A-Cell

True Cell Simulations Inc.

Introduction to A-Cell

—Functional Description—

Document version：Ver1.2.2E
Release：July 22, 2016

http://engl.tc-simulations.com/
Questions：q@tc-simulations.com

True Cell Simulations Inc. Tokyo Japan

CONTENTS

How to read this document

All functions of A-Cell are described in this document. They might not be used in your simulation, or there might be functions that you are not interested in. In these cases, you can skip those functional descriptions in this document. For example, there are descriptions on membrane potential and action potential, which are important for neuronal and cardiac muscle cells. In other cells, however, membrane potential might not be the major focus of modeling and simulation. In this case sections for membrane potential can be skipped.

One may want to know quickly how A-Cell model is constructed before reading this document. To this purpose, a movie of the constructing Michaelis-Menten enzymatic reaction and its simulation by A-Cell is uploaded on YouTube (YouTube https://youtu.be/IdwjjmXJl1s). You can know a part of A-Cell functions within about 6 minutes. A-Cell model used in the movie (MM_Reaction.Csim) is downloadable from our homepage, and a simulation can be run by the free edition of A-Cell, which is also downloadable from our homepage.

The biggest feature of A-Cell is its capability of spatio-temporal 4D simulation by constructing spatial 3D shape model to which biochemical reactions, stimuli, and other models are embedded. 4D simulation is essentially different from traditional pathway simulation, which focuses temporal change of concentrations. An example of the difference between temporal and 4D simulation is shown in a later chapter of this document. We first describe temporal simulation by A-Cell, and next 4D simulation is described.

A-Cell provides four editions including a free edition, which is freely downloadable from our homepage. Functions in lower editions are limited. In this document, all functions available in A-Cell are described. Available functions for each edition are listed on the last pages of this document, which will help you to select editions.

For those who want to know an example of 4D cell simulation, modeling and simulation of the initial step of cancer invasion in 3D space is described in the last chapter. You can find an example of modeling by chemical reaction starting at pathway diagram, and simulation results and a prediction that are obtained only by 4D simulation. This example helps you to learn what you can obtain by 4D simulation.

This document is described assuming Windows OS. Operations for A-Cell in Windows can be interpreted into other OSs (e.g. Mac OS).

Introduction

A-Cell is software for cell simulation. A-Cell employs chemical reactions to express a signal transduction, and it has a capability to construct 3D morphological models enabling spatio-temporal 4D simulation. To run cell simulation, typically you need following four steps: 1) constructing a model, 2) generating a simulation program and initial conditions, 3) running simulation on a computer, and 4) viewing and analyzing simulation results. Among these, steps 2) and 3) are done by A-Cell almost automatically with minimum operations by users. In addition, users can go through all these four steps by GUI[1] of A-Cell. Thus A-Cell is easy-to-use software.

Four steps for running simulation are briefly explained below:

1) In the step of model construction, signal transduction networks are constructed as chemical reactions. For example, when a cell receives a stimulus from outside, this triggers the activation cascades in the protein network. A modeling by chemical reaction is a direct consequence of this scientific picture of signal transduction in a cell. Since A-Cell employed GUI for constructing chemical reactions, users can construct a model as if they are drawing chemical reactions on a paper. In many research papers of wet experiments, a pathway diagram is drawn as a summary of a study. Although this cartoon is shown in abstracted form of chemical reactions in many cases, these are helpful for the construction of a model, and detail chemical reactions can be drawn by adding existing knowledge. Thus, one can construct a detail model starting at a pathway diagram. This way of modeling will not be difficult for researchers who studied molecular cell biology and/or biochemistry[2]. In addition, A-Cell enables users to construct structured chemical reaction models, in which many reactions related to a protein in focus are connected as reaction diagram, and the relationship between reactions are easily understood when one looks at this diagram. Importance of structured chemical reactions is described later.

2) In the present curriculum of molecular cell biology, computer programming for the simulation of a chemical reaction network is not taught. So most molecular cell biologists do not know how a computer program for the simulation can be written. But don't worry this, because A-Cell automatically generate simulation program from chemical reactions. The work you require is to draw chemical reactions on A-Cell. Then A-Cell generates a simulation program written in c language, which is the most common computer language used from note book computer to world fastest super computer and is running on Windows, Mac OS, and Linux. To run a simulation program, initial conditions such as simulation time, calculation step, output symbols (e.g. name of

[1] Graphical User Interface

[2] One may know a method to transform a chemical reaction to ordinary differential equations (ODE). ODE is another method of model construction, and in fact, many models are constructed by using ODEs. However, a list of ODEs is very hard to capture overall structure of a model, and in addition, easily leads to mistake(s). Therefore, A-Cell employs modeling by chemical reactions.

proteins), etc. should be defined, which is also done by GUI of A-Cell. Thus the step of generating a simulation program goes through semi-automatically, and is an easiest step in A-Cell.

3) The third step is running simulation program. A simulation program automatically generated by A-Cell in step 2) requires compilation. You need to learn this operation. However, it is easy, and the operation is regular for different simulation programs. If you have a friend who knows c language, he/she will teach you how to compile and run simulation program only within 30 min. In addition, there is a much easier way of simulation using A-Cell, if the number of chemical reactions is not large. In this case, you can run simulation without generating and compiling a simulation program. A-Cell directly runs simulation by using “in situ simulation” function.

4) Simulation results, which are a set of numerical values such as concentrations of proteins, are saved as disk files. A-Cell reads a disk file and shows us results visually on A-Cell windows as a time course, 3D spatial distribution, and a movie. These are exported to disk files in common format for further analyses, and you can get high quality graphs using Excel, Origin, and other commercially available or free software.

Thus, A-Cell provides the easiest way for cell simulation. This document describes functions of A-Cell showing its capabilities. The actual operation for above four steps are described in separate document “A-Cell operation manual”, which is downloadable by those who purchased A-Cell. You will know how you can realize what you want in cell simulation by reading this document, “Introduction to A-Cell”.

1. A basis of Simulation: Chemical Reactions and Differential Equations

A chemical reaction can be converted into differential equations, and differential equations can be converted back to chemical reactions. Thus, these are two different ways of expression of a model, and these are interchangeable. This is the basis for pathway simulations and also for A-Cell. Differential equations are difficult to capture overall structure of a model if you are not a mathematician. However, it is important to know how a chemical reaction is converted to differential equations for deep understanding of the simulation.

Let's start by the following simple reaction:

$$A \xrightarrow{k} B \qquad 1)$$

This is a first order reaction, in which molecule A is converted to B by the first order kinetics. This reaction indicates that the rate of the decrease in A is proportional to its own concentration, and *k* is the constant of proportionality. B increases by the same rate as the decrease in A. The rate of the change in A is expressed mathematically by dA/dt, then we get the following differential equation:

$$\frac{dA}{dt} = -k \cdot A \qquad 2)$$

A "-" on the right side of Eq.2) indicates the decrease in A. Larger *k* results in quicker decrease in A. The rate of the change in B is dB/dt, and is expressed by the same differential equation by deleting "-" on the right side in Eq.2) indicating the increase in B.

This example clearly shows that a chemical reaction is converted to a set of ordinary differential equations (ODE). This is the basis for the automatic generation of differential equations by A-Cell. There are methods for calculating (i.e. numerical integration) of differential equations, and hence we can obtain time-evolved changes in concentrations. Any chemical reaction including complicated one with many species can be converted to a set of ODEs.

If chemical reactions are complex, however, the generation of ODEs by human leads readily to mistakes. So A-cell generates them automatically by pressing a button in the menu. This is quite helpful both for beginners and specialists. In addition, A-Cell users have no need to know how to generate ODEs. But this knowledge will be powerful for analyzing simulation results in some cases[3]. Refer to textbooks for further discussion of chemical reactions and ODEs[4].

[3] See Ohshima, D., et al., PLoS ONE, 2015 for an example.

[4] There are many text books describing relations between chemical reactions and ODEs.

2. A-Cell Main Window

Figure 1 shows A-Cell main window.

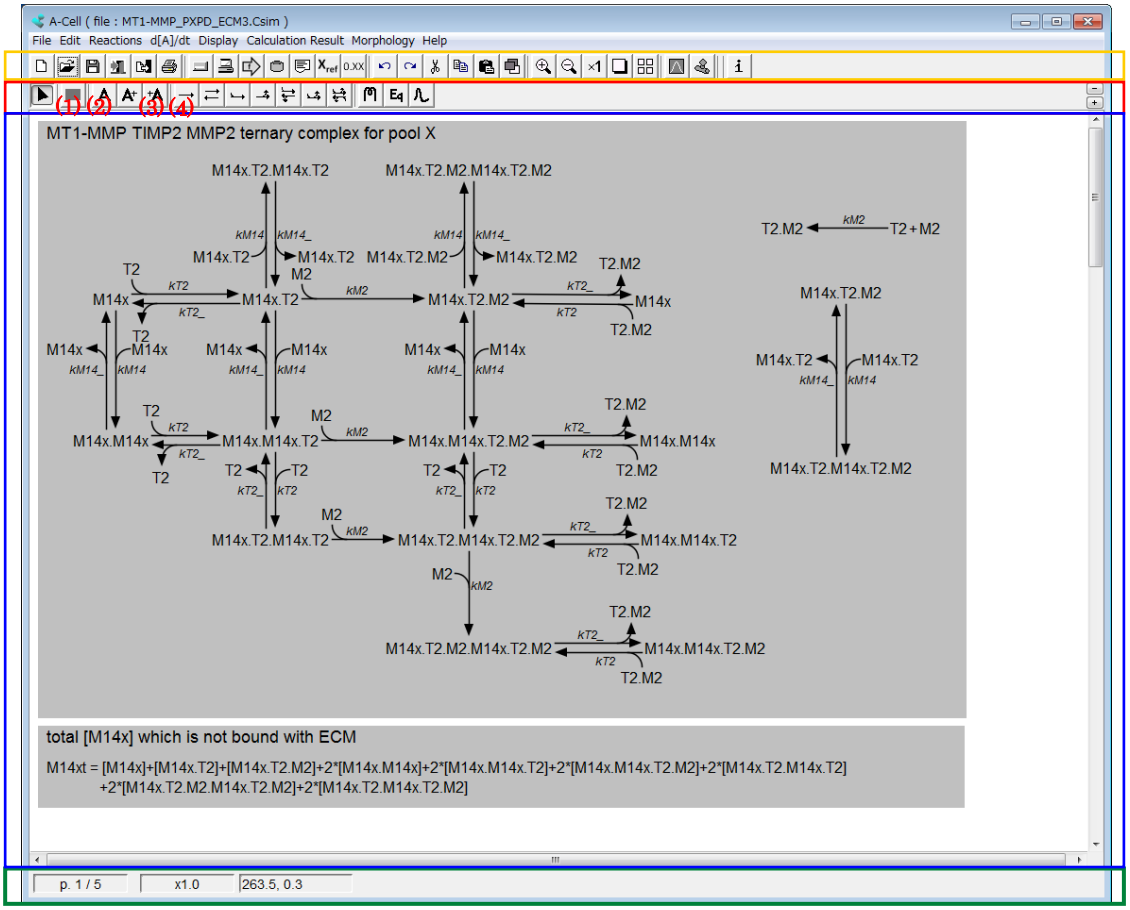


Figure 1 A-Cell main window.

Models such as chemical reactions are constructed and shown in the area indicated by a blue rectangle. Structured chemical reactions (in larger gray rectangle) are shown together with a model of equations (in smaller gray rectangle). The reactions show regulation of MT1-MMP (M14x in the model), which is a membranous protein that degrade ECM (extracellular matrix) proteins. Endogenous inhibitor TIMP-2 (T2) of MT1-MMP and another ECM-degrading protein MMP-2 (M2), which is activated by MT1-MMP are included in the model. The structured chemical reactions shown in Fig.1 are not employed for visualization purpose only. A-Cell users “draw” structured chemical reactions exactly the same format as shown in Figure 1 when constructing a model by A-Cell.

A gray area is called a “group”, and there are groups of “MT1-MMP TIMP2 MMP2 ternary complex for pool X” and “total [M14x] which is not bound with ECM” in Figure 1. It is the user’s decision what reactions are included in a single group. However, reactions included in a group

should be carefully selected, because embedding models to 3D shape is done by choosing a group as a unit of embedding. Embedding a part of reactions in a group is not allowed.

An equation model is constructed as shown in the lower gray rectangle in Figure 1. In this example, total concentration of MT1-MMP and its complex that do not form complex with ECM is calculated for the purpose of analyzing the model. If you write equations calculating MT1-MMP and its all complexes, you will be able to assure conservation of MT1-MMP during simulation. A-Cell allows you to write differential equations as an equation model. In case that the phenomenon is not expressed by chemical reactions, you can construct a model by differential equations. If you want, differential equations instead of chemical reactions are used for describing a pathway model

Menu buttons for constructing a model are shown in red rectangles of Figure 1. A detail usage of these buttons is explained in a separate document "A-Cell operation manual". But a simple example is shown here. Let's assume the following reaction:

$$A + B \xrightarrow{k} A:B \qquad 3)$$

A:B indicates a complex formed by the binding of molecules A and B[5]. To construct a reaction shown in Eq.3), first select button (1) in the read rectangle of Figure 1, and drag mouse to draw gray group area for Eq.3). Then put symbol "A" for molecule A using button (2). Select button (3) to construct "A+B". Put symbol for a complex "A:B". Finally select button (4) for connecting symbols "A+B" and "A:B". You can construct Eq.3) only by these four steps. See "A-Cell operation manual" for these operations in detail[6].

General operation menu buttons are shown in a yellow rectangle in Figure 1. These include buttons for start of a model construction[7], read and save of a model, generation of a simulation program, in situ calculation, construction of 3D shape, read and graphic display of simulations results, and etc.

Green rectangle at the bottom in Figure 1 is used for showing various information including page number, magnitude of display (i.e. enlargement or reduction), and coordinates of the mouse cursor. A-Cell allows multiple page model enabling constructing a large model[8].

There is an important rule in A-Cell, where molecular species are discriminated by their names. The same molecular name appeared in the different groups is recognized as the same molecular species. For example, if Eq.3) is drawn in a group and Eq.4) is drawn in a different group, then species "A" is recognized as the same species in A-Cell.

---

[5] There is no rule for expressing a complex. Instead of A:B, A.B or AB is used.

[6] You can also see operations in YouTube (https://youtu.be/IdwjjmXJl1s).

[7] A-Cell also allows reading multiple model files and copying groups between model files.

[8] There is a comment in a review article that A-Cell is a tool for beginners and large model cannot be constructed. The author of this review totally misunderstood the capability of A-Cell as you have realized here.

$$A \xrightarrow{k_{\mathrm{deg}}} A_{\mathrm{deg}} \qquad 4)$$

Thus “A” decreases both by a second order and a first order reactions according to Eqs.3) and 4). A-Cell generates a combined differential equation for both reactions as shown below:

$$\frac{dA}{dt} = -k \cdot A \cdot B - k_{\mathrm{deg}} \cdot A \qquad 5)$$

Thus, users do not need to worry about combining differential equations about a symbol that appears in different groups. These are automatically combined together by A-Cell generating a single differential equation. This rule is true for all model groups including equation group and others. There is an equation group “total [M14x] which is not bound with ECM” in Figure 1 calculating a summed concentration of ECM-free MT1-MMP (M14xt). The same names as used in a chemical reaction group “MT1-MMP TIMP2 MMP2 ternary complex for pool X” appear on the right side of the equation. The rule of A-Cell described above enables calculating M14xt by using M14x, M14x.T2, etc. whose concentrations are derived by chemical reactions above.

3. A big feature of A-Cell: Structured Reaction Diagram

Many reactions are involved in a model. There are many ways of expressions for these complex reaction diagrams. Relatively simple examples are shown in Figure 2.

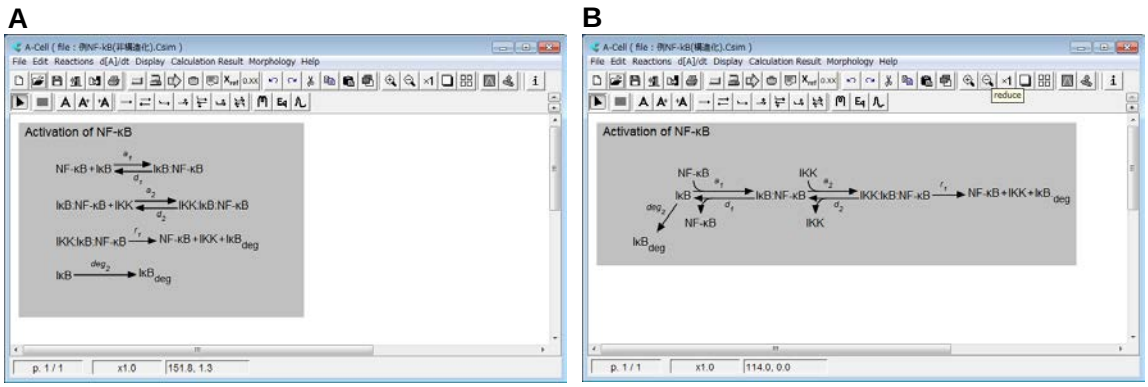


Figure 2 Different two expressions for the same reactions. Discrete (A) and structured (B) expressions.

There are two expressions, discrete (Figure 2A) and structured expressions (Figure 2B), for the activation of a transcription factor NF-κB, which regulates gene expression profiles of many proteins. Briefly, NF-κB, which is well known in innate immune response, is activated by IKK protein. In the absence of stimulus, NF-κB is bound with IκB, which is an inhibitory protein of NF-κB, forming a NF-κB:IκB complex. By the activation of IKK, however, IKK binds to NF-κB:IκB forming IKK:NF-κB:IκB complex leading to the degradation of IκB and activation of NF-κB, which is free from IκB. This leads to the regulation of gene expressions. While the example shown in Figure 2 is a part of the whole picture of regulating NF-κB, the importance of structured reaction diagram is easily recognized if one compares Figures 2A and B describing the same reactions. The reactions from the IKK activation to the generation of free NF-κB are a simple cascade reaction. In Figure 2A, the cascade reaction is separated into four reactions. Because of this unstructured expression, it is not easily recognized that these four reactions form a simple cascaded reaction. In contrast, the expression in Figure 2B is very easy to recognize it. These two expressions generate exactly the same differential equations, and hence give us the same simulation results. However, the expression of Figure 2B is far more understandable. This is very important in the analysis of simulation results. When we try to find reason(s) why we have obtained those simulation results, we refer to the original model. In this case, structured expression is much easier in obtaining a hypothesis, and thus much easier in finding reason(s). A-Cell allows both expressions shown in Figures 2A and B. However, one advantage of A-Cell is to allow users to express models as shown in Figure 2B, and we recommend strongly constructing models using expression as shown in Figure2B.

4. System Requirements, Install/Uninstall, and the First Execution

The performance of CPU and the capacity of main memory and HDD are still increasing. So A-cell can be run on any computer, and simulation can be run even on a notebook computer. However, the resolution of a display is important for ease of use. We recommend 1920x1080 or higher. OS can be Windows, Mac OS, or Linux. However, Java is required for running A-Cell[9]. Connection to the internet is required for running A-Cell except the free edition.

Installation is easy. Download and unzip the installation package, and then just double click A-Cell.jar. Then A-Cell will start. To uninstall, simply delete the folder where A-Cell download files exist. When you first run A-Cell, a window shown in Fig.3 appears after license authorization process. (The license authorization process does not run in the free edition.) You are requested to set user name and affiliation. The information can be modified at any time. But the correct information is very important because it is used for the authorization of the model you have constructed. Detail information will be shown in the next section.

