sum_{i=1}^{n} N_{Si} = \sum_{i=1}^{n} \sum_{k=1}^{K} N_{Si}v_k \] (4.2)

The joint probability distribution for the number of interacting population objects of different types in the entire system is based on the state of each population in each finite volume:

\[ P = P(N_{S_1v_1}, ..., N_{S_1v_K}, ..., N_{S_1v_1}, ..., N_{S_1v_K}, ..., N_{S_nv_1}, ..., N_{S_nv_K}) \] (4.3)

The state of the model is described by the probability of having objects of population \( S_i \) in each finite volume. Below we develop foundation for exact simulation of population interactions using finite volumes.

Let's assume that the following events \( A \) can occur in every finite volume: random walk/diffusion, collision/reaction, birth/death, chemo attraction/repulsion.

\[ A = \{ \]
\( R_{11}, ..., R_{1M_R}, D_{11}, ..., D_{1M_D}, \]
\( C_{h11}, ..., C_{h1M_{Ch}}, D_{e11}, ..., D_{e1M_{De}}, B_{11}, ..., B_{1M_B}, \]
\( R_{(K-1)1}, ..., R_{(K-1)M_R}, D_{(K-1)1}, ..., D_{(K-1)M_D}, \]
\( C_{h(K-1)1}, ..., C_{h(k-1)M_{Ch}}, D_{e(K-1)1}, ..., \]
\( D_{e(K-1)M_{De}}, B_{(K-1)1}, ..., B_{(K-1)M_B}, \]
\( R_{K1}, ..., R_{KM_R}, D_{K1}, ..., D_{KMD}, \]
\( C_{hK1}, ..., C_{hKM_{Ch}}, D_{eK1}, ..., D_{eKM_{De}}, B_{K1}, ..., B_{KM_B}, \]
\} = \{ R, D, Ch, De, B \} \]

25
where,

$$R = \{R_{11}, ..., R_{1M_R}, R_{K1}, ..., R_{KM_R}\}$$

is a set of all $M_R$ reactions in the space across all finite volumes. Our main assumption is that the population objects are distributed evenly in a given finite volume, and reaction in a finite volume does not (directly) affect any process in neighboring volumes.

Let $D = \{D_{11}, ..., D_{1M_D}, D_{K1}, ..., D_{KM_D}\}$ is a set of all $M_D$ diffusions in a finite volume. Here the diffusion event means that an object left a finite volume. Since the event of leaving one finite volume is equivalent to the event of entering into another finite volume (see the boundary conditions below, which imply that no leaving event happens without a corresponding entering event), then by modeling the events leaving the finite volume, we have to also model the events entering the volumes. The treatment of this condition is provided in the same section below (the notation comes from Birth and Death processes):

$$De = \{De_{11}, ..., De_{1M_{De}}, De_{K1}, ..., De_{KM_{De}}\}$$

$$B = \{B_{11}, ..., B_{1M_B}, B_{K1}, ..., B_{KM_B}\}$$

is a set of all $M_{De}$ deaths and $M_B$ births processes of objects that occur in all finite volumes of the space. Although the processes of death and birth can be modeled using reactions of the first order, we will group these processes into a separate group in the introduction, and then treat the two processes as reaction based processes.

Let $Ch = \{Ch_{11}, ..., Ch_{1M_{Ch}}, Ch_{K1}, ..., Ch_{KM_{Ch}}\}$ is a set of all $M_{Ch}$ chemotraction events. Similar to diffusion, modeling an event of leaving a finite
volume due to a chemo-atraction event that is due to the presence of the chemo-attractants in the neighboring volumes, is equivalent to the event of entering into a volume with chemo-attractants. The boundary conditions described below support this statement.

Events in the set \( A = \{R, D, Ch, De, B\} \) fully describe all possible events due to processes of random walk, chemo-atraction, interaction, and birth and death in the space of finite volumes. A state of the system will then be defined by sequence events, and the sequence of the events depends on the probabilities of occurrence of events. The laws that govern processes and define probabilities are described in the next section.

As diffusion and chemo-atraction involve interchanging the objects between finite volumes, event \( D \) is represented by two events the model \( D_+ \) and \( D_- \).

\[
D = \{D_{11}, \ldots, D_{1M_B}, D_{K1}, \ldots, D_{KM_B}\} \\
= \{D_{11+}, \ldots, D_{1M_B+}, D_{K1+}, \ldots, D_{KM_B+}, D_{11-}, \ldots, D_{1M_B-}, D_{K1-}, \ldots, D_{KM_B-}\}
\]

There are constraints on the number of objects leaving finite volumes and entering volumes due to diffusion:

\[
\sum_{i=1}^{Q_-} D_{i-} = \sum_{i=1}^{Q_+} D_{i+}
\]

i.e., is the number of objects leaving the finite volumes equals to the number of the objects entering finite volumes through the course of simulation due to diffusion.
In addition there are local restrictions at the boundaries of the finite volumes that lead to constraints. For any given change of state due to an object leaving a finite volume, the adjacent finite volumes must be updated to reflect the migration of the object from one finite volume to another. That is, if only a diffusion process occurs during the entire simulation for a time period of any length then the number of objects in population $S_n$ remains unchanged.

Similar to diffusion, every chemo-attraction event is a process that changes two events in adjacent finite volumes:

$$C_h = \{Ch_{11}, ..., Ch_{1M_D}, Ch_{K1}, ..., Ch_{KM_D}\}$$

$$= \{Ch_{11+}, ..., Ch_{1M_D+}, ..., Ch_{K1+}, ..., Ch_{KM_D+},$$

$$Ch_{11-}, ..., Ch_{1M_D-}, ..., Ch_{K1-}, ..., Ch_{KM_D-}\}$$

### 4.2 Random walk/diffusion.

In a finite volume $V_k$, objects $A_{jV KS_i} = A_{1SIVk}, ..., A_{NnSIVk}$ of population $S_i = S_1, ..., S_n$ move according to the laws of the random walk in three dimensions, and the position of a population object is distributed according to a normal distribution of total variance:

$$\sigma^2_{A_{jVKS_i}} = \frac{t}{\tau} l_{S_i}^2$$

where $t$ is the total time, $l_{S_i}$ is the length of the step for objects of population $S_i$, and $\tau$ is the time between two consecutive steps.

Every population $S_i$ has its own unique set of parameters for the random walk. The random walk leads to diffusion of objects across the entire volume through the finite volumes. The boundary conditions between the volumes are
Dirichlet type [88] for population $S_i$ in any finite volume $V_k$: The boundary conditions of the experimental physical volume are Cauchy type [88], however, we model the conditions as Neumann type [88] for simplicity.

Within each finite volume distribution of population objects is uniform. Therefore, for the inner volumes with the boundary conditions of Dirichlet type, the distribution of the distance in random walk remains Raleigh and the distribution of $x$, $y$ coordinate remains Gaussian, which extends to three dimensions as follows:

$$
P(D_i|x_i, y_i, z_i, x_{vk}, y_{vk}, z_{vk}, \sigma) =
FD(x_i, x_{vk}, h, \sigma)FD(y_i, y_{vk}, h, \sigma)FD(z_i, z_{vk}, h, \sigma)
$$

where $FD(a_i, a_{vk}, h, \sigma) = ERF\left(\frac{a_i-a_{vk} - \frac{h}{2}}{\sigma}, \frac{a_i-a_{vk} + \frac{h}{2}}{\sigma}\right)$

Where $FD$ is the function of $ERF$ - the “error function” encountered in integrating the normal distribution. This expression provides the probability of finding a population object at location $(x_i, y_i, z_i)$ and the probability that a population object leaves the finite volume is therefore expressed as

$$
P(D_i|x_{vk} - \frac{h}{2} > x > x_{vk} + \frac{h}{2}, y_{vk} - \frac{h}{2} > y > y_{vk} + \frac{h}{2})=
\int_{x - \frac{h}{2}}^{x + \frac{h}{2}} \int_{y - \frac{h}{2}}^{y + \frac{h}{2}} \int_{z - \frac{h}{2}}^{z + \frac{h}{2}} FD(x, x_{vk}, h, \sigma)FD(y, y_{vk}, h, \sigma)FD(z, z_{vk}, h, \sigma)dx dy dz
$$

The probability that a population object will be found in any point in space is 1.0:
\[
\int \int \int FD(x, x_{vk}, h, \sigma)FD(y, y_{vk}, h, \sigma)FD(z, z_{vk}, h, \sigma)dxdydz = 1.0
\]

Therefore probability that population object will leave the volume is expressed as follows:

\[
P(D_i) = \int \int \int FD(x, x_{vk}, h, \sigma)FD(y, y_{vk}, h, \sigma)FD(z, z_{vk}, h, \sigma)dxdydz - \int_{x_{vk}-\frac{h}{2}}^{x_{vk}+\frac{h}{2}} \int_{y_{vk}-\frac{h}{2}}^{y_{vk}+\frac{h}{2}} \int_{z_{vk}-\frac{h}{2}}^{z_{vk}+\frac{h}{2}} FD(x, x_{vk}, h, \sigma)FD(y, y_{vk}, h, \sigma)FD(z, z_{vk}, h, \sigma)dxdydz,
\]

and

\[
P(D_i) = 1 - \int_{x_{vk}-\frac{h}{2}}^{x_{vk}+\frac{h}{2}} \int_{y_{vk}-\frac{h}{2}}^{y_{vk}+\frac{h}{2}} \int_{z_{vk}-\frac{h}{2}}^{z_{vk}+\frac{h}{2}} FD(x, x_{vk}, h, \sigma)FD(y, y_{vk}, h, \sigma)FD(z, z_{vk}, h, \sigma)dxdydz
\]

(4.6)
The ERF, a scaled cumulative normal p.d.f., with two arguments supports both upper and lower limits. It can be re-written as

$$ ERF(a_0, a_1) = ERF(a_1) - ERF(a_0) $$

To solve (4.6) we substitute the one argument error function and using integral of the error function as well as the property of the random walk to move independently in three dimensions we obtain the following:

$$ P(D_t) = 1 - \int_{x_{vk} - \frac{h}{2}}^{x_{vk} + \frac{h}{2}} FD(x, x_{vk}, h, \sigma)dx $$

$$ \int_{y_{vk} - \frac{h}{2}}^{y_{vk} + \frac{h}{2}} FD(y, y_{vk}, h, \sigma)dy $$

$$ \int_{z_{vk} - \frac{h}{2}}^{z_{vk} + \frac{h}{2}} FD(z, z_{vk}, h, \sigma)dz $$

$$ \int_{x_{vk} - \frac{h}{2}}^{x_{vk} + \frac{h}{2}} FD(x, x_{vk}, h, \sigma)dx = \int_{x_{vk} - \frac{h}{2}}^{x_{vk} + \frac{h}{2}} ERF\left(\frac{x - x_{vk} + \frac{h}{2}}{\sigma}\right)dx - \int_{x_{vk} - \frac{h}{2}}^{x_{vk} + \frac{h}{2}} ERF\left(\frac{x - x_{vk} - \frac{h}{2}}{\sigma}\right)dx $$

$$ \int_{x_{vk} - \frac{h}{2}}^{x_{vk} + \frac{h}{2}} FD(x, x_{vk}, h, \sigma)dx = \int_{x_{vk} - \frac{h}{2}}^{x_{vk} + \frac{h}{2}} ERF\left(\frac{x - x_{vk} + \frac{h}{2}}{\sigma}\right)dx - \int_{x_{vk} - \frac{h}{2}}^{x_{vk} + \frac{h}{2}} ERF\left(\frac{x - x_{vk} - \frac{h}{2}}{\sigma}\right)dx $$

$$ = \int_{x_{vk} - \frac{h}{2}}^{x_{vk} + \frac{h}{2}} ERF\left(\frac{x - x_{vk} + \frac{h}{2}}{\sigma}\right)\sigma d\frac{x - x_{vk} + \frac{h}{2}}{\sigma} - \int_{x_{vk} - \frac{h}{2}}^{x_{vk} + \frac{h}{2}} ERF\left(\frac{x - x_{vk} - \frac{h}{2}}{\sigma}\right)\sigma d\frac{x - x_{vk} - \frac{h}{2}}{\sigma} = $$

$$ = \sigma \left[ \left(\frac{x - x_{vk} + \frac{h}{2}}{\sigma}\right)ERF\left(\frac{x - x_{vk} + \frac{h}{2}}{\sigma}\right) + \frac{e^{-\left(\frac{x - x_{vk} + h}{\sigma}\right)}}{\sqrt{\pi}} \right]_{x_{vk} - \frac{h}{2}}^{x_{vk} + \frac{h}{2}} - $$

$$ \sigma \left[ \left(\frac{x - x_{vk} - \frac{h}{2}}{\sigma}\right)ERF\left(\frac{x - x_{vk} - \frac{h}{2}}{\sigma}\right) + \frac{e^{-\left(\frac{x - x_{vk} - h}{\sigma}\right)}}{\sqrt{\pi}} \right]_{x_{vk} - \frac{h}{2}}^{x_{vk} + \frac{h}{2}} $$

$$ = 2 \left( h ERF\left(\frac{h}{\sigma}\right) - \frac{e^{-\left(\frac{h}{\sigma}\right)^2}}{\sqrt{\pi}} \right) $$

(4.9)
Figure 4.2: This figure shows a two dimensional section of space. The center finite volume contains a population of objects that experience two processes: diffusion and chemo-attraction. Part A of the figure shows process of diffusion when multiple diffusion events occur, and the objects of the population leave the finite volume (in the center), and enter the finite volumes adjacent to the center finite volume. Part B of the figure shows the process of chemo-attraction, when multiple chemo-attraction events occur during the chemo-atraction process. Part C of the figure shows superimposed processes of diffusion and chemo-attraction. The effects of all processes are additive as the vectors of the movements due to diffusion and chemo-attraction are summed.
Finally, the probability that a population object leaves the finite volume is:

\[ P(D_i) = 8 \left( h_{\text{ERF}} \left( \frac{h}{\sigma} \right) - \frac{e^{-(h/\sigma)^2} - 1}{\sqrt{\pi}} \right)^3 \]  

(4.10)

4.3 Chemo-atraction.

In addition to the diffusion/random walk a population object changes its position due to processes of chemo-attraction or/and chemo-repulsion. This is a deterministic process of movement of a population object in the direction of the highest concentration of the chemo-attractant. To model chemo-atraction we assume that an object of the chemo-attractant population influences the chemo-attracted object with a force proportional to \( r_a \):

\[ f = \frac{e_1 e_2}{\varepsilon r^a} \]

where \( r \) is the distance between objects and an arbitrary \( a > 0 \), \( e_1 \) is the chemoattraction capacity of objects of the first population which chemoattracts the second population, and \( e_2 \) is chemoattraction capacity of objects of the second population towards the objects in the first population.

4.4 Reaction upon collision, birth and death of objects.

For simplicity, let us consider reactions between two populations of objects. We assume that objects of every population in the system move according to the laws of the random walk. This assumption allows not only to model processes of diffusion and chemo-attraction, but also process of reaction between two interacting populations. The reaction between objects from two different populations occurs only if there is a collision between the object. As the movement of the objects is subject to the law of random walk the positions of the
Figure 4.3: This figure shows processes that occur in one subsection. The problem is to figure out the state of the system for all sections in time $t$

objects change constantly. As the positions of the objects change new colliding pairs form. Every colliding pair is then evaluated for a possibility of reaction using the rate provided for the reaction.

Let us now consider modeling a system of interacting, dynamic populations using ordinary differential equations. Every model is based on underlying modeling assumptions, and there are several different mathematical modeling methods that can be used to model a system of reacting populations, involved in the processes of diffusion and chemo-attraction. It is possible to describe the system using a set of linear ordinary differential equations, which describe all possible states of the system. The states are described using probabilities, and it is possible to describe the state of the system using a state vector: $X(t) \equiv$ state of system at limits described by $n$ dimensional vector.
Figure 4.4: This figure shows processes of reaction, birth and death. Part A shows reaction between two populations. The reaction between objects occurs upon collision. In addition, not all collisions lead to reactions. There is a prior probability assigned to each reaction. This is analogous to having a reaction rate in chemical reaction equations. Part B of the figure shows process of birth in the finite volume. The process of birth is modeled as a property of population. Every population has this property, which is implemented as rate of introducing a new object given population size. Part C of the figures shows the process of death in population in the finite volume. Similar to birth rate, every population in the system has death rate.
Then the system of populations $S_{iV_k} = \{S_0V_k, ..., S_nV_k\}$ in a finite volume is described by the states not by the velocities, positions of the population objects. For example, the following state describes the event of diffusing an object outside a finite volume by $n$-dimensional vector:

$$\zeta_{di} = \begin{bmatrix}
0 \\
0 \\
\vdots \\
-1 \\
\vdots \\
0
\end{bmatrix}$$

As a result of diffusing an object out of the finite volume, the state of the finite volume changes:

$$X(t + dt) = X(t) + \zeta_{di}$$

Chemo-atraction is an event which as in the case of diffusion an object leaves the finite volume.

$$\zeta_{cht} = \begin{bmatrix}
0 \\
0 \\
\vdots \\
-1 \\
\vdots \\
0
\end{bmatrix}$$
The corresponding state change is reflected in this equation:

\[ X(t + dt) = X(t) + \zeta_{ch_i} \]

An event of birth of an object in population Si will change the system by adding an additional object in the following way:

\[
\zeta_i = \begin{bmatrix}
0 \\
0 \\
0 \\
\vdots \\
1 \\
\vdots \\
0
\end{bmatrix}
\]

where 1 is position in the i-th row of the matrix.

An event of death an object in population Si will change the system by removing the object in the following way:

\[
\zeta_i = \begin{bmatrix}
0 \\
0 \\
0 \\
\vdots \\
-1 \\
\vdots \\
0
\end{bmatrix}
\]

where -1 is at the position in the ith row of the matrix.

An event of reaction that involves 2 or more populations, will result in matrix where 2 or more population change: For example the following vector
shows a reaction that involves three populations:

\[
\zeta_n = \begin{bmatrix}
0 \\
\vdots \\
M_j \\
\vdots \\
M_i \\
\vdots \\
M_k \\
\vdots \\
0
\end{bmatrix}
\]

where \(M_i, M_j, M_k\), are positive or negative depending on the nature of reaction.

For example, for reaction of the form:

\[3S_k \rightarrow 2S_i + S_j\]

\(M_i = 2\)

\(M_j = 1\)

\(M_k = -3\)

States for chemo-attraction and diffusion of an object into the finite volume are modeled similarly. For example to model the reaction between two colliding objects and simulate it we need to estimate the number of colliding objects between two populations in time \(dt\).
The birth and death processes can be approximated by the first order reactions of following types:

\[ S_i \xrightarrow{c_{bi}} 0 \]
\[ S_i \xrightarrow{c_{di}} 2S_i \]

where \( c_{bi} \) and \( c_{di} \) define the rates of the birth and death processes correspondingly, such that the probability of the birth and death processes (more precisely, we assume that the probability of this reaction taking place in the infinitesimal time interval \([t, t + dt])\) are given by the following expressions:

\[ P(B_i) = c_{bi}X_i(t)dt \]
\[ P(De_i) = c_{di}X_i(t)dt \]

Similarly, for the reactions involving two different populations, the probability of a reaction taking place after collision between the objects of two populations is given by the following expression:

\[ P(R_{ij}) = c_{ri}X_i(t)X_j(t)dt \]

Extending the interactions to diffusion and chemo-attraction processes, we can derive the probability that an object of a population will exit or enter the system is given by the following expression [71]:

\[ P(D_i) = d_{li}X_i(t)dt \]

where \( d_{li} \) is diffusion rate for leaving the finite volume from the neighboring finite volumes. The law of total probability says that the probability that \( X(t) \)
is in a given state $x$ times the probability that event $A$ occurs, for all positive
(disjoint) events $A$.

