The First International Symposium on Optimization and Systems Biology (OSB'07)
Beijing, China, August 8–10, 2007
, pp. 13–20

Loosely Coupled Architecture for Bio-Network Reverse Engineering

Sungwon Jung Sangwoo Kim Doheon Lee

Department of BioSystems, KAIST, Republic of Korea

Abstract Bio-network reverse engineering is inferring networks of biological entities with given observed data. In biology, our main concern is to find out the systematic architecture of biological entities and their functional role in living organisms. Thus it is very important to provide proper systematic approaches for such reverse engineering for bio-networks. Previously, we developed an information fusion platform for bio-network inference and analysis, named BioCAD. BioCAD provided a good workplace where various network inference and analysis techniques were integrated for user-specific variants of bio-network inference problems. The previous version of BioCAD has a tightly coupled architecture where the integrated usage of provided modules is available only in the form of given workflows. Even though it has provided good workplace for bio-network reverse engineering, there are some limitations in adding new modules and selective usage of provided modules. To overcome such limitations, we are designing and developing BioCAD 2.0 which has a loosely-coupled architecture. We propose our current approach of building new BioCAD 2.0 and BioCAD 2.0 will be a powerful workbench for bio-network reverse engineering.

1 Introduction

Revealing the functional networks of biological entities is a key objective in understanding living organisms. To fulfill the objective, many studies have been conducted in various fields. Among those approaches, it is common to use network inference approaches with various data sources [1-15]. Such approaches are called as bio-network reverse engineering. Previously, we proposed an information fusion platform named BioCAD [16], which is for bio-network inference and analysis. BioCAD is a standalone program associated with various functional modules. BioCAD has four key module categories (Figure 1): data preprocessing module, statistical analysis module, network inference module and network analysis module. In the process of inferring bio-networks with BioCAD, users can apply various different module programs to each stage.

In this paper, we propose an extension of the previous BioCAD to a portal of bioinformatics resources and name it as BioCAD Ver. 2.0. For the previous BioCAD, it was easy to use the provided network inference and analysis workflow for various problems. However, the accessibility of BioCAD was limited when users need only part of provided modules. Further, the extensibility of BioCAD was not sufficient in the perspective of high-end users who wish to develop their own modules and