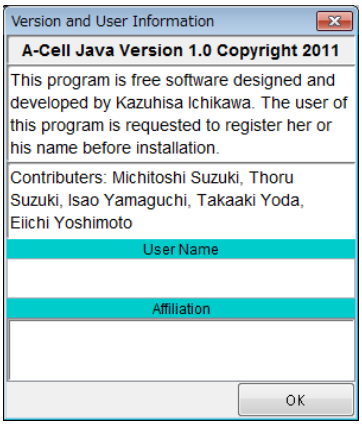


Figure 3 A window appeared after the first running of A-Cell.

5. Cyphering A-Cell Model File

A-Cell model files are cyphered. In cell simulations, complicated reaction networks of many proteins are described in a large model. In such cases, constructing a large model only by one researcher is unrealistic, and collaborative construction and combining models by many researchers will be required. To realize this, model files constructed by other researchers should be copied. A-Cell enables copy of groups between different model files. On the other hand, the originator information describing the name and the affiliation of the researcher who constructed the model group should be kept in a model group when it is copied, and should not be modified. To enable these, the information is embedded to every group and A-Cell models are saved by cyphering all data. The embedded originator information to a group is what you entered by a window shown in Figure 3. The original information can be viewed in the group property window, but there is no means to modify it. These functions of A-Cell ensure two contradictory requirements of free copy and keeping originator information.

[9] Java 8 Update 66 or higher is recommended for running A-Cell. In almost all PCs, Java is preinstalled. If you are not sure, check if Java is installed in your computer by visiting at https://java.com/en/download/installed8.jsp. If not installed, visit http://java.com/en/download/windows_xpi.jsp for installation.

6. Constructing Models by A-Cell

In this chapter, we show what models are constructed by A-Cell. You can construct following types of models.

1) Chemical reactions
2) Stimuli
3) Equations including ordinary differential equations
4) Membrane potential including action potential by Hodgkin-Huxley equations

6.1 Chemical reactions

6.1.1 What type of reactions can we model?

We have already shown several types of chemical reactions in Chapters 2 and 3. In addition to these, various types of reactions can be modelled in A-Cell by using following 7 buttons (Figure 4).

(1) (2) (3) (4) (5) (6) (7)

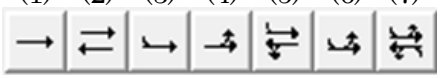


Figure 4 Seven buttons constructing biochemical reactions.

By using button (1), you can construct the following type of reactions:

$$A \rightarrow A^* \qquad 6)$$

$$A + B \rightarrow A:B \qquad 7)$$

$$A:B \rightarrow A + B \qquad 8)$$

In Eq.6), molecule A is converted to A^* by a first order reaction, which is the same type of reaction as Eqs.1) and 4). Eq.7) is a binding reaction of molecules A and B forming a complex A:B. This is a second order reaction[10]. Eq.8) is a dissociation reaction of a complex A:B generating molecules A and B.

By using button (2), following equilibrium reaction with forward and backward reactions is constructed:

$$A + B \rightleftarrows A:B \qquad 9)$$

Eq.9) can be written as follows:

[10] Theoretically third order or even higher order reactions can happen, and simultaneous binding of even more species can happen. However, probability to proceed third or higher order reaction is thought to be very low, because three or more species should collide simultaneously. It is natural to resolve third order reaction A+B+C→A:B:C into a set of second order reactions of A+B→A:B, B+C→B:C, C+A→C:A, A:B+C→A:B:C, A+B:C→A:B:C, and C:A+B→A:B:C. Although third order reaction can be constructed in A-Cell, great care is required for its usage. Note that A:B:C and A:C:B are recognized as different species by A-Cell.

$$A + B \rightarrow A:B$$
$$A:B \rightarrow A + B \qquad 9')$$

Eqs9) and 9') give the same simulation result. But it is obvious that Eq.9) is much easier to capture the overall structure of the reaction. So, we recommend strongly to use Eq.9) instead of Eq.9').

Button (3) is used to construct the same binding reaction as in Eq.7), but in the different form of expression as follows:

$$A \xrightarrow{B} A:B \qquad 10)$$

Button (3) enables you to construct structured reaction, and a reason why button (3) is useful for structured reaction is shown below. Left and right sides of an arrow of a reaction are recognized as a whole in A-Cell. For example, if left side is A+B, then the reaction originated from this should be complex (A:B) forming as an inevitable consequence. This means that a reaction originated from the left side A+B leading to degradation of only B without any change in A is not allowed (Figure 5).

Figure 5 Correct and false reactions.

The same is true in A+B→A:B+C→A:B:C. One may intended to form complex A:B from A and B, and A:B binds with C forming another complex A:B:C. However, reaction A+B→A:B+C results in the formation of A:B and C only by the binding of A and B, which is very curious. To avoid this misled modeling, one should construct these reactions by the following separated reactions if one uses button (1):

$$A + B \rightarrow A:B$$
$$A:B + C \rightarrow A:B:C \qquad 11)$$

This expression is not good in terms of understandability of reaction structure. By using button (3), however, one can draw these as follows:

$$A \xrightarrow{B} A:B \xrightarrow{C} A:B:C \qquad 12)$$

Eq.12) is far more understandable, and we can capture overall structure of reactions. Thus, button (3) is useful for this type of cascaded reaction.

Button (4) is used for dissociation reaction. An example is shown below:

$$A:B \xrightarrow{B} A \qquad 13)$$

This is also useful for cascaded reaction as in button (3).

Button (5) enables you to construct equilibrium reaction combining buttons (3) and (4) as follows:

$$A \underset{B}{\overset{B}{\rightleftharpoons}} A:B \qquad 14)$$

Figure 2B was drawn using button (5).

Button (6) is used for simplified Michaelis-Menten-like enzymatic reaction as follows:

$$A \xrightarrow{X} A' \qquad 15)$$

This example shows an enzymatic reaction converting A into A' by an enzyme X. Since there is no change in X, differential equations for X is not generated by A-Cell.

Button (7) is used for simplified forward and backward enzymatic reaction as follows:

$$A \underset{Y}{\overset{X}{\rightleftharpoons}} A' \qquad 16)$$

A is converted into A' by an enzyme X, and A' is converted back into A by a different enzyme Y.

Reactions shown above are drawn horizontally. However, reactions can be drawn vertically or obliquely. This capability of A-Cell allows you to draw many reactions from a single species as shown in Figure 6.

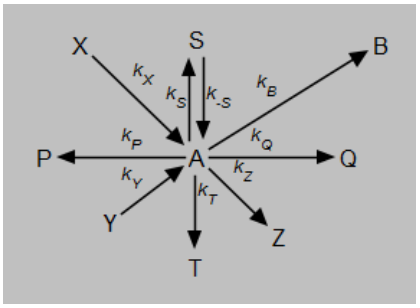


Figure 6 Many reactions with respect to molecule A can be drawn starting from and/or converging to a single species "A".

It is easy to draw reactions shown in Figure 6. Users first determine the position of molecule P or X

relative to A for example, and next connect between P or X by reactions. If P is aligned horizontally to A, then resulting reaction is drawn horizontally, and if X is aligned obliquely, then resulting reaction is drawn obliquely. The position can be changed by drag operation of a mouse. This allows you to repositioning and realignment of any symbol after the construction.

6.1.2 Setting and changing parameter values

Parameter values such as concentration, rate constants, etc. are specified and changed by symbol property windows (Figure 7).

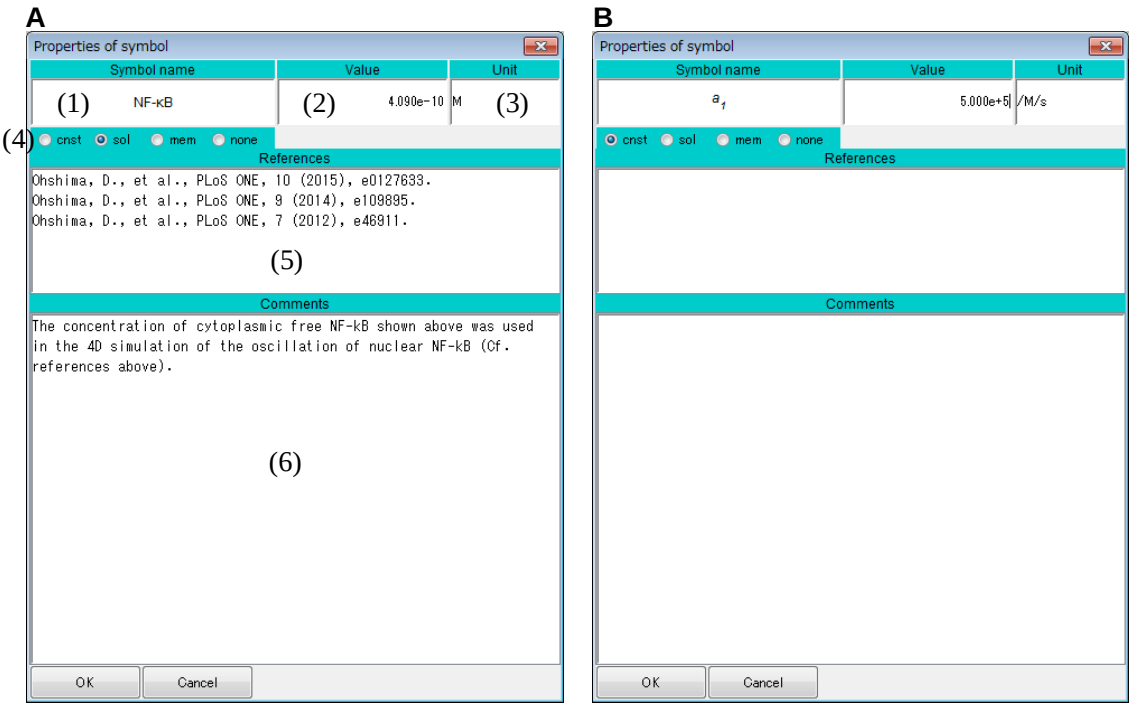


Figure 7 Property windows for NF-κB (A) and a rate constant a_1 (B).

These are examples of property windows for NF-κB and rate constant a_1. Textbox (1) shows a symbol name, which is modified by clicking the textbox. Textbox (2) shows values of the symbol, which are concentration for NF-κB and rate of the reaction for a_1. These are altered by directly clicking the textbox. Textbox (3) shows unit of the symbol. Although setting of unit has no effect on the simulation, it is important for recognizing that the model is properly constructed using the same system of unit. Radio buttons in (4) are used to indicate solubility of the symbol. These include “const”, “sol”, “mem”, and none”. NF-κB is a soluble protein, and then “sol” is selected. a_1 is a rate constant, and then “const” is selected. Symbols indicated by “const” are kept constant during the simulation. If NF-κB is selected as a “const” symbol, its concentration is not changed during the simulation, which might be helpful in some cases for analyses of simulation results.

Textboxes (5) and (6) are used for references and comments, respectively. Several different

values in a concentration might have been reported, or a value can be estimated by combining several reports when there is no direct report describing the value. In these cases it is important to clarify the bases for the value. Using these textboxes, you can record the bases. Setting and changing in (1)-(6) are influential for symbols having the same name appeared on different groups.

6.1.3 Group property

Figure 8 shows a group property window. By clicking textbox (1), the name of the group is changed. Textbox (2) is used for comments of the group, which might be used for indicating the explanation of reactions in the group. Textbox (3) show us the originator name and affiliation indicating who the first was to construct the model shown in the group. The information cannot be modified.

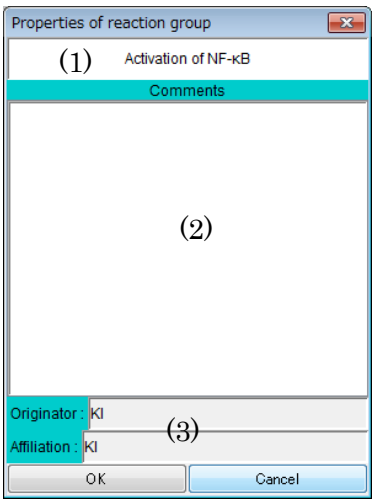


Figure 8 Property window for a group.

6.2 Stimulus

Cells receive various stimuli such as chemicals. Electrical stimulus might be applied for measuring responses of neurons. Therefore, modeling these stimuli is mandatory for simulating a cell. A-Cell allows you to construct stimuli as a model groups by a button surrounded by a red square in Figure 9.

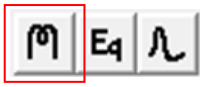


Figure 9 Button for constructing stimulus model (red square).

An example of stimulus model is shown in a group surrounded by a red rectangle of Figure 10.

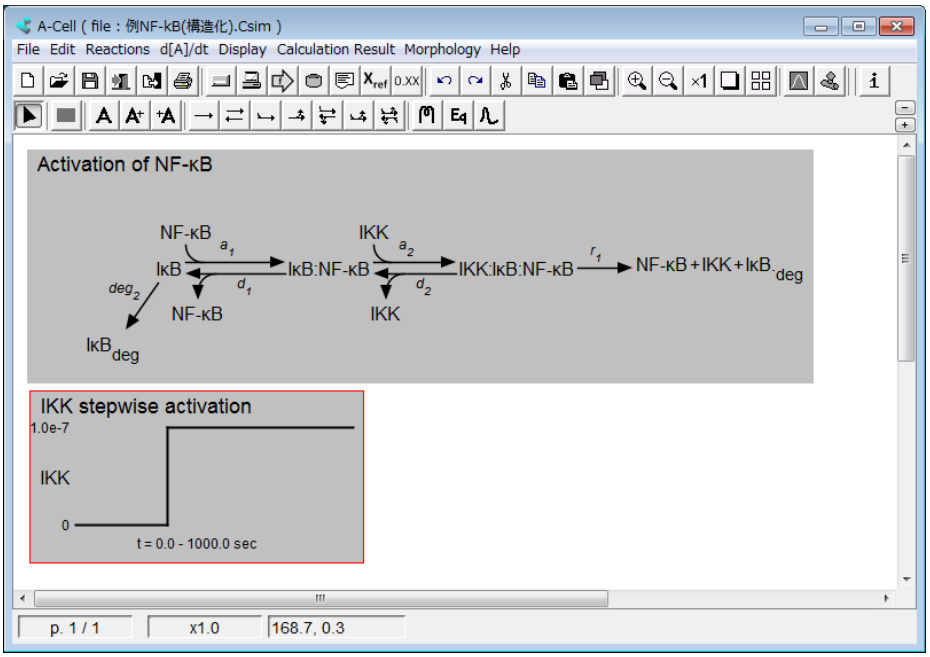


Figure10 An example of stimulus model (red rectangle).

In this example, the concentration of protein IKK jumps up by step function from 0 to 100 nM at time 0 s lasting for 1,000 s. As described previously, the same symbol name is recognized as the same symbol even if it is described in a different group. Therefore, IKK in the reaction group "Activation of NF-κB" and that in stimulus group "IKK stepwise activation" is recognized as the same symbol by A-Cell, and the concentration of IKK in the reaction group is also jumps up by step function leading to the activation of NF-κB at time 0 s.

This rule is true for all groups including membrane potential group. Let's assume that there is a symbol name I_{Inj} in a stimulus group, and a membrane potential group is constructed including the same symbol name I_{Inj} as injection current for electrical stimulus. By the rule of a symbol name in A-Cell, symbol I_{Inj} in the stimulus group and in the membrane potential group is recognized as the same symbol, and behaves exactly the same manner[11]. Thus, if I_{Inj} in the stimulus group changes in a step wise function, that in membrane potential group changes in the same manner. This rule for symbol in A-Cell is important for constructing a model by A-Cell.

There is a property window for stimulus group, by which users can change and set parameters for the stimulus including mathematical functions of stimulus waveform. Thus, step function shown in Figure 10 can be changed to pulse, sinusoidal, ramp, and other functions (Figure 11).

[11] Such symbol is called "global symbol" in programming. Thus, all symbols in A-Cell are global symbols.

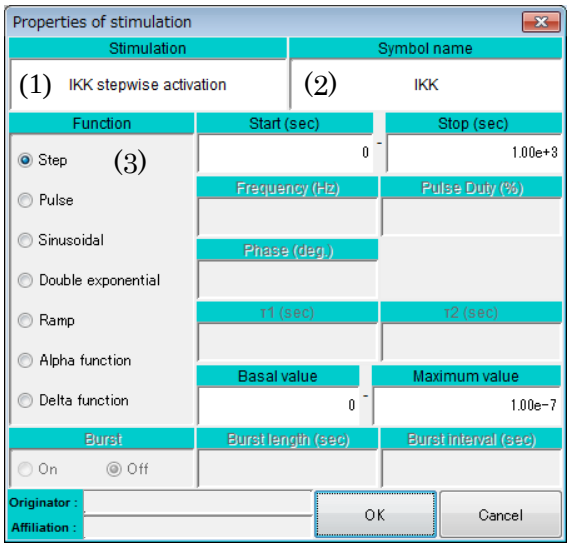


Figure 11 Property window of a stimulus group.

Stimulus group name, symbol name, and mathematical function can be set and changed by (1), (2), and (3), respectively. You may know mathematical functions in (3). These are explained in "A-Cell operation manual" in detail. Multiple stimulus groups can be constructed in a single model file.

6.3 Equation

During simulation studies, one may need calculations in addition to ODEs of chemical reactions. For example one may want to calculate total concentration of several different proteins that have the same role in a chemical reaction network. Or simplified way of modeling by a single equation instead of detail and complex reaction network might be needed, because it is not the major focus of the study, and the simplification is judged not to affect focusing point of study seriously. In other case, one may want to make a model describing events that cannot be expressed by chemical reactions. In these cases, modeling by equation is required. Button surrounded by red square in Figure 12 is used to construct equation group for these purposes.

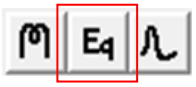


Figure 12 Button constructing equation model (red square).

An example of equation model is shown in Figure 1. Here property window of equation model group is shown (Figure 13).

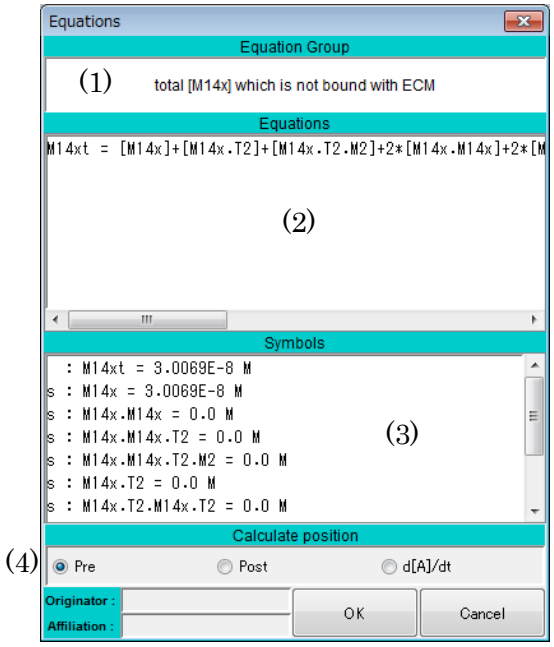


Figure 13 Property of an equation group.

As in property windows in other groups, group name can be changed by textbox (1). Equations are directly written in textbox (2). [M14x] indicates the concentration of M14x. In addition to four arithmetic operations (addition (+), subtraction (-), multiplication (*), and division (/)), other mathematical operator such as power (^), exponential function (exp), trigonometric functions (e.g. sin), etc. can be used. All symbols in the equation group are listed in textbox (3). You can modify values here directly. The change here is subjected to the same symbol name appeared in other groups. Type of equations in the group is defined by the radio button in box (4). If “d[A]/dt” is selected, all equations in textbox (2) are recognized as ODEs, which are numerically integrated by A-Cell simulation program. Thus you can construct a reaction network model by writing ODEs here instead of drawing chemical reactions in the main window of A-Cell. “Pre” and “Post” button are used to specify whether equations are calculated before or after the numerical integration. If you want to calculate symbol value before numerical integration, you should select “Pre” button. “Pre” or “Post” specification is applied to all equations written in textbox (2). See “A-Cell operation manual” for more detail.

