This work introduces MACACO, a macroscopic calcium currents simulator. It provides a parameter-sweep framework which computes macroscopic $Ca^{2+}$ currents from the individual aggregation of unitary currents, using a stochastic model for $L$-type $Ca^{2+}$ channels. MACACO uses a simplified 3-state Markov model to simulate the response of each $Ca^{2+}$ channel to different voltage inputs to the cell. In order to provide an accurate systematic view for the stochastic nature of the calcium channels, MACACO is composed of an experiment generator, a central simulation engine and a post-processing script component. Due to the computational complexity of the problem and the dimensions of the parameter space, the MACACO simulation engine employs a grid-enabled task farm. Having been designed as a computational biology tool, MACACO heavily borrows from the way cell physiologists conduct and report their experimental work.

Keywords: Computational Cell Biology; Calcium Channels; Markov Model; Parameter-sweep; Algorithmic Skeleton

1. Introduction

Excitable cells are the subject of intense study in electrophysiology, cell biology, and biophysics. They are comprised of a cytoplasm and a surrounding membrane, and contain ions such as $Ca^{2+}$, $Na^+$, and $K^+$ in highly-specific concentrations. $Ca^{2+}$-selective channels, known as calcium channels, modify their conformation in response to depolarisation from a resting potential typically ($-90mV$, $-70mV$). When this membrane voltage becomes greater than its threshold potential, channels allow $Ca^{2+}$ to cross the cell membrane producing a measurable ionic current, named calcium current. The charge transfer/storage process, known as calcium regulation,

*Accepted for publication by Prof Limsoon Wong, Managing Editor – JCBB
is of particular importance since it greatly influences neuronal and cardiac muscle activity.

Furthermore, calcium concentration in cells, predominantly defined by the \( \text{Ca}^{2+} \) ion presence, impacts on crucial biological processes such as the excitation-contraction coupling and the regulation of gene expression. \( \text{Ca}^{2+} \) ions can enter heart cells through two types of voltage-dependent calcium channels: L-type and T-type. L-type calcium channels are characterised by their slow activation, fast deactivation, strong calcium-dependent inactivation, and have a prominent physiological relevance.

This work describes macaco, a Markov-based simulator of macroscopic calcium currents which aggregates individual calcium channel contributions for a given channel population. Under the assumption that one cell can have thousands of ionic channels, MACACO can calculate L-type \( \text{Ca}^{2+} \) currents. Furthermore, as a result of its parallel implementation, it is able to work concurrently with multiple parametric executions containing thousands of calcium channels on the membrane of a spherical cell.

Rather than using a statistical approximation, MACACO actually considers each individual channel for different populations, making each parametric execution computationally expensive. Furthermore, since we are interested in analysing these currents due to different voltage stimuli (amplitude and duration), MACACO generates a set of executions for a given parameter space and then executes this set concurrently in a parameter-sweep fashion.

The parameter-sweep mechanism is implemented as a task farm, which exploits the independent nature of the workload to generate a multiplicity of tasks to compute results. The task farm operation can be adumbrated as a farmer process which spawns a number of independent worker processes to run simultaneously. Each worker executes a series of tasks by applying the given worker function. Different task farm implementations assign different amounts of tasks per worker based on their scheduling method.

The task farm structure can be efficaciously abstracted using algorithmic skeletons, which synthesise commonly-used patterns of parallel computation, communication, and interaction. Furnishing top-down design composition and control inheritance, skeletons are typically implemented as parameterisable building blocks, in the form of patterns, templates, or higher-order functions, and complete programs are expressed using these blocks analogously to the way sequential structured programs are constructed.

Therefore, we have incorporated the aforementioned factors into the MACACO design in order to provide an efficient grid-enabled parameter-sweep environment, based on the skeletal paradigm, for the \( \text{Ca}^{2+} \) exchange stochastic modelling.