The resulting total probability

$$P(X(t) = x) = \sum_A P(X(t) = x|A)P(A)$$

where $A$ is the events in the set defined earlier.

A more detailed expression for the total probability accounts for each type of process:

$$P(X(t) = A) = \sum P(X(t) = A|R)P(R) + \sum P(X(t) = A|Ch)P(Ch) +$$

$$\sum P(X(t) = A|D)P(D) + \sum P(X(t) = A|B)P(B) +$$

$$\sum P(X(t) = A|De)P(De)$$

For reactions between two populations the conditional probability that system is in state $X(t)$ given that reaction $R_i$ occurred in time $dt$ is the probability of occurrence of a reaction $R_i$ that leads to the state $X(t)$:

$$P(X(t) = A|R_i) = c_i(X(t) - \zeta_i)dt$$

where, $\zeta_i$ is state vector describing the event of reaction.

Similarly, the probability of an object leaving the finite volume due to diffusion is given by the following:

$$P(X(t) = A|D_i) = d_i(X(t) - \zeta_i)dt$$

The corresponding conditional probability for chemo-attraction is more complex because the chemo-attraction coefficient depends on the concentrations in the
neighboring finite volumes. For simplicity we assume that for any given volume the chemo-attracting objects that are further than adjacent volumes to the given volume does not chemo-attract the objects in the given volume. The objects in the finite volume experience chemo-attracting force from the neighboring volumes only.

\[ P(X(t) = A | Ch_i) = ch_i(X(t) - \zeta_{chi}) dt \]

where \( ch_i \) is a chemo-attracting coefficient defined as a function of the concentrations in the neighboring volumes:

\[ \mu_{k,adj} = \{ \frac{N_{SVij(k-1)}}{V_k}, \frac{N_{SVij(k+1)}}{V_k}, \ldots \} \]

are vector of concentrations in the adjacent finite volumes. The function that reflects the force of chemo-attraction is assumed to have quadratic relationship between the distance and the force. Such assumptions support the fact that the closer the object to the chemo-attracting objects the faster it moves toward. The movement direction is assumed to be of gradient type, that is the movement occurs in the direction of the maximum force. The direction of the maximum force is determined as the direction of the resultant vector that forms by summing all the vectors forms the neighboring finite volumes.

The birth and death processes are described using the state matrices for the birth and death processes as follows:

\[ P(X(t) = A | B_i) = b_i(X(t) - \zeta_{bi}) dt \]

\[ P(X(t) = A | D_e_i) = d_e_i(X(t) - \zeta_{dei}) dt \]

Since the processes of birth and death can be approximated by the first order reactions, the birth and death are assumed to be modeled in the same fashion.
as we model reactions between populations.

The total probability of finding the system of populations in state $X(t)$ in time $dt$, is given by the following probability:

$$P(X(t + dt) = A) = (P(X(t + dt) = A|A)P(A)) + (P(X(t + dt) = A|A^c)P(A^c))$$

Probability that the state $P(X(t + dt) = A)$ will occur due to no event is the probability that no interaction occurred in time $t + dt$ times the probability of being in state $X(t)$ at time $t$ (noevent), that is $P(X(t))$:

$$P(X(t + dt) = A|\text{noevent}) = 
(1 - \sum_{i=1}^{MR} r_i X(t) dt - \sum_{i=1}^{MC} c_{hi} X(t) dt - \sum_{i=1}^{MD} d_i X(t) dt - \sum_{i=1}^{MD_e} d_{ei} X(t) dt - \sum_{i=1}^{MR} b_i X(t) dt) P(X(t))$$

Probability that the state $P(X(t + dt) = A)$ will occur due to an event is defined as follows:

$$P(X(t + dt) = A|A)P(A) = 
\sum_{i=0}^{MR} r_i (X(t) - \zeta_{ri}) dt P(X(t) - \zeta_{ri}) + 
\sum_{i=0}^{MD} d_i (X(t) - \zeta_{di}) dt P(X(t) - \zeta_{di}) + 
\sum_{i=0}^{MC} c_{hi} (X(t) - \zeta_{chi}) dt P(X(t) - \zeta_{chi}) + 
\sum_{i=0}^{MD_e} d_{ei} (X(t) - \zeta_{dei}) dt P(X(t) - \zeta_{dei}) + 
\sum_{i=0}^{MR} b_i (X(t) - \zeta_{bi}) dt P(X(t) - \zeta_{bi})$$
Therefore, the total probability of finding the system of populations in state \( X(t) \) in time \( dt \), in all finite volumes \( V \) containing populations \( S_{iV_k} = S_0V_k, ..., S_nV_k \) is given by the following probability:

\[
P(X(t + dt) = A) = \left(1 - \sum_{i=1}^{M_R} r_i X(t) dt - \sum_{i=1}^{M_{Ch}} c_{hi} X(t) dt - \sum_{i=1}^{M_D} d_{i}(X(t) dt - \sum_{i=1}^{M_{de}} d_{ei} X(t) dt - \sum_{i=1}^{M_R} b_{i} X(t) dt P(X(t)) \right)
\]

\[
= \sum_{i=0}^{M_R} r_i (X(t) - \zeta_{ri}) dt P(X(t) - \zeta_{ri}) + \sum_{i=0}^{M_D} d_i (X(t) - \zeta_{di}) dt P(X(t) - \zeta_{di}) + \sum_{i=0}^{M_{Ch}} c_{hi} (X(t) - \zeta_{chi}) dt P(X(t) - \zeta_{chi}) + \sum_{i=0}^{M_{de}} d_{ei} (X(t) - \zeta_{dei}) dt P(X(t) - \zeta_{dei}) + \sum_{i=0}^{M_R} b_i (X(t) - \zeta_{bi}) dt P(X(t) - \zeta_{bi})
\]

Rearranging we have

\[
\lim_{dt \to 0} \left[ \frac{P(X(t + dt) = A) - P(X(t) = A)}{dt} \right] = \frac{dP(X(t))}{dt} = \sum_{i=0}^{M_R} r_i (X(t) - \zeta_{ri}) P(X(t) - \zeta_{ri}) - \sum_{i=1}^{M_R} R_i X(t) P(X(t)) + \sum_{i=0}^{M_D} d_i (X(t) - \zeta_{di}) P(X(t) - \zeta_{di}) - \sum_{i=1}^{M_D} D_i X(t) P(X(t)) + \sum_{i=0}^{M_{Ch}} c_{hi} (X(t) - \zeta_{chi}) P(X(t) - \zeta_{chi}) - \sum_{i=1}^{M_{Ch}} C_{hi} X(t) P(X(t)) + \sum_{i=0}^{M_{de}} d_{ei} (X(t) - \zeta_{dei}) P(X(t) - \zeta_{dei}) - \sum_{i=1}^{M_{de}} D_{ei} X(t) P(X(t)) + \sum_{i=0}^{M_R} b_i (X(t) - \zeta_{bi}) dt P(X(t) - \zeta_{bi}) - \sum_{i=1}^{M_R} B_i X(t) P(X(t))
\]

In the probability equation above we assume that \( P(X(t + dt) = A) - P(X(t) = A) = dP \) at \( dt \), which allows us to define a derivative for the probability. This
equation is equivalent the Chemical Master Equation, and each state is represented as ordinary differential equation. The high dimensionality of this equation makes it computationally intractable to solve for systems with large number of interacting populations. A system of ordinary differential equations (ODEs) with one ODE for each possible state of the system leads to a NP-hard computational problem and therefore is not practical.

To avoid this we choose to simulate the system by tracing probabilistic paths through the states thus avoiding expensive computations of all possible states in the system. This approach is a widely accepted alternative to simulation and is employed in many simulation systems. The differences between our system and previously published systems are:

- We provide rigorous mathematical treatment of the simulation algorithms to provide exact methods for the simulation.

- Based on the mathematical methods developed here we can achieve fast and practical simulation of the population objects in the space.

Using the results achieved in the previous section, we will derive a foundation for the stochastic simulation algorithm, which follows a probabilistic path across the sequence of the state; see Figure 4.5.

Only for demonstration purposes, we introduce the end result of the simulation as a state when the number of objects in a population comes to zero. This may not be the case in many simulations, however, for illustration purposes, let us consider a system which has a population modeled in such a way that one of the populations terminates. This way we define more than one criterion for measuring the ability of the system to reproduce the results. For example, al-
Figure 4.5: This figure shows an example path out of many paths possible that the stochastic simulation algorithm will simulate. Due to the nature of the algorithm, every simulation will result in a new path, thus every simulation most likely be a unique simulation of the system of interacting populations. However, due to the large number of objects in the populations, the end result of the simulation is almost always identical (see details in text).
Figure 4.6: This figure shows the end results of an example simulation system provided in the Appendix C. As seen, the system arrives to 4 different end results, which can be possible course of treatment of a particular disease.

though the stochastic simulation algorithm uses different simulation paths, the end result with high probability will be the same, or there can be several end results that the simulation system will arrive at various times. This intuitively supports flow of the processes in living organisms (for the course of a particular disease) or in industrial production (for example, the life time of cars).

We introduce the quantity $X(t) = \text{number of events in } [0, t]$ and

$$P(\text{noevent}, \tau) = P\{X(t + \tau) - X(t) = 0\}$$

as the probability that no reaction takes place in the time interval $[t, t + \tau)$.

Probability of no event for the following time period:

$$P\{X(\tau + \delta\tau) = 0\} =$$

$$1 - \sum_{i=1}^{M_b} r_i X(t) dt - \sum_{i=1}^{M_{CH}} c_i X(t) dt - \sum_{i=1}^{M_D} d_i X(t) dt - \sum_{i=1}^{M_{DS}} d_{i,} X(t) dt -$$

$$\sum_{i=1}^{M_B} b_i X(t) dt)P\{X(\tau) = 0\}$$
\[
P \{ X(\tau + \delta \tau) = 0 \} - P \{ X(\tau) = 0 \} = \\
- \left( \sum_{i=1}^{M_r} r_i X(t) + \sum_{i=1}^{M_{Ch}} c_{i1} X(t) + \sum_{i=1}^{M_D} d_{i1} X(t) + \sum_{i=1}^{M_{De}} d_{i2} X(t) + \sum_{i=1}^{M_B} b_i X(t) \right) P \{ X(\tau) = 0 \}
\] (4.12)

The right hand side of the equation is assigned to \( a_{sum}(X(t)) \), and the left hand side of the equation is the derivative at the limit of the delta time:

\[
\frac{dP \{ X(\tau) = 0 \}}{d\tau} = -a_{sum}(X(t))
\]

and

\[
P \{ X(\tau) = 0 \} = e^{-a_{sum}(X(t))\tau}
\]

The probability that the next event in time interval \([\tau, \tau + d\tau]\) will be event \( A_i \) from any of the following types \( \{R, D, Ch, De, B\} \) given the system is in the state \( X(t) \):

\[
P(\tau, A_i | X(t), t) d\tau = P \{ X(\tau) = 0 \} a_i(X(t)) d\tau
\]

Here \( P \{ X(\tau) = 0 \} \) is the probability that no reaction occurred in time \([\tau, \tau + d\tau]\), and \( a_i(X(t)) d\tau \) is the probability that event \( A_i \) from \( \{R, D, Ch, De, B\} \) occurred in the \([t + \tau; t + \tau + d\tau]\).

Using previously achieved result on probability of having no event, we have

\[
P(\tau, A_i | X(t), t) = a_i(X(t)) e^{-a_{sum}(X(t))\tau}
\]

To make the separation between time and the event more clear and convenient to handle during simulation we have:

\[
P(\tau, A_i | X(t), t) = \frac{a_i(X(t))}{a_{sum}(X(t))} a_{sum}(X(t)) e^{-a_{sum}(X(t))\tau}
\]
Term $\frac{a(X(t))}{a_{\text{sum}}(X(t))}$ can be used to select the next event in the system, and
$a_{\text{sum}}(X(t))e^{-a_{\text{sum}}(X(t))}\tau$ can be used to derive the time when the reaction will occur.

As before, executing a simulation step the rates for reactions, diffusion, chemo-attraction, birth and death are updated based on the positions of the population object. Such simulation results in a very slow process due to the large number of events, and simulating a system with many interactions is extremely slow. Therefore, this approach is not practical.

The algorithm works as follows: 1. Evaluate vector of rates for all possible events in the system of interacting population objects across all finite volumes in the space. Rates update for diffusion, chemo-attraction, reactions, and birth and death processes for all $K$ volumes, and the sum of all rates as follows:

$$\{r_{k,0}, r_{k,1}, \ldots, r_{k,Mr}\}, \{d_{k,0}, d_{k,1}, \ldots, d_{k,Md}\}, \{c_{h_k,0}, c_{h_k,1}, \ldots, c_{h_k,Mch}\},$$

$$\{de_{k,0}, de_{k,1}, \ldots, de_{k,Mde}\}, \{b_{k,0}, b_{k,1}, \ldots, b_{k,Mb}\}$$

and the sum of rates across all $K$ volumes:

$$a_{\text{sum}} = \sum_{k=0}^{K} r_{k,\text{sum}} + \sum_{k=0}^{K} d_{k,\text{sum}} + \sum_{k=0}^{K} c_{d_{k,\text{sum}}} + \sum_{k=0}^{K} de_{k,\text{sum}} + \sum_{k=0}^{K} b_{k,\text{sum}}$$

2. Using a uniform random number generator create two random variables $r_1$ and $r_2$ for selecting the event and the time at which the event occurs. For selecting an event index we use condition:

$$\sum_{i=0}^{j} a_i(X(t)) < r_1()a_{\text{sum}}$$

For selecting time we set $\tau$ to be equal to:

$$\tau = \ln\left(\frac{1}{r_2()}\right)/a_{\text{sum}}(X(t))$$
3. Using the selected time and event, advance the system by defining the new state of the system as follows:

\[ X(t + \tau) = X(t) + \zeta_{ai} \]

As seen from the algorithm, the simulation of the system happens one event at a time.

4.5 Tau Leaping Algorithm

The tau leaping method avoids the problem by allowing larger steps at a cost of accuracy of simulation. However, the benefits of being able to actually run the simulation all the way through from the initial state to the end result seems to overweigh the disadvantage of the system. In addition, it is always possible to minimize the error by choosing \( \tau \) to be small.

Instead of simulation every event the tau leaping method advances the system by the given time \( \tau \), and updates the quantities of the populations a number of events that would have happened during the interval. The assumption is that the rates of the events are constant during the time interval \( \tau \).

\[ X(t + \tau) = X(t) + \sum_{i=1}^{M} \zeta_{i} Y(a_{i}(X(t)), \tau) \]

where, \( Y(a_{i}(X(t)), \tau) \) is the number of events, which occur during the tau leap step. The number of events \( Y \) depends on the length of the step (\( \tau \)) and the rate of the reaction \( a \). The rate of the reaction \( a \) depends on the population state \( X(t) \). Obviously, if the rate changes very slightly, (which can happen only when a few objects are added or removed in a population with large number of objects), then using the method does not introduce large errors. Therefore, the larger the number of objects, the larger the time period that can be adopted.
To evaluate how many events \( Y(a_i(X(t)), \tau) \) of type \( A_i \) happened over the time period we do the following. Since we assume that the rate is constant then events happen at regular intervals and therefore the number of the events that happened during the time interval follow a Poisson distribution. Therefore, the number \( Y(a_i(X(t)), \tau) \) is estimated by drawing a sample from a Poisson distribution with parameter \( a_i(X(t))d\tau \), which is the probability the event \( A_i \) will occur in time \( d\tau \).

The algorithm works as follows:

1. Generate vectors of the sample numbers using Poisson distributions with the following parameters:

\[
\{r_{k,0}(X(t))\tau, r_{k,1}(X(t))\tau, \ldots, r_{k,Mr}(X(t))\tau\},
\]

\[
\{d_{k,0}(X(t))\tau, d_{k,1}(X(t))\tau, \ldots, d_{k,Md}(X(t))\tau\},
\]

\[
\{c_{k,0}(X(t))\tau, c_{k,1}(X(t))\tau, \ldots, c_{k,Mch}(X(t))\tau\},
\]

\[
\{d_{k,0}(X(t))\tau, d_{k,1}(X(t))\tau, \ldots, d_{k,Mde}(X(t))\tau\},
\]

\[
\{b_{k,0}(X(t))\tau, b_{k,1}(X(t))\tau, \ldots, b_{k,Mb}(X(t))\tau\}
\]

for each event in the system of interacting population objects across all finite volumes in the space.

\[
\{Y_{r_{k,0}}, Y_{r_{k,1}}, \ldots, Y_{r_{k,Mr}}\},
\]

\[
\{Y_{d_{k,0}}, Y_{d_{k,1}}, \ldots, Y_{d_{k,Md}}\},
\]

\[
\{Y_{c_{k,0}}, Y_{c_{k,1}}, \ldots, Y_{c_{k,Mch}}\},
\]

\[
\{Y_{d_{k,0}}, Y_{d_{k,1}}, \ldots, Y_{d_{k,Mde}}\},
\]

\[
\{Y_{b_{k,0}}, Y_{b_{k,1}}, \ldots, Y_{b_{k,Mb}}\}
\]
\[ \{ Y_{b_k,0}, Y_{b_k,1}, \ldots, Y_{b_k,MB} \} \]

2. Using the computed sample numbers, advance the system by defining the new state of the system as follows:

\[ X(t + \tau) = X(t) + \sum_{i=1}^{M} \zeta_{ai} Y_i \]

Switching from Poisson \( \text{Poisson}(a_i(X(t)), \tau) \) to normal distribution \( a_i(X(t))\tau + \sqrt{a_i(X(t)), \tau} Z_i \) can be done in case the number of events occurred over the time interval is large, based on limiting distribution of Poisson for larger mean \( \text{Poisson}(\mu) \approx N(\mu, \mu) \).

\[ X(t + \tau) = X(t) + \sum_{i=1}^{M} \zeta_i \left[ a_i(X(t))\tau + \sqrt{a_i(X(t)), \tau} Z_i \right] \]

In this case we arrived at Chemical Langevin Equation, the Euler-Maruyama discretization of the continuous time problem [74]:

\[ dX(t) = \sum_{i=1}^{M} \zeta_i \left[ a_i(X(t))d\tau + \sqrt{a_i(X(t))}dW_i(t) \right] \]

where \( W_i(t) \) are Brownian motions. The simulation algorithm using Chemical Langevin Equation is identical to the algorithm with tau leaping:

1. Generate vectors of the sample numbers using Normal distributions for each event in the system of interacting population objects across all finite vol-
umes in the space.

\[ N_{S_1V_1} + \ldots + N_{S_1V_k} + \ldots + N_{S_1V_{k-1}} + N_{S_1V_K} \]
\[ \vdots \]
\[ N_{S_iV_i} + \ldots + N_{S_iV_k} + \ldots + N_{S_iV_{k-1}} + N_{S_iV_K} \]
\[ \vdots \]
\[ N_{S_{n-1}V_{n-1}} + \ldots + N_{S_{n-1}V_k} + \ldots + N_{S_{n-1}V_{k-1}} + N_{S_{n-1}V_K} \]
\[ N_{S_nV_1} + \ldots + N_{S_nV_k} + \ldots + N_{S_nV_{k-1}} + N_{S_nV_K} \] (4.13)

2. Using the computed sample numbers, advance the system by defining the new state of the system as follows:

\[ X(t + \tau) = X(t) + \sum_{i=1}^{M} \xi_i G_i \]

In both cases, the number of objects in the system should be large to produce accurate results.

