Systems biology

Grid Cellware: the first grid-enabled tool for modelling and simulating cellular processes

Pawan K. Dhar*, Tan Chee Meng, Sandee Somani, Li Ye, Kishore Sakharkar, Arun Krishnan, Azmi B. M. Ridwan, Sebastian Ho Kok Wah, Mandar Chitre and Zhu Hao

Systems Biology Group, Bioinformatics Institute, 30 Biopolis Street, #07-01, Matrix, Singapore 138671

Received on November 17, 2003; revised on April 18, 2004; accepted on April 22, 2004

Advance Access publication November 16, 2004

ABSTRACT

Summary: Modelling and simulation of complex cellular transactions involve development of platforms that understand diverse mathematical representations and are capable of handling large backend computations. Grid Cellware, an integrated modelling and simulation tool, has been developed to precisely address these niche requirements of the modelling community. Grid Cellware implements various pathway simulation algorithms along with adaptive Swarm algorithm for parameter estimation. For enhanced computational productivity Grid Cellware uses grid technology with Globus as the middleware.

Availability: https://www.bii-sg.org/sbg/cellware

Contact: pk@bii-sg.org

Supplementary information: User Manual is available as PDF file. Examples are attached to expedite mastering of the software.

INTRODUCTION

Systems Biology takes a panoramic view of the cellular transactions brought about by the interaction of genes, mRNAs, proteins, etc. Owing to limitations in the range and accuracy of experiments it is not feasible to test every type of possibility or hypothesis in vitro and/or in vivo. The need for alternative approaches is especially felt when attractive hypothesis command experimental confirmation, or when non-intuitive phenomena emerge from the system that cannot be understood using a traditional wet lab approach. Systems Biology offers a unique opportunity of weaving the molecules and their messages into an in silico fabric that can be tested against a wide range of conditions and combinations. To create an in silico model, a tool is needed that understands not only the topology of the network but also the language(s) that it’s constituent components speak. In order to capture diverse interactions among objects, various mathematical representations and modelling platforms are required. In an ideal setting, the modelling community would need a computational tool that integrates and graphically models biological information of varying complexity, conducts large-scale simulations that are computationally very heavy and offers services over the web. Thus, as the complexity of the model grows, the future computational needs of modelling community will move from serial to distributed computing. To address this imminent requirement, Grid Cellware—a grid-based modelling and simulation tool, has been developed by the Systems Biology Group of Bioinformatics Institute in Singapore.

Grid Cellware has not only been designed to conduct modelling and simulation of biochemical pathways but it also offers an integrated environment for diverse mathematical representations, parameter estimation and optimization, user-friendly graphical display and capability to run large and complex models. Grid Cellware is the first grid-based modelling and simulation tool in the field of Systems Biology, to the best of our knowledge.

FEATURES

Simulation algorithms

Grid Cellware provides a simulation environment for heterogeneous cell models like gene regulatory network, signal transduction and metabolic pathways. In addition, the flexible software framework facilitates speedy implementation of more algorithms in future versions. The simulation algorithms currently available in Grid Cellware are as follows:

- Stochastic algorithms, such as Gillespie, Gibson and TauLeap.
- Deterministic algorithms based on ordinary differential equation solvers, such as Euler Forward, Runge-Kutta schemes and stiff-ode solver like Dormand Prince scheme.
- Hybrid algorithms flexible architecture.

The availability of various deterministic and stochastic algorithms gives us an opportunity to develop and evaluate hybrid algorithms. Such algorithm development is planned for the future.

Parameter estimation

Grid Cellware incorporates the Swarm algorithm (Ray and Saini, 2001) for parameter estimation jobs. Swarm algorithm is based on simulation of social behaviour where each individual in a swarm adjusts its flight according to its own flying experience as well as that of its companions’ flying experience (Ray and Saini, 2001). Swarm algorithm is highly suitable for constrained multi objective optimization problems, thus, it is well suited for biochemical pathway optimization. Currently, Cellware uses Swarm algorithm for parameter estimation of biochemical pathway based on time-series data. The user interface provides a friendly and intuitive environment for speedy inverse engineering of biochemical pathways and cell model construction.