Despite the fact that distinct solutions to different computational problems have been demonstrated, algorithmic skeletons have been scantily employed by the scientific community due to their apparent 'restricted' application for large problem instances executing in distributed, heterogeneous environments. This work provides
further evidence that the skeletal paradigm is highly applicable for solving relevant computational problems in grids.

This paper proceeds as follows. Firstly, we discuss the concept of \( Ca^{2+} \) currents in cells and how they are empirically estimated; followed by a description of our stochastic model. Then we introduce our parameter-sweep implementation based on a skeletal task-farm approach. Then we present some of the results obtained for the model. Finally, we conclude with some future directions.

2. Motivation

In a seminal work, Hodgkin and Huxley \(^{18}\) experimentally verified the flow of electric current in a membrane, and developed a mathematical framework for the fibre conduction and excitation, which is the foundation of most ionic current models of excitable tissues. Based on their work, it has been demonstrated that calcium currents are generated in response to a change in the permeability of the cell membrane, due to the opening of \( Ca^{2+} \) channels. In fact, the whole cell current is the result of the contribution of all open channels at a given time in response to an electric stimulus, and the opening or closing of a channel can be conceived as a stochastic process where it is not possible to predict transitions between different conformational states \(^{9}\).

Voltage-activated \( Ca^{2+} \) channels are proteins that change their permeability (conductance) in response to variations in membrane potential and also in the internal calcium cell concentration. The macroscopic calcium current, \( I_{Ca} \), is the sum of all \( Ca^{2+} \) ions crossing the cell membrane through calcium channels at a given time. Nevertheless, the \( I_{Ca} \) depends primarily on the neighbouring membrane voltage (\( V_m \)) and the contributions of each open single channel current (\( i_{Ca} \)). The \( i_{Ca} \) amplitude could be individually defined as an averaged peak amplitude from a reference current level measured in a membrane patch \(^{5}\).

From an experimental standpoint, there are two commonly-used approaches to measuring \( I_{Ca} \):

- The total \( I_{Ca} \), or whole cell current, is measured and the single channel conductance and the number of open channels are estimated using the mean current amplitude and its associated variance.
- The individual \( i_{Ca} \) is recorded from many patches having one channel each. Then the \( I_{Ca} \) is extrapolated as the product of the averaged \( i_{Ca} \) by an estimation of the open channels under specific conditions.

However, at best, both methods roughly estimate individual channel contributions. Nevertheless, the properties of single channel currents are crucial for the diagnosis of cardiac dysfunction such as the abnormal excitation-contraction coupling or even human heart accidents, where the \( I_{Ca} \) does not show detectable variations but the \( i_{Ca} \) does \(^{24}\). Different experimental work has obtained results with different degrees of accuracy depending on the techniques employed \(^{2,3,22}\).
Fig. 1. The proposed 3-state Markov model for each calcium channel. States: Open (O), Close and active (C₁), and Close and inactive (C₂). Transitions: α, β, γ, and ω.

From a computational perspective, macroscopic currents can be calculated using (1), provided the transition rates are constant.

\[ p(t) = p(0) \times \exp(Qt) \]  

(1)

where \( p(t) \) is the vector of state occupancies at time \( t \) and \( Q \) is the matrix of transition rates.

Nonetheless, if the transition rates are not constant and calcium diffusion in the channel vicinity is taken into account, current computation methods fall short of accurately estimating the macroscopic calcium current for a large number of channels. On the other hand, experimental methodologies cannot accurately record individual channel contributions. Thus, we argue the importance of calculating the macroscopic \( \text{Ca}^{2+} \) concentration by individually aggregating each channel contributions while tracing the variability induced by the dynamic calcium concentration.

