ISSN 1392–124X (print), ISSN 2335–884X (online) INFORMATION TECHNOLOGY AND CONTROL, 2014, T. 43, Nr. 2

Continuous Time Markov Chain Models of Voltage Gating of Gap Junction Channels

Henrikas Pranevicius

Department of Applied Informatics, Vytautas Magnus University,
Department of Business Informatics Research in Systems, Kaunas University of Technology,
Studentų St. 56, LT – 51424 Kaunas, Lithuania,
e-mail: henrikas.pranevicius@ktu.lt

Mindaugas Pranevicius

Department of Anestesiology, Albert Einstein College of Medicine,
1300 Morris Park Ave., Bronx, NY 10461, USA,
e-mail: pranevicius@gmail.com

Osvaldas Pranevicius

Department of Anestesiology, New York Hospital Queens,
56-45 Main Street, Flushing, NY 11355, USA,
e-mail: opranevicius@aol.com

Mindaugas Snipas

Department of Business Informatics Research in Systems and
Department of Mathematical Research in Systems, Kaunas University of Technology
Studentų St. 50, LT – 51368 Kaunas, Lithuania
e-mail: minsnip@ktu.lt

Nerijus Paulauskas

Institute of Cardiology, Lithuanian University of Health Sciences,
Department of Business Informatics, Kaunas University of Technology,
Studentų St. 56, LT – 51424 Kaunas, Lithuania,
e-mail: ne.paulauskas@gmail.com

Feliksas Bukauskas

Department of Neuroscience, Albert Einstein College of Medicine,
1300 Morris Park Ave., Bronx, NY 10461, USA,
e-mail: feliksas.bukauskas@einstein.yu.edu

crossref http://dx.doi.org/10.5755/j01.itc.43.2.3198

Abstract. The major goal of this study was to create a continuous time Markov chain (CTMC) models of voltage gating of gap junction (GJ) channels formed of connexin protein. This goal was achieved by using the Piece Linear Aggregate (PLA) formalism to describe the function of GJs and transforming PLA into Markov process. Infinitesimal generator of CTMC was used to automate construction of Markov chain model from description of the system using PLA formalism. Developed Markov chain models were used to simulate gap junctional conductance dependence on transjunctional voltage. The proposed method was implemented to create models of voltage gating of GJ channels containing 4 and 12 gates. CTMC modeling results were compared with the results obtained using a discrete time Markov chain (DTMC) model. It was shown that CTMC modeling requires less CPU time than an analogous DTMC model.

Keywords: Continuous time Markov chain; PLA formalism; gap junction channel; steady-state probabilities.

1. Introduction

Connexins (Cxs) is large family of integral membrane proteins that provide a direct pathway for electrical and metabolic signaling between cells [2]. 21 Cx isoforms in humans [14] form gap junction (GJ) channels. Each GJ channel is composed of two hemichannels (HCs), each oligomerized of six Cxs. Cxs have four alpha helical transmembrane domains (M1 to M4), intracellular N- and C-termini (NT and CT), two extracellular loops (E1 and E2), and a cytoplasmic loop (CL) [15]. Docking of HCs from neighboring cells leads to formation of the GJ channels composed of 12 Cxs.

Sensitivity to transjunctional voltage (V_j), called voltage-gating, appears to be common to all GJ channels. Symmetric reductions in junctional conductance (g_j) for either polarity of V_j have been explained by the presence of a V_j-sensitive gate in each apposed hemichannel [1]. Gap-junctional communication plays important roles in many processes, such as impulse propagation in the heart, communication between neurons and glia, metabolic exchange between cells in the lens lacking blood circulation, organ formation during development, and regulation of cell proliferation.

Earlier, we developed stochastic 4- and 16-state models of voltage gating, containing 2 and 4 gates in series in each GJ channel, respectively. These models contain a certain number (>10) of parameters and to estimate them global optimization (GO) algorithms should be used [4]. Typically, thousands of iterations should be used in performing GO to estimate a global minimum. If a single iteration of the model lasts up to 10 s, then a search for a global minimum can take several hours or days. Thus, the reduction of computation time necessary to perform a single simulation is an important task.

Preliminary studies showed [6] that modeling of GJ channels gating using the Markov chain formalism requires over 100 – fold less CPU time than a simulation using DTMC model [5,13] describing the GJ channel containing 12 gates. In this model, differently from 4- and 16-state models, it is assumed that each connexin protein of GJ channel contain the gate. Since all 12 gates operate at the same time, construction of the transition matrix is not a trivial task. Therefore, transition matrix **P** is dense, and the run-time complexity of calculation of steady-state probabilities is $O(n^3)$ if direct methods, i.e. Gaussian elimination, are applied.

In this study, we use CTMC, instead of DTMC, to model gating of GJ channels. The CTMC model has an advantage over DTMC, because construction of infinitesimal generator (transition rates matrix) is relatively easy, since no more than a few transitions can happen at any state of an infinitesimal time period. Moreover, the run-time complexity of a steady-state solution is significantly lower than with DTMC, since the infinitesimal generator matrix **Q** is sparse. In some cases, matrix **Q** is tridiagonal, which allows achieving run-time complexity in the order of $O(n)$ to calculate steady state probabilities.

We used **Piece Linear Aggregate** formalism (PLA) [11, 12] in order to describe system behavior and to create an infinitesimal generator matrix of CTMC model of the GJ channel. Formal specification can be applied for verification of liveness of CTMC model, using special tools, e.g., Simple Promela Interpreter (SPIN) model checker [3, 8, 9].

