MULE: An Efficient Algorithm for Detecting Frequent Subgraphs in Biological Networks

Mehmet Koyutürk, Ananth Grama,
& Wojciech Szpankowski

http://www.cs.purdue.edu/homes/koyuturk/pathway/

Purdue University
Department of Computer Sciences

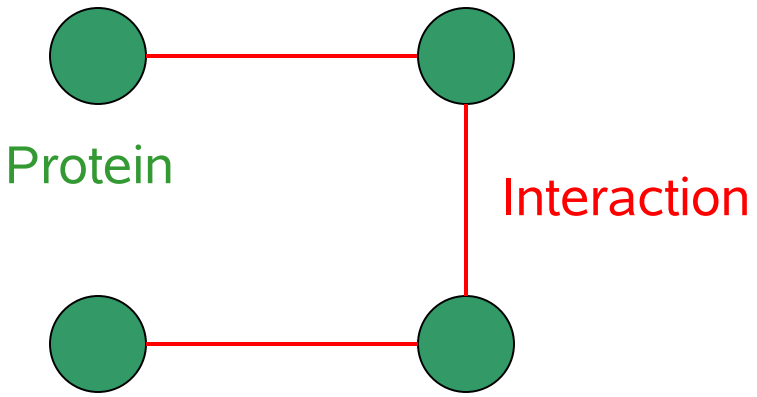
August 4, 2004

Biological Networks

- Interactions between biomolecules that drive cellular processes
  - Mass & energy generation, information transfer…
  - Genes, proteins, enzymes, chemical compounds…
- Coarser level than sequences in life's complexity pyramid
  - Modular analysis of cellular processes
  - Understanding evolutionary relationships at a higher level
- Experimental data in various forms
  - Protein interaction networks
  - Gene regulatory networks
  - Metabolic & signaling pathways

Protein Interaction Networks

- Protein-protein interaction
  - Proteins that cooperate in a process bind to each other
- Pairs of aminoacid chains that bind to each other can be discovered experimentally
  - Two-hybrid
  - Mass spectrometry
  - Phage display

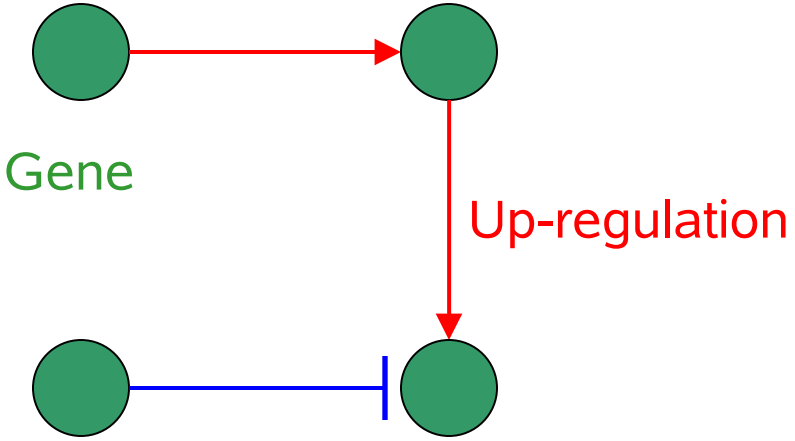


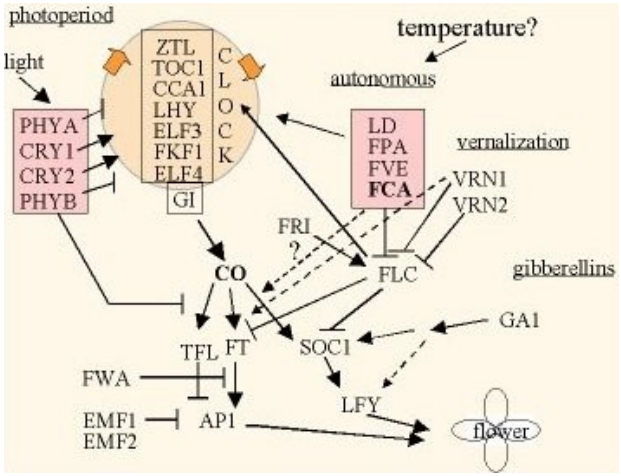
Yeast protein interaction network

Source: Jeong et al. *Nature* 411: 41-42, 2001.