Equation \( X(t + \tau) = X(t) + \sum_{i=1}^{M} \xi_i \left[ a_i(X(t)) \tau + \sqrt{a_i(X(t))} \tau Z_i \right] \) without the stochastic part is a set of ordinary differential equations, which are widely used for modeling reaction rate systems:

\[ \frac{X(t + \tau) - X(t)}{\tau} = \frac{dX(t)}{d\tau} = \sum_{i=1}^{M} \xi_i a_i(X(t)) \]

when \( \tau \to 0 \).

Simulating systems using the reaction rates ODEs achieve computation of the thermodynamic states of the system. All methods - stochastic simulations on event at a time, tau leaping, Langevin equations and reaction rate ODEs are closely related as shown above.
In our simulation system Vixdum we deploy Langevin equations, and the rates of interactions are updated after every simulation time step.

4.6 Model of Infiltration Demyelination

This model is an attempt to model T-cell penetration from blood vessels to brain tissue [75] thus destroying myelin layer in axons. Activated T-Cells from the beginning are present only in blood vessels. Activated T-cells do not have enough kinetic energy to pass through the wall of a blood vessel and pass into the brain tissue. The wall of the blood vessel consists of very dense endothelial cells. Activated T-cells produce proteins called cytokines. Cytokines are contacting with blood vessel wall. During the contact cytokines are touching the endothelial cells and endothelial cells are getting activated. Activated endothelial cells begin to produce a special, very small size, protein. The special feature of these proteins is that they construct a gradient field, which influences those cells that are sensitive to these proteins. The proteins of this group are called chemokines. Macrogliia are cells that are found in brain tissue and are chemo-attractive to chemokines produced by endothelial cells. Because of the diffusion and chemo-attraction processes, chemokines travel to the brain tissue, where they contact and activate macroglia. Activated macroglia cells produce chemokines which can attract activated T-cells. These chemokines construct a gradient field which not only covers brain tissue, but also walls of vessels. Activated T-cells moving in the gradient field and having more kinetic energy as a result of chemo-attraction, penetrate through the walls of blood vessels to the brain tissue and destroy myelin.

4.7 Identification.
Figure 4.7: Different stages of infiltration and demialination in the model.
The program for the simulation was tested and verified. Nevertheless, with the aim of further research, we traced the simulation interaction of particles for those examples where an explicit solution is possible through elementary functions. Consider the first transformation of the type \( A \rightarrow B \) the simplest scheme with two types of particles and complex interaction. Omitting solutions in the Kolmogorov’s equations, we present the exact solutions to mathematical expectations:

\[
E_A(t) = A_0 e^{-\lambda t} \quad E_B(t) = A_0 (1 - e^{-\lambda t})
\]

where \( \lambda \) is the speed of transition and \( A_0 \) is the initial value of the process.

A somewhat complex example is a transformation of a type \( A \rightarrow B \rightarrow C \) with rates of transformation - \( \lambda, \mu \). We present solutions for expectation.

\[
E_A(t) = A_0 e^{-\lambda t},
\]

\[
E_B(t) = A_0 \frac{\lambda}{\lambda - \mu} \left( e^{-\mu t} - e^{-\lambda t} \right),
\]

\[
E_C(t) = A_0 \left( 1 + \frac{\mu}{\lambda - \mu} e^{-\lambda t} - \frac{\lambda}{\lambda - \mu} e^{-\mu t} \right)
\]

where \( E_A + E_B + E_C = A_0, \quad \lambda \neq \mu \). The figure 4.8 below presents estimated (theoretical) and experimental 4.9 (obtained through the simulation) data, when \( A_0 = 10000, \lambda = 2, \mu = 1.5 \), and it shows an excellent agreement for a very typical stochastic realization of this type of transformation. The theoretical values of variance (figure 4.8) are calculated by equations

\[
D_A(t) = A_0 e^{-\lambda t} (1 - e^{-\lambda t})
\]

\[
D_B(t) = A_0 \frac{\lambda}{\lambda - \mu} \left( e^{-\mu t} - e^{-\lambda t} \right) \left( 1 - \frac{\lambda}{\lambda - \mu} \left( e^{-\mu t} - e^{-\lambda t} \right) \right)
\]

\[
D_C(t) = A_0 \left( \frac{\mu}{\lambda - \mu} e^{-\lambda t} - \frac{\lambda}{\lambda - \mu} e^{-\mu t} + 1 \right) \left( \frac{\lambda}{\lambda - \mu} e^{-\mu t} - \frac{\mu}{\lambda - \mu} e^{-\lambda t} \right)
\]

The figures 4.8 and 4.9 show excellent agreement for a very typical stochastic realization of this type of transformation.
Figure 4.8: Experimental curve of the average result of 20 simulations: the variance of the number of particles of all types and the simulations. Series 1, 2, 3 are cells of type A, B, C respectively.
Figure 4.9: Theoretical curve of the variance of the number of particles of all types and the simulations. Series 1, 2, 3 are cells of type A, B, C respectively.
5. Elements of theory and research tasks

5.1 Markov branching processes with a finite number of types of particles

Let us denote $N = \{0, 1, 2, \ldots\}$ as the set of all non-negative integers and $N^n$ as the set of all vectors with non-negative integer components. Further, for $\alpha, \beta, \gamma \in N^n$ let us denote $\gamma = \alpha - \beta$ if $\gamma_1 = \alpha_1 \beta_1, \ldots, \gamma_n = \alpha_n - \beta_n$; $\alpha \geq \beta$, if $\alpha_1 \geq \beta_1, \ldots, \alpha_n \geq \beta_n$; $|\alpha| = \alpha_1 + \cdots + \alpha_n$; the sum $\sum_{\alpha \in N^n}$ is denoted as $\sum_{\alpha}$.

5.2 Markov process on the $N^n$. Kolmogorov equations

The state of the Markov process is a vector $\xi(t) = (\xi_1(t), \ldots, \xi_n(t))$, $t \in [0, \infty)$, $k$th component of which $\xi_k(t)$ shows that at the time $t$ there are $\xi_k(t)$ particles of type $T_k$. $\xi(t)$ is a Markov process if its transition probabilities

$$P_{\alpha\beta}(t) = P\{\xi(t_1 + t) = \beta | \xi(t_1) = \alpha\}$$

satisfy the conditions

$$P_{\alpha\beta}(t) \geq 0$$

for all $\alpha, \beta \in N^n$, $t \in [0, \infty)$ (non-negativity condition);

$$\sum_{\beta} P_{\alpha\beta}(t) = 1$$

in any $\alpha \in N^n$, $t \in [0, \infty)$ (normalization condition)

$$P_{\alpha\beta}(t + \tau) = \sum_{\gamma} P_{\alpha\gamma}(t) P_{\gamma\beta}(\tau)$$

for all $\alpha, \beta, \gamma \in N^n$, $t, \tau \in [0, \infty)$ (Markovian property).
The primary condition is

\[ \lim_{t \to \infty} P_{\alpha\beta}(t) = P_{\alpha\beta}(0) = \delta_{\alpha\beta} = \begin{cases} 1, & \text{if } \alpha = \beta \\ 0, & \text{if } \alpha \neq \beta \end{cases} \]

It is anticipated that the process \( \xi(t) \) is stochastically continuous. That is, the condition

\[ \lim_{t \to 0^+} P_{\alpha\alpha}(t) = 1, \quad \lim_{t \to 0^+} P_{\alpha\beta}(t) = 0, \quad (\alpha \neq \beta), \]

is satisfied.

There are always finite or infinite limits

\[ a_{\alpha\alpha} = \lim_{t \to 0^+} \frac{P_{\alpha\alpha}(t) - 1}{t}, \quad a_{\alpha\beta} = \lim_{t \to 0^+} \frac{P_{\alpha\beta}(t)}{t}, \quad (\alpha \neq \beta), \alpha, \beta \in \mathbb{N}^n \]

These limits are called infinitesimal characteristics and can be written as

\[ a_{\alpha\alpha} = \left. \frac{dP_{\alpha\beta}(t)}{dt} \right|_{t=0^+}, \quad \alpha, \beta \in \mathbb{N}^n \]

where \( a_{\alpha\alpha} \) can be interpreted as density of transition probability.

If \( \alpha \neq \beta \), then \( a_{\alpha\beta} \) is finite; \( a_{\alpha\alpha} \) is either finite or \( a_{\alpha\alpha} = -\infty \). In all cases, \( \sum_{\beta \neq \alpha} a_{\alpha\beta} \leq -a_{\alpha\alpha} \). Using \( a_{\alpha\beta} \) classify the states of the process as: the state \( \alpha \) is called instantaneous if \( a_{\alpha\alpha} = -\infty \). If the state is not instantaneous it is called regular, when \( \sum_{\beta \neq \alpha} a_{\alpha\beta} = -a_{\alpha\alpha} \).

The probabilistic meaning of characteristics \( \{a_{\alpha\beta}, \alpha, \beta \in \mathbb{N}^n\} \) is explained as follows. The Markov process \( \xi(t) \) starts in state \( \alpha \) at random time \( \tau_\alpha \) which has an exponential distribution with mean \( 1/a_{\alpha\alpha} \)

\[ P\{\tau_\alpha \leq t\} = 1 - e^{a_{\alpha\alpha}t} \]
If \( a_{\alpha \alpha} = 0 \) then the process is absorbing. If \( a_{\alpha \alpha} < 0 \), then at time \( \tau_\alpha \) with distribution

\[
\{ p_{\alpha \beta} = -a_{\alpha \beta}/a_{\alpha \alpha}, \quad \beta \neq \alpha; \quad p_{\alpha \alpha} = 0 \}
\]

the process transitions to the state \( \beta \); in the state \( \beta \) the process spends time \( \tau_\beta \); further similar evolution process is observed.

Let all of the states of the process be recurrent, and, given \( \beta \), the condition

\[
\sum_\gamma a_{\alpha \gamma} P_{\gamma \beta}(t) > -\infty
\]

is satisfied, and given \( \alpha \), the condition

\[
\sum_\gamma P_{\alpha \beta}(t) a_{\gamma \beta} > -\infty
\]

also satisfied. Then, given \( \alpha, \beta \) the first (backward) and second (forward) system of Kolmogorov’s differential equations for transition probabilities \( P_{\alpha \beta}(t) \) is satisfied.

\[
\frac{dP_{\alpha \beta}(t)}{dt} = \sum_\gamma a_{\alpha \gamma} P_{\gamma \beta}(t), \quad \alpha \in \mathbb{N}^n \quad (5.1)
\]

\[
\frac{dP_{\alpha \beta}(t)}{dt} = \sum_\gamma P_{\alpha \gamma}(t) a_{\gamma \beta}, \quad \beta \in \mathbb{N}^n \quad (5.2)
\]

with initial condition \( P_{\alpha \alpha}(0) = 1, \ P_{\alpha \beta}(0) = 0 \), where \( \alpha \neq \beta \). It is not difficult to prove the conditions of existence and uniqueness of the solution of the system of equations (5.1) and (5.2).

5.3 Branching process with \( n \) types of particles

Assume there are \( n \) types of particles

\[ A_1, \ldots, A_n. \]

If there is a set of particles, consisting of particles \( \alpha_1 \) of type \( A_1 \), \( \alpha_2 \) particles of type \( A_2, \ldots \), \( \alpha_n \) particles of type \( A_n \); assume that the random process is in state \( \alpha = (\alpha = (\alpha_1, \ldots, \alpha_n)). \)
In the set of Markov processes we separate a special class: branching processes, which are determined by the probabilities of transition, $P_\gamma(t)$, which are equal to probabilities of one particle of type $A_i$ in time $t$ to transition into the totality of the particles $S_\gamma = \gamma_1 s_1 + \ldots + \gamma_n s_n$, corresponding to the vector $\gamma = (\gamma_1, \ldots, \gamma_n) \in \mathbb{N}^n$. The totality of particles $S_\eta$, generated as a result of $i$th particle's transformation, is determined by a random vector $\gamma$ with the distribution of probabilities $\left\{ P_\gamma(t), \sum_\gamma P_\gamma(t) = 1 \right\}$.

The main property that makes a Markov process also a branching process, is that the particles that exist in time $t_1$, in any other moment of time $t_1 + t, t > 0$, produce offsprings independently of one another.

The transition probabilities of the branching process $P_{\alpha\beta}(t)$ satisfy the branching condition

$$P_{\alpha\beta}(t) = \left( P_{\alpha}(t) \right)^{\beta_1} \left( P_{\alpha}(t) \right)^{\beta_2} \cdots \left( P_{\alpha}(t) \right)^{\beta_n} \quad (5.3)$$

for all $\alpha, \beta \in \mathbb{N}^n$, $t, \in [0, \infty)$. The equation (5.3) indicates that the distribution $\{ P_\beta(t) \}$ is the collection of $\beta_1$ distributions $\{ P_{\alpha}(t) \}$, $\beta_2$ distributions $\{ P_{\alpha}(t) \}$, ..., and $\beta_n$ distributions $\{ P_{\alpha}(t) \}$.

5.4 Multidimensional generating functions

Let $g_\alpha$ be the value of a numeric function at $\alpha$, defined on $\mathbb{N}^n$. The multidimensional generating function $F(s), s = (s_1, \ldots, s_n)$, corresponding to $\{ g_\alpha \}$, is the sum of series

$$F(s) = \sum_\alpha g_\alpha s_1^{\alpha_1} \ldots s_n^{\alpha_n} \quad (5.4)$$

Going forward we will use a shorter notation $s^\alpha = s_1^{\alpha_1} \ldots s_n^{\alpha_n}$. For the vectors $s = (s_1, \ldots, s_n)$ we use the notation $0$, if all components are equal to 0, and
1 if all $s_i = 1$; we use notation $|s|$ for vector $(|s_1|, ..., |s_n|)$. We also use notation:

$$
\alpha! = \alpha_1!, ..., \alpha_n!, \quad \alpha^{[\beta]} = \alpha_1^{[\beta_1]}, ..., \alpha_n^{[\beta_n]},
$$

where $\alpha^{[\beta]} = \alpha_i (\alpha_i - 1), ..., (\alpha_i - \beta_i + 1)$, $i = 1, ..., n$, and

$$
\frac{\partial^n F(s)}{\partial s^\alpha} = \frac{\partial^{[\alpha]} F(s)}{\partial s_1^{\alpha_1}, ..., \partial s_n^{\alpha_n}}
$$

(5.5)

The generating function $F(s)$ is called positive if all $g_\alpha \geq 0$. A positive generating function is called probabilistic if $F(1) = 1$.

If series (5.4) converges at some point $s_0 \neq 0$, then it converges for all $|s| < |s_0|$, and if $s$ is complex, then the sum $F(s)$ in that region is the analytical function over all variables $s_1, ..., s_n$. In this case the function $F(s)$ is clearly defined by coefficients $g_\alpha$

$$
g_\alpha = \frac{1}{\alpha!} \left. \frac{\partial^n F(s)}{\partial s^\alpha} \right|_{s=0}
$$

(5.6)

Thus, one-to-one correspondence is established between $\{g_\alpha\}$ and generating function $F(s)$.

Let $\beta \in N^n$. If $F(s)$ is a generating function for $\{g_\alpha\}$, then $\left(\partial^\beta F(s)/\partial s^\beta\right)$ is a generating function for $\{\alpha^\beta g_\alpha\}$. The probabilistic function $F(s)$ corresponds to $n$-dimensional probability distribution $\{g_\alpha\}$ on $N^n$. Also, probability generating functions do not have to relate to the distribution of probabilities $\{g_\alpha\}$. Instead they can relate to a random vector $\xi = (\xi_1, ..., \xi_n)$, having $\{g_\alpha\}$ as its distribution. Using a random vector $\xi$ helps in determining an equivalent definition of the probabilistic generating function

$$
F(s) = E s^\xi
$$

(5.7)
The expectation $E_{\xi^{[\beta]}}$ is called the factorial moment of $|\beta| = \beta_1 + \cdots + \beta_n$ order. $E_{\xi^{[\beta]}}$ is also called $\beta$-moment. We establish the equality

$$E_{\xi^{[\beta]}} = \frac{\partial^3 F(s)}{\partial s^\beta} igg|_{s=1}$$  \hspace{1cm} (5.8)

where the derivative in the point $s = 1$ is considered to be the corresponding left derivative over all coordinates $s_i$, $i = 1, \ldots, n$. In particular, for expectation the component of a random vector $\xi$ is

$$E_{\xi_i^{[\beta]}} = \frac{\partial F(s)}{\partial s_i} igg|_{s=1}, i = 1, \ldots, n$$  \hspace{1cm} (5.9)

The expression for the variance takes the form

$$D_{\xi_i} = \left[ \frac{\partial^2 F(s)}{\partial s_i^2} + \frac{\partial F(s)}{\partial s_i} \left( \frac{\partial F(s)}{\partial s_i} \right)^2 \right] \bigg|_{s=1}, i = 1, \ldots, n$$

The following property of generating functions is frequently used: if $\xi, \eta$ are independent random vectors, then the generating function of sum $\xi + \eta$ is equal to the product of generating functions summands; i.e.

$$F_{\xi+\eta}(s) = F_\xi(s) F_\eta(s)$$

5.5 Equations of branching processes

Let us introduce the generating processes $(z = (z_1, \ldots, z_n))$

$$G_\beta (t; z) = \sum_\alpha P_{\alpha\beta} (t) \frac{z_\alpha}{\alpha!}; \quad F_\alpha (t; s) = \sum_\beta P_{\alpha\beta} (t) s^\beta;$$

$$h_i (s) = \sum_\gamma p_i^\gamma s^\gamma, \quad i = 1, \ldots, n; \quad |s| \leq 1$$

and the linear differential operator with constant coefficients

$$h_i \left( \frac{\partial}{\partial z} \right) = \sum_\gamma p_i^\gamma \frac{\partial^\gamma}{\partial z^\gamma}, \quad i = 1, \ldots, n$$
Here \( \{p_i^\beta \geq 0, \ i = 1, ..., n; \sum_\gamma p_i^\gamma \} \) is the given probability distributions on \( N^n \).

The exponential generating function of transition probabilities \( G_\beta(t; z) \) of the branching process, for all \( \beta \in N^n \), satisfies the following linear partial differential equation

\[
\frac{\partial G_\beta(t; z)}{\partial t} = \sum_{i=1}^n \lambda_i z_i \left( h_i \left( \frac{\partial}{\partial z_i} \right) - \frac{\partial}{\partial z_i} \right) G_\beta(t; z)
\]

with initial condition \( G_\beta(0; z) = z^\beta / \beta! \). Equation (5.10) follows from the backward system of Kolmogorov’s partial differential equations (5.1).

