BIOINFORMATICS APPLICATIONS NOTE Vol. 26 no. 14 2010, pages 1794–1796
doi:10.1093/bioinformatics/btq276

Systems biology

Advance Access publication May 26, 2010

DA 1.0: parameter estimation of biological pathways using data assimilation approach

Chuan Hock Koh[1,2,3,†], Masao Nagasaki[3,*,†], Ayumu Saito[3], Limsoon Wong[2] and Satoru Miyano[3]

[1]NUS Graduate School for Integrative Sciences and Engineering, Singapore 117597, [2]School of Computing, National University of Singapore, Computing Drive, Singapore 117417 and [3]Human Genome Center, Institute of Medical Science, the University of Tokyo, 4-6-1 Shirokanedai, Minato-ku, Tokyo 108-8639, Japan

Associate Editor: Olga Troyanskaya

ABSTRACT

Summary: Data assimilation (DA) is a computational approach that estimates unknown parameters in a pathway model using time-course information. Particle filtering, the underlying method used, is a well-established statistical method that approximates the joint posterior distributions of parameters by using sequentially generated Monte Carlo samples. In this article, we report the release of Java-based software (DA 1.0) with an intuitive and user-friendly interface to allow users to carry out parameters estimation using DA.

Availability and Implementation: DA 1.0 was developed using Java and thus would be executable on any platform installed with JDK 6.0 (not JRE 6.0) or later. DA 1.0 is freely available for academic users and can be launched or downloaded from http://da.csml.org.

Contact: masao@ims.u-tokyo.ac.jp

Received on January 19, 2010; revised on April 21, 2010; accepted on May 23, 2010

1 INTRODUCTION

Simulating and modeling of biological pathways have been gaining popularity in an attempt to better understand complex biological relationships (Fisher and Henzinger, 2007). However, before a pathway model can be simulated, all parameters in the model must be known. Unfortunately, some parameters are usually either unknown or do not agree in literature.

To address this problem, we previously introduced a computational method called Data assimilation (DA) to estimate unknown parameters in a pathway model using the time-course information based on a well-established statistical method, particle filtering (Nagasaki *et al*., 2006). This method has been successfully used to do parameter estimation for the circadian clock model (Nagasaki *et al*., 2006) and phosphotyrosine-dependent signaling networks in the epidermal growth factor receptor (EGFR) pathway (Tasaki *et al.*, 2006). Furthermore, in a recent paper, it has also been shown that particle filtering can be deployed to utilize parallel computing infrastructure that makes it scalable for large, complex models (Nakamura *et al.*, 2009).

*To whom correspondence should be addressed.

†The authors wish it to be known that, in their opinion, the first two authors should be regarded as joint first authors.

2 DA

DA is an approach widely used especially in the field of geophysics. It combines observations and numerical simulation models to estimate the unknown parameters. Two advantages of DA are its compatibility with parallelism and its ability to reveal the posterior distributions of unknown parameters.

The approach in brief is as follows; given a pathway model, the list of parameters to be estimated and N observed time points, a set of M particles is drawn either randomly or through user-specified distribution. At each time point N, the M particles will be resampled with a probability directly proportional to a fitness score. The fitness score is computed as a function of difference between the simulated and observed data at each time point. A higher score is given to particles with simulated results closer to the observed results. At the end of the algorithm, users are given the distribution plots of the M particles' values. It is recommended that the mode of the distribution be chosen as the estimated value for downstream applications. For more details on the algorithm, please refer to Nagasaki *et al.* (2006).

The power of DA largely depends on two factors: (i) the number of observed time points and (ii) size of particles and number of parameters to be estimated. From a statistical point of view, the more time points observed (higher frequency and longer duration), the higher the accuracy would be. However, time points are often limited by current experimental techniques. The size of particles should also be exponentially proportional to the number of parameters to be estimated in order to obtain a high accuracy. However, if the number of particles is large, it is likely to cause either out of memory error or slow running time on standard desktop computers.

In the next section, we will describe practical ways to work around the two limitations above in order to obtain an accurate estimation of parameters in a normal desktop environment.

3 SOFTWARE FEATURES

DA 1.0 contains implementation of the particle filter methodology and several other features for ease of use, including a drawing utility that will be particularly useful when observed data is limited. The user interface is deliberately minimalistic so that the usage would be intuitive (Fig. 1, main dialog). The required inputs include a pathway model, observed data and the range of the parameters to estimate. The output consists of a distribution plot of the particles, simulation results of the fitted models and Cell System Markup Language (CSML) format (http://www.csml.org) of the fitted models.