Gene Regulatory Networks

- Genes regulate each others' expression
  - A simple model: Boolean networks
- Can be derived from gene expression data



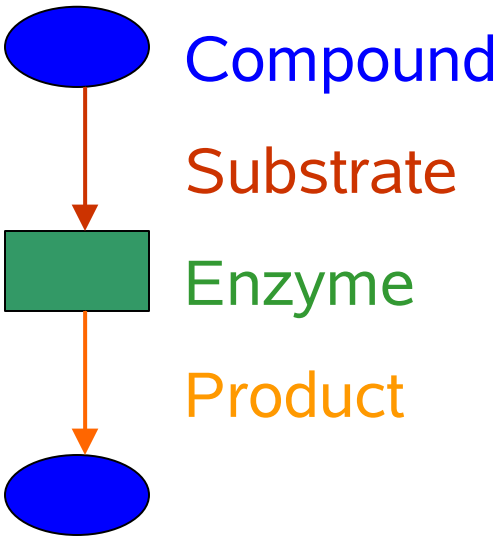


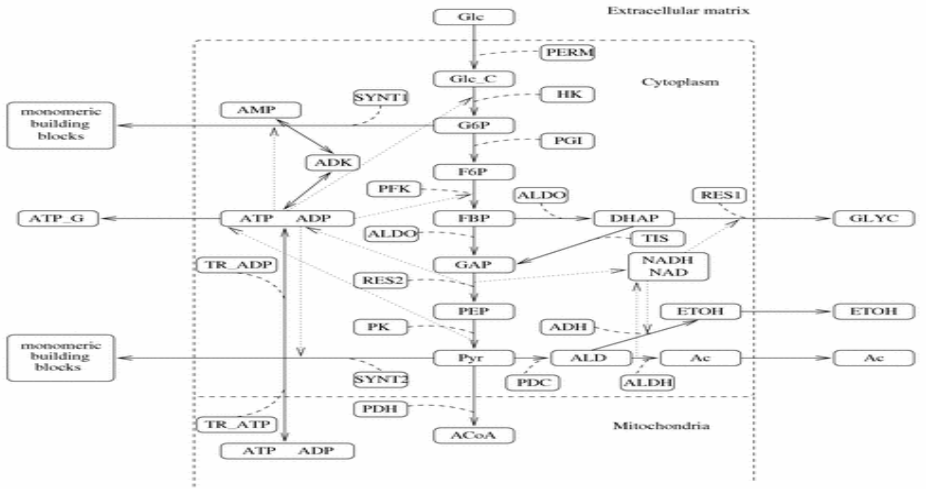
Genetic network that controls flowering time in Arabidopsis

Source: Blazquez et al. *EMBO Reports* 2: 1078-1082, 2001

Metabolic Pathways

- Chains of reactions that perform a particular metabolic function
  - Reactions are linked to each other through substrate-product relationships
- Directed hypergraph/ graph models





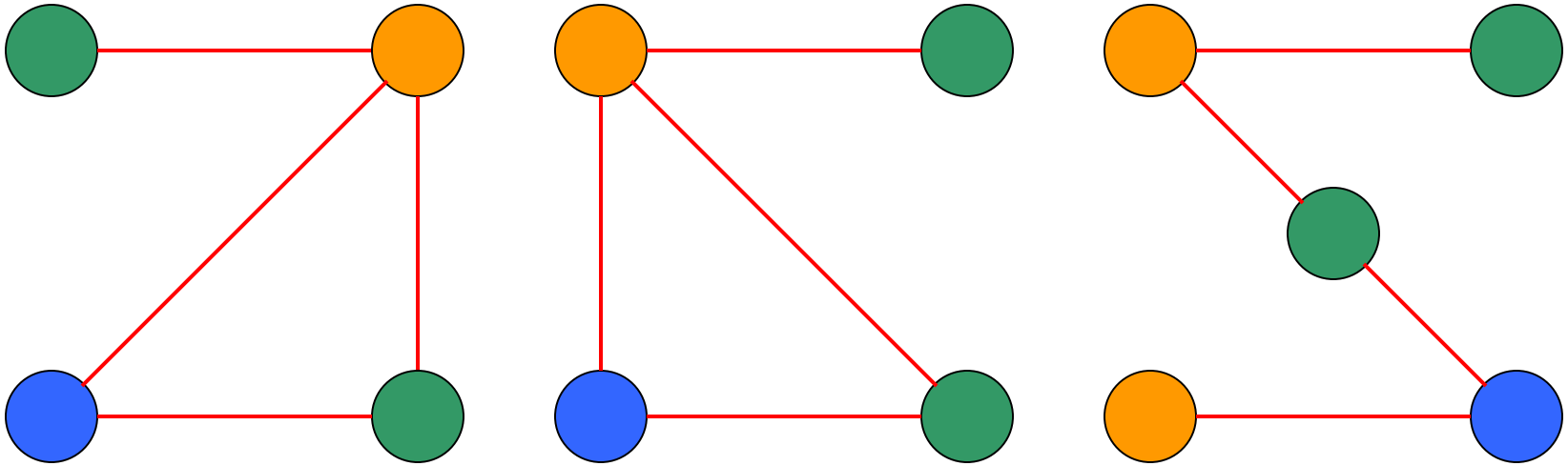
Glycolysis pathway in S. cerevisiae

Source: Rizzi et al. *Biotechnology & Bioengineering* 55: 592-608, 1997.