This work describes, to a high degree of accuracy, the voltage-dependence, the open-closed transitions, and the \( \text{Ca}^{2+} \)-dependent inactivation. Its main differentiator is the capacity to individually aggregate vast amounts of channels with high time resolution in affordable processing times. In order to evaluate its applicability, this paper provides a series of experiments under different conditions (parameters) that are typical when studying channel electrophysiology, and extends our initial approach for a simplified Markov model by presenting a more comprehensive set of results and incorporating calcium diffusion in the channel vicinity.

3. Methods

\( \text{Ca}^{2+} \) channels can be modelled as a set of states representing points of specific conductance and a series of transitions among them which depend on voltage and calcium values. Due to the stochastic nature of channel transitions, voltage dependence and \( \text{Ca}_i \) dependence of the rate constants between states, we have modelled the system as a three-state Markov model. The three discrete states (Open O, Close and active C₁, and Close and inactive C₂) have four different rate constants (α, β, γ, and ω) representing transitions between states as shown in Fig. 1.

The probability values for each state are derived from the rate equations of the calcium kinetic model, as presented in (2), (3), (4).

\[ \frac{dP_{C_1}}{dt} = (\beta P_O + \omega P_{C_2}) - (\alpha P_{C_1} + \gamma P_{C_1}) \]  

(2)

\[ \frac{dP_{C_2}}{dt} = \gamma P_{C_1} - \omega P_{C_2} \]  

(3)
\[
\frac{dP_O}{dt} = \alpha P_{C1} - \beta P_O
\]  (4)

The stable-state solution of this variable set provides the probability that the channel can be in one of the three states. The equations are expressed in terms of rate constants in (5), (6), (7), (8), constrained by probability conservation (9). \(P_O, P_{C1}, \) and \(P_{C2}\) are the associated probabilities to each state respectively, and \(P_x\) is the crossed probability.

\[
P_O = \frac{\alpha \omega}{P_x}
\]  (5)

\[
P_{C1} = \frac{\beta \omega}{P_x}
\]  (6)

\[
P_{C2} = 1 - (P_{C1} + P_O)
\]  (7)

\[
P_x = \alpha \omega + \gamma \beta + \beta \omega
\]  (8)

\[
\sum_{i=0}^{2} P_i = 1
\]  (9)

The \(\alpha\) and \(\beta\) rates are voltage-dependant, i.e. transitions to/from the \(O\) are decided by the membrane potential. \(\gamma\) absorbs calcium dependence for inactivation of the channel, and \(\omega\) represents recovery from calcium inactivation.

As is usually the case with stochastic methodologies, this model relies heavily on the quality of the random number generator, as well as the experiment resolution, in order to produce reliable results.

4. Implementation

A cell has, typically, thousands of channels, and simulating their transitions implies the processing of a large number of random elements for a significant period of time. Additionally, the current, obtained as the cell response, is modified by the difference between the base (membrane) and peak depolarising voltages as well as its duration. This parameter space is summarised in Table 1.

Hence, every problem instance in an Macaco parameter sweep is constructed as follows: firstly, the initial conditions are fixed eliciting a basal calcium current. Secondly, after a certain period of time, a fixed-time depolarising voltage pulse is applied to the cell membrane in order to promote stochastic channel transitions for a large number of elements. Thirdly, macroscopic cell current is calculated using single channel contributions and calcium concentration.

Macaco builds upon these experiences and provides an independent, component-based framework, developed in C with MPI calls, comprising:

- experiments generator,
- central simulation engine, and
- post-processing scripting
The main technical requirement of MACACO is an underlying infrastructure with a grid-enabled MPI and support to parallel I/O. While most MPI implementations support concurrent I/O and the MPI 2.0 standard includes extensive support, it is important that the grid/metacomputer filesystem also supports concurrent writing/reading from any node.

A functional view of the system is depicted in Fig. 2.

The experiment generator allows the user to define the accuracy of the parameter-sweep. It allows the division of the simulation space presented in Table 1, by parameterising the model in terms of: number of channels, time resolution, base voltage (in mV), peak voltage (in mV) and peak duration (in seconds). The time resolution, which is the quotient of the total duration of the simulation and the size of the individual time slice, determines the total number of steps. The $k_0, \ldots, k_5$ physiological constants are determined using standardised experimental values $^{25,14}$.