PLA, in essence, is equivalent to piece-linear Markov process. If duration of operations used in PLA specifications are distributed by exponential law, then the piece-linear Markov process becomes a linear CTMC process with a discrete set of states. These presumptions allow transforming a PLA specification to the specification of Markov processes and the automatic creation of a state-space graph of the analyzed system.

As reported earlier [7], automated creation of an infinitesimal generator matrix can be achieved in the following steps: 1) formal specification of the system, 2) creation of state-space graph, and 3) state-space graph transformation into the infinitesimal generator matrix. PLA formalism and its theoretical background are presented in Section 2. CTMC models of GJ channels are presented in Section 3. We also present formal specification of CTMC models using PLA formalism.

2. PLA formalism for creation of continuous time Markov chain models

PLA formalism is widely used for formal modelling and creation of models of complex systems. In this section, a brief description of PLA formalism is presented. We also define necessary conditions allowing transformation of PLA model into CTMC.

2.1. PLA formalism

Using the aggregate approach, the system can be represented as a set of interacting PLAs. The PLA is characterized by a set of states $z \in Z$, input signals $x \in X$, and output signals $y \in Y$, which varies over a set of time moments $t \in T$. To describe proper changes of PLA properties over time, transition H and output G operators must be known.

The state $z \in Z$ of the PLA is the same as the state of a piece-linear Markov process, i.e. $z(t) = (\upsilon(t),\ z_\upsilon(t))$, where $\upsilon(t)$ is a discrete state component taking values on a countable set of values, and $z_\upsilon(t)$ is a continuous component comprising of $z_{\upsilon 1}(t), z_{\upsilon 2}(t), \ldots, z_{\upsilon k}(t)$ coordinates.

When there are no inputs, the state of the aggregate changes in the following manner:

$$\upsilon(t) = \text{const}, \quad \frac{dz_\upsilon(t)}{dt} = -\alpha_\upsilon, \tag{1}$$

where $\alpha_\upsilon = (\alpha_{\upsilon 1}, \alpha_{\upsilon 2}, \ldots, \alpha_{\upsilon k})$ is a constant vector.

The state of the aggregate can change when an input signal arrives or when a continuous component acquires a definite value.

Controlling sequence approach permits to define the continuous coordinates of PLA as follows:

$$z_\nu(t_m) = \{w(e''_1, t_m), w(e''_2, t_m), \ldots, w(e''_f, t_m)\}; \tag{2}$$

where $w(e''_i, t_m)$ is the time moment in which the event e''_i occurs.

When the state of the system is known ($z(t_m)$, m = 0, 1, 2, …), then the moment, t_{m+1}, is determined by an input signal arrival or by the following equation:

$$t_{m+1} = \min\{w(e''_i, t_m)\},\ 1 \le i \le f. \tag{3}$$

The class of the next event, e_{m+1}, is determined by an input signal if it arrives at the time moment t_{m+1} or by the control coordinate, which acquires minimal value at the moment t_m, i.e. if $w(e''_i, t_m)$ acquires minimal value, then $e_{m+1} = e''_i$.

The operator H conditions the new state:

$$z(t_{m+1}) = H[z(t_m), e_i],\ e_i \in E' \cup E''. \tag{4}$$

The output signal y_i from the set of output signals ($Y = \{y_1, y_2, \ldots, y_m\}$) can be generated by an aggregate only at moments of events from the subsets of internal and external events – E' and E'', respectively. The operator G determines the content of the output signals:

$$y = G[z(t_m), e_i],\ e_i \in E' \cup E'',\ y \in Y. \tag{5}$$

PLA can be changed to Markov process with continuous time discrete state if durations of operations are distributed according to exponential law

$$F_i(t) = P(\xi^i_j < t) = 1 - e^{-\lambda_i t}, \tag{6}$$

where $1/\lambda_i$ is an average duration of the i^{th} operation, k is the number of active operations at the state z(t). Probability, that an event will occur at system state z(t), is equal to

$$P(t_{m+1} - t_m < t) = 1 - e^{-t\sum_k \lambda_k}. \tag{7}$$

Then, it follows that

$$z(t) = \{\gamma(t); w(e''_1, t), \ldots, w(e''_f, t)\}, \tag{8}$$

here $\gamma(t)$ is the discrete component of the state, and $w(e''_i, t)$ is defined as follows:

$$w(e''_i, t) = \begin{cases} \lambda_i, \text{ if the } i^{th} \text{ operation is active at moment } t; \\ 0, \text{ otherwise.} \end{cases} \tag{9}$$

Meta-model of PLA formalism is presented below and is described using the UML notation as reported in [10].

The class diagram is demonstrated in Fig. 1. It includes inputs, outputs as well as discrete and continuous variables of an aggregate.

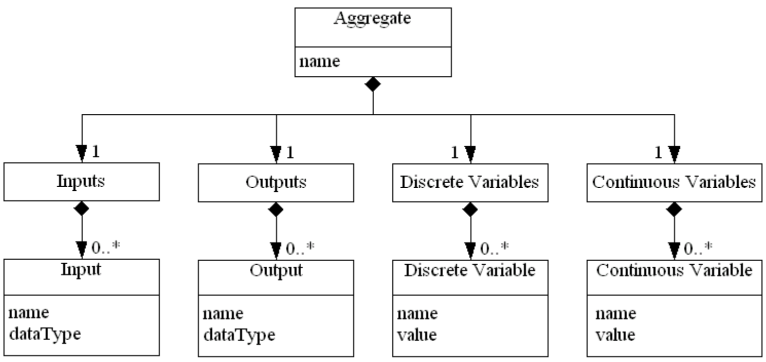

Figure 1. The schematic structure of the aggregate

Continuous variables are used to describe moments during which the internal events occur after operations terminate.

Input signals cause external events.