Analysis of Biological Networks

- Understanding cellular processes at a higher level
  - Functional modules & protein complexes
  - Evolutionary conservation
- Computational analysis: from sequences to graphs
  - Graph clustering
    - Functionally related entities are densely connected
  - Multiple/pairwise graph alignment
    - Conservation-divergence of modules and pathways
  - **Graph mining**
    - Common topological motifs, **frequent molecular interaction patterns**

Graph Mining



Graph Database

Subgraphs with frequency 3

A Basis for Graph Mining: Frequent Itemset Mining

- Given a set of transactions, find sets of items that are frequent
- Algorithms exploit downward closure property
  - A set is frequent only if all of its subsets are frequent

T1: {bread, butter, milk, beer}
T2: {bread, butter, diaper}
T3: {bread, butter, milk, soda}
T4: {bread, soda, beer}

Frequency threshold : 3
⇨Frequent itemsets: {bread, butter}

Frequency threshold : 2
⇨Frequent itemsets: {bread, butter, milk}, {bread, soda}, {bread, beer}

Downward closure: {bread, butter} is frequent only if {bread} and {butter} are frequent.

Graph Mining Challenges

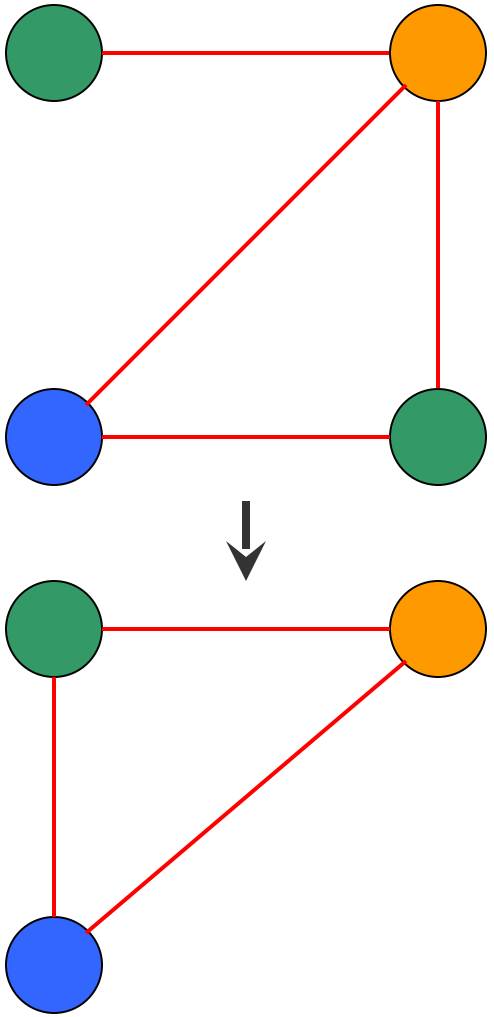
- Subgraph Isomorphism
  - For counting frequencies, it is necessary to check whether a given graph is a subgraph of another one
  - NP-complete
- Cannonical labeling
  - To avoid redundancy while generating subgraphs, canonical labeling of graphs is necessary
  - Equivalent to subgraph isomorphism
- Connectivity
  - Patterns of interest are generally connected, so it is necessary to only generate connected subgraphs

Existing Graph Mining Algorithms

- Adapting frequent itemset mining algorithms to graph mining
  - AGM (2000), FSG (2001)
- Efficient canonical labeling to reduce redundancy
  - gSpan (2002), CloseGraph (2003), FFSM (2003)
- Mining and extending simple subgraphs (trees, paths)
  - SPIN (2004), GASTON (2004)
- Summarizing graphs to prune out search space
  - Ghazizadeh & Chawathe, 2001