The generating function of transition probabilities \( F_\alpha(t; s) \) of branching process for each \( \alpha \in N^n \) satisfies the linear partial differential equation for each \( |s| \leq 1 \)

\[
\frac{\partial F_\alpha(t; s)}{\partial t} = \sum_{i=1}^n \lambda_i (h_i(s) - s_i) \frac{\partial F_\alpha(t; s)}{\partial s_i}
\]

with initial condition \( F_\alpha(0; s) = s^\alpha \). Equation (5.11) follows from the second system of differential equations (5.2). The branching property (5.3) for generating functions \( F_\alpha(t; s) \) can be written as

\[
F_\alpha(t; s) = F_1^{\alpha_1}(t; s), ..., F_n^{\alpha_n}(t; s),
\]

where \( F_i(t; s) \) is the generating function of the process, starting with one particle of type \( i \) (single-particle generating function). Single-particle generating functions satisfy the ordinary system of nonlinear differential equations

\[
\frac{\partial F_i(t; s)}{\partial t} = \lambda_i (F_1(t; s), ..., F_n(t; s)) - F_i(t; s), \ i = 1, ..., n,
\]

with initial conditions \( F_i(0; s) = s_1, ..., F_n(0; s) = s_n \).
The branching process corresponds to the following transformations

\[
\begin{align*}
A_1 & \to \gamma_1^1 A_1 + \ldots + \gamma_1^n A_n; \\
& \ldots \\
A_t & \to \gamma_t^1 A_t + \ldots + \gamma_t^n A_n.
\end{align*}
\]

5.6 Probabilities of extinction

If there is a branching process \( \xi(t) = 0 \), it means that it has extinct by the time \( t \). The probability \( P_{0}^{i}(t) \), \( i = 1, \ldots, n \) is called the probability of extinction at time \( t \).

If the branching process \( \xi(t) \) becomes 0 in some finite \( t \), then the process extincts. The probabilities of extinction \( q^i \) of the branching process are equal to the limit \( q^i = \lim_{t \to \infty} P_{0}^{i}(t) \).

The branching process, for which \( q^1 = \ldots = q^n = 1 \), is called extinctor. If for any \( i \) the probability of extinction is \( q^i < 1 \) then the process is not extincting.

The probability of extinction \( q = (q^1, \ldots, q^n) \) of the branching process satisfies the system of equations

\[
h(q^1, \ldots, q^n) - q^i = 0, \quad i = 1, \ldots, n \quad (5.14)
\]

5.7 Equations for generating functions

In this work I consider branching processes over the following set

\[ N^2 = \{(\alpha_1, \alpha_2), \alpha_1, \alpha_2 \in N\} \]

(with two types of particles). For now, limit the type of transformations to:

\[
\begin{align*}
A & \to 0, B, A + B, 2A + B, 2A; \\
B & \to 0, A, A + B, A + 2B, 2B;
\end{align*}
\]
The intensity of transformations are \( \lambda \) and \( \mu, \lambda > 0, \mu > 0 \), the backward partial differential equation takes the form:

\[
\frac{\partial G_{\beta}(t; z_1, z_2)}{\partial t} = \lambda z_1 \left( p_0 + p_1 \frac{\partial}{\partial z_1} + p_2 \frac{\partial^2}{\partial z_1 \partial z_2} + p_3 \frac{\partial}{\partial z_1} \frac{\partial}{\partial z_2} + p_4 \frac{\partial^2}{\partial z_1^2} - \frac{\partial}{\partial z_1} \right) G_{\beta}(t; z_1, z_2) + \\
+ \mu z_2 \left( q_0 + q_1 \frac{\partial}{\partial z_1} + q_2 \frac{\partial^2}{\partial z_1 \partial z_2} + q_3 \frac{\partial}{\partial z_1} \frac{\partial}{\partial z_2} + q_4 \frac{\partial^2}{\partial z_1^2} - \frac{\partial}{\partial z_1} \right) G_{\beta}(t; z_1, z_2)
\]

with initial condition \( G_{\beta}(0; z_1, z_2) = z_1^{\beta_1} z_2^{\beta_2} / (\beta_1! \beta_2!) \). Here \( p_0 \geq 0, p_1 \geq 0, p_2 \geq 0, p_3 \geq 0, p_4 \geq 0, p_0 + p_1 + p_2 + p_3 + p_4 = 1; q_0 \geq 0, q_1 \geq 0, q_2 \geq 0, q_3 \geq 0, q_4 \geq 0, q_0 + q_1 + q_2 + q_3 + q_4 = 1 \)

The forward equation expressed through a generating function takes the form:

\[
\frac{\partial F_{\alpha}(t; s_1, s_2)}{\partial t} = \lambda \left( p_0 + p_1 s_2 + p_2 s_1 s_2 + p_3 s_1^2 s_2 + p_4 s_1^2 - s_1 \right) \frac{\partial F_{\alpha}(t; s_1, s_2)}{\partial s_1} + \\
+ \mu \left( q_0 + q_1 s_2 + q_2 s_1 s_2 + q_3 s_1^2 s_2 + q_4 s_2^2 s_2 - s_2 \right) \frac{\partial F_{\alpha}(t; s_1, s_2)}{\partial s_2}
\]

with initial condition \( F_{\alpha}(0; s_1, s_2) = s_1^{\alpha_1} s_2^{\alpha_2} \). The generating function \( F_{\alpha}(t; s_1, s_2) \) satisfies the non-linear condition (5.12); i.e., \( F_{\alpha}(t; s_1, s_2) = F_{\alpha_1}^{\alpha_1}(t; s_1, s_2) \cdot F_{\alpha_2}^{\alpha_2}(t; s_1, s_2) \) The systems of equations (5.13) for one-particles’ generating functions is

\[
\begin{align*}
\frac{\partial F_1}{\partial t} &= \lambda \left( p_0 + p_1 F_1 + p_2 F_1 F_2 + p_3 F_1^2 F_2 + p_4 F_1^2 - F_1 \right) \\
\frac{\partial F_2}{\partial t} &= \mu \left( q_0 + q_1 F_1 + q_2 F_1 F_2 + q_3 F_1^2 F_2 + q_4 F_1^2 - F_2 \right)
\end{align*}
\]

with initial conditions \( F_1(0; s_1, s_2) = s_1, F_2(0; s_1, s_2) = s_2 \).

5.7.1 Equations for expectations

The expectations of the number of particles are found by the formula

\[
E_i(t) = E \xi_i = \left. \frac{\partial F(t; s_1, s_2)}{\partial s_i} \right|_{s_1 = s_2}, \quad i = 1, 2
\]

Differentiating equation (5.15), we get the system of equations:

\[
\frac{\partial E_1(t)}{\partial t} = \lambda \left( p_2 + 2p_3 + 2p_4 - 1 \right) E_1(t) + \mu \left( q_1 + q_2 + q_3 \right) E_2(t)
\]

\[
\frac{\partial E_2(t)}{\partial t} = \lambda \left( p_1 + p_2 F_1 + p_3 F_1^2 F_2 + p_4 F_1^2 - F_1 \right) E_1(t) + \lambda \left( 2p_3 + 2p_4 - 1 \right) E_2(t) + \\
+ \mu \left( 2q_2 + q_3 F_1 F_2 + q_4 F_1^2 - F_2 \right) E_2(t)
\]
\[
\frac{dE_2(t)}{dt} = \lambda (p_1 + p_2 + p_3) E_1(t) + \mu (q_2 + 2q_3 + 2q_4 - 1) E_2(t)
\]

with initial conditions \( E_1(0) = \alpha_1, \ E_2(0) = \alpha_2 \)

It is easy to build exact solutions of the system of equations (5.16) using specifically tailored algorithms.
6. Transformations of the form \( A \rightarrow \gamma_1 A + \gamma_2 B; \ B \rightarrow 0 \)

6.1 Transformation \( A \rightarrow A + B, 2A + B; \ B \rightarrow 0 \)

Transformation of particles of the branching process can be represented in the following form:

\[
\begin{align*}
A & \rightarrow A + B, 2A + B \\
B & \rightarrow 0
\end{align*}
\]

The intensity of the transformations are \( \lambda \) and \( \mu, \lambda > 0, \mu > 0 \)

The backward partial differential equation has the following form

\[
\frac{\partial G_{\beta}(t; z_1, z_2)}{\partial t} = \lambda z_1 \left( p_2 \frac{\partial^2}{\partial z_1^2} + p_3 \frac{\partial^3}{\partial z_1^3} - \frac{\partial}{\partial z_1} \right) G_{\beta}(t; z_1, z_2) + \\
+ \mu z_2 \left( 1 - \frac{\partial}{\partial z_2} \right) G_{\beta}(t; z_1, z_2),
\]

with initial condition \( G_{\beta}(0; z_1, z_2) = z_1^{\beta_1} z_2^{\beta_2} / (\beta_1! \beta_2!) \). Here \( p_2 \geq 0, p_3 \geq 0, p_2 + p_3 = 1 \).

The forward equation represented through generating functions is

\[
\frac{\partial F_\alpha(t; s_1, s_2)}{\partial t} = \\
\lambda \left( p_2 s_1 s_2 + p_3 s_1^2 s_2 - s_1 \right) \frac{\partial F_\alpha(t; s_1, s_2)}{\partial s_1} + \mu \left( 1 - s_2 \right) \frac{\partial F_\alpha(t; s_1, s_2)}{\partial s_2},
\]

with initial condition \( F_\alpha(0; s_1, s_2) = s_1^{\alpha_1} s_2^{\alpha_2} \).

The system of equations for generating functions with one particle is

\[
\frac{\partial F_1}{\partial t} = \lambda \left( p_3 F_2 F_1^2 + p_2 F_1 F_2 - F_1 \right) \tag{6.1}
\]

\[
\frac{\partial F_2}{\partial t} = \mu \left( 1 - F_2 \right) \tag{6.2}
\]
with initial conditions \( F_1(0; s_1, s_2) = s_1, \ F_2(0; s_1, s_2) = s_2 \)

Solving (6.2), shows that \( F_2(t; s_2) = 1 - (1 - s_2)e^{-\mu t} \). Equation (6.1) is a Bernoulli equation, which is solved in quadratures. Standard replacement

\[ H(t) = \frac{1}{F_1(t; s_1, s_2)} \]

leads to a linear equation

\[ \frac{dH}{dt} = \lambda \left( p_3 + p_2 (1 - s_2) e^{-\mu t} \right) H - \lambda p_3 \left( 1 - (1 - s_2) e^{-\mu t} \right) \]

This is an ordinary first order differential equation, which can be solved by the method of variations of constants. The corresponding homogeneous equation takes the form

\[ \frac{dH}{dt} = \lambda \left( p_3 + p_2 (1 - s_2) e^{-\mu t} \right) H \]

Solving this equation:

\[ H(t) = C \exp \{ \lambda p_3 t - a (1 - s_2) e^{-\mu t} \} \]

\( C = C(t) \) can be obtained from the initial condition.

The final form is

\[
F_1(t; s_1, s_2) = \left\{ \begin{array}{c}
\frac{1}{s_1} e^{-a} - 1 F_1(b; b + 1; a(1 - s_2)) + \frac{b}{b+1} (1 - s_2) \times \\
\times 1 F_1(b + 1; b + 2; a(1 - s_2)) \\
+ 1 F_1(b; b + 1; a(1 - s_2) e^{-\mu t}) \\
- (1 - s_2) e^{-\mu t} 1 F_1(b + 1; b + 2; a(1 - s_2) e^{-\mu t})
\end{array} \right\} e^{\lambda p_3 t + \{ a(1 - s_2) e^{-\mu t} \}}^{-1}
\]

The generating function of the transition probabilities of the branching process with arbitrary initial state \((\alpha_1, \alpha_2)\) is determined by the branching property

\[ F_{\alpha}(t; s_1, s_2) = F_{\alpha_1}^{\alpha_1}(t; s_1, s_2) \ F_{\alpha_2}^{\alpha_2}(t; s_2) \]
The exact solution of this form of transformation cannot be found. The above system is simulated using single iteration with various rates, where \( r_1, \ldots, r_n \) is relative growth rate (percentage), \( i_1, \ldots, i_n \) is interaction rate (percentage) for populations 1, \ldots, \( n \).

\[
\begin{align*}
A \rightarrow A + B & \Rightarrow A \text{ relative growth } r_2 \\
A \rightarrow A + B & \Rightarrow A \text{ produces } B \text{ } i_3 \\
A \rightarrow 2A + B & \Rightarrow A \text{ relative growth } r_1 \\
A \rightarrow 2A + B & \Rightarrow A \text{ produces } B \text{ } i_2 \\
B \rightarrow 0 & \Rightarrow B \text{ self destroys } i_1
\end{align*}
\]

\( r_1 = 2, \ r_2 = 2, \ i_1 = 30, \ i_2 = 30, \ i_3 = 30; \)
\( r_1 = 2, \ r_2 = 2, \ i_1 = 50, \ i_2 = 50, \ i_3 = 50; \)
\( r_1 = 2, \ r_2 = 2, \ i_1 = 70, \ i_2 = 70, \ i_3 = 70; \)
\( r_1 = 7, \ r_2 = 7, \ i_1 = 30, \ i_2 = 30, \ i_3 = 30; \)
\( r_1 = 7, \ r_2 = 7, \ i_1 = 50, \ i_2 = 50, \ i_3 = 50; \)
\( r_1 = 7, \ r_2 = 7, \ i_1 = 70, \ i_2 = 70, \ i_3 = 70; \)
\( r_1 = 7, \ r_2 = 7, \ i_1 = 20, \ i_2 = 30, \ i_3 = 50; \)
\( r_1 = 5, \ r_2 = 5, \ i_1 = 50, \ i_2 = 10, \ i_3 = 10; \)
6.2 Transformation: $A \rightarrow 0, B, A + B, 2A; B \rightarrow 0$

6.2.1 System of non-linear equations

The schema of the transformation of the branching process is

\[
\begin{cases}
A \rightarrow 0, B, A + B, 2A \\
B \rightarrow 0
\end{cases}
\]
The intensity of transformation is \(\lambda\) and \(\mu\), \(\lambda > 0, \mu > 0\). The backward equation expressed through generating functions can be written as

\[
\frac{\partial G_{\beta}(t,z_1,z_2)}{\partial t} = \lambda z_1 \left( p_0 + p_1 \frac{\partial}{\partial z_2} + p_2 \frac{\partial^2}{\partial z_1 \partial z_2} + p_4 \frac{\partial^2}{\partial z_1^2} - \frac{\partial}{\partial z_1} \right) G_{\beta}(t,z_1,z_2) + \\
+ \mu z_2 \left( 1 - \frac{\partial}{\partial z_2} \right) G_{\beta}(t,z_1,z_2)
\]

with initial condition \(G_{\beta}(0; z_1, z_2) = z_1^{\beta_1} z_2^{\beta_2} / (\beta_1! \beta_2!)\). Here \(p_0 \geq 0, p_1 \geq 0, p_2 \geq 0, p_4 \geq 0, p_0 + p_1 + p_2 + p_4 = 1\). Kolmogorov’s forward equation expressed in generating functions is

\[
\frac{\partial F_{\alpha}(t,z_1,z_2)}{\partial t} = \lambda \left( p_0 + p_1 s_2 + p_2 s_1 s_2 + p_4 s_1^2 - s_1 \right) \frac{\partial F_{\alpha}(t,z_1,z_2)}{\partial s_1} + \\
+ \mu \left( 1 - s_2 \right) \frac{\partial F_{\alpha}(t,z_1,z_2)}{\partial s_2}
\]

with initial condition \(F_{\alpha}(0; s_1, s_2) = s_1^{\alpha_1} s_2^{\alpha_2}\).

The non-linear system of equations for generating functions having one-type particles takes the form

\[
\frac{\partial F_1}{\partial t} = \lambda(p_4 F_1^2 + p_2 F_1 F_2 + p_1 F_2 + p_0 - F_1) \\
\frac{\partial F_2}{\partial t} = \mu(1 - F_2)
\]

with initial conditions \(F_1(0; s_1, s_2) = s_1, F_2(0; s_1; s_2) = s_2\). The initial condition: \(F_2(0; s_1; s_2) = s_2\), which leads to:

\[
F_2(t; s_2) = 1 - (1 - s_2) e^{-\mu t}
\]

Replacing \(z = e^{-\mu t}\):

\[
\frac{\partial F_1}{\partial z} = \frac{d}{z} F_1 + \frac{a + b + d + c(1 - s_2)z}{z} F_1 - \frac{a + b - b(1 - s_2)z}{z} F_1
\]

(6.3)

where \(a = \lambda p_0 / \mu, b = \lambda p_1 / \mu, c = \lambda p_2 / \mu, d = \lambda p_4 / \mu\).
The Riccati equation (6.3) is written in the form of a linear differential equation of second order when substituting:

\[ F_1(z; s_1, s_2) = \frac{z}{d y(z; s_2)} \frac{\partial y(z; s_2)}{\partial z} \]

The general solution for equation (6.3) takes the form:

\[ F_1(z; s_1, s_2) = \frac{z}{d y(z; s_2)} \frac{d y(z; s_2)}{d z} = \frac{C W_1(t; s_2) + W_2(t; s_2)}{C H_1(t; s_2) + H_2(t; s_2)} \tag{6.4} \]

with the notations

\[ W_1(t; s_2) = (a + b) \times \]
\[ \times e^{-(a+b) \mu t}, F_2(a + b + bd/c, a + b + 1; a + b - d + 1, a + b; c(1 - s_2)e^{-\mu t}) \; ; \]
\[ W_2(t; s_2) = de^{-d \mu t} F_2(d + bd/c, d + 1, -a - b + d + 1, d; c(1 - s_2)e^{-\mu t}) \; ; \]
\[ H_1(t; s_2) = de^{-(a+b) \mu t} F_1(a + b + bd/c, a + b - d + 1, d; c(1 - s_2)e^{-\mu t}) \; ; \]
\[ H_2(t; s_2) = de^{-d \mu t} F_1(d + bd/c, -a - b + d + 1, d; c(1 - s_2)e^{-\mu t}) \; . \]

The constant \( C \) is derived from the initial condition \( F_1(0, s_1, s_2) = s_1 \),

\[ C = \frac{W_2(0; s_2) - s_1 H_2(0; s_2)}{W_1(0; s_2) - s_1 H_1(0; s_2)} \]

Initial conditions for generating function \( F_\alpha(t, s_1, s_2) \) is determined by the equation

\[ F_\alpha(t s_1, s_2) = F_1^{\alpha_1}(t; s_1, s_2) F_2^{\alpha_2}(t; s_2) = \]
\[ \left( \frac{C W_1(t; s_2) + W_2(t; s_2)^{\alpha_1}}{W_1(0; s_2) - s_1 H_1(0; s_2)} \right)^{\alpha_1} (1 - (1 - s_2)e^{-\mu t})^{\alpha_2} \]

The value of \( F_1(t, s_1, s_2) \) when \( s_1 = 1, s_2 = 1 \) is:

\[ C|_{s_1=1,s_2=1} = 0; \quad W_2(t; 1) = e^{-dt}; \quad H_2(t; 1) = e^{-dt} \]
Then

$$F_1(t;1,1) = 1$$

The exact solution of this form of transformation cannot be found. The above system is simulated using single iteration with various rates:

$$\begin{cases} 
A \rightarrow 0 & \Rightarrow \text{A self destroys } i_4 \\
A \rightarrow B & \Rightarrow \text{A transforms into B } i_3 \\
A \rightarrow A + B & \Rightarrow \text{A relative growth} \\
A \rightarrow A + B & \Rightarrow \text{A produces B } i_2 \\
A \rightarrow 2A & \Rightarrow \text{A relative growth} \\
B \rightarrow 0 & \Rightarrow \text{B self destroys } i_1 
\end{cases}$$

$$r_1 = 3, r_2 = 2, i_1 = 20, i_2 = 20, i_3 = 20, i_4 = 10;$$
$$r_1 = 3, r_2 = 2, i_1 = 20, i_2 = 50, i_3 = 50, i_4 = 10;$$
$$r_1 = 3, r_2 = 2, i_1 = 20, i_2 = 50, i_3 = 70, i_4 = 0.5;$$
$$r_1 = 4, r_2 = 2, i_1 = 20, i_2 = 20, i_3 = 50, i_4 = 0.5;$$
$$r_1 = 5, r_2 = 2, i_1 = 50, i_2 = 50, i_3 = 70, i_4 = 0.5;$$
$$r_1 = 7, r_2 = 2, i_1 = 70, i_2 = 70, i_3 = 20, i_4 = 0.5;$$
6.3 Transformation $A \to 0, A + B, 2A + B, 2A; B \to 0$

Let the system of transitions of the branching process be:

$$
\begin{align*}
\begin{cases}
A \to 0, A + B, 2A + B, 2A; \\
B \to 0.
\end{cases}
\end{align*}
$$

The intensity of transitions is $\lambda$ and $\mu$, $\lambda > 0$, $\mu > 0$.