The model is implemented as a linear algorithm where the number of channels ($\text{channels}$) and time resolution, defined as the number of steps ($\text{steps}$), determine its temporal complexity, ergo the number of daxpy operations and the storage space on a per-experiment basis as shown in equations (10) and (11).

\begin{align}
\text{Time(model)} &= O(\text{channels} \times \text{steps}) \quad (10) \\
\text{Space(model)} &= O(\text{steps}) \quad (11)
\end{align}

Thus, a typical experiment involving the simulation of $10^5$ channels for a second in intervals of a microsecond ($10^6$ steps) will have $O(10^{11})$ running time and $O(10^6)$ storage space functions respectively. The overall functionality of the model is abstracted in Algorithm 1.

Once a series of experiments is generated, the MACACO central simulation engine executes the parameter sweep concurrently through an adaptive skeletal task farm $^{13}$.

In order to assure randomness in the simulation, we have employed the Linux C library pseudo number generator $\text{rand}()$ and the initialisation call

\begin{table}[h]
\centering
\begin{tabular}{|l|c|c|c|}
\hline
\textbf{Parameter} & \textbf{Unit} & \textbf{Affects computational complexity?} \\
\hline
Number of Channels & – & X \\
Steps & – & X \\
Base Voltage & Millivolts & X \\
Peak Voltage & Millivolts & X \\
Peak Duration & Seconds & X \\
\hline
\end{tabular}
\caption{MACACO Parameter Space}
\end{table}
Data:
$\Delta t$: Time slice;
$\text{sim}$: Simulation time;
$\text{channels}$: Total number of channels;
$\text{layer}$: Number of layers in the L-type spherical cell;
$k_0, \ldots, k_5$: Physiological constants;

Result:
$\text{Ca}^{2+}$ data and control files;

$$/* I: \text{Current} $; \text{V: Voltage} $; A: \text{accumulator} */$$

Initialise $\vec{I}, \vec{V}, \vec{A}$;
$$I_0 \leftarrow I_{\text{base}}; \quad V_0 \leftarrow V_{\text{base}};$$

$$/* R: \text{Radius} */$$
$$R_n \leftarrow R_{\text{core}} + \text{layer} \times \Delta R; \quad R_{n-1} \leftarrow R_n - \Delta R;$$

$$/* n-layer volume */$$
$$\text{Vol}_n = \frac{4}{3} \times \pi (R_n^3 - R_{n-1}^3);$$

steps $\leftarrow \text{sim}/\Delta t$

$$i \leftarrow 0$$

while $i < \text{steps}$ do

$$V_{\text{norm}} \leftarrow V_i - k_0; \quad \alpha \leftarrow 2 \times \exp(k_1 \times V_{\text{norm}}); \quad \beta \leftarrow k_2 \times \exp(k_3 \times V_{\text{norm}}); \quad \gamma \leftarrow k_4 \times I_i; \quad P_{\text{tot}} \leftarrow \alpha \times \omega + \gamma \times \beta + \beta \times \omega; \quad P_a \leftarrow \alpha \times \omega/P_{\text{tot}};$$

$$/* \text{open is the accumulated number of open channels} */$$
$$\text{open} \leftarrow 0;$$

$$j \leftarrow 0;$$

while $j < \text{channels}$ do

$$/* \text{True random value} */$$
$$\text{if } \text{random} < P_a \text{ then}$$
$$\text{open} \leftarrow \text{open} + 1;$$
$$\text{else}$$
$$I_{i+1} \leftarrow I_i;$$
$$\text{end}$$

$$a_{i+1} \leftarrow I_{\text{unicellular}} \times \text{open}; \quad I_{i+1} \leftarrow I_i - ((a_{i+1} \times \Delta t)/(2 \times k_5 \times V_n));$$
$$\text{end}$$

return ($\text{Ca}^{2+}$ data and control files);

**Algorithm 1**: Macroscopic Calcium Currents Model
srand((unsigned) time(NULL)). While Linux random generation has drawn some criticism in terms of security\textsuperscript{17}, we consider its stability adequate for these experiments.