6.4 Membrane potential and action potential by Hodgkin-Huxley equations

In neuronal cells, membrane and action potentials are important. A-Cell provides the modeling and simulation environment of membrane and action potentials, which are simulated together with chemical reactions. For example, transmitter molecule glutamate is released from the presynaptic terminals of a neuron, and bound with ion channels on a postsynaptic neuron. This binding results in the opening of the ion channel leading to the change in the membrane potential of a neuron. This

further causes opening of voltage-sensitive Ca^{2+} channels, and Ca^{2+} flows into a postsynaptic neuron, and leads to the activation of kinases and phosphatases resulting in the modification of various proteins. Thus, the change in the membrane potential is related to biochemical changes in a neuron. These are a part of the mechanisms of synaptic plasticity. Therefore, the simultaneous simulation of membrane potential and chemical reaction is essential in the simulation of neuronal cells. This will also true in simulating mitochondria. In this section, we show how membrane and action potential models are constructed using A-Cell. A method relating the change in the membrane potential to the change in protein activities is shown elsewhere.

It is common to use electrical equivalent circuits for the modeling of membrane and action potentials. Equivalent circuits for both of these potentials are basically the same (Figure 14). A difference exists in the alteration of membrane resistance due to the change in the membrane potential. In the modeling of membrane potential, in which no voltage-sensitive ion channels is involved, the membrane resistance is constant as shown in Figure 14A. In contrast, the modeling of action potential requires voltage-dependent membrane resistance (Figure 14B). You will find details of the theory in text books.

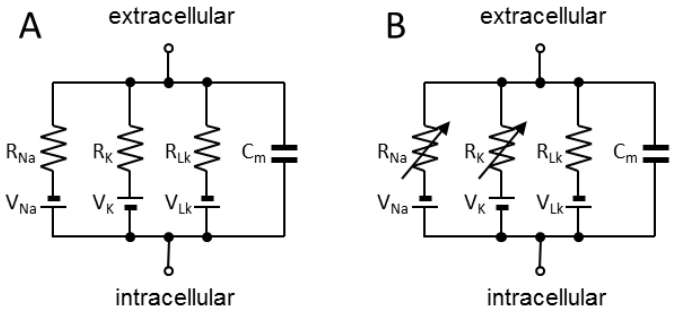


Figure 14 Equivalent circuit models of cell membrane potential.
(A) A model for membrane potential. (B) A model for action potential.

R_{Na} and R_K are membrane resistances for Na^+ and K^+, respectively, and R_{Lk} is for leak current through cell membrane. V_{Na}, V_K, and V_{Lk} are potentials for corresponding current, which originate from imbalances of corresponding ions between intra- and extracellular space, and C_m is membrane capacitance. The membrane resistances for Na^+ and K^+ in Figure 14B change according to a membrane potential which are described by complicated equations. In both cases shown in Figures 14 A and B, the membrane potential V_m is calculated by the following differential equation:

$$C_m \frac{dV_m}{dt} = I_{N_a} + I_K + I_{Lk} \qquad 17)$$

where I_{Na}, I_K, and I_K are current components for each ion species. In the simulation of action potential, Eq.17) is numerically integrated together with differential equations for each

voltage-sensitive membrane resistance. Thus, the modeling of action potential is not simple and is not easy from scratch.

In A-Cell, however, user does not need to describe differential equations, because A-Cell provides equations and differential equations for ion species and ion channels. A model for membrane and action potentials are constructed only by constructing equivalent circuits shown in Figure 14. All equations and differential equations required for the simulation including Eq.17) are automatically generated by A-Cell. To do this, use the button shown in red square in Figure 15.

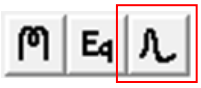


Figure 15 Button for constructing membrane and action potential model (red square).

Membrane potential model is also a “group”, and a group area should be prepared by dragging a mouse, and enter the group name. Then a window shown in Figure 16 appears. When it appears, textboxes (2), (4), and (5) are blank.

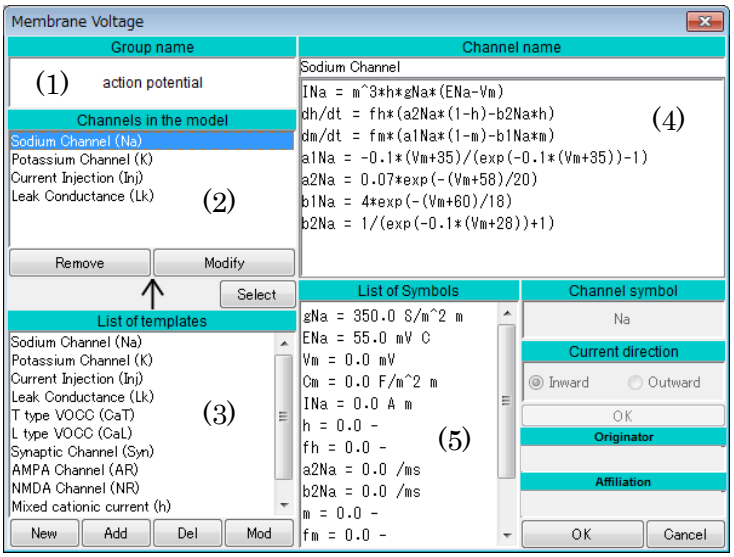


Figure 16 Property window for membrane potential group.

To construct membrane potential model, first select current involved in the equivalent circuit from textbox (3). Selected current component is added to textbox (2). Second, select current component in textbox (2) for modification. All equations and differential equations, and parameter values concerning the selected current component appear in textboxes (4) and (5), respectively. By changing equations and/or parameter values directly in these textboxes, users can construct original models. One such example, which is a Hodgkin-Huxley model, is shown in Figure 17.

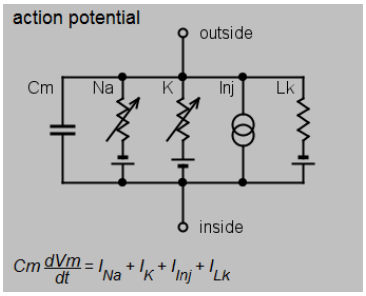


Figure 17 An example of model for action potential.

As shown here, users can construct Hodgkin-Huxley model only by selecting current component. Figure 17 appears on the A-Cell main window when you constructed a membrane potential model by a window shown in Fig.16. If there are chemical reaction “group” in your model, membrane potential is numerically integrated together with chemical reactions.

6.5 Saving a model and name of a model file

A-Cell model file is saved as a disk file with the extension Csim. Thus the model file is stored by xxx.Csim. xxx is a file name specified by a user. Don’t change the extension. As shown in the Chapter 5, the A-Cell model file is ciphered.

6.6 System of unit in A-Cell

In the last section of Chapter 6, the system of unit employed in A-Cell is described. SI unit is applied to A-Cell. Units used for each physical value are listed below:

Length: meter (m)

Weight (mass): kilogram (kg)

Time: second (s)

Current: ampere (A)

Temperature: kelvin (K)

Concentration: mole (M)

It should be noted that order of magnitudes is not expressed by using μ (10^{-6}), n (10^{-9}), M (10^{6}), T (10^{12}), etc. Instead, 1 μM is expressed as 1e-6 M for example. This is also true in the expression of length: 50 μm is expressed by 5e-5 or 50e-6 m in A-Cell. The expression in articles of molecular cell biology and simulation might be different from that employed in A-Cell, and if you utilize value in published articles without changing the system of unit and the order of magnitude, the simulation results will not be valid.

7. Generating Simulation Program and Running It

In previous chapters, we have shown what kinds of models are constructed by A-Cell. Model construction is an important step, but it's a first step. Simulation according to the constructed model should be run to obtain simulations results. We show here how this is accomplished.

7.1 Basics of computer simulation and numerical simulation in A-Cell

Computer simulation is a method to obtain results starting at initial conditions by applying rules that are executable by a computer. Rules are described by equations and/or algorithms. In the present version of A-Cell, numerical calculations are executed in a simulation program to obtain numerical solution of equations that are generated according to the model. Thus, A-Cell simulation program executes numerical integration of differential equations. Many methods for numerical integration are known including 4^{th}-order Runge-Kutta method, which A-Cell employs. This method manages both calculation speed and accuracy, and is most popular among many simulation methods. Although the algorism is more complicated than Euler method, which you might have studied in beginner's course of mathematics in universities, both computing speed and accuracy are much better in 4^{th}-order Runge-Kutta method than Euler method.

To run simulations by A-Cell, there is no need to know details of 4^{th}-order Runge-Kutta method. If you are interested in this, you can find many text books or information on the net. If you want to modify simulation program generated by A-Cell, however, you might have to know about it together with programming. Here we show how you can run simulations by A-Cell without any knowledge of 4^{th}-order Runge-Kutta method and programming.

7.2 Two methods of running simulations

A-Cell provides two simulation methods: one is running simulation by automatically generating simulation program by A-Cell, which we call "independent simulation", and another is running simulation using a module included in A-Cell without generating simulation program, which we call "in situ simulation" (Figure 18). There are pros and cons in "independent simulation" and "in situ simulation". Although independent simulation requires an additional step (compilation of a program) before running it, simulation runs much faster than "in situ simulation". In addition, the simulation can be run in parallel computers such as high performance computers and even super computers. In contrast, simulation speed in "in site simulation" is much slower. But no compilation is required, and you can run simulation only by A-Cell. You need not to install compiler software in your computer. Selecting these two simulation methods depends on the size of a model, and performance of computers you are using. If the model is described within one page and there is no morphological model, you can run simulation by "in situ simulation". In Chapter 13, the difference in simulation speed in an actual case will be shown. In 4D simulation described in later chapters, we need longer simulation time, and in site simulation is not allowed.

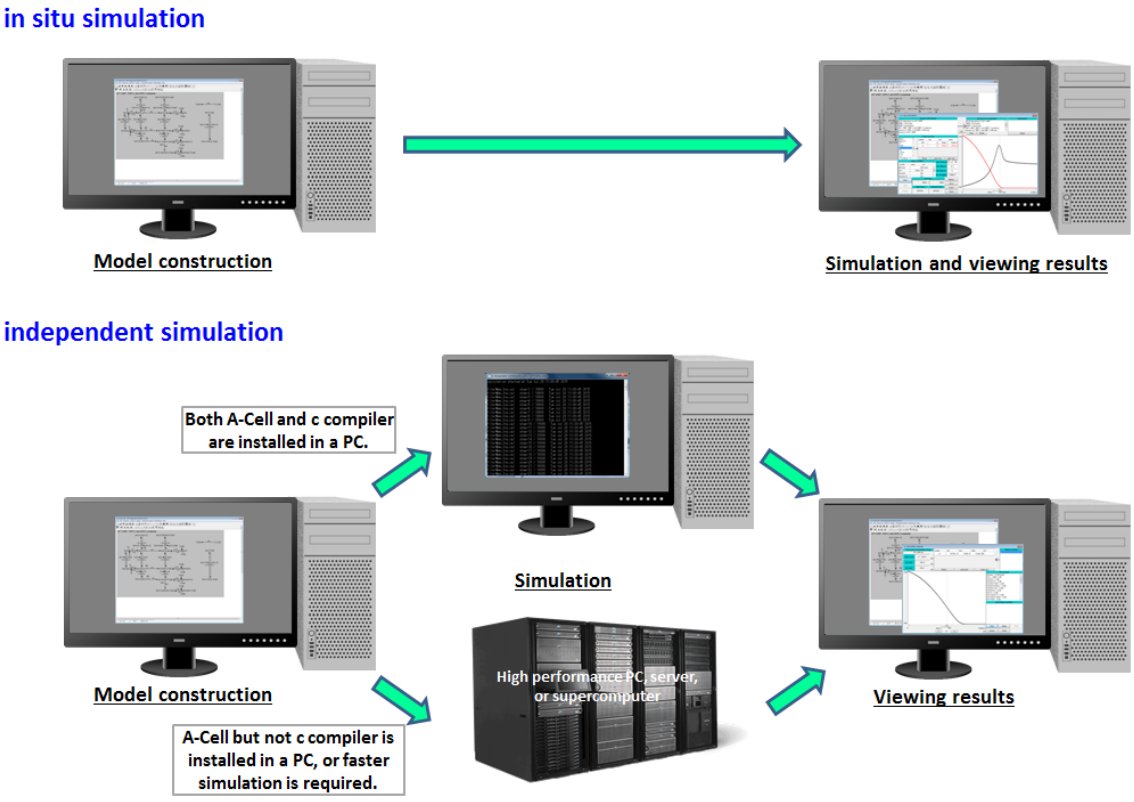


Figure 18 “in situ simulation” and “independent simulation”

In both methods of simulation, setting initial conditions is required. First we show how initial conditions are set.

7.3 Setting initial conditions

There are following four kinds of initial conditions:

1) initial values of symbols,
2) output symbols,
3) calculation time,
4) output file name.

These setting are common to two simulation methods of A-Cell.

1) Initial values of symbols: Values of concentration, rate constants, and parameters in stimulus and membrane potential model are 0 just after creation of a model[12] and nothing will happen by the simulation. These values should be set by property windows (Figures.7, 11, 13, and 16) before running simulations.
2) Output symbols: Time series values for symbol(s) that you specified are stored in xxx.cal file, and can be displayed later in “independent simulation”. The number of symbols in a model often

[12] There are some exceptions.

exceeds 100. In such case, you may want to know concentrations of proteins in focus, and you may not be interested in other proteins. Thus in A-Cell, you can specify symbols you are focusing. Calculated results for symbol(s) not specified here cannot be known. You can specify all symbols, or any symbols including constants (e.g. rate constants).

3) Calculation time: Period of simulation (simulation time), calculation step, which specifies iteration time step, and output step, which specifies output time step to a disk file are specified. The calculation step[13] is important in numerical integration, because if it is too large, error is large and in an extreme case overflow occurs. On the contrary, if it is too small, longer computational time is required. It is common to set calculation step to be very small value such as 10^{-6} s (1 μs)[14]. If calculated values of symbols are stored every 1 μs, large memory space in HDD will be required. This is not good both for reduction of file volume and reading time of a simulation result in a disk file to A-Cell. Thus users are requested to specify output step by an adequate value. If user specifies calculation time, calculation step, and output step to be 100 s, 10^{-6} s, and 10^{-1} s, respectively, one output event occurs every 10^5 iterations, and there are 1,000 outputs during the simulation[15].
4) Output file name: This specifies a name of a file, which is xxx.cal, storing calculation results. Since calculation results are stored as simple text data, they are read by 'note pad' of Windows, or by commercially available software such as Excel or Origin for further analysis. However, there are header and trailer information in a file, these should be deleted before importing to these software (Cf. Chapter 8).

Settings 1)-4) are required both for "in situ simulation" and "independent simulation" simulations.

7.4 in situ simulation

In this section, the first method of simulation, "in situ simulation", is described, which enables users to run simulations without generating simulation program using a module of 4th order Runge-Kutta calculation in A-Cell software. You can use in situ simulation by a button shown in red square in Figure 18.

Fig.18 Button for running in situ simulation (red square) .

Figure 19 shows the "in situ simulation" window.

13 Numerical integration starts from certain time (commonly from time 0), and increase small time step (iteration time step) to calculate values at the next time. The difference between the previous and next time depends on differential equation you are calculating.

14 The calculation step is model-dependent, and users are required to specify it.

15 Thus, change during a time smaller than 0.1 s cannot be detected in this setting. This is the same as sampling problem.

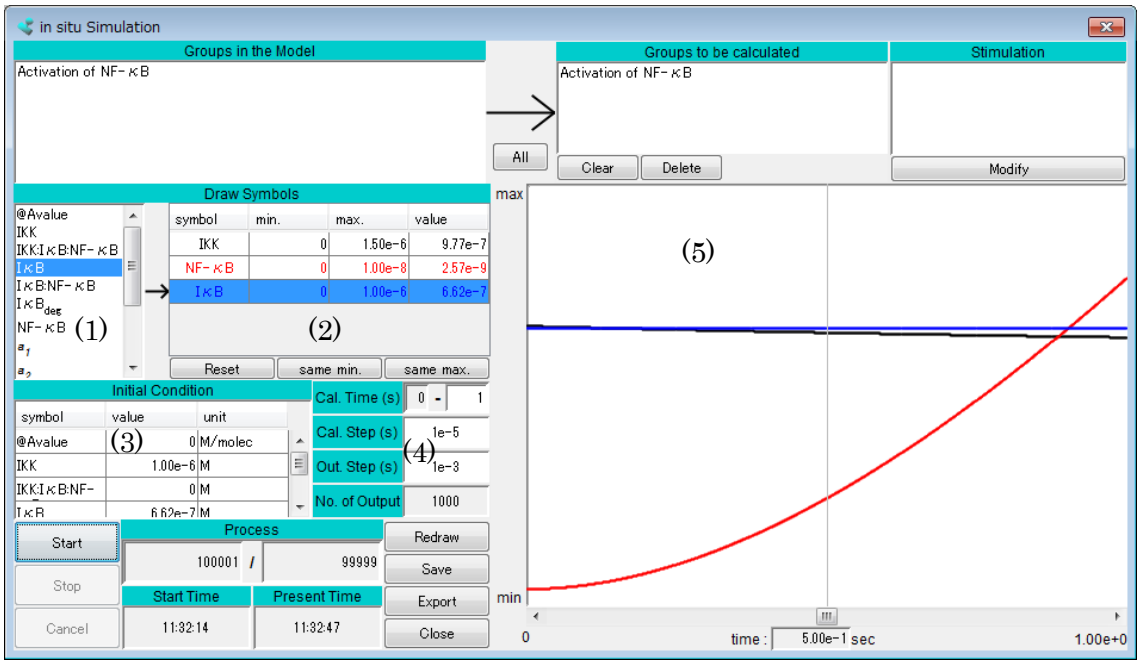


Figure 19 “in situ simulation” window.

Select output and display symbols from listbox (1), and set max. and min. values in textboxes (2) for graphic display shown in picture box (5), change initial values in textboxes (3) if needed, and set calculation time, calculation step, and output step in textboxes (4). By pressing “Start” button, a simulation starts and time courses are drawn in (5) as shown in Figure 19. This is a result of “in situ simulation” for the model shown in Figure 2B. If you move the cursor at the bottom of the picture box (5), values for each symbol are displayed in “value” textbox in (2). Thus, simulation can be run just after the completion of model construction without generating simulation program.

7.5 Independent simulation

7.5.1 Generating simulation program from differential equations

To run independent simulation, there are several steps. Use buttons shown in Figure 20.

Figure 20 Buttons used for independent simulation. Buttons in the left and right red squares are for generating simulation program and for setting initial conditions, respectively.

To this purpose, first use a button in the left red square of Figure 20. This automatically generates differential equations from chemical reaction, equation, and membrane voltage groups, combining all terms in differential equation with respect to a symbol. In A-Cell, simulation program is automatically generated. Thus, it seems that there is no need to show differential equations to a user. However, one may want to know differential equations for later analyses of simulation results. Or one may want to compare generated differential equations with chemical reactions. To this purpose, A-Cell takes two steps: first, generation of differential equations based on constructed model as shown here, and second, generation of simulation program based on generated differential equations.

By the button in the left red square of Figure 20, a window shown in Figure 21 appears. All group names in the model appear in a textbox (1). In this example, there is only one group name, because a model shown in Figure 2B is used. If one uses Figure 2A, resulting differential equations are the same as shown in Figure 21. Thus, even if the expression of chemical reactions is different, a resulting differential equation is same, and hence we obtain the same simulation results. To generate differential equations, first, select group(s) you want to generate from the textbox (1), and selected group(s) are listed in the textbox (2). Next, push "Generate" button, and you get differential equations in the textbox (3).