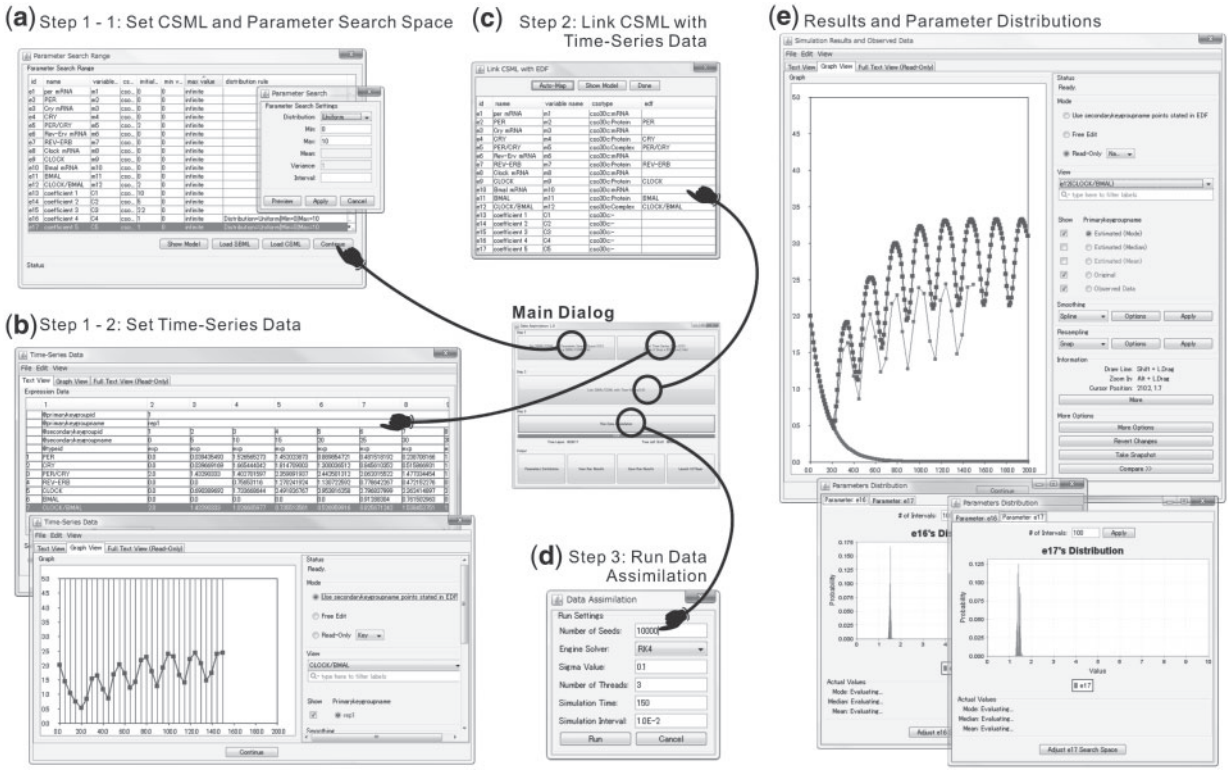

Fig. 1. (**a**) This step is to load the model file (CSML or SBML) and define the distribution and range for parameters that users wish to estimate. (**b**) This step is to input the observed time-series data. Accepted formats include EDF, CSV and TSV. Functions such as smoothing and sampling are included to improve the quality of observed data for better estimation results. (**c**) This step is needed to pair the model entities with observed data. An auto-map function is available to match corresponding entities and observed data with same names. (**d**) A variety of settings for the particle filter and simulation are enabled to allow for flexibility based on the user's needs. (**e**) After running the particle filter algorithm, the simulation runs results using estimated parameters will be plotted for ease of comparison between the original and fitted models. The parameters' distribution plot is also displayed.

3.1 Inputs

The required format for the pathway model is CSML as DA 1.0 is built to run particle filtering on hybrid functional Petri net with extension (HFPNe; Nagasaki *et al.*, 2004), which uses the CSML format. However, support has also been extended to another format, Systems Biology Markup Language (SBML) in the form of a SBML2CSML converter. Thus, it is possible to input the pathway model in either CSML or SBML (Fig. 1a). If the latter is being provided, it will be automatically converted into CSML format.

As for observed data, EDF (expression data format) would be required. EDF (http://da.csml.org) was developed in our laboratory for the ease of representing time series expression data that usually include replicates and annotation data. Similarly, to support the commonly used tab (or comma) separated format, a convertor to convert tab (or comma) separated format into EDF is included.

Finally, users would have to set the range for the parameters they wish to estimate (Fig. 1a). This need not be precise. It would be sufficient to simply give it a rough range that is biologically possible.

3.2 Outputs

Unlike parameter estimation methods based on using optimization method (Yoshida *et al.,* 2008), particle filtering gives a distribution plot of the possible values for the parameter. This information is particularly useful for repeat runs to obtain a better estimation.

The simulation results of the original model, observed data and fitted model are also plotted on one graph for ease of comparison (Fig. 1e). Finally, users can save the fitted models in CSML format, which can be displayed and replayed on Cell Illustrator Player that is available for free. Direct launch of the fitted models in Cell Illustrator Player from DA 1.0 is also possible. Additionally, users can run Cell Illustrator to apply more comprehensive downstream analysis. Cell Illustrator (Nagasaki *et al*., 2003) is available for free as a 1 month trial at https://cionline.hgc.jp. Users also have the freedom to utilize other software of their choice, and they can obtain the estimated values from the distribution plot (Fig. 1e).