This task farm enhances grid programmability by allowing the rapid inclusion of any change in the model without any disturbance to the control and the structure of the application. The farm control is based on a self-scheduled workqueue, and provides an acceptable load balancing strategy since the number of experiments is typically greater than the number of available processing nodes. Since complexity varies from instance to instance, the greedy nature of self-scheduling permits the work assignment to idle nodes which balances itself over time. In essence, every worker is adaptively assigned a series of experiments based on the amount of its available resources, generating the data output and the post-processing files. The skeletal API is presented in Fig. 3.

It is important to emphasise that this task-farm based parameter sweep environment has two main advantages:

- it can be seamlessly adapted to perform another scientific parameter sweep by substituting the main calcium currents algorithm with another function,
Fig. 3. Application program interface to the algorithmic skeleton of the adaptive task farm

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Channels</td>
<td>$10^4, 10^5, 10^6$</td>
</tr>
<tr>
<td>Steps</td>
<td>$10^4, 10^5, 10^6$</td>
</tr>
<tr>
<td></td>
<td>(Sim. time=1ms, 100ms, 1s, $\Delta t = 1 \mu s$)</td>
</tr>
<tr>
<td>Base Voltage</td>
<td>$-80mV$</td>
</tr>
<tr>
<td>Peak Voltage</td>
<td>$[-40mV, 60mV]$ (in 5mV intervals)</td>
</tr>
<tr>
<td>Peak Duration</td>
<td>$20%, 40%, 60%$ of Sim. time</td>
</tr>
</tbody>
</table>

Table 2. 756-experiment instance of the macaco parameter space

and

- it can statistically adapt its scheduling, based on the application structure, according to the dynamic characteristics of the computational grid in turn

Every experiment generates a data and control file, each of which is used in turn by the macaco post-processing module to automatically produce an encapsulated postscript graph using the gnuplot utility, depicting the $Ca^{2+}$ current values versus time (in the order of milliseconds) as shown in Section 5. Having been designed as a computational biology tool, macaco heavily borrows from the way cell physiologists conduct and report their experimental work.

5. Results

In order to provide a significant experimental resolution, we have chosen a $\Delta t$ of 1 $\mu$s and a membrane potential of $-80mV$. This basal potential was selected due to its close resemblance to a physiological potential, since depolarisation from this point produces a gradual increase and a slow decay of the calcium current, allowing us to detect calcium current activation (negative current peak).

The selected channel populations are $10^4, 10^5, 10^6$. Depolarising voltage varies from $-40mV$ to $60mV$ in 5$mV$ steps with a duration from $20\%, 40\%$, and $60\%$ of the total simulation time, using total simulation times of 10ms, 100ms, 1s. It is important to mention that $-40mV$ has been selected as the initial depolarising voltage because it is the accepted threshold for L-type calcium channels opening. This whole parametrisation accounts for 756 experiments and is summarised in...
The aforementioned set of experiments implies problem instances with temporal complexities from $O(10^8)$ to $O(10^{12})$, and spatial complexities from $O(10^4)$ to $O(10^6)$.

Moreover, Fig. 4 reaffirms the scalability of the model. Figure 4 (a) shows a linear correlation between the time resolution $r$ and the execution time and the storage space. Figure 4 (b) demonstrates a linear relation between the number of channels $c$ and the execution times. Thus, this experimentally iterates the temporal and spatial complexities shown in (10) and (11). All times shown were measured within the problem instances and include additional values for completeness.