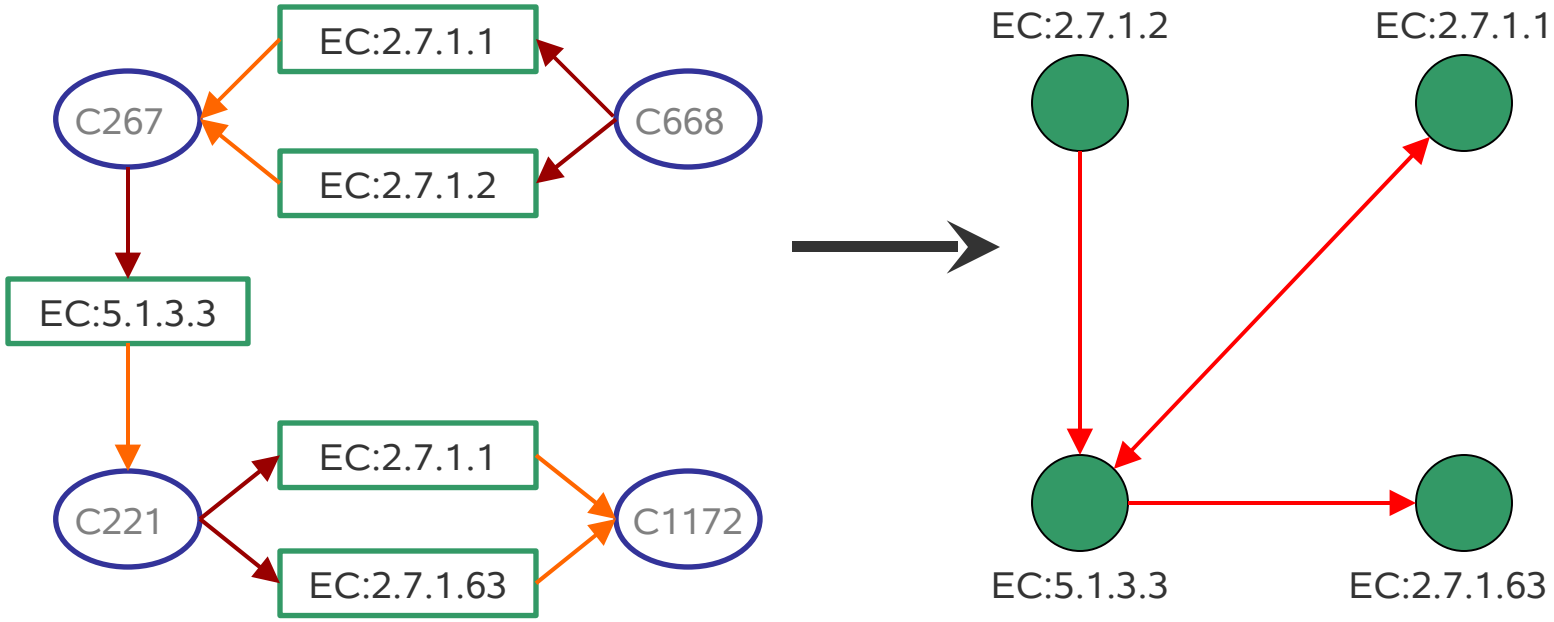
MULE Basics: Uniquely-labeled Graphs

- Contract nodes with identical label into a single node
  - No subgraph isomorphism
    - Much simpler to extract frequent subgraphs
  - Graphs are uniquely identified by their edge sets
  - Frequent subgraphs are conserved
    - Subgraphs that are frequent in general graphs are also frequent in uniquely-labeled graphs
  - Discovered frequent subgraphs are still biologically interpretable!



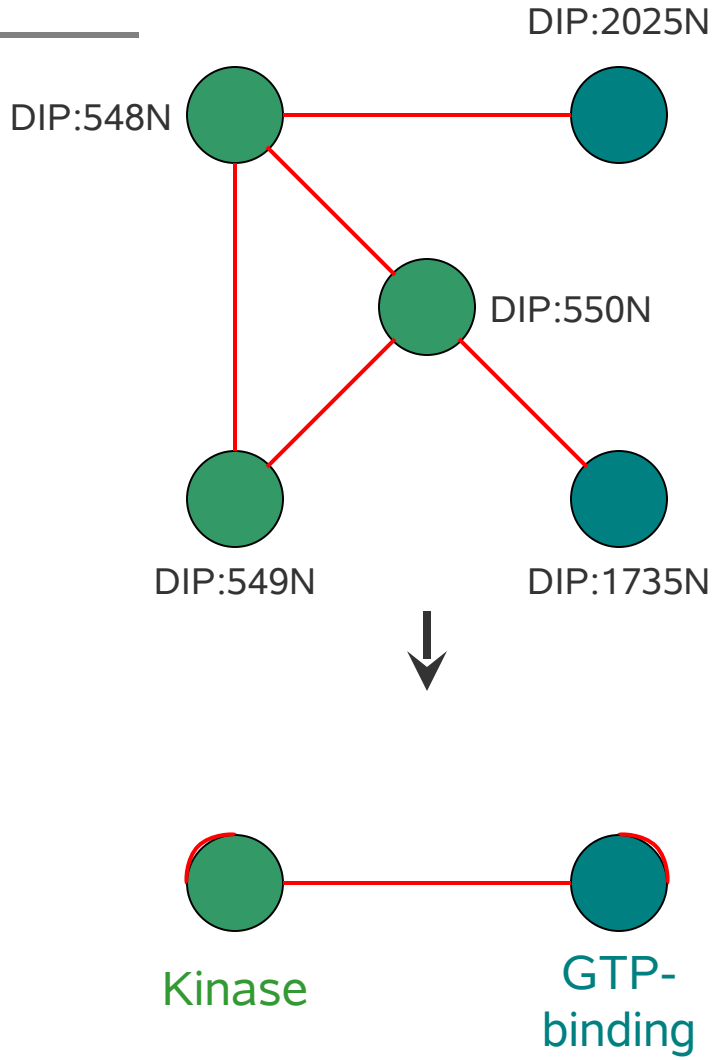
Contraction in Metabolic Pathways

- Uniquely-labeled directed graph model
  - Nodes represent enzymes
    - Global labeling by enzyme nomenclature (EC numbers)
  - A directed edge from one enzyme to the other implies that the second consumes a product of the first