The modeling system consists of interconnected aggregates as shown in Fig. 2.

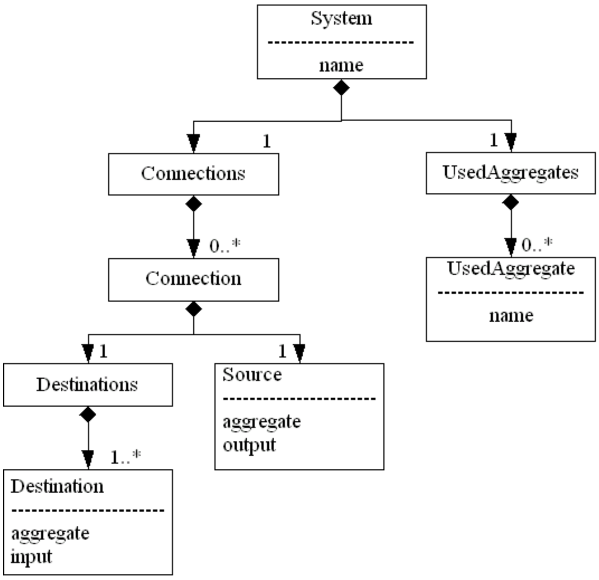

Figure 2. The system of aggregates interacting throughout shown connections

Fig. 3 shows the structure of input and output signals' data.

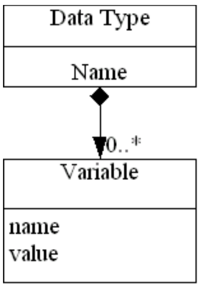

Figure 3. The structure of input and output signals

2.2. Generation of state graph of CTMC

The main concept for generation of open and closed states of gates from PLA, referred as a *state graph,* is based on the fact that the subset of internal events, initiating transition among states, is known in every state.

The states graph of CTMC is described as $G=(Z,\Lambda)$, where Z is a set of system states and Λ is a set of transition rates ($\Lambda : Z \times Z \to \Re_0^+$).

The generation of the Markov chain graph involves the following steps:

1. Collection of information about:
   1.1. The set of internal events *E*.
   1.2. The initial state of the system *z*(0).
   1.3. The set of generated states of the system Z_g, which initially has only one member, *z*(0), i.e., $Z_g = \{z(0)\}$.
   1.4. The set of analyzed states Z_a, which initially is empty, i.e., $Z_a = \varnothing$.
2. The search of neighboring states:
   2.1. Choose a non-analyzed state $z \in Z_g \setminus Z_a$
   2.2. Search for $\forall e \in E \mid \omega(e,t) > 0$, which can occur in the state *z:*
      2.2.1. Generate the set of states Z^+ into which the system can pass in a single step: $Z^+ = \{z^+ \mid z^+ = H(z,e_i), \forall e_i : w(e_i) > 0\}$;
      2.2.2. $\forall z^+ \in Z^+$:
         2.2.2.1. Add z^+ to Z_a, i.e., $Z_a := Z_a \cup \{z^+\}$
         2.2.2.2. Add the new transition rate from state z to z^+, i.e., $\Lambda := \Lambda \cup \{w(e_i)\}$.
3. Testing criteria defining termination of the algorithm: $Z_a = Z_g$.

3. The model of the GJ channel containing 12 connexins

Here we present CTMC model of the GJ channel consisting of two gates in series, each containing six subgates, one per connexin. We used numerical methods for calculation of steady-state probabilities for different values of V_js and obtained data compared with results acquired using DTMC model [13].

3.1. Conceptual model of the GJ channel containing 12 connexins

Gap junctions form clusters (junctional plaques) of individual channels arranged in parallel in the junctional membrane of two adjacent cells. The GJ channel is composed of 2 hemichannels (left and right) arranged in series. Each hemichannel is composed/oligomerized from six Cxs forming a hexamer with the pore inside. We envision that each hemichannel forms the gate, which is composed of six subgates arranged in parallel, i.e. to each connexin the subgate is attributed and the GJ channel contains two gates and 12 subgates. Each subgate operates between open (*o*) and closed (*c*) states. For simplicity, we assume that only subgates in the left hemichannel operate [4], while subgates in the right hemichannel are always open (see Fig. 4).

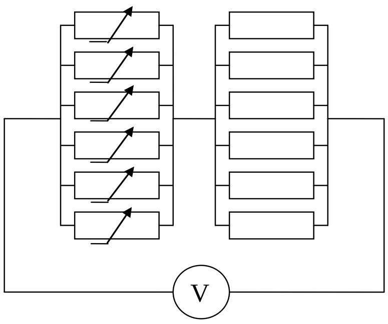

Figure 4. Electrical scheme of the GJ channel composed of two hemichannels each formed of 6 connexins. Transjunctional voltage (V_j) controls both hemichannels from which only Cxs in the left hemichannel operate between open and closed states, while Cxs in the right hemichannel are always open

The GJ channel gates in response to V_j by performing $o \leftrightarrow c$ transitions for each subgate. Each subgate has a possibility for four transitions as shown in Fig. 5:

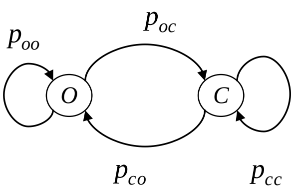

Figure 5 The graph illustrating open (*o*) and closed (*c*) states of the gate and probabilities of transitions