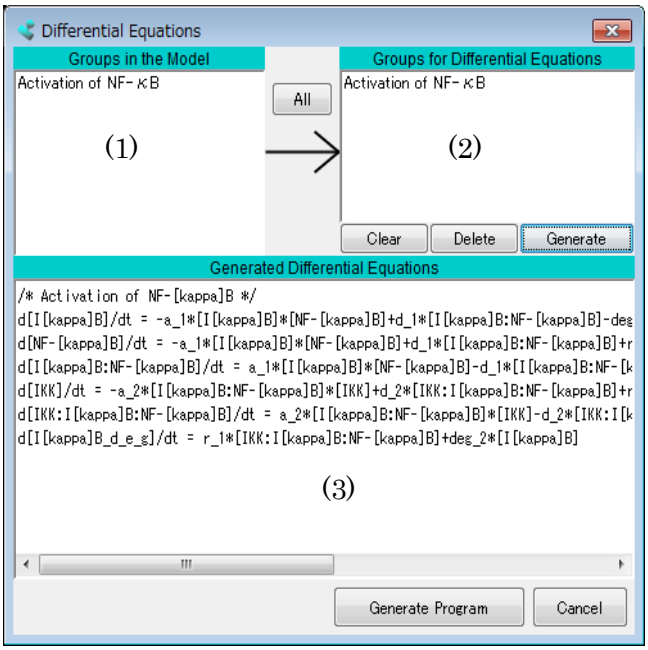


Figure 21 Window showing automatically generated differential equations from a model.

The generation of simulation program by 4th order Runge-Kutta method is realized according to a rule, and A-Cell generates simulation program automatically by "Generate Program" button in Figure 21. Then one gets simulation program for the constructed model as shown in Figure 22.

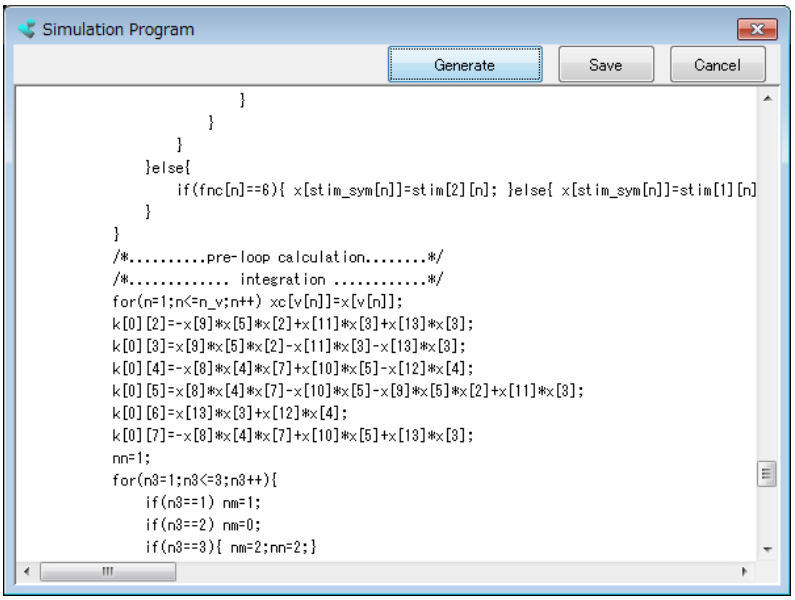


Figure 22 Part of a simulation program automatically generated by A-Cell.

The simulation program is written in a basic style of c language, and no special library is used. Thus, simulation program generated by A-Cell can be run on various computers[16], and at the same time, simulation proceeds at higher speed. It is common that simulation programs employ function call of the programming language. However, this leads to a significant decrease in the computing speed[17]. Therefore, A-Cell simulation program does not use function call. By compiling the simulation program, you can run simulation[18]. Any name of a simulation program is acceptable, but the default is “Csim file name”.c. Never change the extension.

7.5.2 Compile

Users does not need to concern about programing during generation of a simulation program. However, compilation requires user operation. But the operation is simple and stereotype.

Simulation program generated by A-Cell can be compiled by almost all c compilers. Since A-Cell does not provide a compiler of c language, it’s user’s responsibility to install a compiler. However, it is difficult for many molecular cell biologists to use a compiler. So compilation methods for typical three compilers are shown in “A-Cell operation manual”. Here we list three typical compilers as follows:

16 It is sometimes talked that a program cannot be compiled because of using special library. We believe that such kind of problem will not happen for A-Cell simulation program.

17 We have experienced 10-20% lower speed by using function call in a simulation program.

18 Parallelization is not applied in pathway simulations, because almost all pathway simulations proceed within an acceptable time period. In 4D simulations, however, much longer time is required, and parallelization can be applied (Cf. Chapter 10).

1) Intel compiler (icc)[19]
2) Microsoft Visual Studio (cl)[20]
3) GNU c compiler (GCC)[21]

Compiler command names are shown in parentheses. Intel compiler is very popular, providing evaluation edition, which is free with limited days of usage in addition to a commercial edition. Microsoft Visual Studio provides free edition for students and academic edition in addition to commercial edition. There is free edition for business usage with limited function. GNU c compiler is free software. Download sites for these compilers are shown in the footnotes. You can also find download site by a search by the name of a compiler, or on sites of free software. Refer to manuals for the methods of compiling, or you can find brief explanation in "A-Cell operation manual".

7.5.3 Setting initial conditions

When simulation starts, simulation program reads initial conditions stored in a file. In an initial condition file, information required for the simulation, which was shown in section 7.3, is stored. So, initial condition file should be created before running independent simulation. By pressing a button in the right square of Figure 20, a window for setting initial conditions appears (Figure 23).

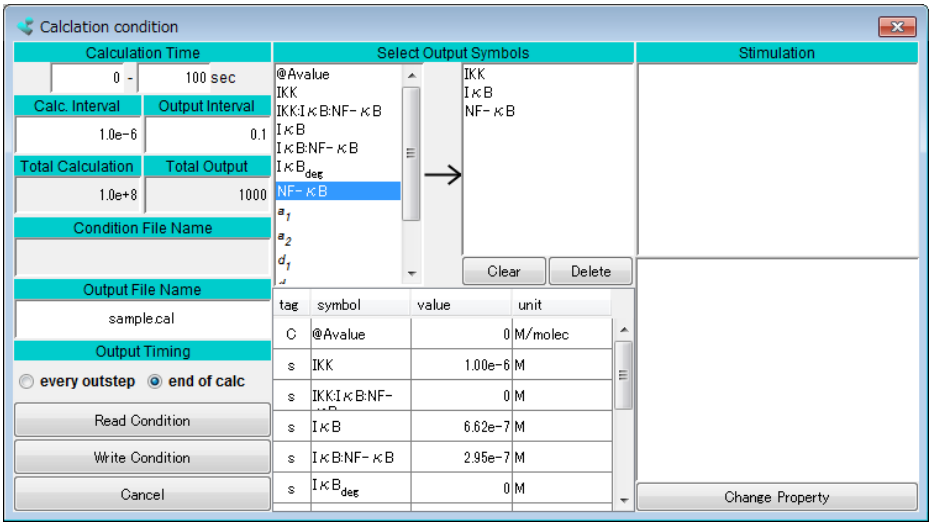


Figure 23 Window for setting initial conditions for independent simulation.

Setting parameters are almost the same as in "in situ simulation", and one can recognize and understand them by watching Figure 23.

By saving initial conditions, two disk files "RK.TMP" and "xxx.cond" are created, where "xxx" indicates a file name you specify for a condition file. This enables users to keep initial conditions for

19 http://www.xlsoft.com/jp/products/intel/purchase/prices.html
20 https://www.microsoft.com/ja-jp/dev/default.aspx
21 http://www.mingw.org/. MinGW is easy-to-use compiler developed for windows.

every simulation, and to refer them in later analyses. Initial conditions are stored in “xxx.cond” as a text file, and file name “xxx.cond” is stored in “RK.TMP” file. Simulation program reads “RK.TMP” file getting a file name of “xxx.cond”, and next it reads “xxx.cond” for getting actual initial conditions. The reason for this two-step access is that simulation program can access any condition files only by reading one “RK.TMP” file.

7.5.4 Running simulations

After compiling simulation program and setting initial conditions, simulation can be run. Method of program execution depends on OS. Follow manuals for each OS. In Windows OS, for example, simulation starts by double clicking compiled simulation program, which should be xxx.exe, in a folder, and following window appears:

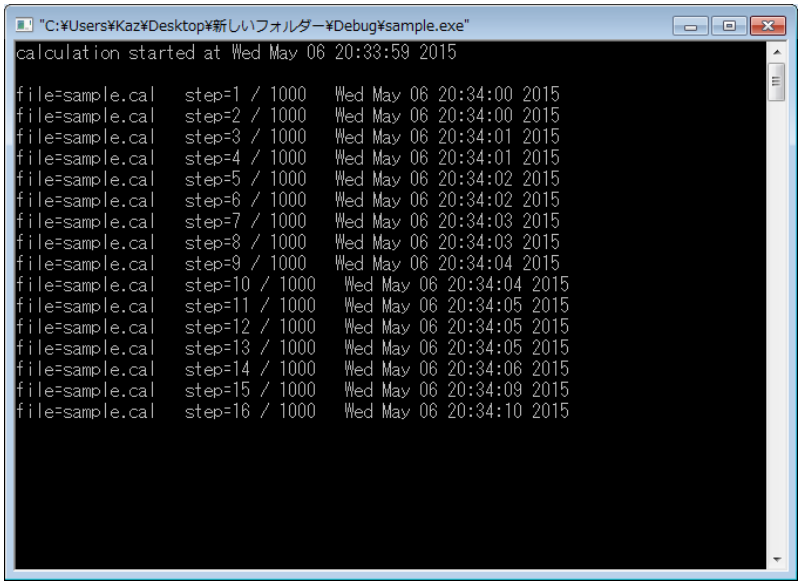


Figure 24 Window during independent simulation.

Figure 24 shows information 11 s after the start of a simulation. From this window, we know that there will be 1,000 outputs to a disk file, and 16 outputs have been saved. The file name is “sample.cal”. By calculating a time required for a single output step, which is ~0.5 s in this example, we can estimate time required to completion of simulation. This information is quite helpful in case that simulation takes a long time (e.g. a week or more). If there is no information from a computer during a long simulation period, we will worry about whether simulation is running or not, and even it is running, we don’t know when the simulation will be completed. To avoid these, A-Cell informs us of processing of a simulation.

7.6 Importance of getting equilibrium before starting simulation

Here, we comment an important point in simulation, the establishment of equilibrium before running simulations. In many cases, simulation starts by an application of a stimulus to a cell at an

equilibrium state[22]. Just after setting initial values to a model, however, it may be not clear whether the model is in equilibrium, because parameter values such as concentrations obtaining from many papers do not guarantee the equilibrium. If the model is simple, one can analytically obtain values for the equilibrium. If the model is large, however, one cannot know how equilibrium can be established. Stimulus is often involved in a model, where the changes in active protein concentration are calculated. If simulation involving stimulus starts without equilibrium, however, the simulation gives us combined results of dynamics toward equilibrium and that of stimulation, and it is impossible to discriminate between these two dynamics. Such simulation results are meaningless. Thus before simulation, equilibrium should be established. A method of establishing equilibrium is simple. First, delete the stimulus in the model, or set stimulus to null value, which will be realized by setting ligand concentration 0 for example. Next, run simulation until equilibrium has reached, which is known by checking that any change happens by further running of simulation. Third, replace initial values of concentration by those at equilibrium thus obtained. These values are true initial values for the simulation. Lastly, run simulations with ligand concentration larger than 0.

[22] A state where there is no change in values (e.g. concentrations) of the model.

8. Display simulation results

Simulation result is stored in “xxx.cal” file, where xxx is a file name you specified. The structure of this file is shown in Figure 25. Although this file is a text file, it is not directly imported to Excel or Origin for analyses, because there are lines of header and trailer information.

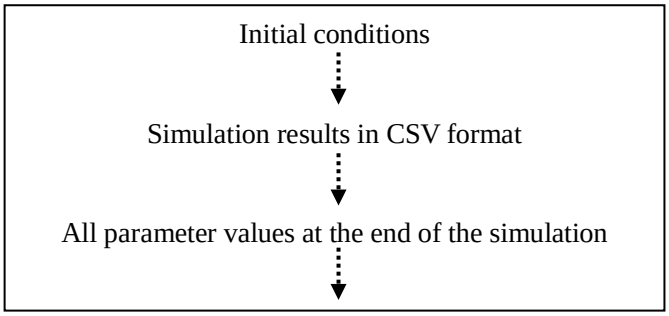


Figure 25 File structure of “xxx.cal”.

Instead, “xxx.cal” file is imported to A-Cell, and A-Cell displays simulation results by using a button in red square of Figure 26.

Figure 26 Button for reading and displaying simulation result (red square).

When reading “xxx.cal” file has finished, a window shown in Figure 27 appears. Time courses of concentrations are displayed on the window.

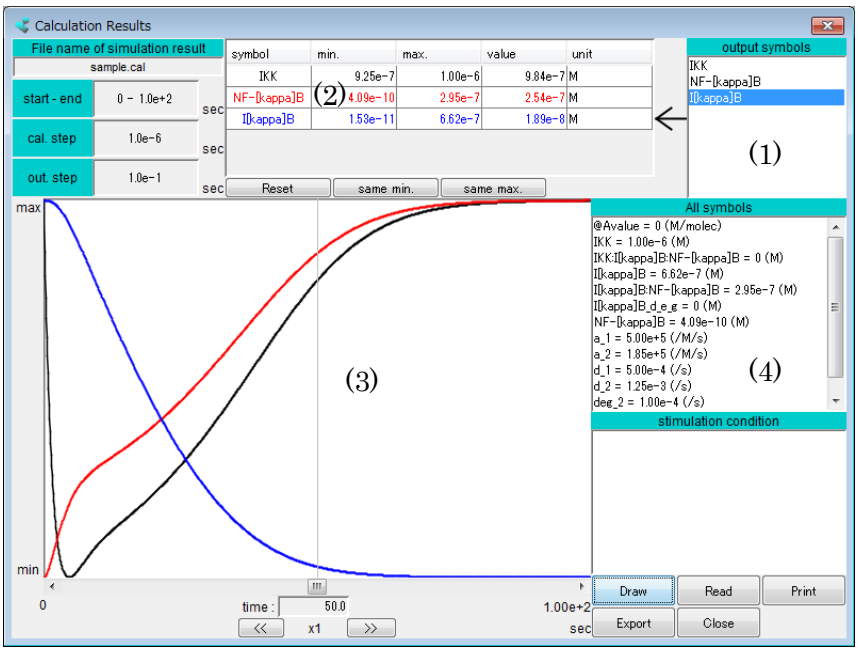


Figure 27 A window showing calculation results graphically.

At a moment the window is just displayed, however, no data is shown in (2) and (3). To display time courses, first select symbol(s) from a textbox (1), next set max. and min. values for each symbol in textbox (2), and finally press “Draw” button in the bottom right of this window. Then you get time courses in a picture box (3). Different max. and min. values can be set for different symbols. Values, such as concentration, at a time indicated by the cursor are shown in (2). Time course data shown in (3) are exported as a CSV file by a button “Export”. All symbols with initial values are shown in the textbox (4). Exported CSV data will be imported to other software for further analysis. Thus, these operations are almost identical to those in “in situ simulation”.

9. Flux analysis[23]

During a course of analysis of simulation results, one may want to know a flux in chemical reaction, which is a rate of change in the concentration with respect to a protein in focus. If one knows this in pathways, one can know which pathway is most influential to the change in a concentration of a protein in focus. Figure 28 shows a simple example.

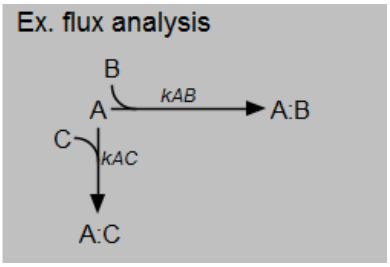


Figure 28 Knowing flux with respect to a focusing protein.

Although this is a simple example, in case where B and C are regulated by reactions not shown here changing their concentration dynamically, it is not easy to know which pathway, A+B→A:B or A+C→A:C, is more influential to the change in the concentration of A. In addition, influential pathway may be switched by time. In this case, flux analysis is useful. To compare these two fluxes, one can realize this by using equation group of A-Cell describing the following equations:

$$\begin{aligned} flux_{A\to A:B} &= -k_{AB}\cdot A\cdot B \\ flux_{A\to A:C} &= -k_{AC}\cdot A\cdot C \end{aligned} \qquad 18)$$

However, it is time consuming to write Eq.18). So A-Cell provides this function as “Flux Analysis”.

A button shown in red square of Figure 29 allows you to show Flux Calculation window (Figure 30).

Figure 29 A button for flux analysis.

[23] This function will be available in the update versions of A-Cell.

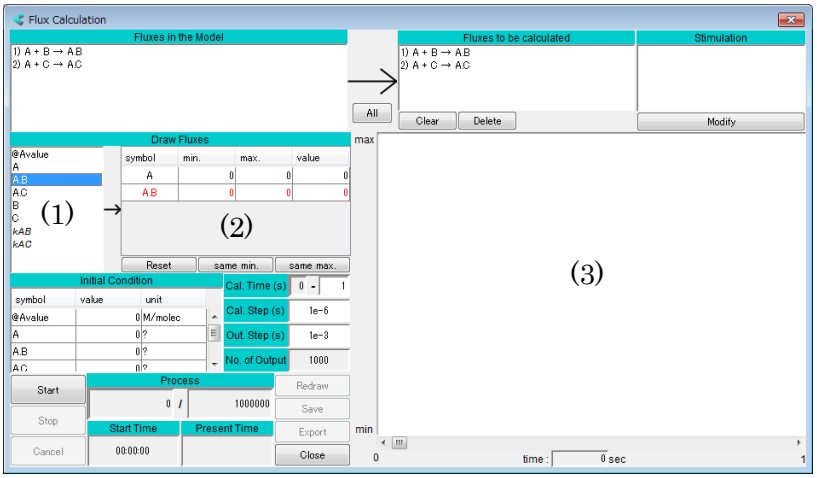


Figure 30 Window for flux analysis.

A window shown in Figure 30 is similar to that for “in situ simulation”. First select path form the textbox (1), next set max. and min. values for display in (2), and then fluxes are shown in a picture box (3) by pressing “Strat” button on the bottom left of this window.

10. Spatio-Temporal Simulation (4D simulation)

10.1 Importance of spatial aspect in cell simulations

Description on A-Cell function shown in the previous chapters does not care about spatial extent of a cell. Models without spatial extent assume implicitly that all species are concentrated in one point or that a pot is well-mixed assuming spatial homogeneity. Thus, the model only concerns about temporal changes of concentrations and is called temporal model. Great majority of pathway simulation is temporal simulation without spatial extent of a cell. It is needless to say, however, that a cell has spatial extent in 3D space, and thus, spatio-temporal change in protein concentration such as local activation of proteins and its spread proceeds in a cell. In fact, spatio-temporal localization is known to be important in cell function such as in embryonic development.

However, it is not intuitive that the spatio-temporal simulation is important. A discussion of this importance is beyond the scope of this document, so we would like to ask readers for referring published articles[24]. But in short, total amounts (but not concentration) and diffusion of molecules have considerable effects on signal transductions in a cell. One example is shown in Chapter 13.

