BioEdge: a tool box for advanced analyses of biochemical networks

Yves Deville[1] Pierre Dupont[1] Grégoire Dooms[2] Jean-Noël Monette[1]
Pierre Schaus[1] Shoshana Wodak[3] Stéphane Zampelli[1]

1 Computing Science and Engineering Department, Université catholique de Louvain, Place Saint-Barbe 2, 1348 Louvain-la-Neuve, Belgium, Yves.Deville@uclouvain.be
2 Department of Computer Science, Brown University, Box 1910, Providence, RI 02912, USA, gdooms@cs.brown.edu
3 Hospital for Sick Children, 555 University Ave. Toronto, Ontario, M5G 1X8, and Depts of Biochemistry and Medical Genetics, University of Toronto, Canada.shoshana@sickkids.ca

The Challenges

- Integration of different types of networks : metabolic, regulatory, signal transduction
- Analyzing these networks provides valuable clues on the system level behavior
- Elaborated analyses
  - Involve solving complex combinatorial problems
  - Size of the network tends to be large
- Need for sophisticated computational methods and algorithms

The BioEdge tools

- Graph properties
- Path and subgraph finding
- Context Extraction
- Motif extraction
- Graph Matching

Research Context

- The BioMaze and TransMaze projects
- Representation, **Analysis** & Visualization of biochemical networks
- Interdisciplinary and interuniverity projects
- Funded by the Walloon Region, Belgium (2003-2008)

Research Approach

- The BioEdge underlying tools are mostly domain independent
- Analysis of large graphs
- Using standard (efficient) approaches
- Using the Contraint Programming and the Machine Learning frameworks

Implementation

- Java technology
- Initial tool box : Eclispe plugins in the BioMaze environment
- New implementation : **Cytoscape** plugins

References

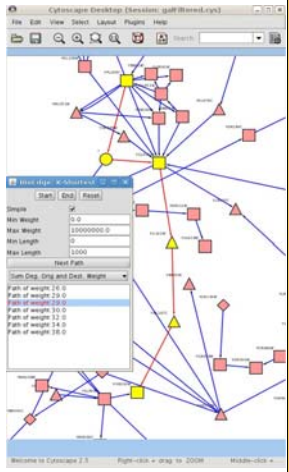
[1] V. Jiménez, A. Marzal. Computing the K Shortest Paths: A New Algorithm and an Experimental Comparison LNCS 1668, Springer Verlag, 1999

[2] Jérôme Callut. First Passage Times Dynamics in Markov Models with Applications to HMM Induction, Sequence Classification and Graph Mining. Ph.D Thesis, UCL/INGI, October 2007

Path and Subgraph Finding

Two complementary approaches

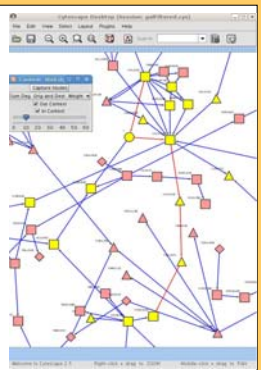
- **K-shortest paths**
  - Implementation of a classical efficient k-shortest paths finding algorithm [1]
  - Various weight policies
  - Directed and undirected networks
  - Can be combined with various filters (path without loop, mutually exclusive nodes, length of the path, …)
- **Constrained path finding**
  - Based on constraint programming
  - Allow sophisticated subgraph extractions



Context Extraction

Two complementary approaches

- **Neighbourhood of a subgraph**
  - Given a path / subgraph, visualize its context
  - Extract the neighbours of the subgraph in its context
  - Specify the width of the context



- **Extraction of a relevant subgraph**
  - Given a set of nodes
  - Extract a subgraph that best explains the relation between these nodes
  - Novel method based on graph data mining [2]
  - Specify the level of relevance