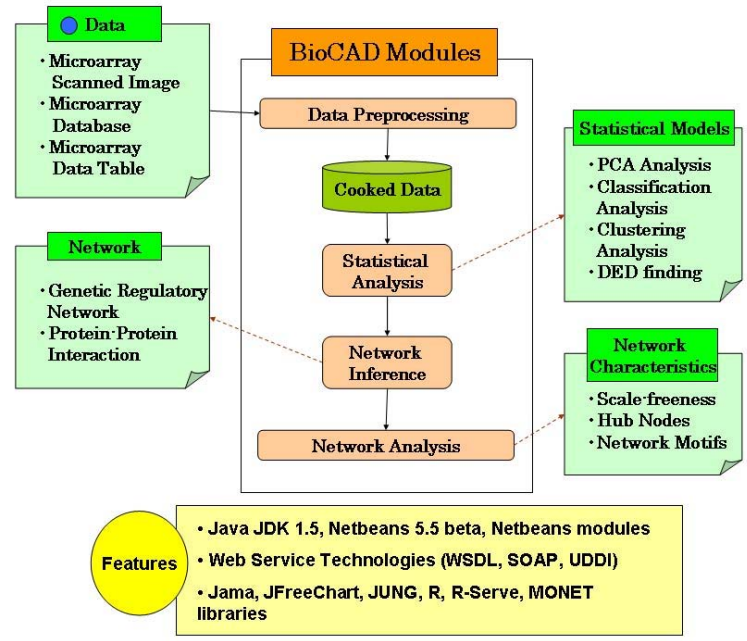

Figure 1: System architecture of BioCAD 1.0

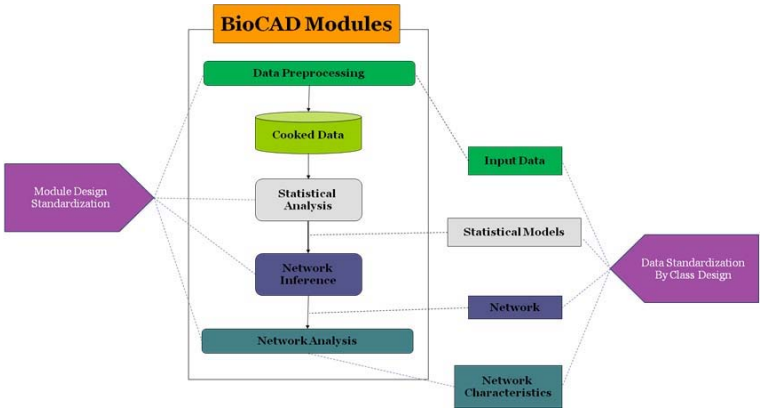

Figure 2: Standardization for loosely coupled architecture of new BioCAD 2.0

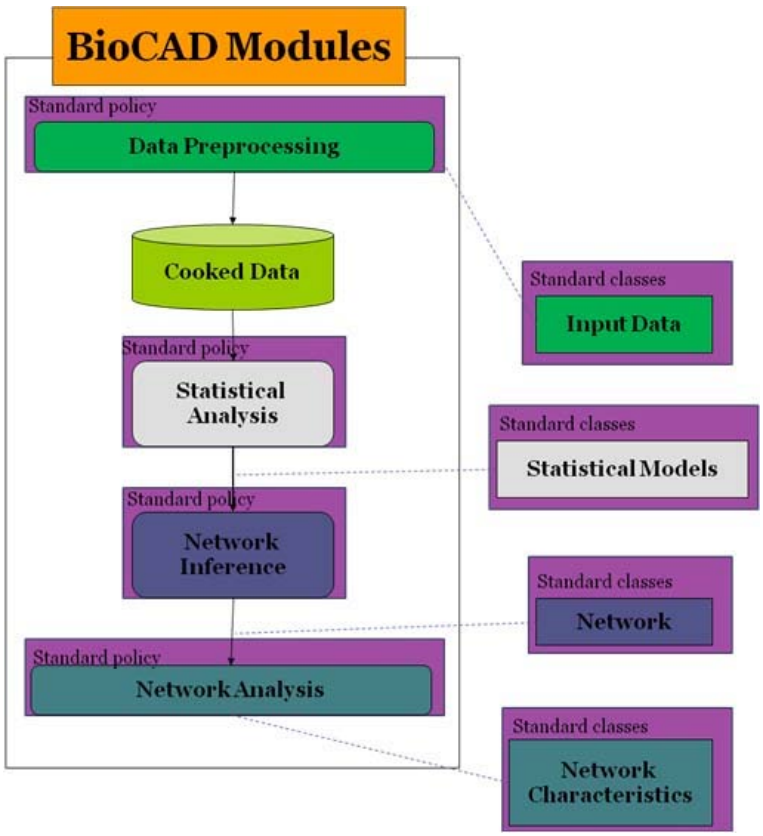

Figure 3: The architecture of BioCAD Ver. 2.0. It provides enhanced extensibility and flexibility by importing the loosely coupled architecture with standard policy and wrapping classes.

integrate them into BioCAD. This is because of the tightly coupled architecture of BioCAD. For these reasons, we extend our previous BioCAD into BioCAD Ver. 2.0 with a loosely coupled architecture. By modifying the previous tightly coupled architecture to the loosely coupled architecture, the new BioCAD 2.0 will provide standardized input and output data formats and equip standardized policies of module development.

2 Designing Loosely Coupled Architecture

The previous version of BioCAD is equipped with tightly coupled architecture which provides workflow of integrated modules. Because there was no standard interface and design policy between modules and data objects, it was not easy for developers who were not familiar with the low level architecture of BioCAD to design and embed new modules into the system. For BioCAD 2.0, we apply loosely coupled

architecture which provides standard data object definitions and standard module development policies. This is achieved by following two approaches.