Here we discuss diffusion, which is the basis for 4D simulation. Diffusion is a phenomenon of random movement of small particles because of heat energy, and hence the position of a particle becomes distant from origin as time elapses. When there are huge numbers of such particles, we can observe a phenomenon of spreading of substances like ink within water. This phenomenon has physical basis and is described mathematically as the diffusion equation. A detail of this is also beyond the scope of this document. So refer to textbooks (there are many good textbooks on this topic). Here we describe only important aspects of diffusion for cell simulation. Those are flux, which describes the rate of bulky movement of substance, and a measure of spreading. These are described by the following equations:

$$flux = D\frac{\partial C}{\partial x} \qquad 19)$$

$$\lambda = \sqrt{2n \cdot D \cdot \Delta t} \qquad 20)$$

Eq.19) is called Fick's equation calculating flux with diffusion coefficient D. $\partial C / \partial x$ is spatial differentiation of concentration showing spatial steepness in the change of the concentration. Thus, larger the diffusion coefficient and spatial steepness in the concentration change are, flux is larger[25].

[24] Ichikawa, K., Neurosci.Res. 25(1996), 137; Ichikawa, K., Neurocomp. 58-60(2004), 709; Goryachev, A.B., et al., FEBS Lett., 582(2008), 1437; Terry, A.J., et al., J.Theor.Biol, 290(2011),7; Hoshino, D., et al., PLoS Comp.Biol., 8 (2012), e1002479; Ohshima, D., et al., PLoS ONE, 7 (2012), e46911; Ohshima, D., et al., PLoS ONE, 9(2014), e109895; Ohshima, D., et al., PLoS Comp.Biol., (2015); Ohshima, D., et al., PLoS ONE, (2015).

[25] Famous diffusion equation is derived from Eq.19).

Eq.20 gives us a measure of spatial spread λ of diffusing substance after the elapsed time Δt. n is spatial dimension, which is 1, 2, and 3 for 1D, 2D, and 3D space, respectively. Reported diffusion coefficient D of proteins is in the order of 10^{-11} m^2/s. If a substances are concentrated at the center of spherical 3D cell of 50 μm in diameter at $t=0$, λ is 24.5 μm after Δt of 10 s by Eq.20. Thus, the substance spreads almost all over a cell within 10 s. 10 s might be short or long depend on phenomenon you are focusing. Localization of substance is obvious at shorter period than this. In addition, the localization, that is inhomogeneity of spatial distribution, becomes quite prominent when there is a source of the substance in the 3D space. In this case, chemical reactions proceeds only within a limited region in a cell, and resulting effect will be much different from that in homogeneous process. At the beginning of embryonic development, it is well known that mRNAs and proteins localize, and this is important for the emergence of cell polarity[26]. Thus, in cell simulation, 4D simulation, which is a simulation in time (1D) and space (3D), is inherently important.

A-Cell has been developed as a tool with capability of spatio-temporal simulation from the beginning. In this chapter, functions prepared by A-Cell for 4D simulation are described.

10.2 An overview of 4D simulation by A-Cell

To realize 4D simulation, following 4 steps are required: 1) constructing 3D shape model, 2) dividing a shape into regions such as the nucleus and the cytoplasm embedding chemical reactions to a corresponding region, 3) generation of simulation program for numerical integration of PDE[27], and 4) display spatio-temporal simulation results. A-Cell provides all these functions (Figure 31).

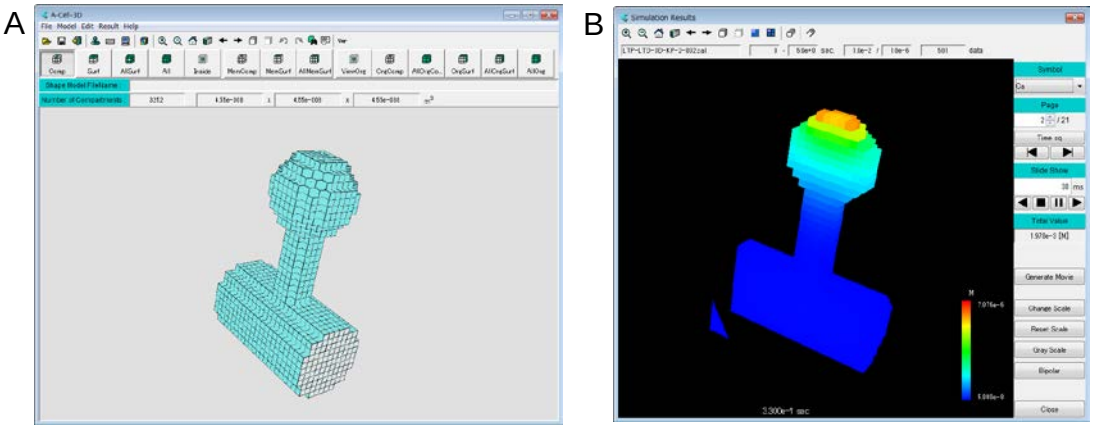


Figure 31 An example of 3D morphological model constructed by A-Cell (A) and a simulation result (B).

This example shows signal transduction in a spine of a neuron. One can construct this 3D shape

[26] In addition to diffusion, other mechanisms of movement such as transportation on microtubules also play important roles.

[27] Partial Differential Equation

model by specifying small number of parameters by using A-Cell. Simulation result is shown as 3D spatial difference in molecular concentration as shown in Figure 31B, where cross section is shown. In addition, A-Cell provides movie display for intuitive capturing of simulation result. A-Cell also provides drawing of spatial profile of concentration.

One may realize that a shape shown in Figure 31A is composed of many cubes. These are called compartments in A-Cell. Division of 3D shape into compartments is a basis of 4D simulation. Within a compartment, homogenous space is assumed, and hence concentrations are assumed to be the same anywhere within a compartment. However, flux of a substance between compartments is calculated according to Eq.19). This is shown in Figure 32 in detail for 1D case.

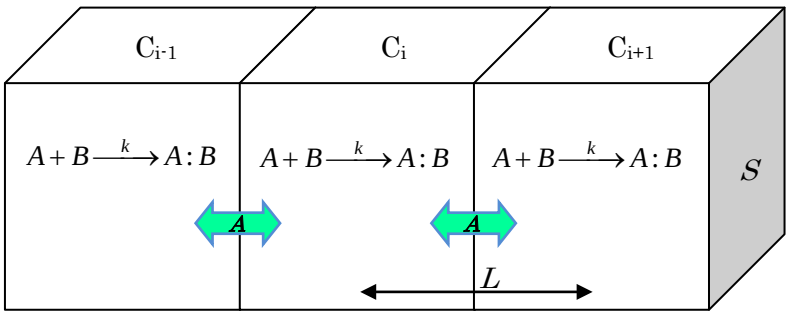


Figure 32 Method of reaction-diffusion simulations in 1D space.

In this example, a binding reaction $A+B \xrightarrow{k} A:B$ proceeds in compartments C_{i-1}, C_i, and C_{i+1}, and at the same time molecule A diffuses. B and A:B is assumed not to diffuse. If concentration of A is 1 μM at compartment C_i, and 0 at other compartments, A at C_i diffuses to adjacent two compartments, where a reaction $A+B \xrightarrow{k} A:B$ proceeds at the same time. Thus, the concentration of A in compartment C_i changes by two factors, which are binding of A with B and diffusion to adjacent compartments C_{i-1} and C_{i+1}. This is expressed by differential equations as follows:

$$\frac{dA_i}{dt} = -k \cdot A_i \cdot B_i + D\frac{A_{i-1} - A_i}{L}S + D\frac{A_{i+1} - A_i}{L}S \qquad 21)$$

where the first term on the right side is the decrease by the reaction, and the second and the third terms are the change by diffusion to adjacent compartments. In Eq.21), flux is calculated using distance between compartments (L) and cross sectional area between the compartments (S). Since two difference concentrations between compartments A_{i-1}-A_i and A_{i+1}-A_i are negative, the second and the third terms are negative. The first term is negative too leading to the decrease in the concentration of A at the center compartment. This is an example for 1D space, and if simulation is for 2D or 3D space, four or six diffusion terms appear because a compartment is connected to

adjacent ones with four or six surfaces. If there are other reactions in each compartment, additional terms for reactions are added. Thus, differential equations for 4D cell simulation are complex, and constructing all differential equations by human is not realistic. A-Cell performs all these formulation automatically from a shape and pathway models constructed by a user.

10.3 Main window for 4D simulation in A-Cell

To construct a 4D simulation model, a button shown in a red square of Figure 33 is used.

Figure 33 Button for the construction of 3D shape model (red square).

Figure 34 is the main window for 4D simulation.

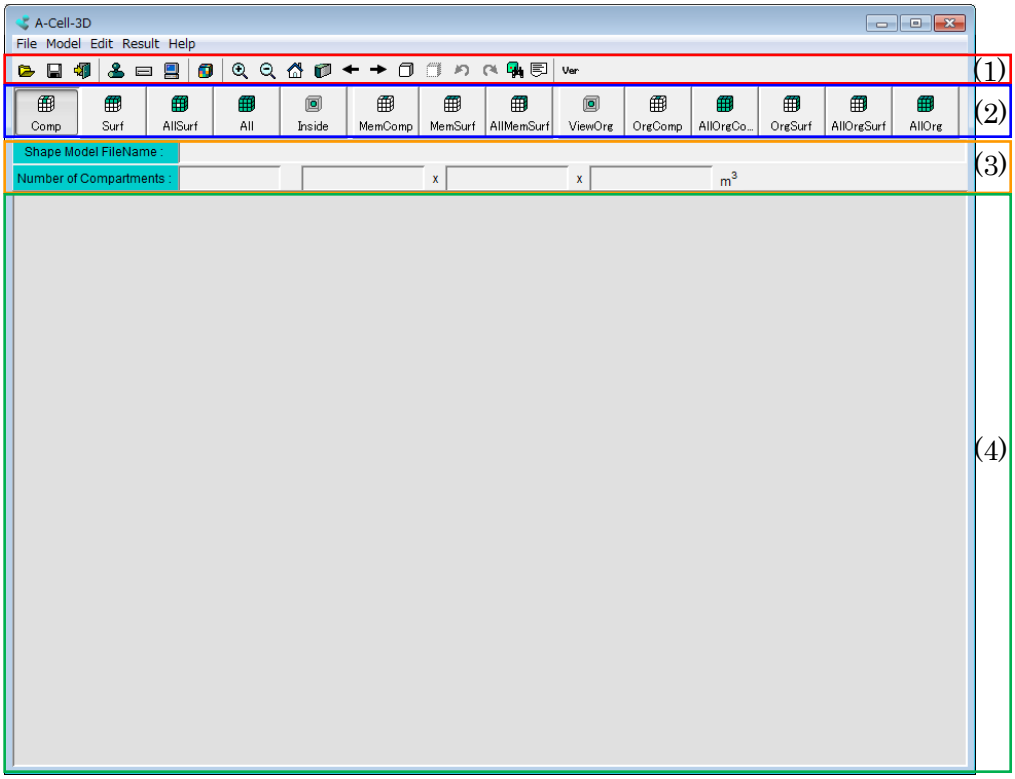


Figure 34 Main window for 4D simulation.

From this window, operations for 4D simulation are performed. Shape model construction, saving constructed model, generation of 4D simulation program, and reading 4D simulation result are executed by buttons in menu (1). Buttons in menu (2) are used for specifying region or compartments in constructed 3D shape, to which chemical reaction groups, stimulus groups, etc. are embedded. This operation is essential in constructing 4D model because this allows you to embed model groups to a specific region or compartments in a shape by specifying compartment(s) by

mouse pointing and/or dragging, selecting all or part of regions, or selecting surface compartments. Selected model groups are embedded to selected region or compartments. Data for the constructed shape are shown in textboxes (3), and constructed model is graphically shown in picture box (4).

10.4 Constructing 3D shape

A button shown in a red square of Figure 35 is used for constructing 3D shape.

Figure 35 Button for construction shape model (red square).

By pressing this, a window shown in Figure 36A appears. First select shape from five templates. Shape of a cell is quite complicated and there are many shapes in different cell types. In the present version of A-Cell, five shapes are prepared as shown in Figure 36A. By selecting one of five shapes, you can construct a shape model only by specifying small number of parameters.

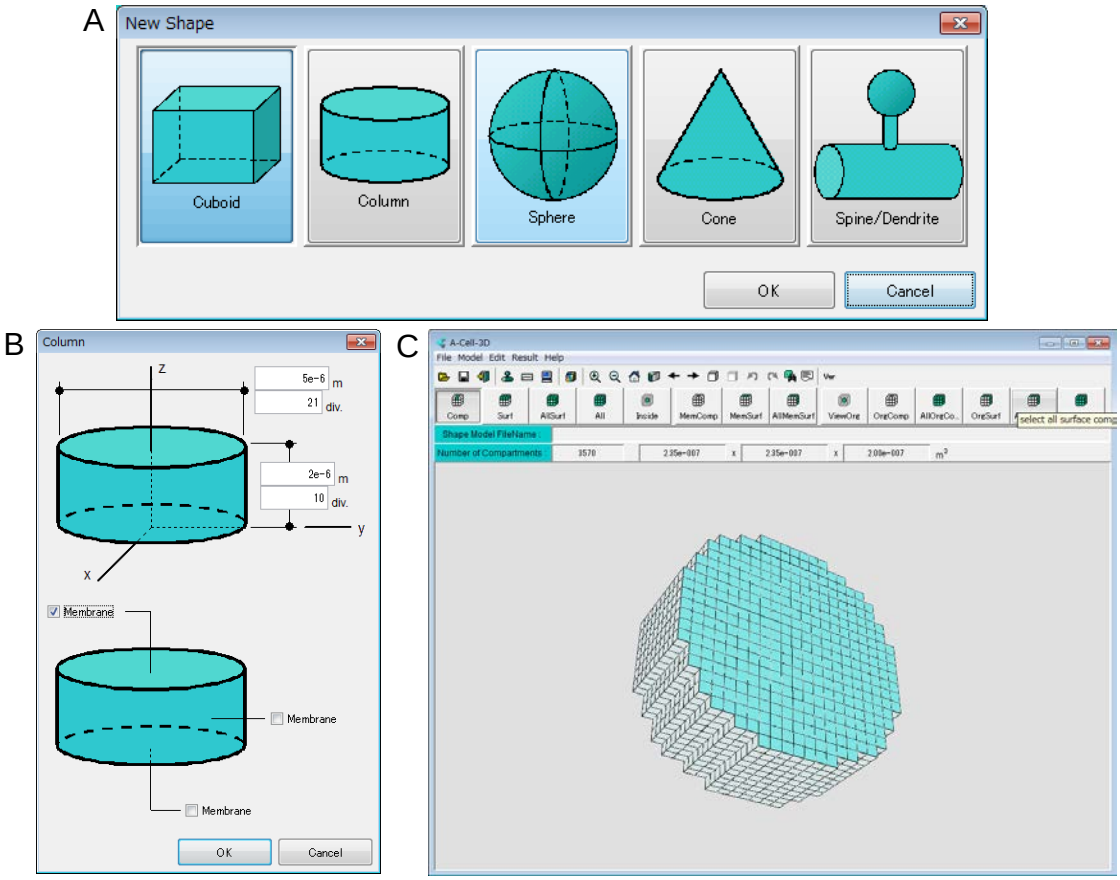


Figure 36 Windows for constructing 3D shape model. (A) Five shapes prepared by A-Cell, (B) a window for specifying parameters of a columnar shape, and (C) constructed columnar shape model as an example.

Figure 36 shows an example of constructing columnar shape model. First specify diameter and height, and next specify number of divisions in radial and height direction (Figure 36B). Then a constructed compartmental model is shown in the main window (Figure 36C). The size of compartments is same in a shape. Since columnar shape is approximated by many cuboid compartments, there is an error in the volume of constructed shape. A-Cell calculated the size of a compartment automatically so that the resulting volume of the shape is close to the original shape. The upper plane of Figure 36C is shown in light blue, which indicates that this plane is a membrane. This is specified by checking the check box in Figure 36B. You can construct both intra- and extracellular spaces in one shape model by adequately managing compartments and model group embedding to them[28].

10.5 Constructing organelle (nucleus) in a shape

When simulating transcription factors, one may want to construct a 3D cell model with the nucleus in it, and chemical reactions will be embedded to the cytoplasmic region and the nuclear region according to reactions in the cytoplasm and in the nucleus. The nucleus can be constructed by specifying compartments one by one. But this is a time consuming. So A-Cell has a mean to construct a nucleus by specifying small number of parameters. An example is shown in Figure 37, in which the cell is 20 μm in diameter with 51 divisions, and the diameter of the nucleus is 7 μm.

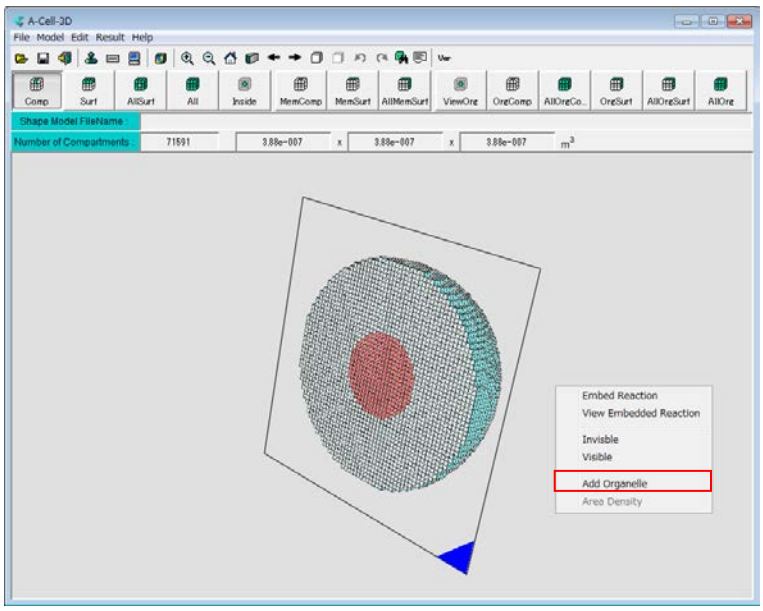


Figure 37 A spherical shape model with the nucleus in it.

As shown in Fig.37, the shape of the nucleus is constructed by using compartments of the parent

[28] If membrane compartments are specified at intermediate region of a shape, you can segregate upper and lower regions as extra- and intracellular regions, for example.

shape. By selecting a menu “Add Organelle” menu in red rectangle, the nucleus is constructed.

10.6 Specifying a region in a shape and embedding models to a region

To embed a chemical reaction group to a destination region or compartments of a shape, A-Cell provides 14 means to specify region(s) in a shape (Figure 38).

Figure 38 Fourteen means to specify region(s) in a shape.

By using “Comp” button at the left most of Figure 38, for example, one can specify a compartment or multiple compartments by mouse click and/or drag (left panel in Figure 39). Selected compartments are shown in red. By using “Surf” button, you can specify all surface compartments by a mouse click (middle panel in Figure 39). “All” button selects all compartments including internal ones, which are shown in orange (right panel in Figure 39).

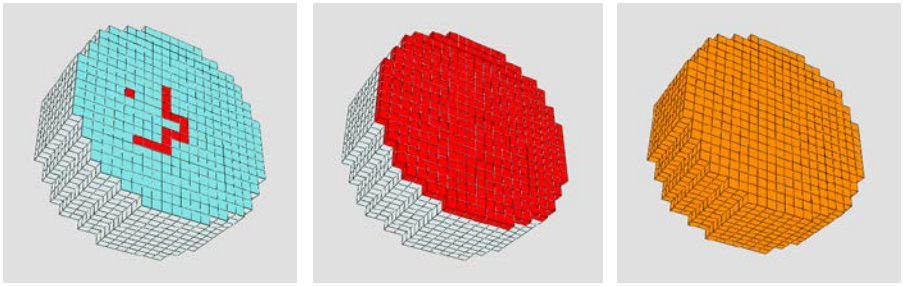


Figure 39 Examples of specifying compartments or a region.

If you want to specify internal compartments of the shape one by one, first make cross section of the shape by a button shown in red square in Figure 40.

Figure 40 Button to make cross section of a shape.