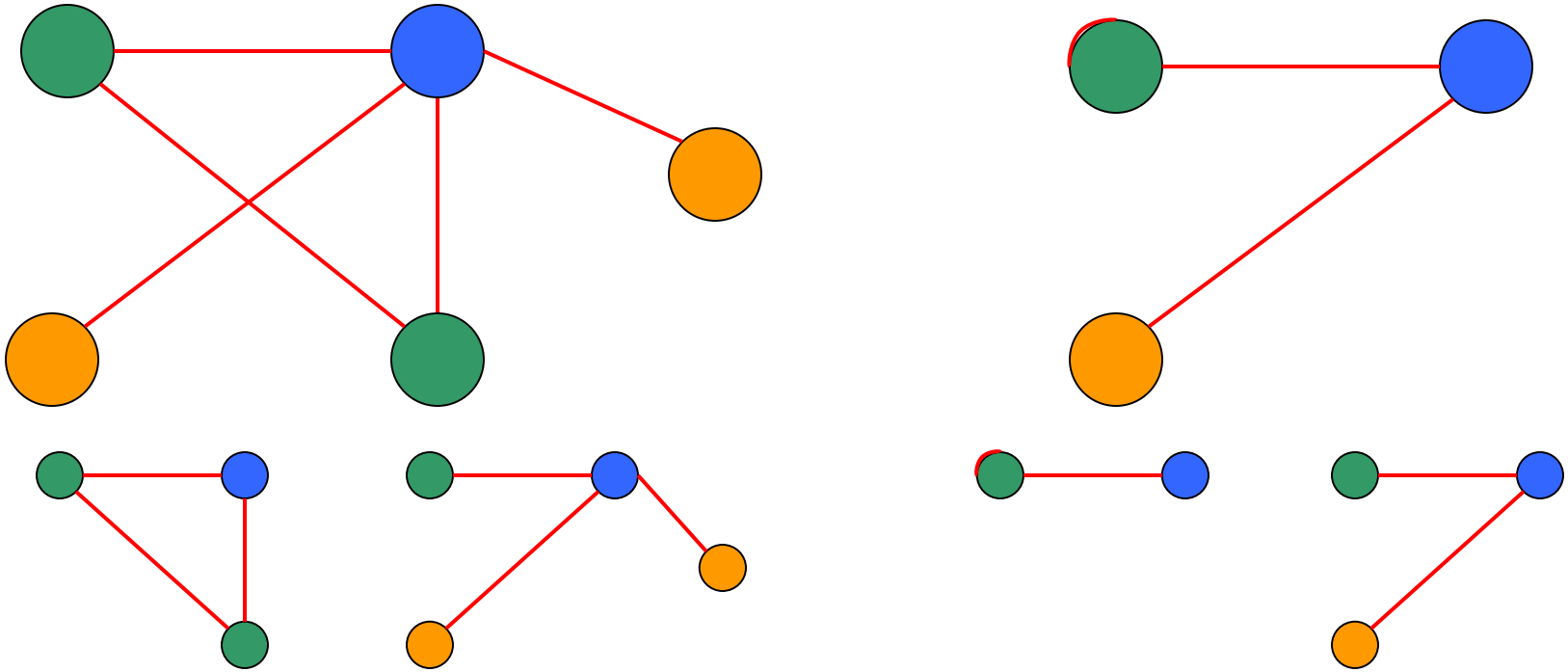
Contraction in Protein Interaction Networks

- Relating proteins in different organisms
  - Clustering
    - Ortholog proteins show sequence similarities
  - Phlyogenetic analysis
    - Allows multi-resolution analysis among distant species
  - Literature, ortholog databases
- Contraction
  - Interaction between proteins becomes interaction between protein families



Preservation of Subgraphs

- The uniquely-labeled version of any frequent subgraph is frequent in the set of uniquely-labeled graphs
  - A uniquely-labeled graph is uniquely determined by the set of its edges



Problem Formulation

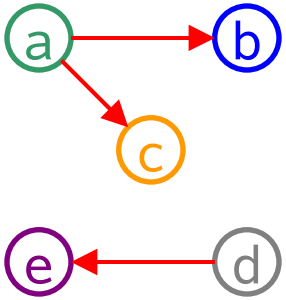
General Graph Mining Problem:

Given a set of graphs $\{G_1, G_2, \ldots, G_m\}$, find all connected graphs S such that S is a subgraph of at least σm of the graphs (is frequent) and no supergraph of S is frequent.