As reported earlier [5], probabilities of shown transitions can be described as follows:

$$p_{oc}(A,P,V_{left},V_0) = \frac{K \cdot k(A,P,V_{left},V_0)}{1+k(A,P,V_{left},V_0)}, \quad (10)$$

$$p_{oo}(A,P,V_{left},V_0) = 1 - p_{oc}(A,P,V_{left},V_0), \quad (11)$$

$$p_{co}(A,P,V_{left},V_0) = \frac{K}{1+k(A,P,V_{left},V_0)}, \quad (12)$$

$$p_{cc}(A,P,V_{left},V_0) = 1 - p_{co}(A,P,V_{left},V_0). \quad (13)$$

In (10) and (12), k is

$$k(P,V_{left},V_0) = e^{A\cdot(P\cdot V_{left}-V_0)}, \quad (14)$$

where P is a gating polarity (+1 or -1); A is a coefficient characterizing gating sensitivity to voltage (1/mV); K is a constant used to change kinetics of $c \leftrightarrow o$ transitions (K can accelerate or decelerate $c \leftrightarrow o$ transitions but does not affect conditions of the steady state); V_o is a voltage across the hemichannel/connexin at which probabilities for *o* and *c* states are equal (mV); V_{left} is variable voltage across the subgate (mV).

Each subgate, depending on a voltage across it ($V_{left/right}$), can gate by changing stepwise between the open state with conductance g_o and the closed state with conductance g_c. It was assumed that g_o and g_c values rectify, i.e., depend on $V_{left/right}$ exponentially:

$$g_o(V_{left/right},P) = g_{o,V=0} \cdot e^{\frac{P\cdot V_{left/right}}{Ro}},$$

$$g_c(V_{left},P) = g_{c,V=0} \cdot e^{\frac{P\cdot V_{left}}{Rc}}, \quad (15)$$

where $V_{left/right}$ is a voltage across the left or right hemichannel, $g_{o,V=0}$ and $g_{c,V=0}$ are conductances at $V_{left/right}=0$, and Ro and Rc are rectification constants.

The conductance of the left hemichannel, when n Cxs are closed, can be described as follows

$$g_{left}(n) = n \cdot g_c(V_{left}(n),P) + (6-n)\cdot g_o(V_{left}(n),P). \quad (16)$$

Similarly, the conductance of the right hemichannel is

$$g_{right}(n) = 6 \cdot g_o(V_{left}(n),P). \quad (17)$$

During gating, conductances of subgates range between $g_o(V_{left/right},P)$ and $g_c(V_{left},P)$, and the total conductance of the GJ channel can be found using steady-state probabilities of Markov chain model of the left hemichannel (see the section 3.2):

$$g_{left} = \sum_{n=0}^{6} \pi_n \cdot g_{left}(n), \quad (18)$$

where π_n is a steady-state probability for n Cxs in the left hemichannel to be closed.

Conductance of the GJ channel depends on the voltage, i.e. the circuit is nonlinear. In order to calculate voltage across each Cx, we used an iterative procedure [13]. We assumed that the value of voltage is settled, if a difference between voltage values, calculated at two consecutive iterations is less than 0.1 %. Calculation showed that no more than 5 iterations were needed to achieve aforesaid precision.

3.2. CTMC model of the GJ channel containing 12 connexins

In order to create a CTMC model of GJ, we use the following relation in describing probabilities and rates of subgate's transitions (see Fig. 6):

$$\lambda_{\grave{o}c} = \frac{p_{oc}}{\tau} \text{ and } \lambda_{co} = \frac{p_{co}}{\tau}, \qquad (19)$$

where τ is a short period of time, in which the probability to observe multiple transitions is negligible, i.e. for $i \neq j$, $p_{ij}(\tau) \to 0$ if $\tau \to 0$.

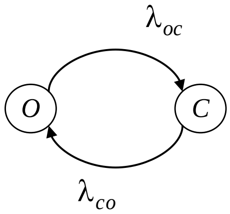

Figure 6. The graph illustrating open (*o*) and closed (*c*) states of a gate with transition rates

Assuming a CTMC model allows using PLA formalism for automatic model creation, an aggregate specification of the continuous time Markov chain model of GJ channel is presented below.

1. The set of input signals: $X = \emptyset$.
2. The set of output signals: $Y = \emptyset$.
3. The set of external events: $E' = \emptyset$.
4. The set of internal events: $E'' = \left\{ e_1'', e_2'' \right\}$,

where e_1'' is a transition from the closed to the open state of the subgate in the left hemichannel; e_2'' is a transition of the subgate in the left hemichannel from the open to the closed state.

5. The transition rates between states of the system: $e_1'' \mapsto n_l(t) \cdot \lambda_{co}(V_{left}(n_l(t)))\; e_2'' \mapsto (6 - n_l(t)) \cdot \lambda_{oc}(V_{left}(n_l(t)))$.
6. The discrete component of the state: $v(t) = \{n_l(t)\}$; $n_l(t) = \overline{0,6}$, where $n_l(t)$ is the number of Cxs in closed state in the left hemichannel.
7. The continuous component of the state:

$z_v(t) = \{w(e_1'', t), w(e_2'', t)\}$.