- Designing classes of data
- Designing policies of module development

In following subsections, we describe those two approaches.

2.1 Designing Classes of Data

Because BioCAD is an information fusion platform for inference and analysis of bio-networks, various types of data sources may be used during the running process of the program. Further, BioCAD is composed of four key module categories as mentioned in the previous section. Because such various data types and modules are in the BioCAD system, the standardization of data objects transferred between modules is essential. By standardizing data formats of modules, users can use each module more easily and there will be no confusion in developing modules.

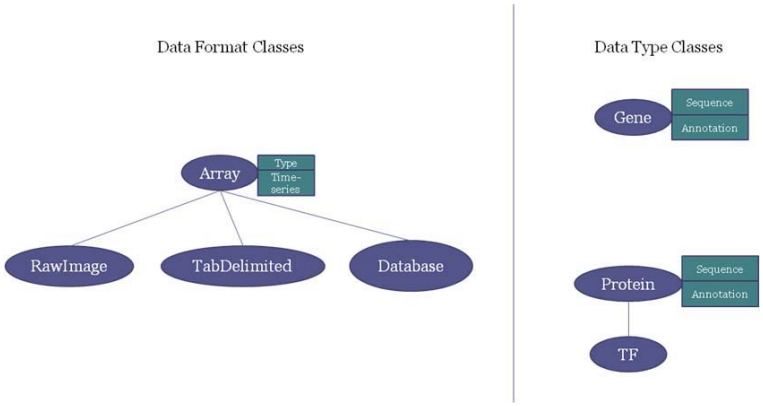

Figure 4: An Example of data format classes and data type classes.

There are two categories of data classes: data format category and data type category. Classes in the data format category are defined according to the file type or the model type. Classes in the data type category are defined according to the data contents itself. An example of those two categories of data classes is shown in Figure 4. In Figure 4, **Array** is a data format class. **RawImage**, **TabDelimited** and **Database** are subclasses of **Array**. **Array** class has a member of *Type* and *Time-series* which describe the content type and the Boolean value to represent whether this is a time-series data or not. This data format classes are defined according to the format of data object. For data type classes, each class is defined according to the content type of data. **Gene** and **Proteins** are different classes and **TF** is a subclass of **Protein**. **Gene** and **Protein** classes have the same members *Sequence* and *Annotation*. For one data object, it is associated with one class for each class

category. For example, one gene expression data of tab-delimited format file will be associated with **TabDelimited** class and **Gene** class.

For data types which have widely known data formats such as NCBI GEO SOFT format [17] and Mage-ML format [18], those public data formats will be considered as a separate class in the data format classes category.

2.2 Policy of Module Development

The policy of module development includes two perspectives. First, the policy of input-output data formats. Second, the policy of user interfaces. For the policy of input-output data formats, there is one simple rule – a module uses input and output data which follows those data standardization described in section 2.1 only. For the policy of user interfaces, each module should keep the same strategy of using command line parameters and configuration files. Thus each module should be used on the command line independently but with the same manner of usage. For example, every module should be used on the command line as follows:

Module-command [option] [option value] [option] [option value] ...

Each option starts with character '-'. If options in different module programs have the same meaning, they should have the same option name.

Besides the command line parameters, there could be configuration files when there are lots of parameters for the module program. Such configurations files should also follow single file format. By standardizing module interfaces in this way, users can be familiar with different modules more easily without tedious training to use them.