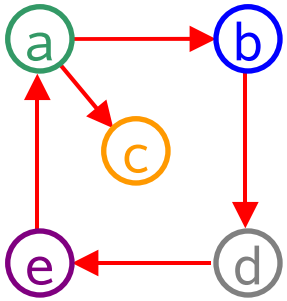
Problem on Uniquely-Labeled Graphs:

Given a set of edge transactions $\{E_1, E_2, \ldots, E_m\}$, find all connected edge sets F such that F is a subset of at least σm of the transactions (is frequent) and no superset of F is frequent.

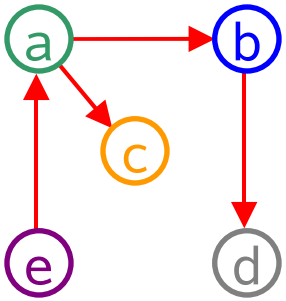
From Graphs to Edgesets



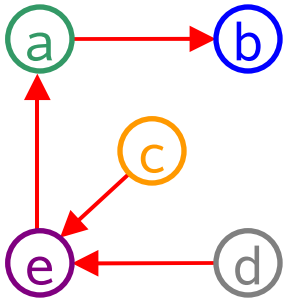
G_1



G_2



G_3



G_4

$\sim$

F_1={ab, ac, de}

F_2={ab, ac, bc, de, ea}

F_3={ab, ac, bc, ea}

F_4={ab, ce, de, ea}

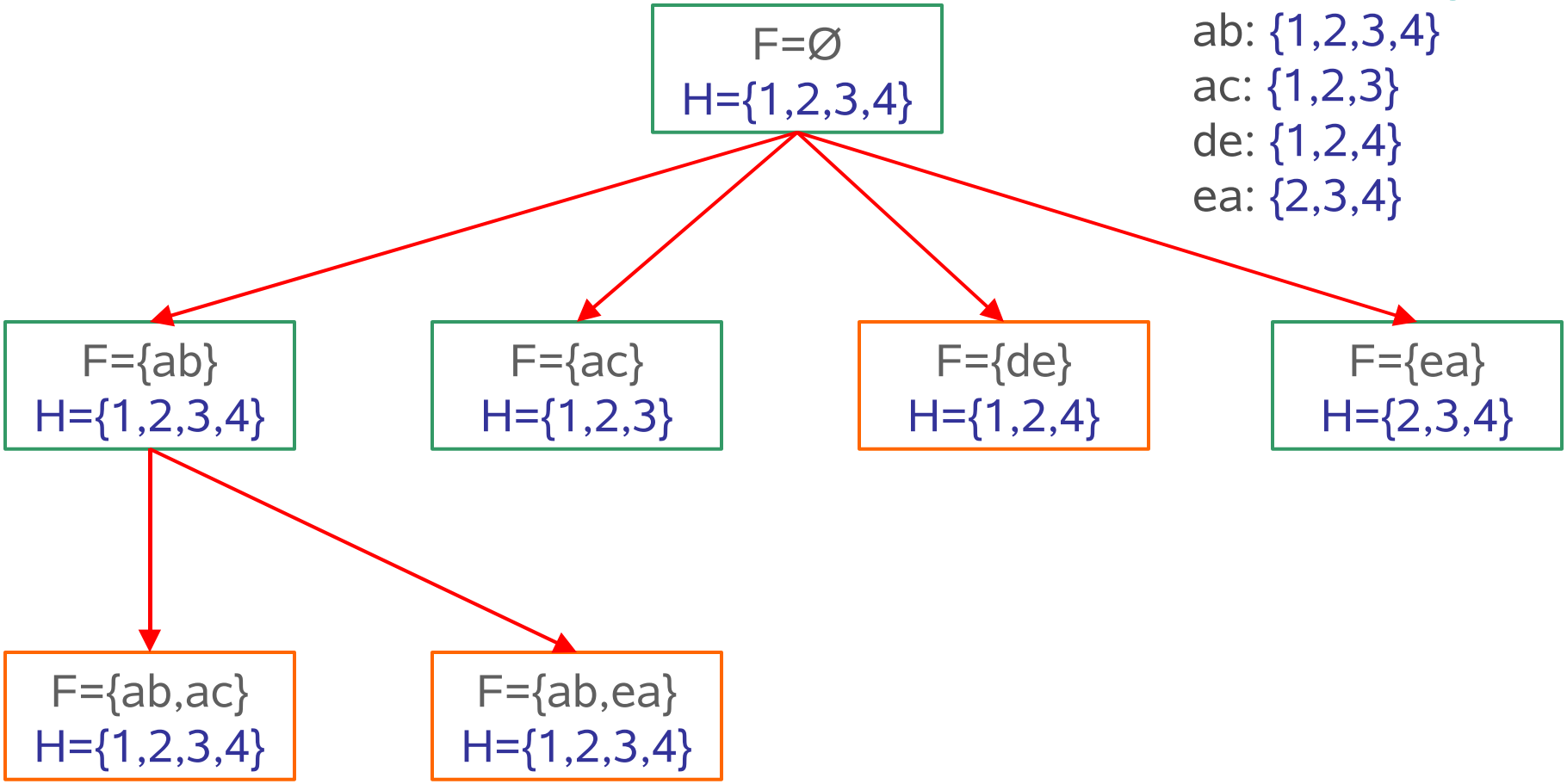
Generalized version of frequent itemset mining !