Kolmogorov’s backward partial differential equation in this case takes the form:

$$
\frac{\partial G_\beta(t; z_1, z_2)}{\partial t} = \lambda z_1 \left( p_0 + p_2 \frac{\partial^2}{\partial z_1 \partial z_2} + p_3 \frac{\partial^3}{\partial z_1^2 \partial z_2} \frac{\partial^2}{\partial z_1^2} \frac{\partial^2}{\partial z_2^2} \right) G_\beta(t; z_1, z_2) + \\
+ \mu z_2 \left( 1 - \frac{\partial}{\partial z_2} \right) G_\beta(t; z_1, z_2)
$$

with initial condition $G_\beta(0; z_1, z_2) = z_1^{\beta_1} z_2^{\beta_2} / (\beta_1! \beta_2!)$. Here $p_0 \geq 0$, $p_2 \geq 0$, $p_2 \geq 0$, $p_4 \geq 0$, $p_0 + p_2 + p_4 + p_4 = 1$.

Kolmogorov’s forward equation, expressed through generating functions is

$$
\frac{\partial F_\alpha(t; s_1, s_2)}{\partial t} =
$$
\[
\lambda(p_0 + p_2 s_1 s_2 + p_3 s_1^2 s_2 + p_4 s_1^4 - s_1) \frac{\partial F_\alpha(t; s_1, s_2)}{\partial s_1} + \mu(1 - s_2) \frac{\partial F_\alpha(t; s_1, s_2)}{\partial s_1}
\]
with initial conditions \(F_\alpha(0; s_1, s_2) = s_1^{\alpha_1} s_2^{\alpha_2}\).

The non-linear system of equations for one-type particles’ generating functions is

\[
\frac{\partial F_1}{\partial t} = \lambda \left( p_4 F_1^2 + p_3 F_1^2 F_2 + p_2 F_1 F_2 + p_0 - F_1 \right),
\]

\[
\frac{\partial F_2}{\partial t} = \mu \left( 1 - F_2 \right)
\]

with initial conditions \(F_1(0; s_1, s_2) = s_1, F_2(0; s_1, s_2) = s_2\).

\(F_2(t; s_2) = 1 - (1 - s_2)e^{-\mu t}\). Replacing in the backward equation of the system \(z = e^{-\mu t}\):

\[
\frac{\partial F_1}{\partial t} = -\frac{c + d - c(1 - s_2)z}{z} F_1^2 + \frac{a + c + d + b(1 - s_2)z}{z} F_1 - \frac{a}{z}
\] (6.5)

where \(a = \lambda p_0 / \mu, b = \lambda p_2 / \mu, c = \lambda p_3 / \mu, d = \lambda p_4 / \mu\).

The equation (6.5) is a Riccati equation and can be transformed to an ordinary linear differential equation of the second order. The downside is that the resulting equation is going to be complex and the recurrent equation for coefficient determination of generalized series is going to be a second order equation with quadratic coefficients. Because of that, before transforming equation (6.5), perform the following substitution

\[
V(z; s_1, s_2) = \frac{1}{F_1(z; s_1, s_2)}.
\]

Then for \(V(z; s_1, s_2)\) the equation becomes:

\[
\frac{dV}{dz} = \frac{a}{z} V^2 - \frac{a + c + d + b(1 - s_2)z}{z} V + \frac{c + d - c(1 - s_2)z}{z}
\]

which is also a Riccati equation.
Introduce the notation

\[ W_1(t; s_2) = ae^{\alpha t} F_2(-a - ac/b, -a + 1; -a + c + d + 1, -a; -b(1 - s_2)e^{-\mu t}); \]

\[ W_2(t; s_2) = (c + d)e^{(c+d)t} F_2(-c - d - ac/b, -c - d + 1; a - c - d + 1, -c - d; -b(1 - s_2)e^{-\mu t}); \]

\[ H_1(t; s_2) = de^{\alpha t} F_1(-a - ac/b, -a + c + d + 1; -b(1 - s_2)e^{-\mu t}); \]

\[ H_2(t; s_2) = de^{\alpha t} F_1(-c - d - ac/b, a - c - d + 1; -b(1 - s_2)e^{-\mu t}). \]

The general solution of can be written as:

\[ F_1(t; s_1, s_2) = \frac{H_1(t; s_2) + CH_2(t; s_2)}{W_1(t; s_2) + CW_2(t; s_2)}, \]

where the constant \( C \) is determined using the initial condition \( F_1(0; s_1, s_2) = 0 \)

\[ C = - \frac{H_1(t; s_2) - s_1 W_1(t; s_2)}{H_2(t; s_2) - s_1 W_2(t; s_2)}. \]

The exact solution of this form of transformation cannot be found. The above system is simulated using single iteration with various rates:

\[
\begin{align*}
A \rightarrow 0 & => A \text{ self destroy } i_1 \\
A \rightarrow A + B & => A \text{ relative growth } r_1, r_2 \\
A \rightarrow A + B & => A \text{ produces } B \ i_4 \\
A \rightarrow 2A + B & => A \text{ relative growth } \\
A \rightarrow 2A + B & => A \text{ produces } B \ i_3 \\
A \rightarrow 2A & => A \text{ relative growth } \\
B \rightarrow 0 & => B \text{ self destroys } i_2
\end{align*}
\]
\[ r_1 = 3, \ r_2 = 2, \ i_1 = 20, \ i_2 = 20, \ i_3 = 20, \ i_4 = 20; \]
\[ r_1 = 3, \ r_2 = 2, \ i_1 = 20, \ i_2 = 50, \ i_3 = 50, \ i_4 = 50; \]
\[ r_1 = 3, \ r_2 = 2, \ i_1 = 20, \ i_2 = 50, \ i_3 = 70, \ i_4 = 70; \]
\[ r_1 = 4, \ r_2 = 2, \ i_1 = 20, \ i_2 = 20, \ i_3 = 50, \ i_4 = 90; \]
\[ r_1 = 4, \ r_2 = 2, \ i_1 = 50, \ i_2 = 50, \ i_3 = 70, \ i_4 = 90; \]
\[ r_1 = 5, \ r_2 = 2, \ i_1 = 50, \ i_2 = 50, \ i_3 = 70, \ i_4 = 20; \]
\[ r_1 = 3, \ r_2 = 4, \ i_1 = 70, \ i_2 = 70, \ i_3 = 70, \ i_4 = 70; \]
\[ r_1 = 3, \ r_2 = 5, \ i_1 = 70, \ i_2 = 50, \ i_3 = 50, \ i_4 = 70; \]
6.4 Transformation $A \rightarrow B, 2A + B; B \rightarrow 0$

The schema of the transformation of the branching process is

\[
\begin{align*}
A & \rightarrow B, 2A + B; \\
B & \rightarrow 0.
\end{align*}
\]

The intensity of transformations is $\lambda$ and $\mu$, $\lambda > 0$, $\mu > 0$. Kolmogorov's backward equation expressed through generating functions is:

\[
\frac{\partial G_\beta(t; z_1, z_2)}{\partial t} = \lambda z_1 \left( p_1 \frac{\partial}{\partial z_2} + p_3 \frac{\partial^3}{\partial z_1^3 \partial s_2} - \frac{\partial}{\partial z_1} \right) G_\beta(t; z_1, z_2) + \mu z_2 \left( 1 - \frac{\partial}{\partial z_2} \right) G_\beta(t; z_1, z_2),
\]

with initial condition $G_\beta(0; z_1, z_2) = z_1^{\beta_1} z_2^{\beta_2}/(\beta_1! \beta_2!)$. Here $p_1 \geq 0$, $p_3 \geq 0$; $p_1 + p_3 = 1$. Kolmogorov’s forward equation expressed through generating functions is

\[
\frac{\partial F_\alpha(t; s_1, s_2)}{\partial t} = \lambda \left( p_1 s_2 + p_3 s_1^2 s_2 - s_1 \right) \frac{\partial F_\alpha(t; s_1, s_2)}{\partial s_1} + \mu \left( 1 - s_2 \right) \frac{\partial F_\alpha(t; s_1, s_2)}{\partial s_2}
\]

with initial condition $F_\alpha(0; s_1, s_2) = s_1^{\alpha_1} s_2^{\alpha_2}$.
The system of equations for one-type particles’ generating functions

\[
\frac{\partial F_1}{\partial t} = \lambda (p_3 F_1^2 F_2 + p_1 F_2 - F_1),
\]
\[
\frac{\partial F_2}{\partial t} = \mu (1 - F_2),
\]

with initial condition \( F_1(0; s_1, s_2) = s_1, F_2(0; s_1, s_2) = s_2 \).

Replacing \( z = e^{-\mu t} \) in the backward equation \( F_2(t; s_2) = 1 - (1 - s_2)e^{-\mu t}; \)
\[
\frac{\partial F_1}{\partial z} = -\frac{a(1 - (1 - s_2)z)}{z} F_1^2 + \frac{a + b}{z} F_1 - \frac{b(1 - (1 - s_2)z)}{z},
\]

where \( a = \lambda p_1/\mu, b = \lambda p_3/\mu \).

Substituting:
\[
f_1(z; s_1, s_2) = \frac{z}{z(1 - (1 - s_2)z)} \frac{dy(z; s_2)}{dz}
\]

Which leads to:
\[
\frac{z^2 \partial^2 y(z; s_2)}{\partial z^2} + \frac{1 - (a + b)(1 - (1 - s_2)z)}{1 - (1 - s_2)z} \frac{\partial y(z; s_2)}{\partial z} - ab(1 - (1 - s_2)z)^2 y(z; s_2) = 0
\]

(6.8)

The exact solution of this form of transformation cannot be found. The above system is simulated using single iteration with various rates:

\[
\begin{align*}
A \rightarrow B & \Rightarrow A \text{ produces } B \ i_3 \\
A \rightarrow 2A + B & \Rightarrow A \text{ relative growth } r_1, \ r_2 \\
A \rightarrow 2A + B & \Rightarrow A \text{ produces } B \ i_2 \\
B \rightarrow 0 & \Rightarrow B \text{ self destroys } i_1
\end{align*}
\]
\[ r_1 = 2, \ r_2 = 2, \ i_1 = 20, \ i_2 = 20, \ i_3 = 20; \]
\[ r_1 = 2, \ r_2 = 2, \ i_1 = 20, \ i_2 = 50, \ i_3 = 50; \]
\[ r_1 = 2, \ r_2 = 2, \ i_1 = 20, \ i_2 = 50, \ i_3 = 70; \]
\[ r_1 = 2, \ r_2 = 2, \ i_1 = 20, \ i_2 = 20, \ i_3 = 50; \]
\[ r_1 = 5, \ r_2 = 5, \ i_1 = 50, \ i_2 = 50, \ i_3 = 70; \]
\[ r_1 = 5, \ r_2 = 5, \ i_1 = 70, \ i_2 = 40, \ i_3 = 40; \]
\[ r_1 = 5, \ r_2 = 5, \ i_1 = 70, \ i_2 = 50, \ i_3 = 20; \]
\[ r_1 = 5, \ r_2 = 5, \ i_1 = 70, \ i_2 = 20, \ i_3 = 50; \]
r1 = 05% r2 = 05% i1 = 70% i2 = 40% i3 = 40% r1 = 05% r2 = 05% i1 = 70% i2 = 50% i3 = 20%
7. Transformations of the form \( A \rightarrow \gamma_1 A + \gamma_2 B; \ B \rightarrow 2B \)

7.1 Transformation \( A \rightarrow 2A + B; \ B \rightarrow 2B \)

Transformations of the particles in the branching process occur by the following schema

\[
\begin{align*}
A & \rightarrow 2A + B; \\
B & \rightarrow 2B.
\end{align*}
\]

The intensity of the transformation is \( \lambda \) and \( \mu, \lambda > 0, \ \mu > 0 \). Kolmogorov’s backward equation expressed through generating functions is

\[
\frac{\partial G_{\beta}(t; z_1, z_2)}{\partial t} =
\]

\[
\lambda z_1 \left( \frac{\partial^3}{\partial z_1^3 \partial z_2} - \frac{\partial}{\partial z_1} \right) G_{\beta}(t; z_1, z_2) + \mu z_2 \left( \frac{\partial^2}{\partial z_2^2} - \frac{\partial}{\partial z_2} \right) G_{\beta}(t; z_1, z_2)
\]

with initial condition \( G_{\beta}(0; z_1, z_2) = z_1^{\beta_1} z_2^{\beta_2} / (\beta_1 ! \beta_2 !) \).

Kolmogorov’s forward equation in generating functions

\[
\frac{\partial F_\alpha(t; s_1, s_2)}{\partial t} = \lambda (s_1^2 - s_1) \frac{\partial F_\alpha(t; s_1, s_2)}{\partial s_1} + \mu (s_2^2 - s_2) \frac{\partial F_\alpha(t; s_1, s_2)}{\partial s_2},
\]

with initial condition \( F_\alpha(0; s_1, s_2) = s_1^{a_1} s_2^{a_2} \).

The system of non-linear equations for one-type particles’ generating functions is as follows

\[
\frac{\partial F_1}{\partial t} = \lambda (F_2 F_1^2 - F_1), \quad (7.1)
\]

\[
\frac{\partial F_2}{\partial t} = \mu (F_2^2 - F_2), \quad (7.2)
\]

with initial conditions \( F_1(0; s_1, s_2) = s_1, \ F_2(0; s_1, s_2) = s_2 \).
Equation (7.2) is a Bernoulli equation:

\[ F_2(t; s_2) = \frac{s_2}{s_2 + (1 - s_2)e^{\mu t}} = \frac{be^{-\mu t}}{1 + be^{-\mu t}} \]

where \( b = s_2/(1 - s_2) \). Substituting in (7.1):

\[ \frac{dF_1}{dt} = \frac{\lambda be^{-\mu t}}{1 + be^{-\mu t}} F_1^2 - \lambda F_1 \]

Substituting \( H(t) = F_1^{-1}(t; s_1, s_2) \):

\[ \frac{dH}{dt} = \lambda H - \frac{\lambda be^{-\mu t}}{1 + be^{-\mu t}} \]

\[ H(t) = C(t)e^{\lambda t}, \]

\[ \frac{dC}{dt} = -\frac{\lambda be^{-(\lambda + \mu)t}}{1 + be^{-\mu t}} \]  \( (7.3) \)

Using sum of geometric progression

\[ \frac{1}{1 + be^{-\mu t}} = \sum_{k=0}^{\infty} (-1)^k b^k e^{-k\mu t} \]

Integrating (7.3):

\[ C(t) = C_0 + be^{-(\lambda + \mu)t} \sum_{k=0}^{\infty} \frac{\lambda}{\lambda + \mu (k + 1)} (-1)^k b^k e^{-k\mu t} = \]

\[ = C_0 + \frac{\lambda b}{\lambda + \mu} e^{-(\lambda + \mu)t} \sum_{k=0}^{\infty} \frac{\Gamma(\lambda/\mu + 1 + k)}{\Gamma(\lambda/\mu + 1)} \frac{\Gamma(\lambda/\mu + 2)}{\Gamma(\lambda/\mu + 2 + k\mu)} \frac{\Gamma(k + 1)}{k!} (-1)^k b^k e^{-k\mu t} = \]

\[ = C_0 + \frac{\lambda b}{\lambda + \mu} e^{-(\lambda + \mu)t} F_1(\lambda/\mu + 1, 1; \lambda/\mu + 2; -b e^{-\mu t}), \]

where \( F_1(\alpha; \beta; \gamma; x) \) is a Gaussian hypergeometric function.

The constant \( C_0 \) is determined from the initial condition \( F_1(0; s_1; s_2) = s_1 \)

\[ C_0 = \frac{1}{s_1} - \frac{\lambda b}{\lambda + \mu} F_1(\lambda/\mu + 1, 1; \lambda/\mu + 2; -b). \]
Finally

\[ F_1(t; s_1, s_2) = \left[ \frac{1}{s_1} e - \frac{\lambda s_2}{(\lambda + \mu)(1 - s_2)} e^{\lambda t} _2F_1\left( \frac{\lambda}{\mu} + 1, 1; \frac{\lambda}{\mu} + 1, \frac{s_2}{1 - s_2} \right) - \frac{\lambda s_2}{(\lambda + \mu)(1 - s_2)} e^{-\mu t} _2F_1\left( \frac{\lambda}{\mu} + 1, 1; \frac{\lambda}{\mu} + 2, \frac{s_2}{1 - s_2} e^{-\mu t} \right) \right]^{-1} \]

The generating function of transition probabilities of branching process with an arbitrary initial condition \((\alpha_1, \alpha_2)\) is determined by the branching property of

\[ F_\alpha(t; s_1, s_2) = F_1^{\alpha_1}(t; s_1, s_2) F_2^{\alpha_2}(t; s_2) \]

The exact solution of this form of transformation cannot be found. The above system is simulated using single iteration with various rates:

\[
\begin{cases}
  A \to 2A + B \Rightarrow A \text{ relative growth } r_1 \\
  A \to 2A + B \Rightarrow A \text{ produces } B \text{ } i_1 \\
  B \to 2B \Rightarrow B \text{ relative growth } r_1
\end{cases}
\]

\[
\begin{align*}
  r_1 &= 0.1, \  i_1 = 0.2; \\
  r_1 &= 0.2, \  i_1 = 0.4; \\
  r_1 &= 0.2, \  i_1 = 0.8; \\
  r_1 &= 0.2, \  i_1 = 1; \\
  r_1 &= 0.2, \  i_1 = 3; \\
  r_1 &= 0.3, \  i_1 = 4; \\
  r_1 &= 0.3, \  i_1 = 5; \\
  r_1 &= 0.3, \  i_1 = 6;
\end{align*}
\]
7.2 Transformation $A \rightarrow 0, B, A + B, 2A; B \rightarrow 2B$
Transformations of the particles in the branching process occur by the following schema

\[
\begin{align*}
A & \rightarrow 0, B, A + B, 2A; \\
B & \rightarrow 2B.
\end{align*}
\]

The intensity of the transformation is \( \lambda \) and \( \mu, \lambda > 0, \mu > 0 \). Kolmogorov's backward equation expressed in generating functions is

\[
\frac{\partial G_\beta(t; z_1, z_2)}{\partial \beta} = \lambda z_1 \left( p_0 + p_1 \frac{\partial}{\partial z_2} - p_2 \frac{\partial^2}{\partial z_1 \partial z_2} + p_4 \frac{\partial^2}{\partial z_1^2} - \frac{\partial}{\partial z_1} \right) G_\beta(t; z_1, z_2) + \\
+ \mu z_2 \left( \frac{\partial^2}{\partial z_2^2} - \frac{\partial}{\partial z_2} \right) G_\beta(t; z_1, z_2)
\]

with initial condition \( G_\beta(0; z_1, z_2) = z_1^{\beta_1} z_2^{\beta_2} / (\beta_1! \beta_2!) \). Here \( p_0 \geq 0, p_1 \geq 0, p_2 \geq 0, p_4 \geq 0, p_0 + p_1 + p_2 + p + 4 = 1 \).