3.3 Tips to improve estimation accuracy

As mentioned in the previous section, observed time points are often limited, which could potentially reduce the estimation accuracy. To overcome this limitation, we have added the ability to increase time points' frequency by means of smoothing and re-sampling (Fig. 1b).

In some cases, it is possible that users do not have any experimental data but have an idea of how a particular biological entity would behave with respect to time, either gleaned from literature or simply testing out a hypothesis. To handle such cases, we have also made it simple to draw expression plots within DA 1.0.

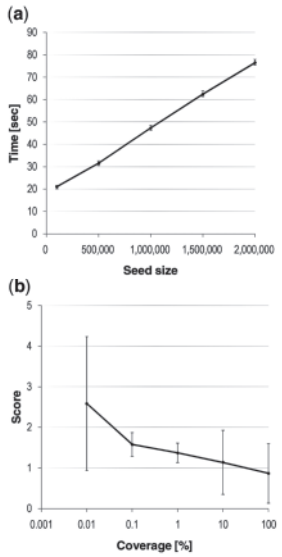

Fig. 2. (**a**) Time versus seed size plot. (**b**) Score versus coverage. Scores close to 1 or <1 indicates a very good match between observed data and simulation results. (Please see supplementary data for experiments details.)

Another important factor that affects estimation accuracy is the seed size (number of particles). However, setting a large seed size causes the program to run slowly. Therefore, we suggest for users to do repeat runs using medium seed size (Fig. 1d) and to adjust the possible range after each run using the distribution plot of the parameters (Fig. 1e).

4 PERFORMANCE

To give users a gauge of the performance of DA 1.0, we have performed some experiments using the circadian clock model (Nagasaki *et al*., 2006) with 17 parameters in total, on a contemporary desktop machine (Intel Core i7 CPU at 3.2 GHz). We focused on the influence of seed size on time used, space needed and estimation power. With respect to the memory needed, every one million seeds require ∼1 GB of RAM (data not shown). From Figure 2a, we can see that the time used increases with the seed size linearly and is able to finish in reasonable time (<80 s for 2 million seeds). Figure 2b demonstrates how the coverage on the search space would affect the estimation power. The standard deviation of the score is large because the seeds are randomly generated. If a 'good' seed is randomly generated, the score would be low. Naturally, with increased coverage, the chance of generating a 'good' seed increases.

However, in the case of a large search space, having good coverage would require a huge amount of memory that might not be available. In such cases, users are encouraged to follow the strategy suggested in Section 3.3.

5 FUTURE WORK

As mentioned, one important factor that affects the estimation power is the size of particles. To empower users with the ability to significantly increase the size of particles yet still be able to complete the estimation within reasonable time, we are currently working on adding the ability to utilize remote computing infrastructure easily using remote method invocation (RMI).

One advantage of DA is that it gives a distribution instead of a single optimal value for each parameter. However, currently only the mean, mode, median or best seed can be used for simulation on Cell Illustrator. Hence, we are now working to further increase the compatibility of Cell Illustrator and DA 1.0 to take full advantage of the DA approach.

ACKNOWLEDGEMENTS

DA 1.0 is developed by using some components in CSMLPipeline application (http://csmlpipeline.hgc.jp).

Funding: Singapore National Research Foundation grant NRF-G-CRP-2997-04-082(d) (to L.W. and C.H.K., in parts); National University of Singapore NGS scholarship (to C.H.K., in parts).

Conflict of Interest: none declared.

REFERENCES

Fisher,J. and Henzinger,T.A. (2007) Executable cell biology. *Nat. Biotechnol.*, **11**, 1239–1249.

Nagasaki,M. *et al.* (2003) Genomic Object Net: I. A platform for modeling and simulating biopathways. *Appl. Bioinform.*, **2**, 181–184.

Nagasaki,M. *et al.* (2004) A versatile Petri net based architecture for modeling and simulation of complex biological processes. *Genome Inform.*, **15**, 180–197.

Nagasaki,M. *et al*. (2006) Genomic data assimilation for estimating hybrid functional Petri net from time-course gene expression data. *Genome Inform.*, **17**, 46–61.

Nakamura,K. *et al*. (2009) Parameter estimation of *in silico* biological pathways with particle filtering towards a petascale computing. *Pac. Symp. Biocomput.*, **14**, 227–238.

Tasaki,S. *et al*. (2006) Modeling and estimation of dynamic EGFR pathway by data assimilation approach using time series proteomic data. *Genome Inform.*, **17**, 226–238.

Yoshida,R. *et al.* (2008) Bayesian learning of biological pathways on genomic data assimilation. *Bioinformatics*, **24**, 2592–2601.