Adapting Itemset Mining to Edgeset Mining

- From itemsets to edgesets
  - Enumerate all edgesets by extending by one edge at each step
  - Maintain connectivity
- **Depth-first** vs breadth-first traversal
  - Graphs in biological network analysis are larger than those in traditional data mining applications
  - Memory is the bottleneck
- **Set intersection** vs set counting
  - We need to return set of organisms that contain a frequent subgraph

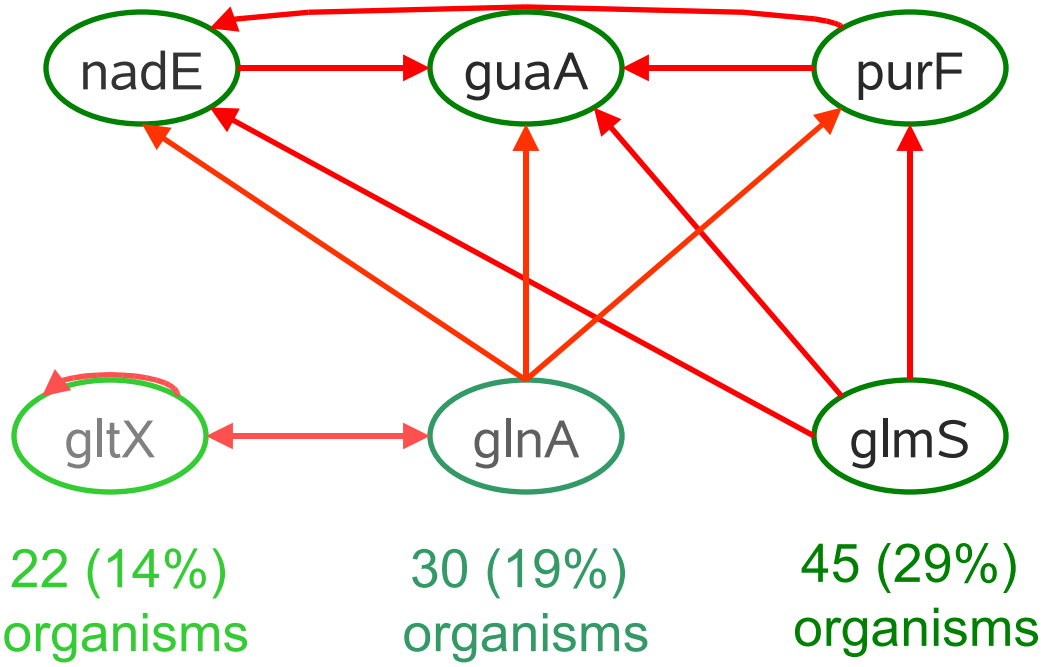
MULE: The Algorithm

Frequent edges:
ab: {1,2,3,4}
ac: {1,2,3}
de: {1,2,4}
ea: {2,3,4}



Frequent Sub-Pathways in KEGG

- Glutamate metabolism
  - 155 organisms



nadE: 6.3.5.1 - NH(3)-dependent NAD(+) synthetase

guaA: 6.3.5.2 – GMP synthase

purF: amidophsphoribosyltransferase

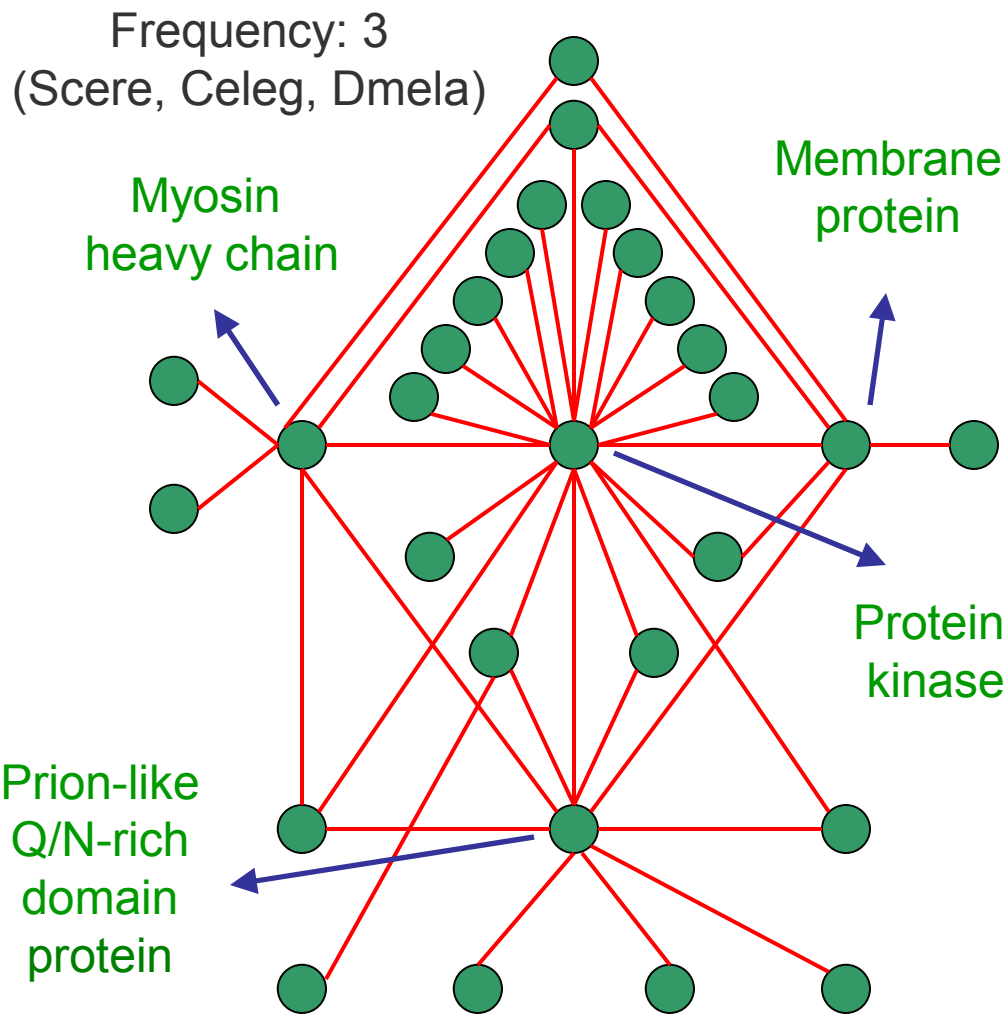
glmS: 2.6.1.5 – glucosamine-fructose-6-phospotase-aminotransferase

glnA: 6.3.1.2 – glutamine synthetase

gltX: 6.1.1.17 – glutamyl-tRNA synthetase

Frequent Interaction Patterns in DIP

- Protein interaction networks for 7 organisms
  - Ecoli, Hpylo, Scere, Celeg, Dmela, Mmusc, Hsapi
  - 44070 interactions between 16783 proteins
- Clustering with TribeMCL & Node contraction
  - 30247 interactions between 6714 protein families



Runtime Characteristics

MULE is up to 3 orders of magnitude faster !

		FSG			MULE		
Dataset	Min. sup. (%)	Runtime (secs.)	Largest pattern	# of Patterns	Runtime (secs.)	Largest pattern	# of Patterns