8. Initial state of the system: $z(t) = \{0, 0, 6\lambda_{oc}(V_{left}(0))\}$.
9. Internal transition operators:

$H(e_1'')$: / transition from closed to open state in the left hemichannel /

$$n_l(t+0) = \begin{cases} n_l(t) - 1, \; if \quad n_l(t) > 0, \\ n_l(t), \quad otherwise; \end{cases}$$

$$w(e_1'', t+0) = (n_l(t) - 1) \cdot \lambda_{co}(V_{left}(n_l(t) - 1));$$

$$w(e_2'', t+0) = (7 - n_l(t)) \cdot \lambda_{oc}(V_{left}(n_l(t) + 1));$$

$H(e_2'')$: / transition from open to closed state in the left hemichannel /

$$n_l(t+0) = \begin{cases} n_l(t) - 1, \; if \quad n_l(t) > 0, \\ n_l(t), \quad otherwise; \end{cases}$$

$$w(e_1'', t+0) = (n_l(t) - 1) \cdot \lambda_{co}(V_{left}(n_l(t) - 1));$$

$$w(e_2'', t+0) = \begin{cases} (n_l(t) - 1) \cdot \lambda_{oc}^{(l)}(V_{left}(n_l(t) - 1)), \; if \quad n_l(t) > 1, \\ 0, \; otherwise. \end{cases}$$

Aggregate specification can be applied to automatically construct the infinitesimal generator $\mathbf{Q}$. Formation of matrix $\mathbf{Q} = (q_{ij}),\; i, j = \overline{1, n},$ by the aggregate specification is achieved using the following relations:

$$n_l(t) \mapsto i; \quad n_l(t+0) \mapsto j; \quad w(e_1''(t)) \mapsto q_{ij}; \qquad (20)$$

where i is row index; j is column index; q_{ij} is entry of the matrix $\mathbf{Q}$.

The infinitesimal generator matrix $\mathbf{Q}^{(12)}$ of CTMC model of the GJ channel with 12 gates is as follows

$$\mathbf{Q}^{(12)} = \begin{pmatrix} * & 6\lambda_{oc} & 0 & 0 & 0 & 0 & 0 \\ \lambda_{co} & * & 5\lambda_{oc} & 0 & 0 & 0 & 0 \\ 0 & 2\lambda_{co} & * & 4\lambda_{oc} & 0 & 0 & 0 \\ 0 & 0 & 3\lambda_{co} & * & 3\lambda_{oc} & 0 & 0 \\ 0 & 0 & 0 & 4\lambda_{co} & * & 2\lambda_{oc} & 0 \\ 0 & 0 & 0 & 0 & 5\lambda_{co} & * & \lambda_{oc} \\ 0 & 0 & 0 & 0 & 0 & 6\lambda_{co} & * \end{pmatrix} \qquad (21)$$

where diagonal entries (denoted as *) are equal to the negated sum of the non-diagonal entries in that row. Transition rates of the matrix $\mathbf{Q}^{(12)}$ in (21) depend on the voltage across the left and right hemichannels, i.e. $\lambda_{oc} = \lambda_{oc}(V_{left}(n_l))$ and $\lambda_{co} = \lambda_{co}(V_{left}(n_l))$.

Since infinitesimal generator $\mathbf{Q}^{(12)}$ is a tridiagonal matrix, it can be stored in compact format as shown in Figure 7.

Memory requirements to store the entire infinitesimal generator matrix $\mathbf{Q}$ are equal to n^2, while the compact storage scheme requires to store only $3n$ elements.

3.3. The comparison of numerical solution and results of DTMC and CTMC models

The steady-state solution of vector $\boldsymbol{\pi}$ of CTMC can be found from

$$\boldsymbol{\pi} \cdot \mathbf{Q} = \mathbf{0}, \qquad (22)$$

where $\mathbf{Q}$ is infinitesimal matrix of transition rates, describing a continuous time Markov chain; $\mathbf{0}$ denotes a zero row vector of length n.

Since $\mathbf{Q}$ is a singular matrix ($rank(\mathbf{Q}) = n - 1$), an additional condition is used to obtain the unique solution

$$\sum_{i=1}^{n} \pi_i = 1. \qquad (23)$$

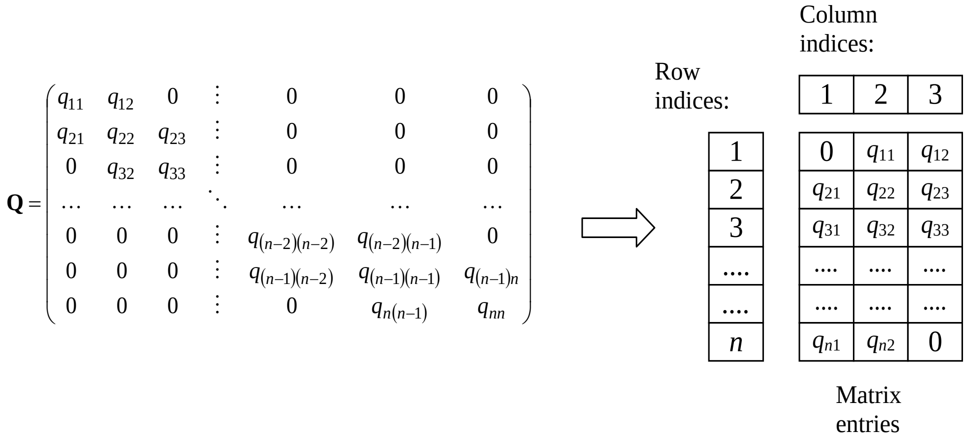

Figure 7. Compact storage scheme of an infinitesimal generator **A**

The transition probability matrix **P** for DTMC model of the GJ channel is dense [13], i.e. the matrix **P** consists of nonzero entries. Therefore, the matrix **P** must be stored in a two-dimensional array of size n^2, and the run-time complexity of a direct algorithm (e.g., Gaussian elimination) for calculating the steady-state solution is equal to $O(n^3)$ [16].