Kolmogorov's forward equation expressed in generating functions

\[
\frac{\partial F_\alpha(t; s_1, s_2)}{\partial t} = \lambda(p_0 + p_1 s_1 + p_2 s_1 s_2 + p_4 s_1^2 - s_1) \frac{\partial F_\alpha(t; s_1, s_2)}{\partial s_1} + \\
+ \mu(s_2^2 - s_2) \frac{\partial F_\alpha(t; s_1, s_2)}{\partial s_2}
\]

with initial condition \( F_\alpha(0; s_1, s_2) = s_1^\alpha s_2^\alpha \). Non-linear system of equations for generating functions that have one-type particles have the following form

\[
\frac{\partial F_1}{\partial t} = \lambda(p_4 F_1^2 + p_2 F_1 F_2 + p_1 F_2 + p_0 - F_1)
\]

\[
\frac{\partial F_2}{\partial t} = \mu(F_2^2 - F_2)
\]

with initial conditions \( F_1(0; s_1, s_2) = s_1, F_2(0; s_1, s_2) = s_2 \).

As before:

\[
F_2(t; s_2) = \frac{s_2}{s_2 + (1 - s_2)e^{\mu t}} = -\frac{f e^{-\mu t}}{1 - f e^{-\mu t}}
\]

87
where \( f = s_2/(s_2 - 1) \). Substituting \( z = e^{-\mu t} \), the Riccati equation is:

\[
\frac{dF_1}{dz} = -\frac{d}{z} F_1^2 + \frac{1}{z} \left( \frac{cfz}{1-fz} + a + b + c + d \right) F_1 + \frac{1}{z} \left( a - \frac{bfz}{1-fz} \right)
\]

where \( a = \lambda p_0/\mu, b = \lambda p_1/\mu, c = \lambda p_2/\mu, d = \lambda p_4/\mu \).

Transforming this into the linear differential equation of the second order:

\[
F_1'(z; s_1, s_2) = \frac{z}{d y(z)} \frac{d y(z)}{dz},
\]

\[
z^2 \frac{d^2 y(z)}{dz^2} + z \left( 1 - a - b - d - \frac{c}{1-fz} \right) \frac{d y(z)}{dz} + d \left( a - \frac{bfz}{1-fz} \right) y(z) = 0
\]

The solution in the form of generalized series with undetermined coefficients is

\[
y(z) = z^0 \sum_{k=0}^{\infty} A_k z^k, \quad A_0 \neq 0,
\]

The general solution can be written as

\[
F_1'(z; s_1, s_2) = \frac{z}{d y(z)} \frac{d y(z; s_2)}{dz} = \frac{C W_1(t; s_2) + W_2(t; s_2)}{C H_1(t; s_2) + H_2(t; s_2)},
\]

where

\[
W_1(t; s_2) = (v - \sqrt{u}/2) z^{(v-\sqrt{u}/2)\mu} F_2(v - \sqrt{u}/2 - d, v - \sqrt{u}/2 - a - b, 1 + v - \sqrt{u}/2; v - \sqrt{u}/2, 1 + \sqrt{u} - \frac{s_2}{1-s_2} e^{-\mu t});
\]

\[
W_2(t; s_2) = (v - \sqrt{u}/2) z^{(v+\sqrt{u}/2)\mu} F_2(v + \sqrt{u}/2 - d, v + \sqrt{u}/2 - a - b, 1 + v + \sqrt{u}/2; v + \sqrt{u}/2, 1 - \sqrt{u} + \frac{s_2}{1-s_2} e^{-\mu t});
\]

\[
H_1(t; s_2) =
\]

\[
d e^{(v-\sqrt{u}/2)\mu} F_1 \left( v - \sqrt{u}/2 - d, v - \sqrt{u}/2 - a - b; 1 + \sqrt{u} - \frac{s_2}{1-s_2} e^{-\mu t} \right); 
\]

\[
H_2(t; s_2) =
\]

88
\[ de^{(v+\sqrt{u}/2)\mu t} {} _2 F_1 \left( v + \sqrt{u}/2 - a, v + \sqrt{u}/2 - a - b, 1 - \sqrt{u}; \frac{s_2}{1 - s_2} e^{-\mu t} \right); \]

The constant \( C \) is determined from the initial condition \( F_1(0, s_1, s_2) = s_1 \)

\[ C = \frac{W_2(0; s_2) - s_1 H_2(0; s_2)}{W_2(0; s_2) - s_1 H_1(0; s_2)}; \]

The exact solution of this form of transformation cannot be found. The above system is simulated using single iteration with various rates:

\[
\begin{align*}
A &\rightarrow 0 \Rightarrow A \text{ self destroy } i_2 \\
A &\rightarrow B \Rightarrow A \text{ produces } B \ i_1 \\
\{ &\text{ } A \rightarrow A + B \Rightarrow A \text{ relative growth } r_2 \\
&\text{ } A \rightarrow A + B \Rightarrow A \text{ produces } B \ i_1 \\
A &\rightarrow 2A \Rightarrow A \text{ relative growth } r_2 \\
B &\rightarrow 2B \Rightarrow B \text{ relative growth } r_1
\end{align*}
\]

\[
\begin{align*}
r_1 &= 0.1, \ r_2 = 0.1, \ i_1 = 0.2, \ i_2 = 0.2; \\
r_1 &= 0.1, \ r_2 = 0.2, \ i_1 = 0.2, \ i_2 = 0.2; \\
r_1 &= 0.2, \ r_2 = 0.2, \ i_1 = 0.2, \ i_2 = 0.2; \\
r_1 &= 0.1, \ r_2 = 0.3, \ i_1 = 0.2, \ i_2 = 0.1; \\
r_1 &= 0.1, \ r_2 = 0.3, \ i_1 = 0.2, \ i_2 = 0.2; \\
r_1 &= 0.1, \ r_2 = 0.2, \ i_1 = 0.2, \ i_2 = 0.1;
\end{align*}
\]
7.3 Transformation $A \to B, 2A + B; B \to 2B$

Transformations of the particles in the branching process occur by the following schema

\[
\begin{align*}
A & \to B, 2A + B; \\
B & \to 2B.
\end{align*}
\]

The intensity of the transformation is $\lambda$ and $\mu, \lambda > 0, \mu > 0$. Kolmogorov’s backward equation expressed in generating functions is

\[
\frac{\partial G_\beta(t; z_1, z_2)}{\partial t} = \lambda z_1 \left( p_1 \frac{\partial}{\partial z_2} + p_3 \frac{\partial^3}{\partial z_1^2 \partial z_2} - \frac{\partial}{\partial z_1} \right) G_\beta(t; z_1, z_2) + \\
+ \mu z_2 \left( \frac{\partial^2}{\partial z_1^2} - \frac{\partial}{\partial z_2} \right) G_\beta(t; z_1, z_2)
\]

with initial condition $G_\beta(0; z_1, z_2) = z_1^{\beta_1} z_2^{\beta_2}/(\beta_1! \beta_2!)$. Here $p_1 \geq 0, p_3 \geq 0, p_1 + p_3 = 1$. Kolmogorov’s forward equation expressed in generating functions

\[
\frac{\partial F_\alpha(t; s_1, s_2)}{\partial t} = \lambda (p_1 s_2 + p_3 s_1^2 s_2 - s_1) \frac{\partial F_\alpha(t; s_1, s_2)}{\partial s_1} + \mu (s_2^2 - s_2) \frac{\partial F_\alpha(t; s_1, s_2)}{\partial s_2},
\]

90
with initial condition $F(0; s_1, s_2) = s_1^{a_1} s_2^{a_2}$.

The system of non-linear equations for one-type particles’ generating functions is

$$\frac{\partial F_1}{\partial t} = \lambda (p_3 F_1^2 F_2 + p_1 F_2 - F_1),$$
$$\frac{\partial F_2}{\partial t} = \mu (F_2^2 - F_2),$$

with initial conditions $F(0; s_1, s_2) = s_1, F(0; s_1, s_2) = s_2$:

$$F(t; s_2) = \frac{fe^{-\mu t}}{1 - fe^{-\mu t}}$$

where $f = s_2/(s_2 - 1)$. Substituting $z = e^{-\mu t}$ the Riccati equation is:

$$\frac{dF_1}{dz} = -\frac{bfz}{1 - fz} F_1^2 + \frac{a + b}{z} F_1 + \frac{afz}{1 - fz},$$

where $a = \lambda p_1/\mu, b = \lambda p_3/\mu$.

The exact solution of this form of transformation cannot be found. The above system is simulated using single iteration with various rates:

$$\begin{align*}
A \rightarrow B \Rightarrow A \text{ produces } B &\leftarrow i_1
\text{ } A \rightarrow 2A + B \Rightarrow A \text{ relative growth } r_2
\text{ } A \rightarrow A + B \Rightarrow A \text{ produces } B &\leftarrow i_1
\text{ } B \rightarrow 2B \Rightarrow B \text{ relative growth } r_1
\end{align*}$$

$$\begin{align*}
r_1 = 0.1, &\text{ } r_2 = 0.1, \text{ } i_1 = 0.1;
\text{ } r_1 = 0.1, &\text{ } r_2 = 0.1, \text{ } i_1 = 0.2;
\text{ } r_1 = 0.1, &\text{ } r_2 = 0.2, \text{ } i_1 = 0.2;
\text{ } r_1 = 0.2, &\text{ } r_2 = 0.2, \text{ } i_1 = 0.2;
\text{ } r_1 = 0.1, &\text{ } r_2 = 0.3, \text{ } i_1 = 0.2;
\text{ } r_1 = 0.3, &\text{ } r_2 = 0.2, \text{ } i_1 = 0.2;
\end{align*}$$

91
8. Other forms of transformations

8.1 Transformation $A \rightarrow B; B \rightarrow A$

Transformations of the particles in the branching process occur by the following schema

\[
\begin{cases} 
A \rightarrow B; \\
B \rightarrow A.
\end{cases}
\]

The intensity of the transformation is $\lambda$ and $\mu$, $\lambda > 0$, $\mu > 0$.

This branching process can be interpreted as heterogeneous over the space random walk on the interval with repulsive ends. Kolmogorov’s backward equation expressed in generating functions is

\[
\frac{\partial G_{\beta}(t; z_1, z_2)}{\partial t} = \lambda z_1 \left( \frac{\partial}{\partial z_2} - \frac{\partial}{\partial z_1} \right) G_{\beta}(t; z_1, z_2) + \mu z_2 \left( \frac{\partial}{\partial z_1} - \frac{\partial}{\partial z_2} \right) G_{\beta}(t; z_1, z_2)
\]

with initial condition $G_{\beta}(0; z_1, z_2) = z_1^{\beta_1} z_2^{\beta_2} / (\beta_1! \beta_2!)$.

Kolmogorov’s forward equation expressed in generating functions

\[
\frac{\partial F_{\alpha}(t; s_1, s_2)}{\partial t} = \lambda (s_2 - s_1) \frac{\partial F_{\alpha}(t; s_1, s_2)}{\partial s_1} + \mu (s_1 - s_2) \frac{\partial F_{\alpha}(t; s_1, s_2)}{\partial s_2},
\]

with initial condition $F_{\alpha}(0; s_1, s_2) = s_1^{\alpha_1} s_2^{\alpha_2}$.

The system of non-linear equations for one-type particles’ generating functions is

\[
\frac{\partial F_1}{\partial t} = \lambda (F_2 - F_1), \quad (8.1)
\]

\[
\frac{\partial F_2}{\partial t} = \mu (F_1 - F_2), \quad (8.2)
\]
with initial conditions \( F_1(0; s_1, s_2) = s_1, F_2(0; s_1, s_2) = s_2 \)

Dividing the equation (8.1) by the equation (8.2):

\[
\frac{dF_1(F_2)}{dF_2} = -\frac{\lambda}{\mu} = -\rho
\]

which leads to \( F_1(t) = -\rho F_2(t) \). After necessary substitutions into equation (8.2):

\[
\frac{dF_2}{dt} = -(\lambda + \mu) F_2 + \mu C_1,
\]

which follows by

\[
F_1(t; s_1, s_2) = C_1 \frac{\mu}{\lambda + \mu} + C_0 \frac{\lambda}{\mu} e^{-(\lambda + \mu) t}
\]

\[
F_2(t; s_1, s_2) = C_1 \frac{\mu}{\lambda + \mu} + C_0 \frac{\lambda}{\mu} e^{-(\lambda + \mu) t}
\]

The constants \( C_0, C_1 \) are determined from the initial conditions

\[
C_0 = \frac{\lambda}{\mu} (s_2 - s_1),
\]

\[
C_1 = s_1 + \frac{\lambda}{\mu} s_2,
\]

Finally,

\[
F_1(z; s_1, s_2) = \frac{\mu s_1 + \lambda s_2}{\lambda + \mu} + (s_1 - s_2) e^{-(\lambda + \mu) t}
\]

\[
F_2(z; s_1, s_2) = \frac{\mu s_1 + \lambda s_2}{\lambda + \mu} - (s_1 - s_2) e^{-(\lambda + \mu) t}
\]

The exact solution of this form of transformation cannot be found. The above system is simulated using single iteration with various rates:

\[
\begin{aligned}
A & \rightarrow B \Rightarrow A \text{ produces } B \ i_2 \\
B & \rightarrow A \Rightarrow B \text{ produces } A \ i_1
\end{aligned}
\]

94
\[
\begin{align*}
&\{ i_1 = 1.1, \ i_2 = 1.1; \\
&i_1 = 1.1, \ i_2 = 1; \\
&i_1 = 1.4, \ i_2 = 1; \\
&i_1 = 1, \ i_2 = 1.4; \\
&i_1 = 1.4, \ i_2 = 1.4; \\
&i_1 = 9, \ i_2 = 9; \\
\end{align*}
\]
8.2 **Transformation** $A \rightarrow A + B; B \rightarrow A$

Transformations of the particles in the branching process occur by the following schema

\[
\begin{align*}
A & \rightarrow A + B; \\
B & \rightarrow A.
\end{align*}
\]

The intensity of the transformation is $\lambda$ and $\mu$, $\lambda > 0$, $\mu > 0$.

Kolmogorov’s backward equation in generating functions is

\[
\frac{\partial G_\beta(t; z_1, z_2)}{\partial t} = 
\lambda z_1 \left( \frac{d^2}{\partial z_1 \partial z_2} - \frac{\partial}{\partial z_1} \right) G_\beta(t; z_1, z_2) + \mu z_2 \left( \frac{\partial}{\partial z_1} - \frac{\partial}{\partial z_2} \right) G_\beta(t; z_1, z_2)
\]

with initial condition $G_\beta(0; z_1, z_2) = z_1^{\beta_1} z_2^{\beta_2}/(\beta_1! \beta_2!)$.

Kolmogorov’s forward equation in generating functions

\[
\frac{\partial F_\alpha(t; s_1, s_2)}{\partial t} = \lambda(s_1 s_2 - s_1) \frac{\partial F_\alpha(t; s_1, s_2)}{\partial s_1} + \mu(s_1 - s_2) \frac{\partial F_\alpha(t; s_1, s_2)}{\partial s_2},
\]

with initial condition $F_\alpha(0; s_1, s_2) = s_1^{a_1} s_2^{a_2}$.

The system of non-linear equations for one-type particles’ generating functions is

\[
\begin{align*}
\frac{\partial F_1}{\partial t} & = \lambda(F_1 F_2 - F_1), \\
\frac{\partial F_2}{\partial t} & = \mu(F_1 - F_2),
\end{align*}
\]

with initial conditions $F_1(0; s_1, s_2) = s_1$, $F_2(0; s_1, s_2) = s_2$.

Dividing the equation (8.4) into the equation (8.3):

\[
\frac{dF_1}{dF_2} = \frac{\lambda F_1(F_2 - 1)}{\mu F_1 - F_2}
\]
Denote $x = F_2$ and substitute $H(x) = F_1(F_2) - F_2 = F_1(x) - x$:

$$\frac{dH(x)}{dx} + 1 = \frac{(H(x) + x)(x - 1)}{H(x)},$$

or

$$H(x) \cdot H'(x) = (x - 2)H(x) + x^2 - x \quad (8.5)$$

Equation (8.5) is an Abel's type equation, where the solving method is generally not known. To bring it to a canonical form, divide both parts of the equations by $(x - 2)$ and substitute the variables

$$\tau = \int (x - 2)dx = \frac{x^2}{2} - 2x + C,$$

$C$ is an arbitrary constant.

$$H(\tau) \cdot H'_\tau - H(\tau) = \varphi(\tau) \quad (8.6)$$

where the function $\varphi(\tau)$ is given by the parametric method

$$\varphi(\tau) = \frac{x^2 - x}{x - 2}, \quad \tau = \int (x - 2)dx.$$

The exact solution of this form of transformation cannot be found. The above system is simulated using single iteration with various rates:

\[
\begin{align*}
A &\rightarrow A + B \Rightarrow A \text{ relative growth } r_1 \\
A &\rightarrow A + B \Rightarrow A \text{ produces } B \quad i_2 \\
B &\rightarrow A \Rightarrow B \text{ produces } A \quad i_1
\end{align*}
\]
\begin{align*}
    r_1 &= 1, \ i_1 = 2, \ i_2 = 2; \\
    r_1 &= 2, \ i_1 = 2, \ i_2 = 2; \\
    r_1 &= 2, \ i_1 = 2, \ i_2 = 1; \\
    r_1 &= 2, \ i_1 = 1, \ i_2 = 1; \\
    r_1 &= 1, \ i_1 = 2, \ i_2 = 1; \\
    r_1 &= 2, \ i_1 = 3, \ i_2 = 1; \\
\end{align*}
8.3 Transformation $A \rightarrow A + B; B \rightarrow A + B$

Transformations of the particles in the branching process occur by the following schema

$$\begin{cases}
A \rightarrow A + B; \\
B \rightarrow A + B.
\end{cases}$$

The intensity of the transformation is $\lambda$ and $\mu, \lambda > 0, \mu > 0$. Kolmogorov’s backward equation expressed in generating functions is

$$\frac{\partial G_3(t; z_1, z_2)}{\partial t} =$$

$$\lambda z_1 \left( \frac{\partial^2}{\partial z_1 \partial z_2} - \frac{\partial}{\partial z_1} \right) G_3(t; z_1, z_2) + \mu z_2 \left( \frac{\partial^2}{\partial z_1 \partial z_2} - \frac{\partial}{\partial z_2} \right) G_3(t; z_1, z_2)$$

with initial condition $G_3(0; z_1, z_2) = z_1^\beta_1 z_2^\beta_2 / (\beta_1! \beta_2!)$.

Kolmogorov’s forward equation expressed in generating functions

$$\frac{\partial F_\alpha(t; s_1, s_2)}{\partial t} = \lambda (s_1 s_2 - s_1) \frac{\partial F_\alpha(t; s_1, s_2)}{\partial s_1} + \mu (s_1 s_2 - s_2) \frac{\partial F_\alpha(t; s_1, s_2)}{\partial s_2},$$

with initial condition $F_\alpha(0; s_1, s_2) = s_1^{\alpha_1} s_2^{\alpha_2}$.