*To whom correspondence should be addressed.
Grid computing is ideal for parallel batch jobs and coarse grained parallelism. A simulation job submitted to the grid environment can utilize a heterogeneous network which may include mainframes, high-performance clusters and even stand-alone computers.

Cellware exploits grid infrastructure for running independent simulation jobs simultaneously. In cases where biologists need to run multiple simulations, e.g. (1) simulate different pathway models, (2) apply multiple algorithms to a single model or (3) perform a parameter sweep on the model parameters, the simulations can be dispatched to several grid nodes concurrently and executed in parallel.

Parameter estimation (Moles et al., 2003) is another ideal computational task for the grid as it requires repeated independent simulations of a model with varying parameters and is apt for coarse grained parallelism. For parameter estimation, at every iteration the algorithm spawns off many simulation jobs which are then sent to different compute nodes.

It should be noted here that the Swarm algorithm implemented in Cellware is MPI based and is constrained by the size of the compute cluster. The utility of grid computing lies in the fact that it can schedule jobs across clusters and retrieve results— all transparent to the user.

SBML compatibility
SBML (http://www.sbml.org) is an XML-based standard for representing biochemical reaction networks designed for smooth exchange and cooperative development of models. Grid Cellware joins the fast increasing SBML community as it supports import and export of SBML Level 1 Version 1 files. Grid Cellware can convert an SBML model into a network graphical model and vice versa. Therefore, Grid Cellware is compatible with all the tools in the SBML community.

Intuitive graphical user interface
The powerful graphical editor plays an important role in Grid Cellware modelling. The features can be summarized as follows:

1. Drawing: The graphical editor provides drawing components for a wide range of graphic primitives. Each drawing component has certain graphical properties, which affect the appearance of the component and can be edited with mouse clicks and menu commands.

2. Components connection: The graphical editor supports a representation of connected components that enables the smooth and consistent connection between objects in the model.

3. Specification: The graphical editor supports the specification of biological components in a pathway model, such as
gene, modifier, metabolite, mRNA, reaction constant and compartment volume.

**Automated graph layout**

Visualization of models in Grid Cellware is made easier by an automated graph layout tool (Fruchtermann and Reingold, 1991) for arranging the various pathway components on the drawing workspace. The tool applies graph layout algorithms that are capable of arranging a neat and visually appealing graph.

**ARCHITECTURE**

Grid Cellware has a highly modular architecture that allows the developer to easily plug in various components into the system. The client application has been developed using Java in order to make use of an object-oriented architecture and provide the client on several platforms. The client is primarily a graphical user interface for specifying the model and simulation parameters while the server is the computational engine which is grid enabled. Java was chosen because cross-platform compatibility was one of the primary design criteria for the client application. As a result the computational algorithms were implemented in this Java version inspite of the performance hit.

For the grid version, the computational engine has been written in standard Ansi C++ (using g++) with special emphasis on performance and portability across various Unix-based clusters. As a result most of the simulation algorithms have a parallel implementation in C++ on the server (grid) side and in Java on the client side. Parameter estimation has only been implemented on the grid.

Communication to the grid was implemented through web services. The client and server API’s for web services were developed.
using Apache Axis as it has desirable features, such as object persistence and session management, which are essential for Cellware. Scheduling and resource management is done by a meta-scheduler which is invoked by Axis. The meta-scheduler evaluates the suitability and availability of grid resources before submitting the job to appropriate machines. Grid Cellware has been tested with GridX, a meta-scheduler developed within Bioinformatics Institute, Singapore. Furthermore, integration with other meta-schedulers is also possible with minor extensions. The underlying grid fabric of Cellware is the Globus Toolkit 2.0 (http://www.globus.org). The meta-scheduler makes use of the APIs provided by Globus. GridX currently uses the Java COG Kit for this purpose.

REFERENCES