Conversely, the infinitesimal generator matrix **Q** for CTMC model of the gap junction channel is a tridiagonal matrix (21). In that case, the run-time complexity of an algorithm for solving (22) is equal to $O(n)$ [16]. It can be achieved by using the following recursive procedure:

$$\begin{cases} r_1 = 1; \\ r_2 = -\dfrac{r_1 \cdot q_{11}}{q_{21}}; \\ r_i = -\dfrac{r_{i-2} \cdot q_{(i-2)(i-1)} + r_{i-1} \cdot q_{(i-1)(i-1)}}{q_{i(i-1)}}, \; i = \overline{3,n}; \\ \pi_i = \dfrac{r_i}{\sum_{i=1}^{n} r_i}, \; i = \overline{1,n}. \end{cases} \quad (24)$$

Recursive procedure (24) can easily be implemented if the infinitesimal generator **Q** is stored in compact format. In that case, j indices of entries q_{ij} must be replaced as follows:

$$j = \begin{cases} 1, & if \quad i - j = 1; \\ 2, & if \quad i - j = 0; \\ 3, & if \quad i - j = -1. \end{cases} \quad (25)$$

We calculated conductance of GJ channel at different V_j values using compact storage schemes and system of equations indicated as (24) to find steady-state probabilities of CTMC. The results were compared to the results of DTMC model presented in [13]. Calculation was performed using a PC with Intel Core i5-3450 CPU @ 3.09 GHz with 4 cores, 3.41 GB of RAM available. We used MATLAB programming language in order to form transition matrix (for DTMC) or infinitesimal generator (for CTMC), to estimate steady state probabilities and to calculate conductance at single voltage value. Modeling results were identical (with 0.0001 precision) to the modeling results obtained by DTMC model [13], but CTMC modeling required significantly less CPU time. It required 15.2 ms on average to calculate conductance of DTMC model at chosen voltage value, while the same calculation took 0.49 ms if CTMC model was used.

3.4. CTMC model of the GJ channel containing 4 gates

The GJ channel is composed of two hemichannels (left and right), with two gates (k1 and k2) in the left and two gates (k3 and k4) in the right hemichannel (see Fig. 8). Each connexin can be in two states, open and closed. In this model, we assume that all gates operate in response to applied voltage.

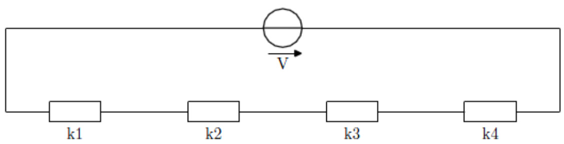

Figure 8 Electrical scheme of the GJ channel composed of two hemichannels each containing two gates

Gating probabilities and conductance of open and closed gates can be calculated as described using equations (10)-(15). Conductance of the GJ channel with four gates, depending on the number of open and closed gates on the left and right side, can be found from:

$$g_{m,n} = \frac{1}{(m+n)\frac{1}{g_c} + (4-m-n)\frac{1}{g_o}}, \quad (26)$$

where m ($m = \overline{0,2}$) is the number of closed gates on the left side; n ($n = \overline{0,2}$) is the number of closed gates on the right side.

Stationary conductance g of the GJ can be found from

$$g = \sum_{m=0}^{2} \sum_{n=0}^{2} \pi_{m,n} \cdot g_{m,n}, \tag{27}$$

where $\pi_{m,n}$ is a steady-state probability for m and n numbers of gates closed in the left and right hemichannel, respectively.

An aggregate specification of the continuous time Markov chain model of GJ channel containing 4 gates is presented below:

1. The set of input signals: $X = \varnothing$.
2. The set of output signals: $Y = \varnothing$.
3. The set of external events: $E' = \varnothing$.
4. The set of internal events: $E'' = \{ e''_1, e''_2, e''_3, e''_4 \}$,

where e''_1 - transition from closed to open state in the left hemichannel,
e''_2 - transition from open to closed state in the left hemichannel,
e''_3 - transition from closed to open state in the right hemichannel,
e''_4 - transition from open to closed state in the right hemichannel.

5. The transition rates between states of the system:

$e''_1 \to (2 - n_l(t)) \cdot \lambda_{co}^{(l)}(V_{left}(n_l(t)))$, $e''_2 \to n_l(t) \cdot \lambda_{oc}^{(l)}(V_{left}(n_l(t)))$, $e''_3 \to (2 - n_r(t)) \cdot \lambda_{co}^{(r)}(V_{right}(n_l(t)))$, $e''_4 \to n_r(t) \cdot \lambda_{oc}^{(r)}(V_{right}(n_l(t)))$.

6. The discrete component of the state: $v(t) = \{ n_l(t), n_r(t) \}$; $n_l(t) = \overline{0,2}$; $n_r(t) = \overline{0,2}$,

where $n_l(t)$ is the number of connexins in closed state in the left hemichannel;
$n_r(t)$ is the number of connexins in closed state in the right hemichannel

7. The continuous component of the state: $z_v(t) = \{ w(e''_1, t), w(e''_2, t), w(e''_3, t), w(e''_4, t) \}$.
8. Initial state of the system: $z(t) = \{ 0,0, <\infty, \infty, <\infty, \infty \}$.
9. Internal transition operators:

$H(e''_1)$: / transition from closed to open state in the left hemichannel /

$$n_l(t+0) = \begin{cases} n_l(t) - 1, & if \quad n_l(t) > 0, \\ n_l(t), & otherwise; \end{cases}$$

$$n_r(t+0) = n_r(t);$$

$$w(e''_1, t+0) = \begin{cases} (n_l(t) - 1) \cdot \lambda_{co}^{(l)}(V_{left}(n_l(t) - 1)), & if \quad n_l(t) > 0, \\ 0, & otherwise; \end{cases}$$

$$w(e''_2, t+0) = \begin{cases} (3 - n_l(t)) \cdot \lambda_{oc}^{(l)}(V_{left}(n_l(t) - 1)), & if \quad n_l(t) > 0, \\ w(e''_2, t), & otherwise; \end{cases}$$