The reported results were obtained employing a departmental grid formed by two non-dedicated Beowulf clusters located across the University of Edinburgh, configured as shown in Table 3. In terms of software, all nodes had Linux Red Hat FC3 with kernel 2.6, gec 3.4.4, LAM/MPI 7.1.1, GSL 1.5 and NWS 2.10.1. All modules were compiled using the “-pedantic -ansi -Wall -O2” flags.

The whole parameter-sweep accounts for a few quadrillion daxpy operations. The experiment generation and the scripting execution times were negligible (<.01%). Furthermore the 756 data files constitute 21GB of storage which are concurrently generated. We have been able to co-allocate resources using the LAM/MPI capabilities without any queue management system in place. All farmer-worker commu-

![Diagram of execution time vs. time resolution and number of channels]

Fig. 4. Graphic Parametric Correlation of the macaco model. Note that the rectangular areas delimit the parameter subspace used in the study.
Fig. 5. $Ca^{2+}$ currents obtained with macaco, using an 1 µsecond time slice and voltage pulses which cover 60% of total simulation time. Membrane potential varies from −80 to 0 mV when current peak is elicited. Time resolution increases column-wise a, d, g while the number of channels increments row-wise a, b, c.

communication took place using non-dedicated Ethernet networks.

Figures 5 and 6 presents two subsets of nine $Ca^{2+}$ current charts obtained using macaco, where time resolution increases column-wise (10ms, 100ms, 1s) and the number of channels increments row-wise (10k, 100k, 1M). The total simulation time defines, in turn, the time resolution and the spatial complexity\(^a\).

The two subsets of charts shown in Fig. 5 and Fig. 6 provide evidence of the impact on the experimental observations not only of the different resolutions but also of the number of channels. The overall effectiveness of the model is schematically corroborated by:

- the noticeable increment in the current peak amplitude when voltage amplitude (column-wise) and channel population (row-wise) are increased;

\(^a\)Note that the graphic quality was deliberately decreased in the 1-second charts (f)–(i) of every set in order to improve the manageability of this document.
Fig. 6. $Ca^{2+}$ currents obtained with macaco, using an 1 µsecond time slice and voltage pulses which cover 60% of total simulation time. Membrane potential varies from $-80$ to $40\text{mV}$ when current peak is elicited. Time resolution increases column-wise $a, d, g$ while the number of channels increments row-wise $a, b, c$.

- the calcium-dependent inactivation when increasing voltage pulse duration (column-wise) or the number of channels (row-wise) leading to a sharp current peak

It is worth noting that the initial boundary conditions are better quantified with higher resolutions. Similarly, a greater channel account increments the $Ca^{2+}$ current readings and the peak sharpness.

6. Discussion

From an stochastic perspective, single-channel macroscopic currents have been simulated with realistic diffusion characteristics. While these approaches calculate the average $Ca^{2+}$ concentration in a macroscopic volume and considers $Ca^{2+}$ release as well, they do not take into account the multiplicity of channels our simulation considers.
The experimental protocols presented in this paper have been selected in order to show how MACACO can accurately reproduce the activation, the deactivation and the strong calcium-dependent inactivation that distinguishes L-type calcium currents in different channel populations. These results reaffirm not only that MACACO can estimate macroscopic currents with high accuracy for channel populations in the order of hundreds of thousands in accordance with other estimations of calcium channel densities, but also that our model is flexible enough to accommodate different parametric executions.

Local $\text{Ca}^{2+}$ concentration variations directly impact on channel inactivation, therefore their correct quantification is vital to the accuracy of the model. These variations can be estimated discretising intracellular space in layers or in subdomains. For this work, we have used a thin-layer approach ($\sim 0.1$ micro) and passive diffusion, but none of the other $\text{Ca}^{2+}$ regulation mechanisms were considered. In particular, the $\text{Na}^+–\text{Ca}^{2+}$ exchanger is one of the most important, and in the past, owing to limited resources, we have been unable to explore its complete function over an extended period of time and evaluate some of its specifics. An extension to this work may explore the inclusion of additional regulation mechanisms.