By specifying compartments with the same operation as shown above, internal compartments are specified (Figure 41 for an example).

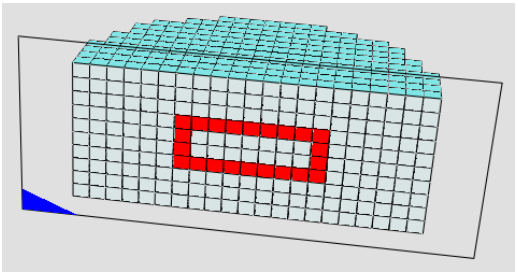


Figure 41 An example of specifying internal compartments in cross-sectioned shape.

Black rectangle displayed in front of cross-sectioned shape indicates the plane of cross section. This plane is moved back and forth, and is selected to x-y, y-z, or z-x cross section. Thus a complicated internal shape can be constructed.

Users can embed selected group(s) to a specific region in a shape by operation shown in Figure 42.

A B

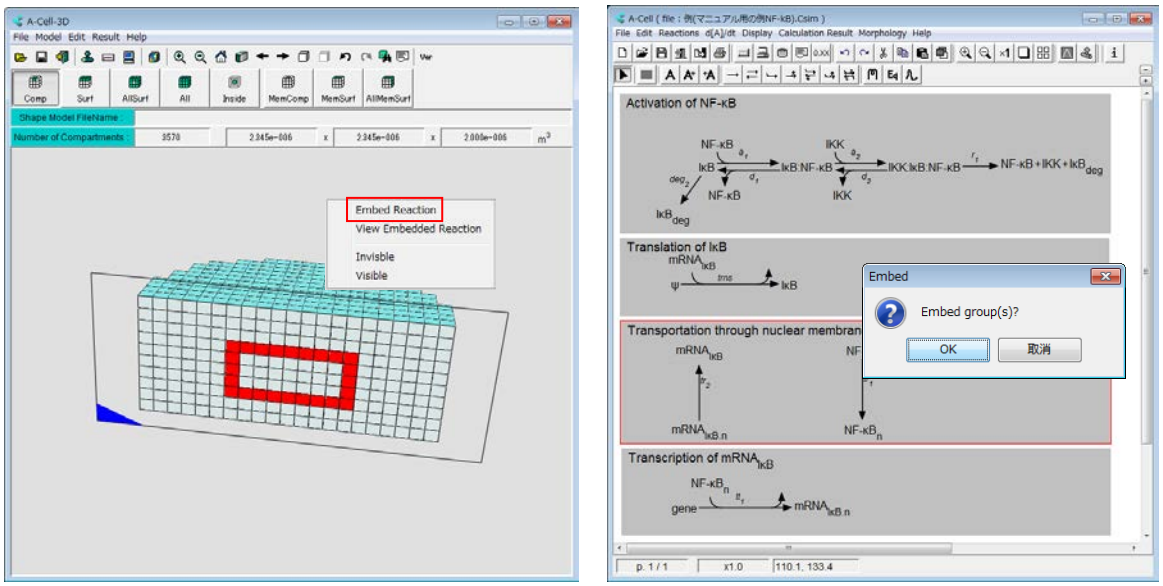


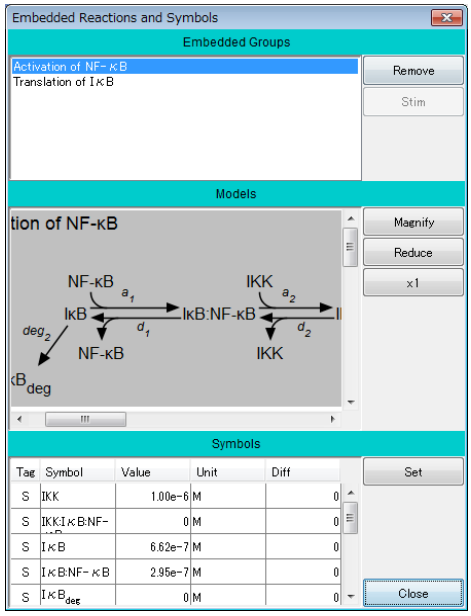
Figure 42 Embedding a reaction group.

To embed a group, first specify destination region where a group is going to be embedded (e.g. red compartments in Figure 42A), and then select “Embed Reaction” menu (red rectangle in Figure 42A). Next return to A-Cell main window and select a group to be embedded. Then a window shown in the inset of Figure 42B appears, and selected group is embedded by pressing “OK”. It is easy to know to which compartments a specific group is embedded. Not only reaction group, any group including equation, stimulation, and membrane voltage groups can be embedded to a shape.

Specifying diffusion coefficient is important in 4D simulation. This is done by a button shown in a red square in Figure 43. This button enables a window shown in Figure 44B. After selecting symbols to be diffused, enter value in the textbox labeled as “Diffusion coefficient”. If a symbol is diffusing, a value of diffusion coefficient appears on the textbox above. Different diffusion coefficient can be specified to different symbols.

Fig.43 Button specifying diffusion coefficient.

A



B

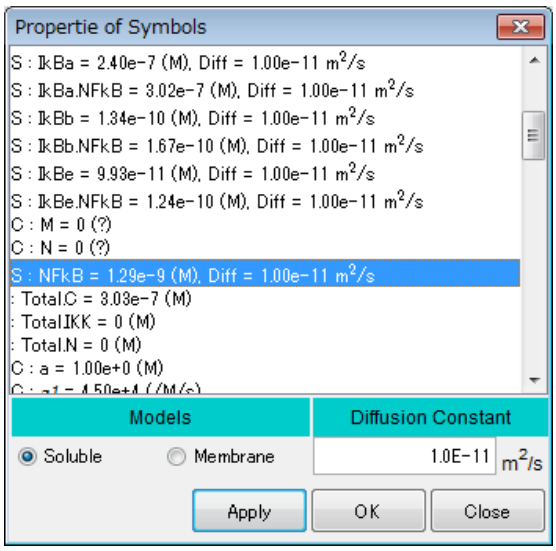


Figure 44 A window showing embedded group to a selected compartment(s) (A) and a window specifying diffusion coefficient (B). (A) In this example, two groups are embedded, and a group “Activation of NF-κB” is shown graphically. (B) In this example, diffusion coefficient 10^{-11} m^2/s is specified to NF-κB and its complexes.

4D simulation model thus constructed are stored by a name “xxx.Vsim”, where “xxx” is a file name you specify. The extension is Vsim, which should not be changed. This file is also ciphered as in xxx.Csim file.

10.7 Generation of simulation program, compiling, and running it

4D simulation takes much longer computational time for simulation in general. So, A-Cell does not provide “in situ simulation” for 4D simulation. All 4D simulation run as “independent simulation”.

10.7.1 Automatic generation of simulation program

Automatic generation of simulation program is done by a left button in a red square of Figure 45. Right button in a red square is used for setting initial conditions.

Figure 45 Buttons for automatic generation of simulation program (left red square) and setting initial conditions (right red square).

Automatically generated simulation program is shown in Figure 46A. After saving and compiling this, one gets executable simulation program.

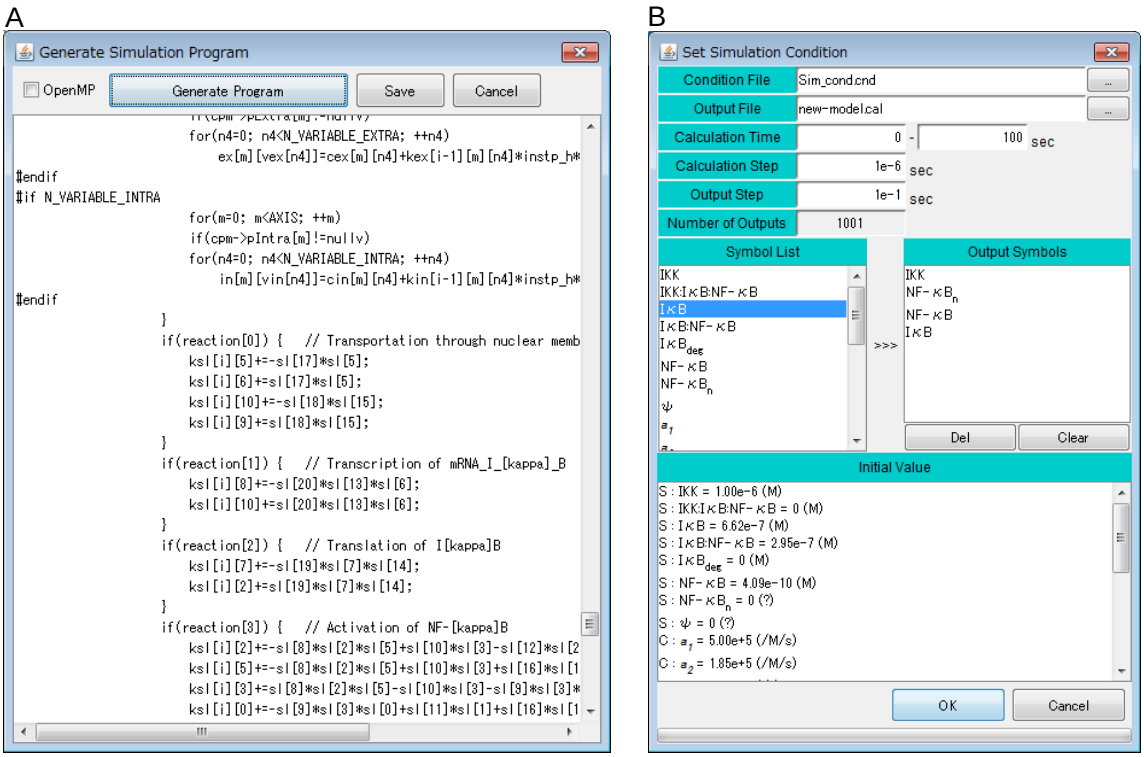


Figure 46 Automatically generated simulation program (A), and a window for setting initial conditions (B).

Generated simulation program includes differential equations for reaction-diffusion simulation as shown by Eq.21). Default name of a generated simulation program is SimProg.cpp, which is modifiable by a user. However, the extension should be cpp, and should be saved as xxx.cpp.

10.7.2 Compiling simulation program, setting initial condition, and running

Users need to compile generated 4D simulation program with the same operation as in pathway simulation, and its operation is described in “A-Cell operation manual” in detail. Setting initial conditions is also almost the same as in pathway simulation (Figure 46B). When saving initial conditions as a file, two files, ”SIM.tmp” and ”xxx.cnd”, are generated. Place these two files at the same folder as that of compiled simulation program. When simulation is started, the simulation program read these files first, and the simulation starts. The same window as Figure 24 appears showing progress of the simulation.

10.7.3 Accelerating simulation speed by parallelization

As described before, 4D simulation takes much longer computational time than pathway simulation, and it is important to reduce the computational time. Recent PC loads multi-core

processor having dual-, quad-, hexa-, and even more cores in one package of CPU. Computers with large number of cores and/or multiple packages of CPU are called high performance computer (HPC). Super computer is the top of HPCs. By using these computers, one can obtain much faster simulation speed.

There are several means to accelerate simulation speed, and one of these is implemented in A-Cell, which is parallelization by openMP. OpenMP is a method of parallelization applied for shared memory parallel computer loading multi-core processor. Although this type of parallelization cannot realize several ten thousands of parallelization, which is realized in a supercomputer, speedup in computers loading Intel Core-i5, Core-i7, or Xeon processor is realized very easily by openMP parallelization. When you want to generate openMP parallelization program by A-Cell, simply check "openMP" check box at the upper left corner of a window shown in Figure 46A. A-Cell generates openMP parallelization program automatically[29].

The question is to what degree the speedup is expected by openMP parallelization. Examples are shown in Table 1. The model was ECM-Deg-3(M14+fn+trnovr_3D)Java.Vsim, which is described in Chapter 13 in detail, and is downloadable from our homepage.

Table1 Speedup by openMP parallelization.

PC	- parallelization	+ parallelization
Core-i7 4770/3.4GHz, 16GB main memory	1	~3.5
Core-i5 3227/1.9GHz, 8GB main memory	1	~2.2

3.5 times faster computation was realized depending on the processor/computer. 7-days simulation without open MP parallelization on Core-i7 was reduced to 2 days with open MP parallelization, and speedup is correlated well with the number of cores. Thus, ten times faster simulation is expected, if we use HPC loading CPU with many cores. In fact, we realized 15.4 faster simulation by HPCs with E5-2699 v4 processor loading 22 cores. This speedup is significant if simulation time without parallelization takes one week or longer, because you can see simulation result within a half day instead of a week. Such long simulation time is often seen in 4D simulation. A-Cell with openMP parallelization capability will be released soon.

10.8 Displaying simulation results

To display 4D simulation results, read xxx.cal file by a button shown in a red square of Figure 47.

Figure 47 Button reading simulation result file.

[29] Present version of A-Cell does not generate MPI parallelized program, by which much more speed up can be realized by HPC having many CPUs including supercomputers.

The simulation result file can be more than 1GB, and transferring (reading) data into main memory can require a long time. In this case progress bar appears. Simulation result shown in Figure 48 was drawn from a same sort of a model as shown in Figure 42[30].

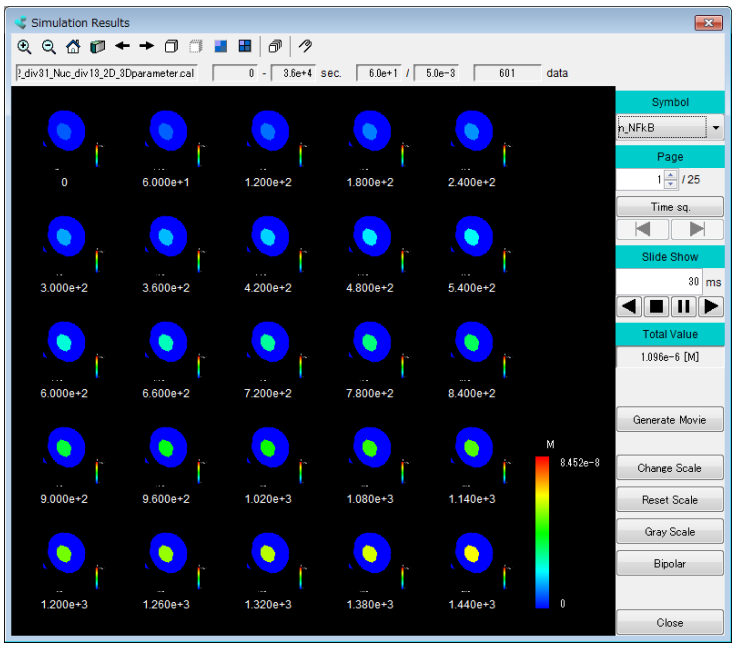


Figure 48 A window for displaying 4D simulation result.

This window shows overall spatio-temporal change for the focusing symbol n_NFkB protein, which is nuclear NF-κB shown in the center circular area. The change in the concentration of n_NFkB is shown in different colors on cross-sectioned spherical cell as time-series data. By advancing the page of this display, further changes in n_NFkB are displayed. In this simulation, cytoplasmic NF-κB flew into the nucleus, and it diffuses in the nucleus. However, no spatial difference in n_NFkB is observed, because n_NFkB diffused quickly within the nucleus in this example.

30 Ohshima, D., et al., PLoS ONE, 7 (2012), e46911.

By clicking one 3D shape in Figure 48, following display appears:

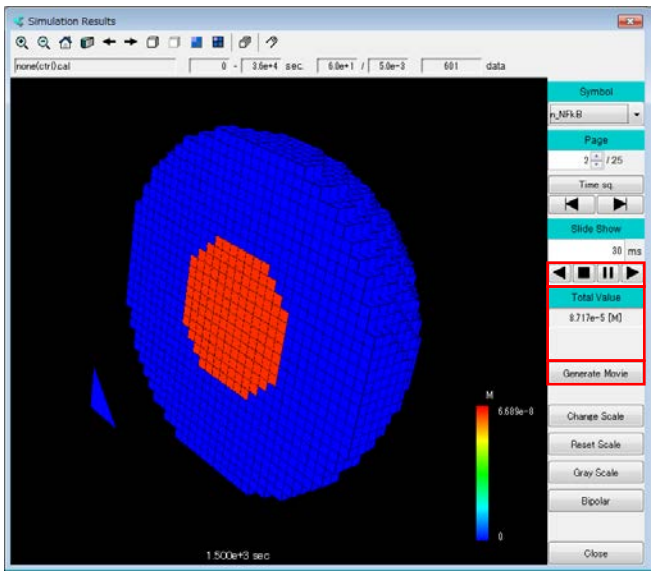


Figure 49 Spatial distribution of a selected symbol at a selected time.

This is a spatial representation of n_NFkB at a selected time. Compartments are shown with their borders in black lines. These can be on or off. By pressing a button in red rectangle (1) in Figure 49, a movie is played on the same window showing spatio-temporal change in the concentration intuitively. This movie can be captured by capturing software, which is embedded on your PowerPoint slide. Or a movie can be made by a button shown in red rectangle (3). Total concentration obtained by summing up concentrations in all compartments is shown in textbox (2). Although this value is not an actual concentration, it is quite helpful for validating your simulation. It may happen that the total amount of substances is not kept constant in a simulation when the simulation and/or a model are not correct. So, one wants to know to what extent the preservation of total value is assured. By checking the total concentration shown here, one can validate how the simulation is running correctly.

Graphical display shown above is useful for capturing overall results. But it is not suitable for quantitative analysis. So, A-Cell provides means to quantify time course and spatial distribution of a specified symbol (Figure 50).

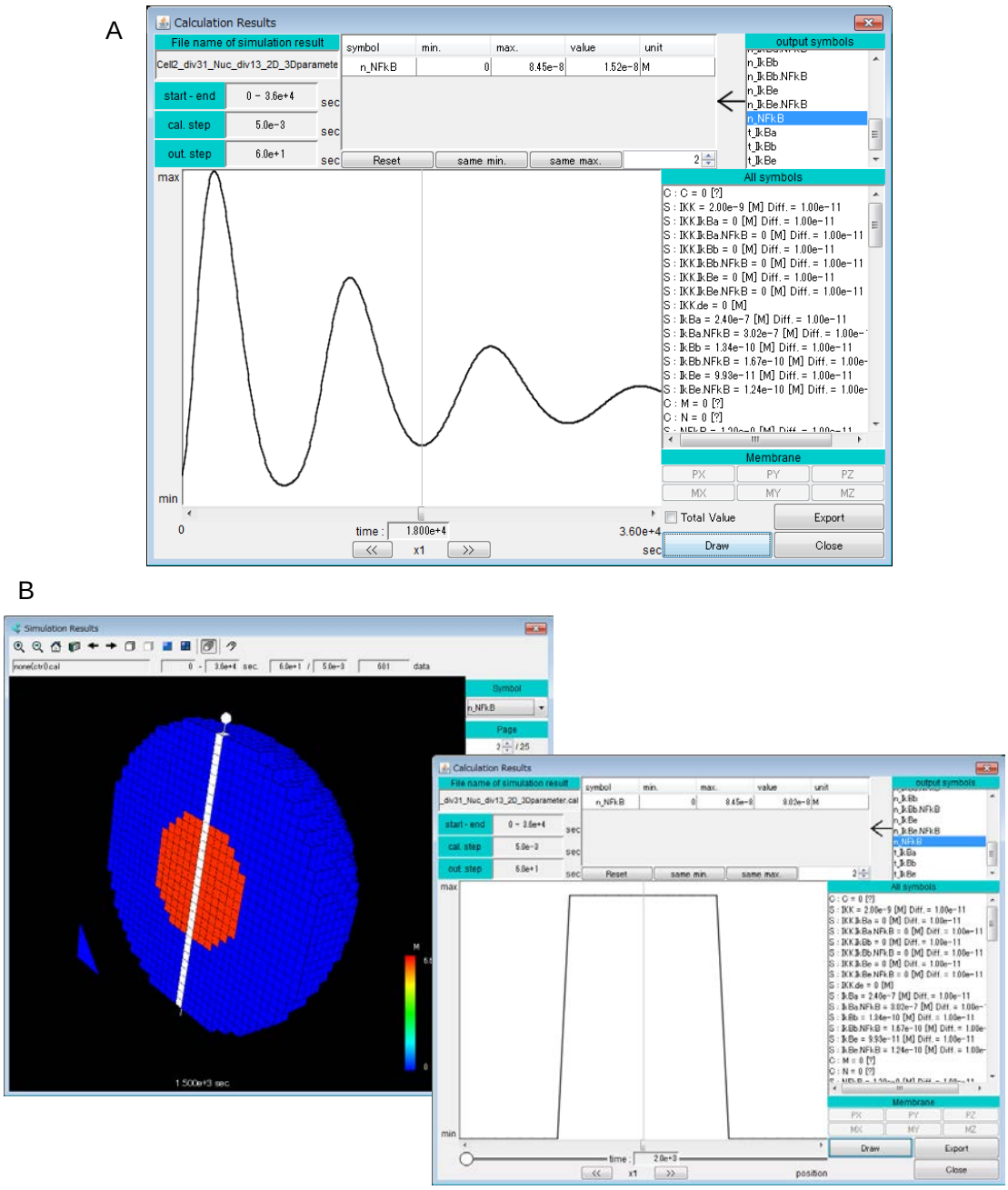


Figure 50 Time course (A) and spatial profile (B) for 4D simulation results.