The system of non-linear equations for one-type particles’ generating functions is

$$\frac{\partial F_1}{\partial t} = \lambda (F_1 F_2 - F_1), \quad (8.7)$$

$$\frac{\partial F_2}{\partial t} = \mu (F_1 F_2 - F_2), \quad (8.8)$$

with initial conditions $F_1(0; s_1, s_2) = s_1, F_2(0; s_1, s_2) = s_2$ Dividing the equation (8.7) by the equation (8.8):

$$\frac{dF_1}{dF_2} = \frac{F_1(F_2 - 1)}{F_2(F_1 - 1)} \quad (8.9)$$
Equation (8.9) is an equation with the divisive variables. Integrating:

\[
\frac{e^{F_1}}{F_1} = C \left( \frac{e^{F_2}}{F_2} \right)^\rho
\]  

(8.10)

where \( \rho = \frac{\lambda}{\mu} \).

To solve (8.10) relative to \( F_1 \) Lambert’s function, \( L(x) \), is used. By definition

\[
L(x) \cdot e^{L(x)} = x
\]

Then

\[
F_1 = -L \left( -\frac{1}{C e^{F_2}} \right).
\]

Substituting the expression to (8.8):

\[
\int \frac{dF_2}{F_2 \left[ 1 + L \left( -\frac{F_2^\rho}{C e^{F_2}} \right) \right]} = -\int \mu dt
\]

The exact solution of this form of transformation cannot be found. The above system is simulated using single iteration with various rates:

\[
\begin{cases}
A \rightarrow A + B \rightarrow A \text{ relative growth } r_1, r_2 \\
A \rightarrow A + B \rightarrow A \text{ produces } B \ i_2 \\
B \rightarrow A + B \rightarrow B \text{ relative growth } r_1, r_2 \\
B \rightarrow A + B \rightarrow B \text{ produces } A \ i_1
\end{cases}
\]

\[
\begin{align*}
& r_1 = 1, 
& r_2 = 1, 
& i_1 = 2, 
& i_2 = 1; \\
& r_1 = 1, 
& r_2 = 1, 
& i_1 = 2, 
& i_2 = 2; \\
& r_1 = 2, 
& r_2 = 2, 
& i_1 = 2, 
& i_2 = 2; \\
& r_1 = 2, 
& r_2 = 2, 
& i_1 = 3, 
& i_2 = 2; \\
& r_1 = 2, 
& r_2 = 2, 
& i_1 = 2, 
& i_2 = 3; \\
& r_1 = 2, 
& r_2 = 2, 
& i_1 = 3, 
& i_2 = 3;
\end{align*}
\]
8.4 Transformation \( A \rightarrow 2A; B \rightarrow A + 2B \)

Transformations of the particles in the branching process occur by the following schema

\[
\begin{align*}
A &\rightarrow 2A + B; \\
B &\rightarrow A + 2B.
\end{align*}
\]
The intensity of the transformation is \( \lambda \) and \( \mu \), \( \lambda > 0, \mu > 0 \). Kolmogorov's backward equation expressed in generating functions is

\[
\frac{\partial G_\beta (t; z_1, z_2)}{\partial t} =
\lambda z_1 \left( \frac{\partial^3}{\partial z_1^2 \partial z_2} - \frac{\partial}{\partial z_1} \right) G_\beta (t; z_1, z_2) + \mu z_2 \left( \frac{\partial^3}{\partial z_1 \partial z_2^2} - \frac{\partial}{\partial z_2} \right) G_\beta (t; z_1, z_2)
\]

with initial condition \( G_\beta (0; z_1, z_2) = \frac{z_1^{\beta_1} z_2^{\beta_2}}{(\beta_1! \beta_2!)}. \)

Kolmogorov's forward equation expressed in generating functions

\[
\frac{\partial F_\alpha (t; s_1, s_2)}{\partial t} = \lambda (s_1^2 s_2 - s_1) \frac{\partial F_\alpha (t; s_1, s_2)}{\partial s_1} + \mu (s_1 s_2^2 - s_2) \frac{\partial F_\alpha (t; s_1, s_2)}{\partial s_2},
\]

with initial condition \( F_\alpha (0; s_1, s_2) = s_1^{\alpha_1} s_2^{\alpha_2}. \)

The system of non-linear equations for one-type particles' generating functions is

\[
\frac{\partial F_1}{\partial t} = \lambda (F_1^2 F_2 - F_1), \tag{8.11}
\]

\[
\frac{\partial F_2}{\partial t} = \mu (F_1 F_2^2 - F_2), \tag{8.12}
\]

with initial conditions \( F_1(0; s_1, s_2) = s_1, \ F_2(0; s_1, s_2) = s_2. \)

Dividing the equation (8.11) by the equation (8.12):

\[
\frac{dF_1}{dF_2} = \frac{\lambda F_1}{\mu F_2}
\]

\[
F_1 = CF_2^{\lambda/\mu}
\]

For determining the function \( F_2 \):

\[
\frac{dF_2}{dt} = \mu \left( CF_2^{\lambda/\mu + 2} - F_2 \right),
\]

102
which is Bernoulli equation. Substitution $H(t) = F_2^{-\lambda/\mu-1}(t; s_1, s_2)$ leads to the following linear differential equation

$$\frac{dH}{dt} = \mu H - \mu C,$$

$$H(t) = C_1 e^{\mu t} + C$$

Following by

$$F_1(t) = C(C_1 e^{\mu t} + C)^{-\lambda/(\lambda+\mu)},$$

$$F_2(t) = C(C_1 e^{\mu t} + C)^{-\mu/(\lambda+\mu)},$$

The constants $C, C_1$ are determined from the initial condition

$$C = s_1 s_2^{-\lambda/\mu}, \quad C_1 = \left(\frac{1}{s_2} - s_1\right) s_2^{-\lambda/\mu}.$$

Finally,

$$F_1(t; s_1, s_2) = s_1 \left((1 - s_1 s_2) e^{\mu t} + s_1 s_2\right)^{-\lambda/(\lambda+\mu)},$$

$$F_2(t; s_1, s_2) = s_2 \left((1 - s_1 s_2) e^{\mu t} + s_1 s_2\right)^{-\mu/(\lambda+\mu)}.$$

To show solution for this transformation type using the model, the set of transformations can be represented in the following form:

$$\begin{cases}
A \rightarrow 2A \Rightarrow A \text{ relative growth } r_2 \\
B \rightarrow A + 2B \Rightarrow B \text{ relative growth } r_1 \\
B \rightarrow A + 2B \Rightarrow B \text{ produces } A \text{ i}_1
\end{cases}$$
\[ r_1 = 1, \ r_2 = 1, \ i_1 = 1; \]
\[ r_1 = 1, \ r_2 = 1, \ i_1 = 2; \]
\[ r_1 = 1, \ r_2 = 2, \ i_1 = 2; \]
\[ r_1 = 2, \ r_2 = 2, \ i_1 = 2; \]
\[ r_1 = 2, \ r_2 = 2, \ i_1 = 1; \]
\[ r_1 = 2, \ r_2 = 1, \ i_1 = 1; \]
\[ r_1 = 2, \ r_2 = 1, \ i_1 = 2; \]
\[ r_1 = 1, \ r_2 = 2, \ i_1 = 1; \]
9. Conclusion

The results of this work can be summarized as follows:

- We developed a software for simulation of possible chemical or biological reactions in a user-friendly format.

- Based on the algorithm of the software we have calculated statistical properties of different types of chemical reactions and investigated several specific types of chemical reactions.

- We implemented a discrete particle dynamics algorithm for simulation of plasma in the vascular space of a blood vessel.

- Some results from the program were compared and verified by theoretical calculations.

- A new approach of calculating statistical properties of collided particles were introduced.

- Using generating functions several important results on the statistical characteristics of some types of allocations were obtained.

Like all software the developed software has some limitations. First, it handles only three predefined compartments in the space: vascular, endothelial and interstitial. This limitation is relatively easy to address: the number of compartments could be 10 or more, depending on the computer power and the
nature of the problem. The number of compartments that can be created is a function of the number of cells and the number of types of interactions in each compartment. Another limitation in this program is that it does not take advantage of multi-tasking or parallel processing. To extend the program to multi-tasking or parallel processing the source code will need to be changed substantially. Adding multi-tasking or parallel processing would add a lot of additional power and speed, and would help in increasing the scale of the problem by calculating interactions in different compartments simultaneously.
Appendix A. The pseudocode

A.0.1 Description of Vixdum

XML document describing the system

The populations and interactions between the populations are described saved using XML markup language designed to describe structured data. The main section of the XML document describing the model is <MODEL>. The two subordinate sections under <MODEL> are <POPULATIONS> and <INTERACTIONS>: <MODEL>

  <POPULATIONS />
  <INTERACTIONS />
</MODEL>

This is the structure of <POPULATION> tag.

<POPULATION>
  <NAME>Activated T-Cells</NAME>
  <TYPE>1</TYPE>
  <ID>1510108310709247258880</ID>
  <OBJECTLIFECOUNT>200000</OBJECTLIFECOUNT>
  <INITIALSIZE>2701</INITIALSIZE>
  <SPACE>2</SPACE>
  <TREENODESTATUS>1</TREENODESTATUS>
  <DISPLAYSETTINGS>
    <OBJECTCOLOR>3</OBJECTCOLOR>
  </DISPLAYSETTINGS>
</POPULATION>
<OBJECTSIZE>40</OBJECTSIZE>
<ObjectShape>8</ObjectShape>
</DISPLAYSETTINGS>
<DYNAMICS>
<STEPRANDOM>
<ENABLED>1</ENABLED>
<VALUE>0.08</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</STEPRANDOM>
<GROUPTHRELATIVE>
<ENABLED>1</ENABLED>
<VALUE>1</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</GROUPTHRELATIVE>
<GROUPTHABSOLUTE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
A population is defined based on various characteristics such population properties:
- Type
  Discrete
  Stochastic
  Adaptive (allow system to choose approach)
- Space
  Interstitium
  Endothelium
  Vascular
- Enable
  Disable
  Enable
- Color
  Various predefined colors
- Size and shape
  Large
  Medium
  Small
- Visible
  Invisible
  Visible
  Partly visible (with transparency factor)
Figure A.1: This figure is a screenshot of the dialog that defines the population properties.

The last three properties define the appearance of the population on the computer monitor.

In addition, population has size dynamics defined using the following set of characteristics in the model:

- Initial population size (this is given across all finite volumes)
- Life cycle (each population objects)
- Step in random direction
  - Size (0, 10, 20, 40, 80)
  - Custom size
  - Apply at every simulation step
- Relative growth
  - Absolute growth (growth does not depend on the size of population)
  - Relative to population size
Figure A.2: This figure is a screenshot of the dialog that defines the size dynamics.

- Constant increment by counts
- Apply at every simulation step
  - Relative clearance
- Absolute clearance
- Relative to population size
- Constant increment by counts
- Apply at every simulation step

In addition, every population can take part in interaction, which has a separate XML tag for its definition <INTERACTION>:

```xml
<INTERACTION>
<NAME>Produce</NAME>
<ID>169710831092321233896</ID>
<POPULATION1> 106251083106908921362144 </POPULATION1>
```
<POPULATION2>32723108310862020915792</POPULATION2>
<INTERACTIONTRIGGER>(None)</INTERACTIONTRIGGER>
<TYPE>1</TYPE>
<PERIOD>1</PERIOD>
<RATE>200</RATE>
<TREENODESTATUS>1</TREENODESTATUS>
</INTERACTION>

<INTERACTION>
<NAME>Attract</NAME>
<ID>231510831092611237832</ID>
<POPULATION1>32723108310862020915792</POPULATION1>
<POPULATION2>1510108310709247258880</POPULATION2>
<INTERACTIONTRIGGER>(None)</INTERACTIONTRIGGER>
<TYPE>4</TYPE>
<PERIOD>1</PERIOD>
<RATE>100</RATE>
<TREENODESTATUS>1</TREENODESTATUS>
</INTERACTION>

<INTERACTION>
<NAME>Transform Into</NAME>
<ID>703110831112151238772</ID>
<POPULATION1>1510108310709247258880</POPULATION1>
<POPULATION2>28371108311115149706400</POPULATION2>
<INTERACTIONTRIGGER>
An interaction can be based on one or two populations. Allowed interaction types are the following:

- None
- Attract
- Collide
- Produce
- Self destroy
- Transform into different cell

In addition an interaction can be triggered by another interaction of any type. For example a transformation of an object of one population type into another population type can be triggered by an interaction event of collision. Such flexible modeling approach allows simulation of complex interactions.

In addition interactions can be edited using interaction tab of the population dialog, which lists all interactions that the populations take part.

An interaction is defined using a set of properties: - First population - Interaction type - Second population - Interaction rate - Trigger on - Apply at every iterations
Figure A.3: This figure is a screenshot of the dialog that defines the interactions.

Figure A.4: This figure shows a screenshot of the dialog that lists all interactions that a populations takes part.
Figure A.5: This figure shows a screenshot of the dialog that lists all interactions that a populations takes part.

The main window of the system with the loaded model of infiltration demyelination modeled using the above described properties of populations and populations is shown on A.6:

Once the populations and interactions are defined the finite volumes are superimposed onto the space where the populations are defined. There are menu options to start and end simulation processes.

Brusselator

Model of the Brusselator is described using the following populations and interactions.
Figure A.6: This figure shows the finite volume mesh superimposed onto the vascular, endothelial and interstitial space.
- X1 transforms into Y1,
- upon collision between Y1 and X2, X2 transforms into Y2
- Population Y1 and Y2 collide and upon collision Y1 transforms into Y2
- Population Y1 transforms into Z2

Using reaction rate equations we can also describe the system as follows:

\[-X_1 \xrightarrow{K_1} Y_1\]
\[-Y_1 + X_2 \xrightarrow{K_2} Y_1 + Y_2\]
\[-Y_1 \xrightarrow{K_3} Y_2\]
\[-Y_1 \xrightarrow{K_4} Z_2\]

Simulation results show oscillating paths that reflect changes in the size of populations of the interacting objects.

A.0.2 Pseudocode

<MODEL>
<POPULATIONS>
<POPULATION>
<NAME>Activated T-Cells</NAME>
Figure A.7: Upper left section of the figure represents a snapshot of the Brusselator model when population B prevails and upper right figure represents when the same population B is depressed. The lower section of the figure shows dynamics of population size.
<TYPE>1</TYPE>
<ID>1510108310709247258880</ID>
<ObjectLifeCount>200000</ObjectLifeCount>
<InitialSize>2701</InitialSize>
<Space>2</Space>
<TreeNodeStatus>1</TreeNodeStatus>
<DisplaySettings>
<ObjectColor>3</ObjectColor>
<ObjectSize>40</ObjectSize>
<ObjectShape>8</ObjectShape>
</DisplaySettings>
<Dynamics>
<StepRandom>
<Enabled>1</Enabled>
<Value>0.08</Value>
<Period>1</Period>
<Type>256</Type>
<Override>0</Override>
<OverrideValue>0</OverrideValue>
</StepRandom>
<GrowthRelative>
<Enabled>1</Enabled>
<Value>1</Value>
<Period>1</Period>
</GrowthRelative>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</CLEARANCEABSOLUTE>
</DYNAMICS>
</POPULATION>
<POPULATION>
<NAME>Infiltrated T-Cells</NAME>
<TYPE>1</TYPE>
<ID>28371108311115149706400</ID>
<ObjectLifecount>1000</ObjectLifecount>
<Initialsize>0</Initialsize>
<Space>1</Space>
<Treenodestatus>1</Treenodestatus>
<Displaysettings>
<Objectcolor>74</Objectcolor>
<Objectsize>40</Objectsize>
<Objectshape>8</Objectshape>
</Displaysettings>
</DYNAMICS>
<Steprandom>
<Enabled>1</Enabled>
<Value>0.02</Value>
<Period>1</Period>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</STEPRANDOM>
<GROWTHRELATIVE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
(TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</GROWTHRELATIVE>
<GROWTHABSOLUTE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
(TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</GROWTHABSOLUTE>
<CLEARANCERELATIVE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<OBJECTCOLOR>0</OBJECTCOLOR>
<OBJECTSIZE>10</OBJECTSIZE>
<OBJECTSHAPE>8</OBJECTSHAPE>
</DISPLAYSETTINGS>
<DYNAMICS>
<STEPRANDOM>
<ENABLED>1</ENABLED>
<VVALUE>0.01</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</STEPRANDOM>
<GROWTHRELATIVE>
<ENABLED>0</ENABLED>
<VVALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</GROWTHRELATIVE>
<GROWTHABSOLUTE>
<ENABLED>0</ENABLED>
<VVALUE>0</VALUE>

126
<PERIOD>1</PERIOD>
<Type>256</Type>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</GROWTHABSOLUTE>
<CLEARANCERELATIVE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<Type>256</Type>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</CLEARANCERELATIVE>
<CLEARANCEABSOLUTE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<Type>256</Type>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</CLEARANCEABSOLUTE>
</DYNAMICS>
</POPULATION>
</POPULATION>
<NAME>Endothelial Cells</NAME>

<TYPE>1</TYPE>

<ID>20817108310716247288544</ID>

<OBJECTLIFECOUNT>200000</OBJECTLIFECOUNT>

<INITIALSIZE>2144</INITIALSIZE>

<SPACE>4</SPACE>

<TREENODESTATUS>1</TREENODESTATUS>

<DISPLAYSETTINGS>

<OBJECTCOLOR>50</OBJECTCOLOR>

<OBJECTSIZE>20</OBJECTSIZE>

<OBJECTSHAPE>8</OBJECTSHAPE>

</DISPLAYSETTINGS>

<DYNAMICS>

<STEPRANDOM>

<ENABLED>1</ENABLED>

<VALUE>0</VALUE>

<PERIOD>1</PERIOD>

<TYPE>256</TYPE>

<OVERRIDE>1</OVERRIDE>

<OVERRIDEVALUE>0.001</OVERRIDEVALUE>

</STEPRANDOM>

<GROWTHRELATIVE>

<ENABLED>0</ENABLED>

<VALUE>1.4</VALUE>

</GROWTHRELATIVE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDGEVALUE>0</OVERRIDGEVALUE>
</GROWTHRELATIVE>
<GROWTHABSOLUTE>
<ENABLED>0</ENABLED>
(VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDGEVALUE>0</OVERRIDGEVALUE>
</GROWTHABSOLUTE>
<CLEARANCERELATIVE>
<ENABLED>1</ENABLED>
(VALUE>1</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDGEVALUE>0</OVERRIDGEVALUE>
</CLEARANCERELATIVE>
<CLEARANCEABSOLUTE>
<ENABLED>0</ENABLED>
(VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</CLEARANCEABSOLUTE>
</DYNAMICS>
</POPULATION>

<POPULATION>
(NAME>Activated Endothelial Cells</NAME>
<TYPE>1</TYPE>
<ID>6872108310725447292392</ID>
<ObjectLIFECOUNT>200000</ObjectLIFECOUNT>
<INITIALSIZE>0</INITIALSIZE>
<SPACE>4</SPACE>
<TREENODESTATUS>1</TREENODESTATUS>
</DISPLAYSETTINGS>
<ObjectCOLOR>85</ObjectCOLOR>
<ObjectSIZE>20</ObjectSIZE>
<ObjectSHAPE>8</ObjectSHAPE>
</DISPLAYSETTINGS>