Glutamate	20	0.2	9	12	0.01	9	12
	16	0.7	10	14	0.01	10	14
	12	5.1	13	39	0.10	13	39
	10	22.7	16	34	0.29	15	34
	8	138.9	16	56	0.99	15	56
Alanine	24	0.1	8	11	0.01	8	11
	20	1.5	11	15	0.02	11	15
	16	4.0	12	21	0.06	12	21
	12	112.7	17	25	1.06	16	25
	10	215.1	17	34	1.72	16	34

Extracting Contracted Patterns

Glutamate (support = 8%)				Alanine (support = 10%)			
Size of contracted pattern	Extraction time (secs.)		Size of extracted pattern	Size of contracted pattern	Extraction time (secs.)		Size of extracted pattern
	FSG	gSpan			FSG	gSpan	
15	10.8	1.12	16	16	54.1	10.13	17
14	12.8	2.42	16	16	24.1	3.92	16
13	1.7	0.31	13	12	0.9	0.27	12
12	0.9	0.30	12	11	0.4	0.13	11
11	0.5	0.08	11	8	0.1	0.01	8
Tot. number of patterns = 56				Tot. number of patterns = 34			
Tot. runtime of FSG alone: 138.9 secs.				Tot. runtime of FSG alone: 215.1 secs.			
Tot. runtime of MULE+FSG: 101.5 secs.				Tot. runtime of MULE+FSG: 162.3 secs.			
Tot. runtime of MULE+gSpan: 17.8 secs.				Tot. runtime of MULE+gSpan: 32.7 secs.			

Conclusions & Future Work

- MULE: An innovative graph mining technique specifically designed for biological networks
  - Conveys significant biological insights at near-interactive rates: A graph equivalent to CLUSTAL-W
  - Can be used as a pre-processor for fast extraction of more detailed patterns
- Improvements on graph mining
  - Statistical significance
    - Accurate probabilistic models
  - Extension to multiple alignment
    - Handling gaps and mismatches