Figure 50A is a time course at a compartment specified by a user. An example of spatial profile of concentration is shown in the right panel of Figure 50B. Horizontal axis in this case is spatial distance instead of time, which is measured along a white line of a pin shown in the left panel. The position and direction of the pin can be selected.

10.9 Summary of 4D spatio-temporal simulation

As shown above, A-Cell provide many functions for 4D spatio-temporal modeling and simulation with easy-to-use interface, because we aim at A-Cell to be used not only by experts in simulation but also by experimental researchers and students. Easy-to-use is also essential for experts, because they can concentrate on analyzing simulation results and finding novel phenomena and their mechanisms. A-Cell helps experts for focusing on these creative works.

In general, analyses of spatio-temporal simulation are much more difficult in comparison to simple pathway simulation. Simulation can run when a model is constructed. However, finding reason(s) and mechanism(s) why one obtained such simulation results is not easy in general. In addition, one may be worrying whether simulation was correct or not. In these cases, we recommend to investigate the result based on basics of spatio-temporal simulation shown by Eqs.19) and 20). Then one may reach to a hypothesis based on these basics, and this enables validating simulation leading to a confirmation of the hypothesis. By circulating this, one can reach a novel finding.

11. The Other Useful Functions

In this chapter, functions not intruduced in previous chapters are described. These functions are used by buttons in red squares of Figure 51.

Figure 51 The other useful functions provided by A-Cell.

11.1 End

To end A-Cell, press a button (1). If model(s) is not saved, a message appears.

11.2 Release a model

As described previously, A-Cell allows reading multiple model files. To release a model file displayed in the main A-Cell window currently, press a button (2). One of remaining A-Cell models appears on the main window.

11.3 Print

By pressing button (3), a window shown in Figure 52 appears.

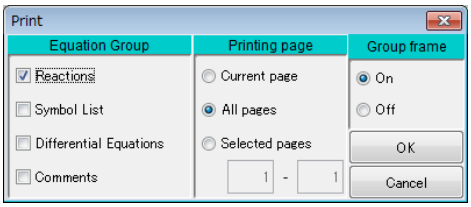


Figure 52 A window for print setting.

Select print category by “Equation Group”, pages to be printed by “Printing page”, and “Group frame” for printing its on or off. Then pages in a model file are printed.

11.4 Symbol list

By pressing button (4), a window shown in Figure 53 appears showing all symbol names, values, and units.

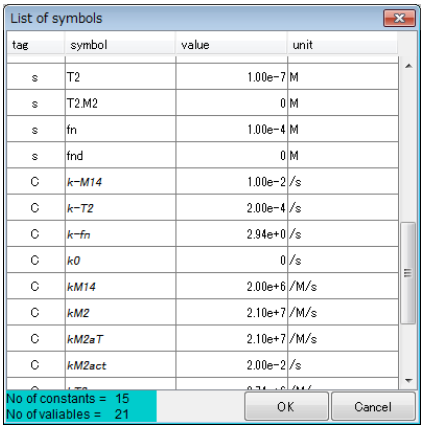


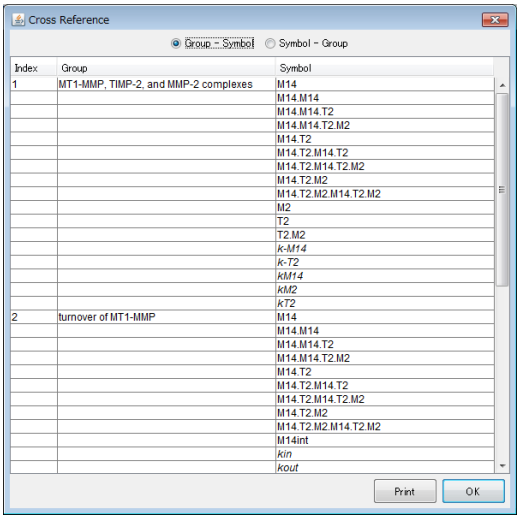
Figure 53 A window showing symbols in a model.

At the top of this list, a symbol name ”@Avalue” appears, which is not shown in Figure 53. This symbol is used for conversion of concentration to number of molecules, and vis vasa. In simulations that do not use this conversion, simply ignore this.

11.5 Cross references

If a model is large, it is not easy to remember in which groups a symbol is used, or conversely, which symbols are used in a group. By pressing button (5), a cross references appears as shown in Figure 54. These can be printed out.

A



B

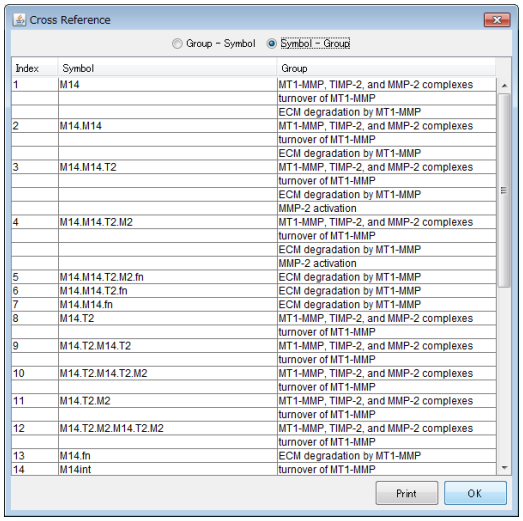


Figure 54 Cross references. Symbols in a group (A) and groups that are using a symbol (B).

11.6 Setting Display number of digits

Button (6) sets number of digits in various windows (Figure 55). The default number is 3.

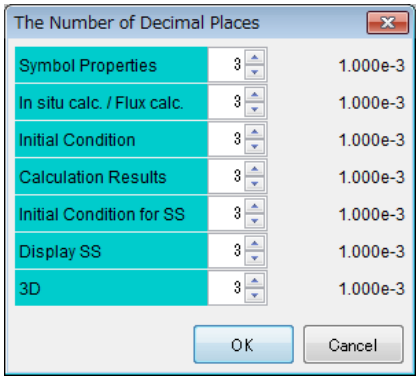


Figure55 A window setting number of digits in various windows.

11.7 Inserting a page

By using button (7), a new page is added following the page now being displayed.

11.8 Page view

Button (8) switches thumbnail view to page view.

11.9 Thumbnail view

Button (9) shows us thumbnail view of a model (Figure 56). Page order of a model can be changed by drag and drop operation.

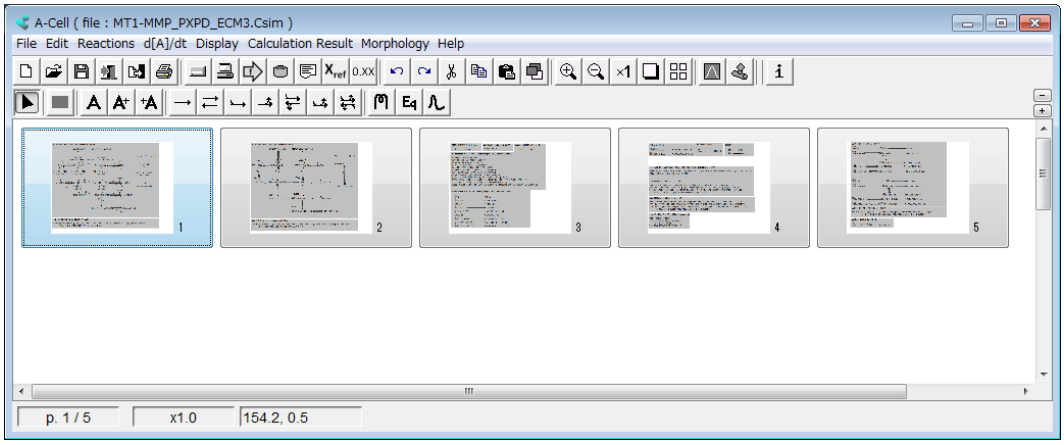


Figure 56 Thumbnail view.

11.10 Information about the A-Cell

Button (10) displays information about the A-Cell you are using (Figure 57).

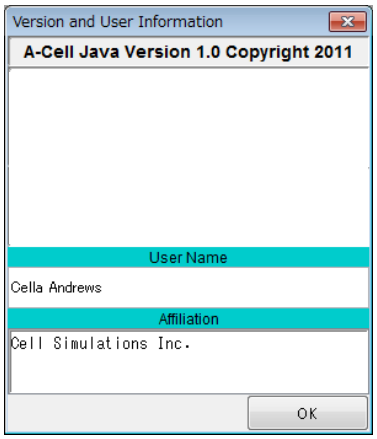


Figure 57 A window showing information about the A-Cell.

From this window, you can change user name and affiliation, which are embedded to groups you are going to construct.

12. A Note for Symbol Names

A-Cell uses following symbol names, which are reserved by A-Cell software.

V_m、C_m、R_m、I_{Na}、I_K、I_{lk}、I_{Inj}、
I_{CaT}、I_{CaL}、I_{Syn}、I_{AR}、I_{NR}、I_h、I_X、
@Avalue

Don't use these in your model. Otherwise, a simulation will give unexpected results. Symbol name starting with "@" such as @xxx can be used. But this is confusing with "@Avalue". So, using such symbol name is not recommended.

13. An Example of A-Cell Simulation: Essential Mechanism for ECM Degradation by Cancer Cells and Novel Intervention Method Preventing Cancer Invasion

In this chapter an example of 4D cell simulation by A-Cell is described. You can learn the advantages of A-Cell simulation. If you are not familiar with proteins appearing in this chapter, we hope that you can capture essence and usefulness of using A-Cell in 4D simulation.

We show here our collaborative research between simulation and wet experiment in cancer invasion. It is thought that more than 90% cancer patients can survive by preventing cancer invasion[31]. Thus, prevention of cancer invasion contributes to cancer therapy greatly. Here we show essential mechanisms of the initial step of cancer invasion revealed by 4D cell simulation using A-Cell, and potential targets to prevent cancer invasion[32].

Several conditions are required for the initiation of cancer invasion. One key condition is the degradation of ECM proteins surrounding cancer cells[33], which proceeds by activation of ECM-degrading proteinases. More than 20 such proteinases are known to date. Among them, MT1-MMP plays critical roles in ECM degradation[34] (Figure 58A). MT1-MMP is heavily expressed on the surface of invadopodia, which are protrusions on the surface of malignant cancer cells[35]. In laboratories, ECM degradation is observed by cancer cell culture on fibronectin (black puncta in Figure 58B).

[31] Hanahan, D., et al., Cell, 100(2000), 57.
[32] Hoshino, D., et al., PLoS Comp.Biol., 8(2012), e1002479; Watanabe, A., et al., PLoS Comp.Biol., 9(2013), e1003086.
[33] ECM is extracellular protein such as collagen, proteoglycan, fibronectin, and etc.
[34] Sato, H., et al., Nature, 370(1994), 61.
[35] Invadopodia are tiny protrusion as small as 0.5 μm in diameter and several μm in length.

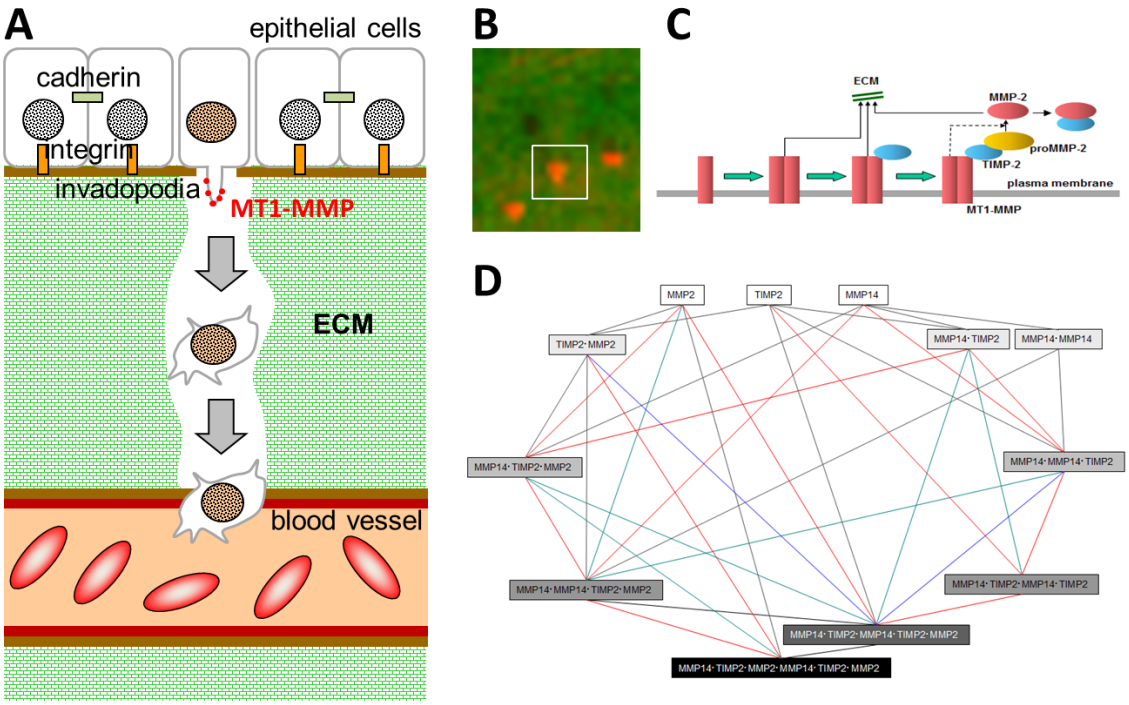


Figure 58 Mechanism of cancer invasion and regulation of MT1-MMP activity. (A) MT1-MMP (red dots) expressed on invadopodia degrades ECM proteins enabling cancer cell to migrate. (B) Experimental setup of ECM degradation (top view). Cancer cells are not visible here. Three orange puncta are MT1-MMP on invadopodia. Invadopodia protrude perpendicular to the surface of the paper in this picture. Green is fluorescent labeled fibronectin, and black puncta indicate degraded fibronectin, from which no fluorescence is detected. (C) Diagram showing regulation mechanisms of MT1-MMP activity. (D) Theoretically comprehensive diagram showing all available complexes formed by MT1-MMP (MMP14 in short here), endogenous inhibitor TIMP-2, and soluble proteinase MMP-2. This diagram shows the regulation mechanism of MT1-MMP activity.

MT1-MMP forms a dimer, and is inhibited by endogenous inhibitor protein TIMP-2 in extracellular space (Figure 58C). It is interesting that TIMP-2 binds with pro-form of MMP-2 (proMMP-2), a soluble extracellular ECM-degrading proteinase, and if one MT1-MMP in a dimer is not inhibited by TIMP-2, proMMP-2 is activated by TIMP-2-free MT1-MMP generating active MMP-2. Thus Figure 58C shows important regulation mechanism of MT1-MMP, which were revealed by extensive wet experiments. However, all possible complexes about regulation of MT1-MMP activity are not shown. So, first we draw theoretically comprehensive complexes and pathways forming them (Figure 58D). By adding chemical reactions degrading fibronectin (fn), an A-Cell model for ECM degradation was constructed as shown in Figure 59.

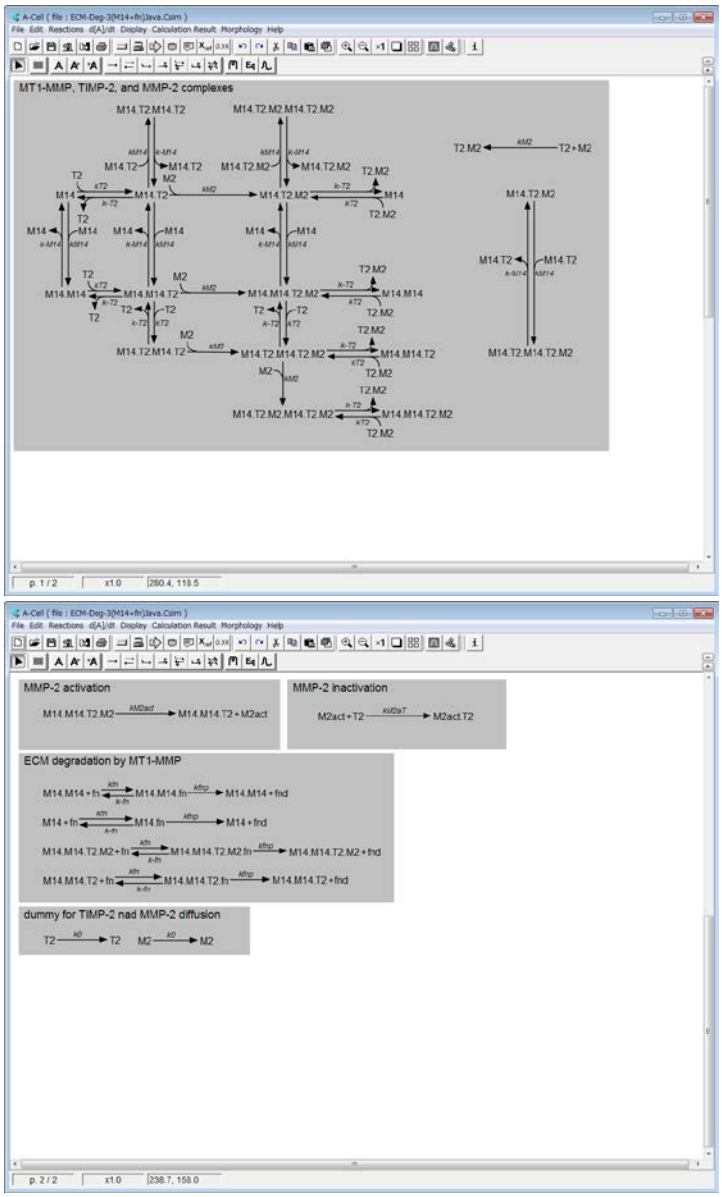


Figure 59 A model of ECM (fn) degradation by MT1-MMP (M14) according to a diagram shown in Figure 58D.

A model group ”dummy for TIMP-2 and MMP-2 diffusion” involves chemical reactions. These are reactions of transition from one species to the same species. Thus, there is no effect on the model. However, these are required for 4D simulation, for which reason is shown later. ECM degradation by MMP-2 is neglected here for simplicity. The name of the model file is ECM-Deg-3(M14+fn)Java.Csim, which is downloadable from our homepage. Many rate constants for reactions shown in Figure 59 were found in published literatures. In case not reported, values could be estimated. List of parameters and their values can be found in reference in footnote 32.