3 Applications

3.1 Inferring Network using Network Inference Tools

BioCAD 2.0 will provide whole processes used for network inference. Currently, two network inference modules – ARACNE[19] and MONET[20] - have been integrated into BioCAD 2.0 and more modules will be integrated in the future. ARACNE is a novel algorithm using microarray expression profiles and mutual information process between a pair of random variables. ARACNE algorithm shows good performance compared to the algorithm complexity and the result represents sufficient information of causes and affections. MONET is basically a Bayesian network algorithm. However, MONET has adopted a divide-and-conquer approach to alleviate the dimensionality problems. MONET shows good usability due to its modularizing processes and noticeable improvement of accuracy. Assuming that a user wants to infer a network starting from a GEO SOFT file from NCBI, the user connects to a Web Services tool that imports the file from the Web. From the microarray database file, microarray expression profile can be extracted. Next, the user can preprocess the extracted profile data using data preprocessing modules either provided in the built-in tools of BioCAD 2.0 or supported Web Services tools. The previous version of BioCAD already provided effective preprocess tools associated with the

Bio-Conductor package. Finally MONET starts inferring process with the user's request. MONET uses Gene Ontology database in its inferring process. The inferred Bayesian network will be shown both in graph and table view. This network data is also a part of BioCAD workflow process, and can be used for subsequent processes.

3.2 Analyzing Networks using Information Fusion Tools

In the BioCAD 2.0, inferred bio-network is treated as a new source for subsequent analysis and validation processes. One good validating method is inspecting relations in the network by utilizing text mining tools. Currently BioCAD 2.0 is equipped with a text mining tool which finds regulation or interaction information between two genes from literature search. The constructed network in the previous step is to be examined through the validating step by putting a pair of genes which are connected in the network into the text mining tool. As a result, we can find out whether each network connection has its supporting literature information and in what kind of relation it is connected. The network analysis tool provided in BioCAD 2.0 will integrate a genetic regulatory network with its corresponding protein-protein interaction map, named Bio-viaduct. Bio-viaduct defines a pathway were a gene can affect another gene via transcriptional regulation and protein-protein interactions. For example, when there is a directed edge from gene A to gene B in the inferred network, it searches paths from expressed protein of gene A to the transcription factor of gene B connected by intermediate protein(s).

4 Summary and Future Works

In this paper, we proposed an approach to develop the loosely coupled architecture for the new BioCAD 2.0, which will be a powerful workbench for bio-network reverse engineering. To enhance the user accessibility and ease of module development, we standardize the input & output data and the policy of developing modules for BioCAD. By standardizing the input & output data of each module through defining classes of data formats and data types, modules that follow given data policy can be integrated and used easily. The standardization of interfaces for modules provides easy accessibility to users. After building complete set of data classes and policies for modules, BioCAD Ver. 2.0 could be a useful workbench for bio-network reverse engineering and further a repository of bioinformatics resources which has very wide range of applicability.

Acknowledgement

This work was supported by the Korea Science and Engineering Foundation (KOSEF) through the National Research Lab. Program (No. 2005-01450) and by the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Advancement) (IITA-2006-C1090-0602-0001). DL was supported in part by the Korea Ministry of Information and Communication under Grant C109006020001. We would like to thank CHUNG Moon Soul Center for BioInformation and BioElectronics for providing research facilities.

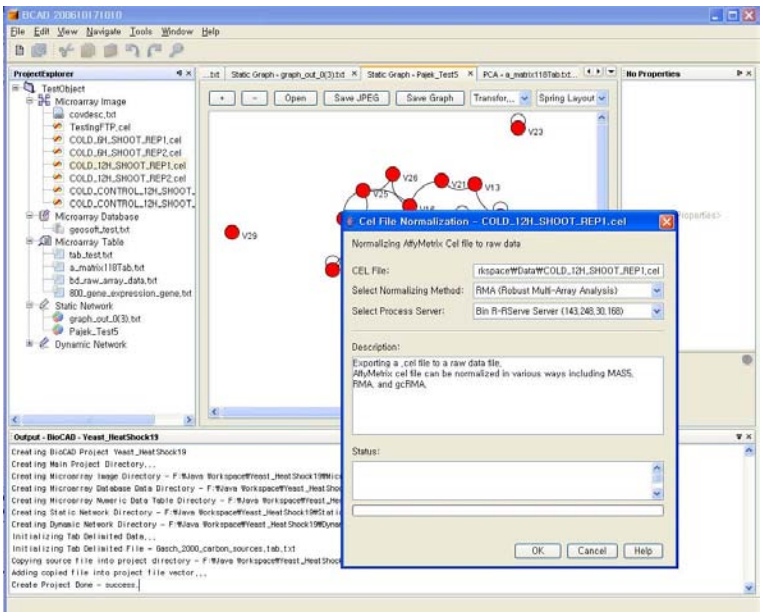

Figure 5: The user interface of previous BioCAD. BioCAD is composed of various data preprocessing and functional modules. Most of jobs are selected and executed in the Project Explorer Window (in the leftmost sub-window).