<DYNAMICS>
<STEPRANDOM>
<ENABLED>1</ENABLED>
<Value>0.01</Value>

<PERIOD>1</PERIOD>
(TYPE>256</TYPE>
<OVERRIDE>1</OVERRIDE>
<OVERRIDEVALUE>0.001</OVERRIDEVALUE>
</STEPRANDOM>
<GROWTHRELATIVE>
<ENABLED>0</ENABLED>
.VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<Type>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</GROWTHRELATIVE>
<GROWTHABSOLUTE>
<ENABLED>0</ENABLED>
.VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<Type>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</GROWTHABSOLUTE>
<CLEARANCERELATIVE>
<ENABLED>0</ENABLED>
.VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</CLEARANCERELATIVE>
<CLEARANCEABSOLUTE>
<ENABLED>0</ENABLED>
(VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</CLEARANCEABSOLUTE>
</DYNAMICS>
</POPULATION>

<POPULATION>
<NAME>Macrophage-Attracting Cytokines</NAME>
<TYPE>1</TYPE>
<ID>10221108310840520635784</ID>
<ObjectLifecount>10</ObjectLifecount>
<InitialSize>0</InitialSize>
<Space>4</Space>
<TreenodeStatus>1</TreenodeStatus>
<DisplaySettings>
<OBJECTCOLOR>122</OBJECTCOLOR>
<OBJECTSIZE>10</OBJECTSIZE>
<OBJECTSHAPE>8</OBJECTSHAPE>
</DISPLAYSETTINGS>
<DYNAMICS>
<STEPRANDOM>
<ENABLED>1</ENABLED>
.VALUE>0.08</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>1</OVERRIDE>
<OVERRIDEVALUE>0.1</OVERRIDEVALUE>
</STEPRANDOM>
<GROWTHRELATIVE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</GROWTHRELATIVE>
<GROWTHABSOLUTE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
</GROWTHABSOLUTE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</GROWTHABSOLUTE>
<CLEARANCERELATIVE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</CLEARANCERELATIVE>
<CLEARANCEABSOLUTE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</CLEARANCEABSOLUTE>
</DYNAMICS>
</POPULATION>
</POPULATION>
<NAME>Macrophage- and T-Cell- Attracting Chemokines</NAME>

<TYPE>1</TYPE>

<ID>32723108310862020915792</ID>

<ObjectLifecount>20</ObjectLifecount>

<InitialSize>1</InitialSize>

<Space>1</Space>

<TreenodeStatus>1</TreenodeStatus>

<DisplaySettings>
<ObjectColor>23</ObjectColor>
<ObjectSize>10</ObjectSize>
<ObjectShape>8</ObjectShape>
</DisplaySettings>

<Dynamics>
<StepRandom>
<Enabled>1</Enabled>
<Value>0.08</Value>
<Period>1</Period>
<Type>256</Type>
<Override>0</Override>
<OverrideValue>0</OverrideValue>
</StepRandom>
</GrowthRelative>

<Enabled>0</Enabled>
<Value>0</Value>
<PERIOD>1</PERIOD>
<Type>256</Type>
<Override>0</Override>
<OverrideValue>0</OverrideValue>
</GrowthRelative>
</GrowthAbsolute>
<Enabled>0</Enabled>
<Value>0</Value>
<Period>1</Period>
<Type>256</Type>
<Override>0</Override>
<OverrideValue>0</OverrideValue>
</GrowthAbsolute>
</ClearanceRelative>
<Enabled>0</Enabled>
<Value>0</Value>
<Period>1</Period>
<Type>256</Type>
<Override>0</Override>
<OverrideValue>0</OverrideValue>
</ClearanceRelative>
</ClearanceAbsolute>
<Enabled>0</Enabled>
<Value>0</Value>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</CLEARANCEABSOLUTE>
</DYNAMICS>
</POPULATION>

<POPULATION>
(NAME>Macroglia</NAME>
<TYPE>1</TYPE>
<ID>6717108310900821352032</ID>
<ObjectLIFECOUNT>40000</ObjectLIFECOUNT>
<INITIALSIZE>1245</INITIALSIZE>
<SPACE>1</SPACE>
<TREENODESTATUS>1</TREENODESTATUS>
</DISPLAYSETTINGS>
<ObjectCOLOR>68</ObjectCOLOR>
<ObjectSIZE>20</ObjectSIZE>
<ObjectSHAPE>8</ObjectSHAPE>
</DISPLAYSETTINGS>
</DYNAMICS>
<STEPRANDOM>
<ENABLED>1</ENABLED>
<VALUE>0.01</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</STEPRANDOM>
<GROWTHRELATIVE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</GROWTHRELATIVE>
<GROWTHABSOLUTE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</GROWTHABSOLUTE>
<CLEARANCERELATIVE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</CLEARANCERELATIVE>
<CLEARANCEABSOLUTE>
<ENABLED>0</ENABLED>
(VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</CLEARANCEABSOLUTE>
</DYNAMICS>
</POPULATION>
<POPULATION>
<NAME>Activated Macroglia</NAME>
<TYPE>1</TYPE>
<ID>10625108310908921362144</ID>
<OBJECTLIFECOUNT>30000</OBJECTLIFECOUNT>
<INITIALSIZE>0</INITIALSIZE>
<SPACE>1</SPACE>
<TREENODESTATUS>1</TREENODESTATUS>
<DISPLAYSETTINGS>
<OBJECTCOLOR>97</OBJECTCOLOR>
<OBJECTSIZE>20</OBJECTSIZE>
<OBJECTSHAPE>8</OBJECTSHAPE>
</DISPLAYSETTINGS>
<DYNAMICS>
<STEPRANDOM>
<ENABLED>1</ENABLED>
<VALUE>0.02</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
< OVERRIDE>0</OVERRIDE>
< OVERRIDENVARIABLE>0</OVERRIDENVARIABLE>
</STEPRANDOM>
<GROWTHRELATIVE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
< OVERRIDE>0</OVERRIDE>
< OVERRIDENVARIABLE>0</OVERRIDENVARIABLE>
</GROWTHRELATIVE>
<GROWTHABSOLUTE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRISEVALUE>0</OVERRISEVALUE>
</GROWTHABSOLUTE>
<CLEARANCERELATIVE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRISEVALUE>0</OVERRISEVALUE>
</CLEARANCERELATIVE>
<CLEARANCEABSOLUTE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRISEVALUE>0</OVERRISEVALUE>
</CLEARANCEABSOLUTE>
</DYNAMICS>
</POPULATION>
<POPULATION>

<NAME>Myelin</NAME>
<TYPE>1</TYPE>
<ID>4277108311123949756424</ID>
<OBJECTLIFECOUNT>400000</OBJECTLIFECOUNT>
<INITIALSIZE>2200</INITIALSIZE>
<SPACE>1</SPACE>
<TREENODESTATUS>1</TREENODESTATUS>
<DISPLAYSETTINGS>
<OBJECTCOLOR>60</OBJECTCOLOR>
<OBJECTSIZE>40</OBJECTSIZE>
<OBJECTSHAPE>8</OBJECTSHAPE>
</DISPLAYSETTINGS>
<DYNAMICS>
<STEPRANDOM>
<ENABLED>1</ENABLED>
.VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>1</OVERRIDE>
<OVERRIDEVALUE>0.001</OVERRIDEVALUE>
</STEPRANDOM>
<GROWTHRELATIVE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</GROWTHRELATIVE>
</GROWTHABSOLUTE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</GROWTHABSOLUTE>
</CLEARANCERELATIVE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</CLEARANCERELATIVE>
</CLEARANCEABSOLUTE>
<ENABLED>0</ENABLED>
<VALUE>0</VALUE>
<PERIOD>1</PERIOD>
<TYPE>256</TYPE>
<OVERRIDE>0</OVERRIDE>
<OVERRIDEVALUE>0</OVERRIDEVALUE>
</CLEARANCEABSOLUTE>
</DYNAMICS>
</POPULATION>
</POPULATIONS>
<INTERACTIONS>
<INTERACTION>
<NAME>Collide</NAME>
<ID>1807810831077151231236</ID>
<POPULATION1>20817108310716247288544</POPULATION1>
<POPULATION2>18667108310877120960424</POPULATION2>
<INTERACTIONTRIGGER>(None)</INTERACTIONTRIGGER>
<TYPE>16</TYPE>
<PERIOD>1</PERIOD>
<RATE>5</RATE>
<TREENODESTATUS>3</TREENODESTATUS>
</INTERACTION>
</INTERACTION>
<NAME>Transform Into</NAME>
<ID>1304710831077231237832</ID>
<POPULATION1>
20817108310716247288544<
/POPULATION1>

<POPULATION2>
6872108310725447292392<
/POPULATION2>

<INTERACTIONTRIGGER>
1807810831077151231236<
/INTERACTIONTRIGGER>

<TYPE>2</TYPE>

<PERIOD>1000001</PERIOD>

<RATE>100</RATE>

<TREENODESTATUS>1</TREENODESTATUS>
</INTERACTION>

<INTERACTION>

<NAME>Produce</NAME>

<ID>2924210831085031237832</ID>

<POPULATION1>6872108310725447292392</POPULATION1>

<POPULATION2>10221108310840520635784</POPULATION2>

<INTERACTIONTRIGGER>(None)</INTERACTIONTRIGGER>

<TYPE>1</TYPE>

<PERIOD>1</PERIOD>

<RATE>50</RATE>

<TREENODESTATUS>1</TREENODESTATUS>
</INTERACTION>
<INTERACTION>
  <NAME>Produce</NAME>
  <ID>1189810831088781237832</ID>
  <POPULATION1>1510108310709247258880</POPULATION1>
  <POPULATION2>18667108310877120960424</POPULATION2>
  <INTERACTIONTRIGGER>(None)</INTERACTIONTRIGGER>
  <TYPE>1</TYPE>
  <PERIOD>3</PERIOD>
  <RATE>5</RATE>
  <TREENODESTATUS>1</TREENODESTATUS>
</INTERACTION>

<INTERACTION>
  <NAME>Produce</NAME>
  <ID>169710831092321233896</ID>
  <POPULATION1>10625108310908921362144</POPULATION1>
  <POPULATION2>32723108310862020915792</POPULATION2>
  <INTERACTIONTRIGGER>(None)</INTERACTIONTRIGGER>
  <TYPE>1</TYPE>
  <PERIOD>1</PERIOD>
  <RATE>200</RATE>
  <TREENODESTATUS>1</TREENODESTATUS>
</INTERACTION>

<INTERACTION>
  <NAME>Attract</NAME>
</INTERACTION>
<ID>231510831092611237832</ID>

<POPULATION1>
32723108310862020915792</POPULATION1>

<POPULATION2>
1510108310709247258880</POPULATION2>

<INTERACTIONTRIGGER>(None)</INTERACTIONTRIGGER>

<TYPE>4</TYPE>

<PERIOD>1</PERIOD>

<RATE>100</RATE>

<TREENODESTATUS>1</TREENODESTATUS>

</INTERACTION>

<INTERACTION>

<NAME>Transform Into</NAME>

<ID>703110831112151238772</ID>

<POPULATION1>
1510108310709247258880</POPULATION1>

/POPULATION1>

<POPULATION2>
28371108311115149706400</POPULATION2>

/POPULATION2>

<INTERACTIONTRIGGER>
2595510881969711231240</INTERACTIONTRIGGER>

<TYPE>2</TYPE>
<PERIOD>1000001</PERIOD>

<RATE>100</RATE>

<TREENODESTATUS>1</TREENODESTATUS>

</INTERACTION>

<INTERACTION>

<NAME>Collide</NAME>

<ID>1522510831095341233896</ID>

<POPULATION1>
10221108310840520635784</POPULATION1>

<POPULATION2>
6717108310900821352032</POPULATION2>

<INTERACTIONTRIGGER>(None)</INTERACTIONTRIGGER>

<TYPE>16</TYPE>

<PERIOD>1</PERIOD>

<RATE>5</RATE>

<TREENODESTATUS>0</TREENODESTATUS>

</INTERACTION>

<INTERACTION>

<NAME>Transform Into</NAME>

<ID>1554210831095671233896</ID>

<POPULATION1>
6717108310900821352032</POPULATION1>

/POPULATION1>

<POPULATION2>
10625108310908921362144<br />
/POPULATION2>

<INTERACTIONTRIGGER>
1522510831095341233896<br />
/INTERACTIONTRIGGER>

<Type>2</Type>

<PERIOD>1</PERIOD>

<RATE>5</Rate>

<TREENODESTATUS>1</TREENODESTATUS>

</INTERACTION>

<INTERACTION>

<NAME>Attract</NAME>

<ID>2839310831113981238772</ID>

<POPULATION1>427710831123949756424</POPULATION1>

/POPULATION1>

<POPULATION2>2837110831115149706400</POPULATION2>

/POPULATION2>

<INTERACTIONTRIGGER>(None)</INTERACTIONTRIGGER>

<Type>4</Type>

<PERIOD>1</PERIOD>

<RATE>100</Rate>

<TREENODESTATUS>1</TREENODESTATUS>

</INTERACTION>

<INTERACTION>
<NAME>Self Destroy</NAME>
<ID>3167510831114141238772</ID>

<POPULATION1>
4277108311123949756424<
/POPULATION1>

<POPULATION2>(None)</POPULATION2>

<INTERACTIONTRIGGER>403010881970451231240</INTERACTIONTRIGGER>

<TYPE>128</TYPE>

<PERIOD>1000001</PERIOD>

<RATE>100</RATE>

<TREENODESTATUS>1</TREENODESTATUS>

</INTERACTION>

<INTERACTION>

<NAME>Self Destroy</NAME>

<ID>502010831108001235156</ID>

<POPULATION1>10221108310840520635784<
/POPULATION1>

<POPULATION2>(None)</POPULATION2>

<INTERACTIONTRIGGER>
1522510831095341233896<
/INTERACTIONTRIGGER>

<TYPE>128</TYPE>

<PERIOD>1</PERIOD>

150
<RATE>5</RATE>
<TREENODESTATUS>1</TREENODESTATUS>
</INTERACTION>

<INTERACTION>
(NAME)Collide</NAME>
<ID>2595510881969711231240</ID>
<POPULATION1>32723108310862020915792</POPULATION1>
<POPULATION2>1510108310709247258880</POPULATION2>
<INTERACTIONTRIGGER>(None)</INTERACTIONTRIGGER>
<TYPE>16</TYPE>
<PERIOD>1</PERIOD>
<RATE>5</RATE>
<TREENODESTATUS>0</TREENODESTATUS>
</INTERACTION>

<INTERACTION>
(NAME)Collide</NAME>
<ID>403010881970451231240</ID>
<POPULATION1>427710831123949756424</POPULATION1>
<POPULATION2>2837110831115149706400</POPULATION2>
<INTERACTIONTRIGGER>(None)</INTERACTIONTRIGGER>
<TYPE>16</TYPE>
<PERIOD>1</PERIOD>
<RATE>5</RATE>
<TREENODESTATUS>0</TREENODESTATUS>
</INTERACTION>
</INTERACTIONS>
</MODEL>
FIGURES

1.1 A sample screen showing objects with random walk that collide in an invisible collection of finite volume, undergo proliferation, death and reactions with other population objects to transform and produce other objects. .............................................. 4

4.1 Cross-section in XZ plane of the heterogeneous space represented by the finite volumes. Enlarged finite volume shows the number of populations that it represents. .............................................. 24

4.2 This figure shows a two dimensional section of space. The center finite volume contains a population of objects that experience two processes: diffusion and chemo-attraction. Part A of the figure shows process of diffusion when multiple diffusion events occur, and the objects of the population leave the finite volume (in the center), and enter the finite volumes adjacent to the center finite volume. Part B of the figure shows the process of chemo-attraction, when multiple chemo-attraction events occur during the chemo-attraction process. Part C of the figure shows superimposed processes of diffusion and chemo-attraction. The effects of all processes are additive as the vectors of the movements due to diffusion and chemo-attraction are summed. ................................................................. 32
4.3 This figure shows processes that occur in one subsection. The problem is to figure out the state of the system for all sections in time $t$.

4.4 This figure shows processes of reaction, birth and death. Part A shows reaction between two populations. The reaction between objects occurs upon collision. In addition, not all collisions lead to reactions. There is a prior probability assigned to each reaction. This is analogous to having a reaction rate in chemical reaction equations. Part B of the figure shows process of birth in the finite volume. The process of birth is modeled as a property of population. Every population has this property, which is implemented as rate of introducing a new object given population size. Part C of the figures shows the process of death in population in the finite volume. Similar to birth rate, every population in the system has death rate.

4.5 This figure shows an example path out of many paths possible that the stochastic simulation algorithm will simulate. Due to the nature of the algorithm, every simulation will result in a new path, thus every simulation most likely be a unique simulation of the system of interacting populations. However, due to the large number of objects in the populations, the end result of the simulation is almost always identical (see details in text).
4.6 This figure shows the end results of an example simulation system provided in the Appendix C. As seen, the system arrives to 4 different end results, which can be possible course of treatment of a particular disease. .................................................. 46

4.7 Different stages of infiltration and demalination in the model. ... 54

4.8 Experimental curve of the average result of 20 simulations: the variance of the number of particles of all types and the simulations. Series 1, 2, 3 are cells of type A, B, C respectively. ...................... 56

4.9 Theoretical curve of the variance of the number of particles of all types and the simulations. Series 1, 2, 3 are cells of type A, B, C respectively. .............................. 57

A.1 This figure is a screenshot of the dialog that defines the population properties. .................................................. 112

A.2 This figure is a screenshot of the dialog that defines the size dynamics. 113

A.3 This figure is a screenshot of the dialog that defines the interactions. 116

A.4 This figure shows a screenshot of the dialog that lists all interactions that a populations takes part. ............................. 116

A.5 This figure shows a screenshot of the dialog that lists all interactions that a populations takes part. .......................... 117

A.6 This figure shows the finite volume mesh superimposed onto the vascular, endothelial and interstitial space. ....................... 118
A.7 Upper left section of the figure represents a snapshot of the Brusselator model when population B prevails and upper right figure represents when the same population B is depressed. The lower section of the figure shows dynamics of population size.

Figure
REFERENCES


[33] Gani, J., Two types of infections among homogeneous IVDU susceptible., MAI, Canberra ACT 0200, Australia

[34] Gani, J., Random Allocation and Urn Models., MSI, Canberra ACT 0200, Australia


[58] Peliaeva, N. B. , analytical study of Markov model of reaction of consistent type, Moscow State Technical University, 2002.


[72] Tolle, Dominic P. and Nicolas Le Novre, Particle-Based Stochastic Simulation in Systems Biology., Current Bioinformatics, 2006, 1


[77] D. S. Skundric, Cai J, Cruikshank WW, Gveric D, Production of IL-16 Correlates with CD4+ Th1 Inflammation and Phosphorylation of Axonal Cytoskeleton in Multiple Sclerosis (MS) Lesions, Journal of Neuroinflammation, http://www.jneuroinflammation.com/content/3/1/13

[78] A Abulrob, H Sprong, P Van Bergen en Henegouwen, D Stanimirovic, The blood-brain barrier transmigrating single domain antibody:


[81] **Desmond J. Highman Modeling and Simulating Chemical Reactions**


[85] **Boryczko K, Dzwiel W, Yuen DA Modeling Heterogeneous Mesoscopic Fluids in Irregular Geometries using Shared Memory Systems Molecular Simulation, 31/1, 45-56, 2005**

[86] **M. M. Dupin, I. Halliday, C. M. Care A multi-component lattice boltzmann scheme: Towards the mesoscale simulation of blood flow Medical engineering and physics, 28 (1), 13-18**

[87] **P. J. Hoogerbrugge, J. M. V. A. Koelman Simulating Microscopic Hydrodynamic Phenomena with Dissipative Particle Dynamics Shell Research B.V. - P.O. Box 60, 2280 AB Rijswijk, The Netherlands**