First we ran pathway simulation without spatial modeling. In this simulation, MT1-MMP is assumed to appear at a concentration of 100 nM at t=0. This is not explicitly shown in the model. This simulation is simply realized by setting MT1-MMP concentration at 100 nM and starting at t=0. This implicit way of modeling and simulation might be one reason that simulation study is not easily understood. However, this is a simple way of simulation. Simulation result is shown in Figure 60. fn was almost completely degraded 3,000 s after the appearance of MT1-MMP, which was consistent with experimental observations.

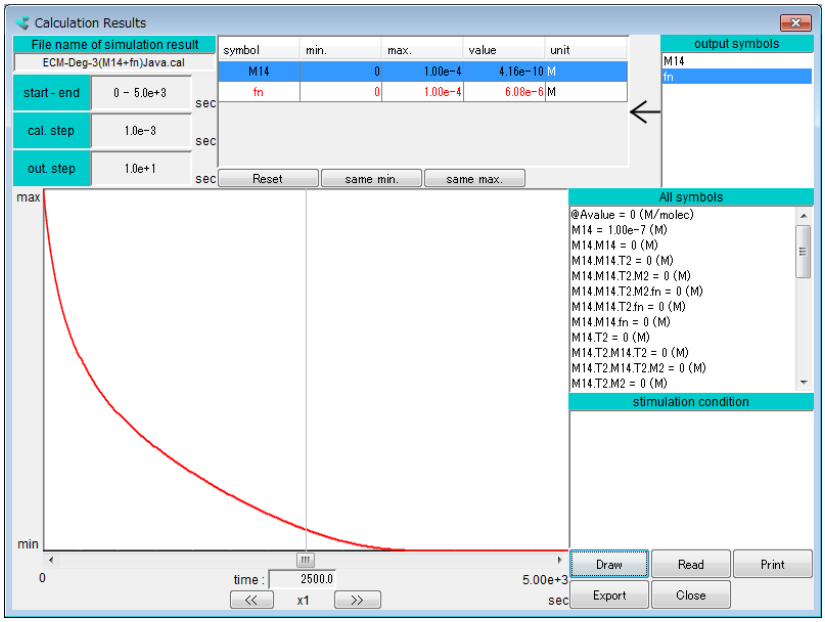


Figure 60 Simulation result of fn degradation by MT1-MMP. This is a simulation without spatial modeling. fn was almost completely degraded 3,000 s after the appearance of MT1-MMP.

Next, we investigated the same ECM degradation with 3D spatial model, which was much more realistic representation of experimental setup shown in Figure 58B. 3D spatial model for white square region in Figure 58B was constructed. A center circular red region was assumed to be an invadopodium surrounded by ECM. Exactly the same reactions with same parameter values were embedded to corresponding region in Figure 61.

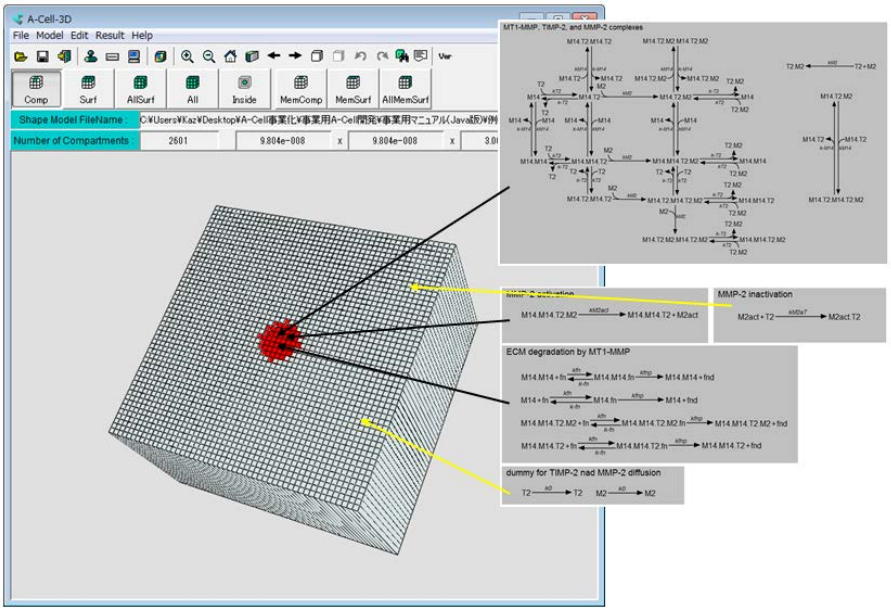


Figure 61 A 4D model for fn degradation by MT1-MMP. To show clearly to which region of 3D model reactions were embedded, reactions groups in Figure 58 are shown here again.

In a red circular region, all reactions with respect to MT1-MMP were embedded. Reactions for TIMP-2, MMP-2 ("MMP-2 inactivation") and dummy reactions were embedded to all regions because TIMP-2 and MMP-2 are soluble diffusing proteins (indicated by yellow arrows in Figure 61). Diffusion coefficient for TIMP-2, MMP-2, and TIMP-2:MMP-2 complexes were 10^{-12} m^2/s, which was smaller than soluble proteins in the cytoplasm, because extracellular space was assumed to be crowded with ECM proteins. In A-Cell, embedding symbols alone is not allowed. Reaction is required for embedding. So, dummy reactions were embedded instead of symbols[36]. This spatio-temporal model is constructed by embedding ECM-Deg-3(M14+fn)Java.Csim to a 3D shape model.

[36] T2 is not required in the dummy group, because it is described in other reactions. However, dummy reaction for T2 was embedded in this model. This had no effect on simulation results. However, a slight elongation of computational time of simulation, which will never be recognized, resulted.

A simulation result is shown in Figure 62.

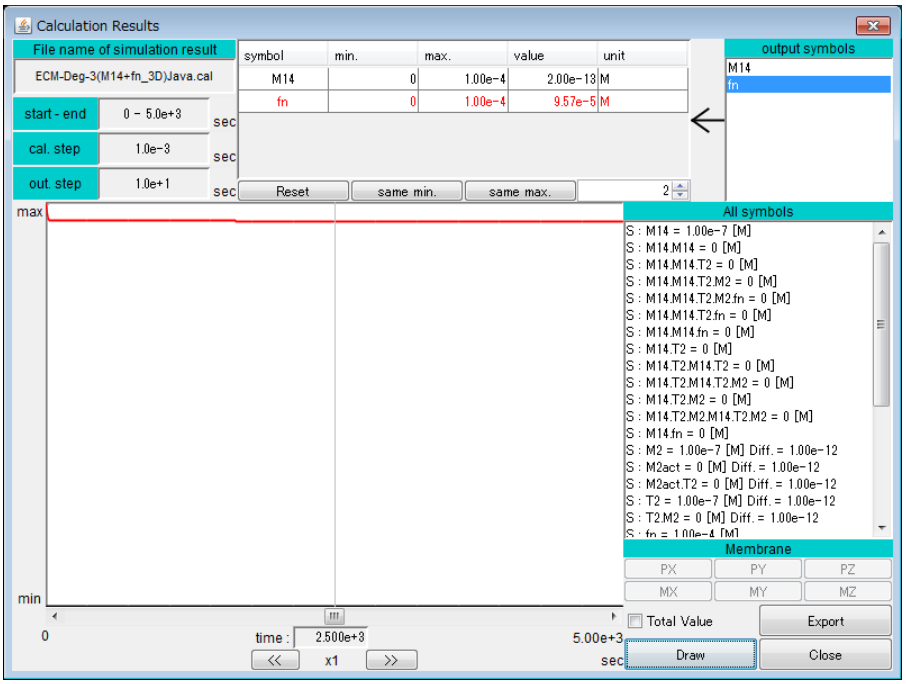


Figure 62 4D simulation result. Although all concentrations and rate constants were identical to pathway model without spatial modeling, simulation result shown here was completely different, in which there was almost no degradation if fn even after 5,000 s.

Simulation result of 4D model was completely different from that of pathway simulation. fn was not degraded even 5,000 s after the appearance of MT1-MMP. This is very different from experimental observation, and at the same time, curious because much more realistic model led to a completely different result. Why did this happen?

We hypothesized that unknown essential mechanism was missing. After an extensive research by simulation, we had found that the duration of MT1-MMP activity was as short as ~4 s. This should be the reason for ineffectiveness in the degradation of ECM, because 4s was too short for ECM degradation that required several thousands seconds. Then we hypothesized that if MT1-MMP was repetitively inserted to invadopodail membrane, ECM should be degraded effectively. So we introduced a turnover of

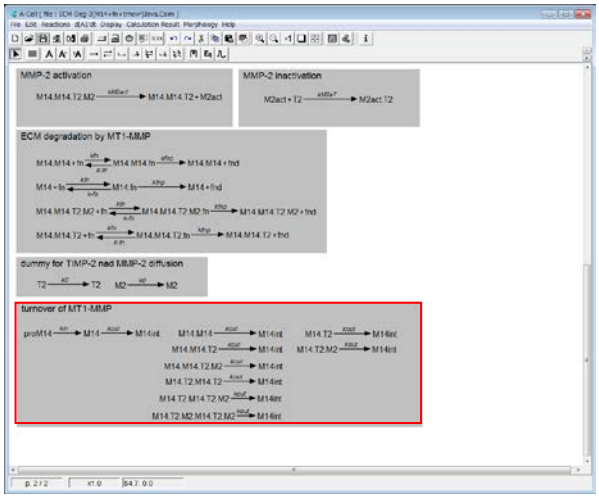


Figure 63 A model of turnover of MT1-MMP (red rectangle), which was added to the model shown in Figure 59.

MT1-MMP in the model (red rectangle Figure 63). Here we assumed that MT1-MMP and its complex that binds to fn were not engaged in the turnover. The names of the model file are ECM-Deg-3(M14+fn+trnovr)Java.Csim and ECM-Deg-3(M14+fn+trnovr_3D)Java.Vsim, which are downloadable from our homepage.

Simulation result is shown in Figure 64.

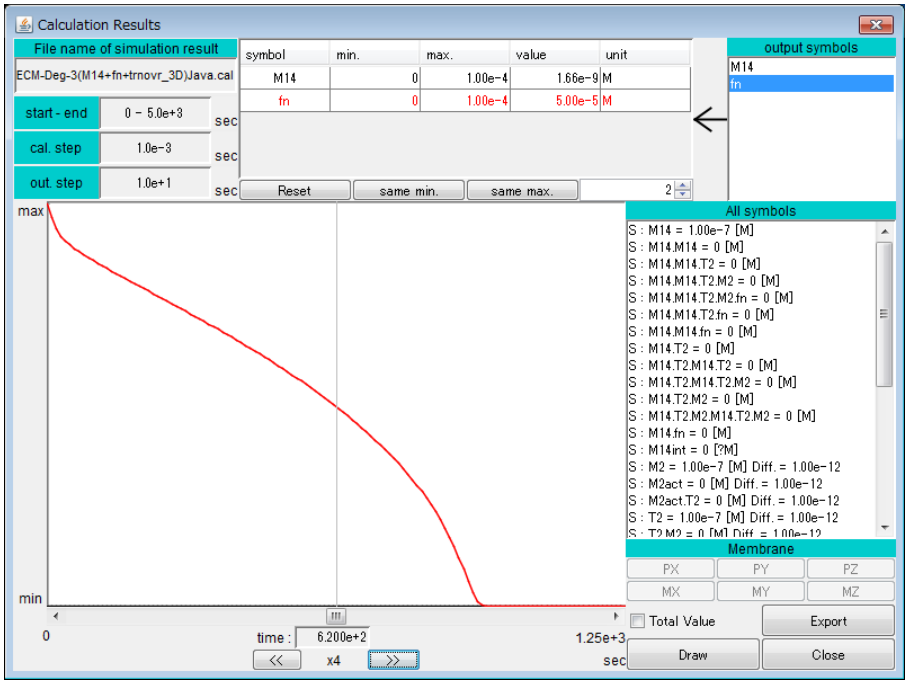


Figure 64 4D simulation result with the turnover of MT1-MMP. fn was effectively degraded, which was consistent with the experimental observation.

In this simulation, fn was effectively degraded, which was consistent with the experimental observation. This strongly suggested that the turnover of MT1-MMP was essential for the degradation of ECM, which was validated by our wet experiments (not shown here, and see our papers in a footnote 32). Modeling of MT1-MMP in Figure 63 assumed continuous turnover. Since MT1-MMP is transported by vesicles, the turnover should be pulsatile instead of continuous. We tested pulsatile insertion obtaining the same results as continuous insertion, which was also reported in our paper shown in a footnote 32.

The essential mechanism in the ECM degradation shown here is not revealed by conventional pathway simulation or readily by wet experiments. In contrast, 4D simulation could reveal the essential mechanism for MT1-MMP to degrade ECM. According to these simulations, novel therapeutic targets could be predicted. As shown in Figure 65A, there are at least three targets for preventing cancer invasion; (1) blockade of MT1-MMP, which is a common method, (2) blockade of the turnover of MT1-MMP, and (3) reduction of MT1-MMP content in a vesicle. Question was

which was the most effective in blocking ECM degradation.

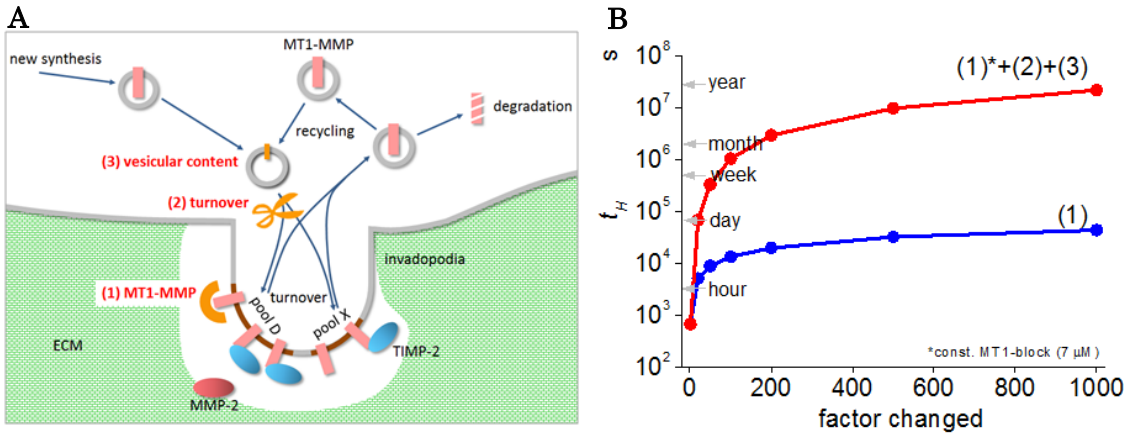


Figure 65 (A) Three potential targets, (1)-(3), for blocking ECM degradation and hence invasion, and (B) simulation results showing synergistically effective blockade of ECM degradation by combined blockade of three targets.

The answer was simple. Any single blockade of three targets was not effective in blocking ECM degradation. By combined blockade of all three targets, however, ECM was effectively blocked. Thus, a strong synergistic effect exists in simultaneous blockade of three targets leading to pronounced elongation of half-degradation time of ECM to more than 3 orders of magnitudes (Figure 65B).

This example clearly shows usefulness and importance of 4D cell simulation, by which novel druggable targets and therapeutic strategies can be derived in addition to elucidating unknown essential mechanisms in cancer invasion. Thus 4D cell simulation is a powerful tool.

In the last section of this chapter, time required for simulation is described. When a simulation for a model shown in Figure 59 was run in a notebook computer with Intel Core-i3 of 2.4Ghz, it took 2.5 h for “in situ simulation” and 41 s by “independent simulation”, respectively. Simulations shown in Figures 62 and 64 took about 2 h for each on the same notebook computer. Thus, models shown in this chapter can be run on a notebook computer.

14. Summary

A-Cell software has been developed for the purpose of using in our research. Our aim of research was to understand cellular mechanisms from chemical and physical bases. In this sense, our research was the same as traditional molecular cell biology. However, we have been using computer simulation, which is not a traditional method in molecular cell biology. The final goal of cell simulation is to construct all mechanisms found in a living cell in a computer. It must take a long time to reach this goal, and in addition, A-Cell should be extended to realizing additional functions from what are available at present. To reach this goal, novel theory and algorisms are required. We have been developing A-Cell by constructing theories and algorisms for cell simulation based on definite physical and chemical mechanisms. This is a great advantage for future extension of A-Cell, because this enables us to develop future A-Cell with scientific bases, which is more than the development of a simulator as an application.

As described in previous chapters, A-Cell is not only easy-to-use but also possesses advantages of 4D cell simulations. A-Cell is a suitable tool for the research on molecular cell biology not only for in silico researchers, but also for wet experimental researchers both in academia and in pharmaceutical companies.

Appendix A. Available Functions by Editions

Function	Edition				
	Free	Entry	Basic	Standard	Parallel
Basic operation					
New model	×	×	×	×	×
Read model	×	×	×	×	×
Save model		×	×	×	×
End	×	×	×	×	×
Discard model	×	×	×	×	×
Print model			×	×	×
Generate simulation program		×	×	×	×
Initial condition of simulation		×	×	×	×
In site simulation	×	×	×	×	×
Calculate flux			×	×	×
Symbol list	×	×	×	×	×
Cross-reference			×	×	×
Change display digits			×	×	×
Undo	×	×	×	×	×
Redo	×	×	×	×	×
Delete group	×	×	×	×	×
Copy group	×	×	×	×	×
Paste group	×	×	×	×	×
Insert page			×	×	×
Read multiple models			×	×	×
Construct multiple models			×	×	×
Enlarge	×	×	×	×	×
Reduce	×	×	×	×	×
Reset magnification	×	×	×	×	×
Copy group between models			×	×	×
Page view	×	×	×	×	×
Thumbnail view			×	×	×
Read simulation results		×	×	×	×
Construct 3D shape				×	×
View A-Cell information	×	×	×	×	×
Modeling operation					
Group	×	×	×	×	×
Symbol	×	×	×	×	×
Add symbol to left	×	×	×	×	×
Add symbol to right	×	×	×	×	×
1st order forward reaction	×	×	×	×	×
1st order equilibrium reaction	×	×	×	×	×
2nd order forward reaction	×	×	×	×	×
2nd order equilibrium reaction	×	×	×	×	×
Forward catalytic reaction	×	×	×	×	×
Equilibrium catalytic reaction	×	×	×	×	×
Stimuli			×	×	×
Equations & differential equations			×	×	×
Membrane & action potential			×	×	×

Function	Edition				
	Free	Entry	Basic	Standard	Parallel
Simulation program					
Automatic generation		×	×	×	×
Export in A-Cell format		×	×	×	×
Export in CSV		×	×	×	×
in situ simulation					
Run simulation	×	×	×	×	×
Modify stimuli			×	×	×
Export in A-Cell format			×	×	×
Export in CSV		×	×	×	×
Temporal cursor		×	×	×	×
Initial condition					
Modify stimuli			×	×	×
Viewing simulation result					
Export in CSV		×	×	×	×
4D simulation program					
Automatic generation				×	×
openMP parallelization					×

Brief description of editions

free: Model construction and in situ simulation are available, but saving a model and simulation result are not allowed. 4D simulation and modeling of equation, stimulation, and membrane potential are not allowed too.

entry: Number of page of a model file is limited to 1 page. However, independent simulation and reading simulation results are allowed in addition to in situ simulation. Equation, stimulation, and membrane potential models are not allowed as in free edition.

basic: All A-Cell functions except 4D simulation and parallelized simulation are available.

standard: All A-cell functions except parallelized simulation are available. 4D modeling and simulation are available.

parallel: Parallelized simulation program for 4D simulation can be generated, by which acceleration of running simulation is realized.