$$w(e''_3, t+0) = w(e''_3, t);$$

$$w(e''_4, t+0) = w(e''_4, t).$$

$H(e''_2)$: / transition from open to closed state in the left hemichannel /

$$n_l(t+0) = \begin{cases} n_l(t) + 1, & if \quad n_l(t) < 2, \\ n_l(t), & otherwise; \end{cases}$$

$$n_r(t+0) = n_r(t);$$

$$w(e''_1, t+0) = \begin{cases} (n_l(t) + 1) \cdot \lambda_{co}^{(l)}(V_{left}(n_l(t) + 1)), & if \quad n_l(t) < 2, \\ w(e''_1, t), & otherwise; \end{cases}$$

$$w(e''_2, t+0) = \begin{cases} (1 - n_l(t)) \cdot \lambda_{oc}^{(l)}(V_{left}(n_l(t) + 1)), & if \quad n_l(t) < 2, \\ 0, & otherwise; \end{cases}$$

$$w(e''_3, t+0) = w(e''_3, t);$$

$$w(e''_4, t+0) = w(e''_4, t).$$

$H(e''_3)$: / transition from closed to open state in the right hemichannel /

$$n_l(t+0) = n_l(t);$$

$$n_r(t+0) = \begin{cases} n_r(t) - 1, & if \quad n_r(t) > 0, \\ n_r(t), & otherwise; \end{cases}$$

$$w(e''_1, t+0) = w(e''_1, t);$$

$$w(e''_2, t+0) = w(e''_2, t);$$

$$w(e''_3, t+0) = \begin{cases} (n_r(t) - 1) \cdot \lambda_{co}^{(r)}(V_{right}(n_r(t) - 1)), & if \quad n_r(t) < 0, \\ 0, & otherwise; \end{cases}$$

$$w(e''_4, t+0) = \begin{cases} (3 - n_r(t)) \cdot \lambda_{oc}^{(r)}(V_{right}(n_r(t) - 1)), & if \quad n_r(t) > 0, \\ w(e''_4, t), & otherwise. \end{cases}$$

$H(e''_4)$: / transition from open to closed state in the right hemichannel /

$$n_l(t+0) = n_l(t);$$

$$n_r(t+0) = \begin{cases} n_r(t) + 1, & if \quad n_r(t) < 2, \\ n_r(t), & otherwise; \end{cases}$$

$$w(e''_1, t+0) = w(e''_1, t);$$

$$w(e''_2, t+0) = w(e''_2, t);$$

$$w(e''_3, t+0) = \begin{cases} (n_r(t) + 1) \cdot \lambda_{co}^{(r)}(V_{rightt}(n_r(t) + 1)), & if \quad n_r(t) < 2, \\ w(e''_3, t), & otherwise; \end{cases}$$

$$w(e''_4, t+0) = \begin{cases} (1 - n_r(t)) \cdot \lambda_{oc}^{(r)}(V_{right}(n_r(t) + 1)), & if \quad n_r(t) < 2, \\ 0, & otherwise. \end{cases}$$

3.5. Modelling results of the GJ channel model containing 4 gates

An infinitesimal generator matrix $\mathbf{Q}^{(4)}$ of CTMC model of the GJ channel is as follows:

$$\mathbf{Q}^{(4)} = \begin{pmatrix} * & 2\lambda_{oc}^{(r)} & 0 & 2\lambda_{oc}^{(l)} & 0 & 0 & 0 & 0 & 0 \\ \lambda_{co}^{(r)} & * & \lambda_{oc}^{(r)} & 0 & 2\lambda_{oc}^{(l)} & 0 & 0 & 0 & 0 \\ 0 & 2\lambda_{co}^{(r)} & * & 0 & 0 & 2\lambda_{oc}^{(l)} & 0 & 0 & 0 \\ \lambda_{co}^{(l)} & 0 & 0 & * & 2\lambda_{oc}^{(r)} & 0 & \lambda_{oc}^{(l)} & 0 & 0 \\ 0 & \lambda_{co}^{(l)} & 0 & \lambda_{co}^{(r)} & * & \lambda_{oc}^{(r)} & 0 & \lambda_{oc}^{(l)} & 0 \\ 0 & 0 & \lambda_{co}^{(l)} & 0 & 2\lambda_{co}^{(r)} & * & 0 & 0 & \lambda_{oc}^{(l)} \\ 0 & 0 & 0 & 2\lambda_{co}^{(l)} & 0 & 0 & * & 2\lambda_{oc}^{(r)} & 0 \\ 0 & 0 & 0 & 0 & 2\lambda_{co}^{(l)} & 0 & \lambda_{co}^{(r)} & * & \lambda_{oc}^{(r)} \\ 0 & 0 & 0 & 0 & 0 & 2\lambda_{co}^{(l)} & 0 & 2\lambda_{co}^{(r)} & * \end{pmatrix}. \tag{28}$$

Transition rates of the matrix $\mathbf{Q}^{(4)}$ in equation (28) depend on voltages across the left and right hemichannels, i.e. $\lambda_{co}^{(l)} = \lambda_{co}^{(l)}\left(V_{left}\left(n_l\right)\right)$, $\lambda_{oc}^{(l)} = \lambda_{oc}^{(l)}\left(V_{left}\left(n_l\right)\right)$ $\lambda_{co}^{(r)} = \lambda_{co}^{(r)}\left(V_{right}\left(n_r\right)\right)$, $\lambda_{oc}^{(r)} = \lambda_{oc}^{(r)}\left(V_{right}\left(n_r\right)\right)$ and $V_{right}\left(n_r\right) = V - V_{left}\left(n_l\right)$.

Results are presented in Table 1, which shows calculated conductances at different V_j values. Modeling results are obtained from CTMC model and simulation of the GJ channel [4].