References

[1] Friedman N, Linial M, Nachman I, Pe'er D: Using Bayesian Networks to Analyze Expression Data. Journal of Computational Biology 2000, 7(3-4):601–620.

[2] Pena JM, Bjorkegren J, Tegner J: Growing Bayesian network models of gene networks from seed genes. Bioinformatics 2005, 21(2):224–229.

[3] Arkin A: A Test Case of Correlation Metric Construction of a Reaction Pathway from Measurements. Science 1997, 277(5330):1275–1279.

[4] Madigan D, Raftery AE: Model Selection and Accounting for Model Uncertainly in Graphical Models Using Occam's Window. Journal of the American Statistical Association 1994, 89(428).

[5] Kim SY, Imoto S, Miyano S: Dynamic Bayesian Network and Nonparametric Regression Model for Inferring Gene Networks. Genome Informatics 2002, 13:371–372.

[6] Kikuchi S, Tominaga D, Arita M, Takahashi K, Tomita M: Dynamic modeling of genetic networks using genetic algorithm and S-system. Bioinformatics 2003, 19(5):643–650.

[7] Kimura S, Hatakeyama M, Konagaya A: Inference of S-system models of genetic networks from noisy time-series data. Chem-Bio Informatics Journal 2004, 4:1–14.

[8] Ladesmai H, Shmulevich I, Yli-Harja O: On Learning Gene Regulatory Networks Under the Boolean Network Model. Machine Learning 2003, 52:147–167.

[9] Bulashevska S, Eils R: Inferring genetic regulatory logic from expression data. Bioinformatics 2005, 21(11):2706–2713.

[10] Saric J: Large-Scale Extraction of Gene Regulation for Model Organisms in an Ontological Context. In Silico Biology 2005, 5:21–32.

[11] Saric J, Jensen LJ, Ouzounova R, Rojas I, Bork P: Extraction of regulatory gene/protein networks from Medline. Bioinformatics 2006, 22(6):645.

[12] Hartemink AJ, Gifford DK, Jaakkola TS, Young RA: Combining location and expression data for principled discovery of genetic regulatory network models. Pac. Symp. Biocomput 2002, 7:437–449.

[13] Kato T, Tsuda K, Asai K: Selective integration of multiple biological data for supervised network inference. Bioinformatics 2005, 21(10):2488–2495.

[14] Xing B, van der Laan MJ: A Statistical Method for Constructing Transcriptional Regulatory Networks Using Gene Expression and Sequence Data. Journal of Computational Biology 2005, 12(2):229–246.

[15] Edgar R, Domrachev M, Lash AE: Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. Nucleic Acids Research 2002, 30:207–210.

[16] Doheon Lee, Sangwoo Kim, BioCAD: An Information Fusion Platform for Bio-Network Inference and Analysis. BMC Bioinformatics 2007, in press.

[17] http://www.ncbi.nlm.nih.gov/projects/geo/info/soft2.html

[18] http://www.mged.org/Workgroups/MAGE/mage-ml.html

[19] Margolin AA, Nemenman I, Basso K, Wiggins C, Stolovitzky G, Dalla Favera R, Califano A, ARACNE: An Algorithm for the Reconstruction of Gene Regulatory Networks in a Mammalian Cellular Context. BMC Bioinformatics 2006, 7 Suppl 1:S7.

[20] Lee PH, Lee D, Modularized Learning of Genetic Interaction Networks from Biological Annotations and mRNA Expression Data. Bioinformatics 2005, 21:2739-2747.