It is considered important to run comprehensive simulations in order to study calcium-dependent inactivation of the channels with elongated pulses. From a physiological standpoint, the number of channels is responsible for determining the overall definition of the currents while the time resolution steers the experimental observational capabilities, e.g., enabling the study of “ultra-slow” $\text{Ca}^{2+}$ processes.

Furthermore, an increasing number of channels or groups of channels can extend the MACACO applicability to the simulation of cell clusters. Hence, the MACACO design, based on structured parallelism, and its current workload distribution for grids constitutes an excellent foundation for the development of more extensive experimental protocols that can shed some light on the fascinating field of cell physiology.

From a more computational standpoint, it is widely acknowledged that parameter-sweep applications are well suited to be implemented in computational grids, as proven by the use of grid template systems such as AppLeS, GridSim, and the EDG framework. While these frameworks have successfully been employed to deploy biomedical and bio-informatics applications, they normally require the inclusion of ad-hoc calls, and do not exploit the structure of the application. On the other hand, the use of a skeleton-based infrastructure such as MACACO has been proven to enhance application performance.

7. Concluding Remarks

In this work, we have introduced the MACACO parameter sweep framework, a component-based environment for the calculation of macroscopic currents. Due to its skeletal design, MACACO provides a suitable component framework for the de-
development of grid-based biomedical applications.

It is important to note that from a biomedical standpoint, the simple fact of affording a comprehensive parameter sweep has important physiological implications. It provides a more accurate exploratory method to study the implications of calcium concentrations over calcium currents.

Moreover, it may be desirable to increase the size of the parameter sweep in order to accommodate a more complete set of physiological requirements which imply a higher experimental resolution. A direct implication of a possible experiment space augmentation is the resource increase, thus careful attention must be paid regarding the physical location of grid resources in order to increase the overall computation to communications ratio while maintaining an even workload distribution. The adaptive nature of the task farm has been carefully designed to ameliorate the resource dynamism.

It is crucial to restate that any improvement in the biological model will not affect the control of the application as a result of the macaco implementation, which is based on structured parallelism and, viceversa, the parallel control structures can be bettered without altering the model behaviour.

The main technical limitation in macaco is the management of the multiplicity of data and control files. Although macaco generates unique filenames for each experiment, it does not provide any indexing or search facility at this point. Users must understand the macaco file nomenclature to accurately retrieve the generated graphs. Furthermore, since there is no interface to a grid portal, program staging must be pre-programmed to avoid costly interruptions.

References


Virginia González–Vélez Received the BSc (summa cum laude) and the MSc degrees from Universidad Autónoma Metropolitana–Iztapalapa in Mexico in biomedical engineering in 1990 and 1994 respectively. After consulting for the pharmaceutical industry for five years, she joined the basic sciences and engineering division of the Universidad Autónoma Metropolitana–Azcapotzalco as a lecturer in 1999. She has published over a dozen peer-reviewed contributions in computational cell biology and biomedical engineering. Her research interests include development and application of cell models, bio-computing, and the use of new modelling languages in biological software simulators. Her e-mail address is vgv@correo.azc.uam.mx.

Horacio González–Vélez A research fellow in the School of Informatics of the University of Edinburgh, he has over 10 years of proven industrial experience working for Sun Microsystems and Silicon Graphics in various management and engineering positions. He earned a BSc and a MSc in computer science from the Universidad Autónoma Metropolitana–Iztapalapa and the University of Essex, and a postgraduate diploma in system analysis from OIC-Japan. He is currently a PhD candidate in the School of Informatics of the University of Edinburgh, and his research interests include structured parallelism, distributed and grid computing, computational biology, and biomedical applications. He can be reached at h.gv@ed.ac.uk.