Table 1. Voltage-dependent conductance in the GJ channel containing 4 connexins: comparison of simulation and CTMC modeling results

Voltage, mV	Conductance, pS	
	Simulation	CTMC model
-200	0.2110	0.2105
-160	0.2643	0.2657
-120	0.3120	0.3176
-80	0.5664	0.5435
-40	0.9304	0.9561
0	0.9833	1.0000
40	0.9451	0.9561
80	0.5332	0.5435
120	0.3164	0.3176
160	0.2609	0.2657
200	0.2085	0.2105

Table 1 shows that CTMC model and simulation produce similar results. Calculation was performed using a PC with Intel Core 2 Duo CPU T9400 @ 2.53 GHz 2.53 GHz and 4 GB of physical RAM. Preliminary data showed [6] that continuous time Markov chain modeling required significantly less CPU time than simulation.

4. Conclusion

Our results shows that the CMCT model of GJ channel is less complex than an analogous DTMC model, so it is possible to apply Markov model creation tools using a/the PLA method. Creation of infinitesimal generator is derived from the description of system behavior.

In this paper, we showed that the use of CTMC (instead of DTMC) to model GJ channels enables to reduce computation time ~30 times. This can be achieved since the infinitesimal generator of CTMC arising from GJ channel model, is sparse. This ensures faster creation of infinitesimal generator **Q** of CTMC model than the transition matrix **P** of analogous DTMC model of GJ channel. Sparsity of **Q** also allows using efficient numerical methods to calculate steady-state probabilities.

Acknowledgments

This work was supported by the Research Council of Lithuania for collaboration Lithuania and USA scientists under grant MIT-074/2012.

References

[1] **F. F. Bukauskas, V. K. Verselis**. Gap junction channel gating. *Biochimica et Biophysica Acta-Biomembranes,* 2004, Vol. 1662, No. 1-2, 42-60.

[2] **J. F. Ek-Vitorin, T. J. King, N. S. Heyman, P. D. Lampe, J. M. Burt**. Selectivity of connexin 43 channels is regulated through protein kinase C-dependent phosphorylation. *Circulation Research,* 2006, Vol. 98, No. 12, 1498-1505.

[3] **K. Bang, J. Choi, C. Yoo**. Comments on "The Model Checker SPIN". *IEEE Transactions on Software Engineering,* 2001, Vol. 27, No. 6, 573-576.

[4] **N. Paulauskas, H. Pranevicius, J. Mockus, F. F. Bukauskas**. Stochastic 16-State Model of Voltage Gating of Gap-Junction Channels Enclosing Fast and Slow Gates. *Biophysical Journal,* 2012, Vol. 102, No. 11, 2471-2480.

[5] **N. Paulauskas, M. Pranevicius, H. Pranevicius, F. F. Bukauskas**. A Stochastic Four-State Model of Contingent Gating of Gap Junction Channels Containing Two "Fast" Gates Sensitive to Transjunctional Voltage. *Biophysical Journal,* 2009, Vol. 96, No. 10, 3936-3948.

[6] **H. Pranevicius, F. F. Bukauskas, O. Pranevicius, M. Pranevicius, S. Vaiceliunas.** Markov models of voltage gating of gap junction channels. *Invited presentation at 25th Conference on Operational Research (EURO-2012),* Vilnius, Lithuania, 2012.

[7] **H. Pranevicius, V. Germanavicius, G. Tumelis.** Automatic creation of numerical models of systems

specified by PLA method. In: *12th International Conference on Analytical and Stochastic Modelling Techniques and Applications (ASTMA 2005)*, Nottingham, United Kingdom, 2005, pp. 118-124.

[8] **H. Pranevicius, R. Miseviciene.** Verification of business rules using logic programming means. In: *Modelling of Business, Industrial and Transport Systems*, Riga, Latvia, 2008, pp. 99-106.

[9] **H. Pranevicius, S. Norgela**. Applications of Finite Linear Temporal Logic to Piecewise Linear Aggregates. *Informatica*, 2012, Vol. 23, No. 3, 427-441.

[10] **H. Pranevicius, V. Pilkauskas, G. Guginis.** Creating simulation models specified by PLA using UML. In: *International Conference on Operational Research: Simulation and Optimisation in Business and Industry*, Kaunas, Lithuania, 2006, pp. 87-92.

[11] **H. Pranevicius, L. Simaitis, M. Pranevicius, O. Pranevicius**. Piece-Linear Aggregates for Formal Specification and Simulation of Hybrid Systems: Pharmacokinetics Patient-Controlled Analgesia. *Elektronika ir Elektrotechnika*, 2011, No. 4, 81-84.

[12] **H. Pranevicius, E. Valakevicius, M. Snipas**. Complexity of embedded chain algorithm for computing steady state probabilities of Markov chain. *Information Technology and Control*, 2011, Vol. 40, No. 2, 110-117.

[13] **A. Sakalauskaite, H. Pranevicius, M. Pranevicius, F. Bukauskas**. Markovian Model of the Voltage Gating of Connexin-based Gap Junction Channels. *Elektronika ir Elektrotechnika*, 2011, No. 5, 103-106.

[14] **G. Sohl, K. Willecke**. Gap junctions and the connexin protein family. *Cardiovascular Research*, 2004, Vol. 62, No. 2, 228-232.

[15] **G. E. Sosinsky, B. J. Nicholson**. Structural organization of gap junction channels. *Biochimica et Biophysica Acta-Biomembranes*, 2005, Vol. 1711, No. 2, 99-125.

[16] **W. J. Stewart.** Introduction to the Numerical Solution of Markov Chains. Princeton University Press, 1994.

Received January 2013.