Novel Computational and Integrative Tools for the Analysis of Gene Co-Expression Data

Michael A. Langston
Department of Computer Science
University of Tennessee
currently on leave to
Computer Science and Mathematics Division
Oak Ridge National Laboratory
USA
Technology Mapping

Analysis Tools
- Ontology
- Cis-Regulatory Elements
- Quantitative Trait Loci
- Combinatorial Algorithms
- Bayesian Networks

Biological Knowledge
- Protein Structure
- Gene Regulatory Networks
- Sequence Homology
- Protein function
- Cell Physiology

Microarray Data
Many Network Actions

• *Cis* and *trans* (direct and indirect) regulation
• Post-transcriptional regulation (e.g., alternate splicing)
• μRNA (e.g., functional RNA, RNAi and gene silencing)
• All are forms of co-regulation.
• Not to be confused with mere differential expression.
• Thus the central problem is clique.
• But it’s *NP*-complete to decide clique.
• In fact it’s *NP*-complete even to approximate clique!
• Nevertheless, with new mathematical tools (FPT) we can solve clique optimally using vertex cover.
• Confines “combinatorial explosion” to the parameter.
A Little Complexity Theory

• The Classic View:

P \subset NP \subset \Sigma_2^P \subset \ldots \subset \text{PSPACE} \subset \ldots

“easy”

“hard”

“fuggetaboutit”
Parameter Sensitivity: Instance$(n,k)$

• Suppose our problem is, say, $NP$-complete.
• Consider an algorithm with a time bound such as $O(2^{k+n})$.
• And now one with a time bound more like $O(2^k+n)$.
• Both are exponential in parameter value(s).
• But what happens when $k$ is fixed?
• FPT confines superpolynomial behavior to the parameter.
A Little Complexity Theory

The Parameterized View:

- "solvable" (even if NP-hard!)
- "heuristics only"
- "fuggettaboutit"

FPT  W[1]  W[2]  ...  XP  ...
On Solving Clique

- Clique is a central problem all right, but it’s not FPT (unless the W hierarchy collapses).
- Fortunately, Vertex Cover is FPT.
- And Vertex Cover is a complementary dual to Clique:
Solving Vertex Cover

COMPLEXITY THEORY
- Problem Classification
- Algorithm Selection

PARALLELISM AND GRIDS
- Speedup
- Collaboration

Intellectual Property

GRAPH ALGORITHMS
- Modeling
- Optimization

Available Technologies

RECONFIGURATION
- Hardware Acceleration
- Fast Prototyping
The Vertex Cover Project

- use preprocessing via degree structures
- then kernelize to reduce to a computational core
- employ branching to explore the core
- finally, interleave all three
Sample Grid Architecture

Key: NetSolve's program description file facility

- NetSolve Servers
- NetSolve Client
- NetSolve Agent
- Distributed Storage

Middleware (NetSolve)
Compute Resources (Grid Service Clusters)
Foundational Fabric (Switches and Depots)
Hardware Acceleration

- Reconfigurable devices
- Very different algorithms
- VHDL versus C
- I/O is often the most critical resource

- With current implementations, we are able to solve sub-instances:
  - of size 512 or less,
  - and with speedups north of about 125
The Clique Compute Engine

Pre-Processed Graph \rightarrow \text{Parametric Tuning, Decomposition, and Refinement} \rightarrow \text{Cliques for Post-Processing}

\text{Highly Parallel Computation}

\text{Recalcitrant Sub-problem}

\text{Reconfigurable Technology}

\text{PE} \quad \text{PE} \quad \text{PE} \quad \text{PE} \quad \text{PE} \quad \text{PE} \quad \text{FPGA} \quad \text{FPGA} \quad \text{FPGA}
A simple mechanism. (Sometimes too simple.)

Vertex Cover Driver

- Splitter
- Job Scheduler
- Initialize Branching
- Job List
- Handle Machine
- Branching
- Processor 1
- Processor 2
- ...
Distributed Subtree Splitting

Processor 1 is still active.
Processor 2 is still active.
Processor 3 is still active.

Send a subtree to the job queue.

Pruning is needed at processor 4.
### Sample Results on Protein Sequence Data

<table>
<thead>
<tr>
<th>Graph Name</th>
<th>Graph Size</th>
<th>Cover Size</th>
<th>Instance Type</th>
<th>Sequential Kernelization</th>
<th>Sequential Branching</th>
<th>Parallel Branching</th>
<th>Dynamic Decomposition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SH2-5</td>
<td>839</td>
<td>399</td>
<td>Yes</td>
<td>34 seconds</td>
<td>7 seconds</td>
<td>Not needed</td>
<td>Not needed</td>
</tr>
<tr>
<td>SH2-5</td>
<td>839</td>
<td>398</td>
<td>No</td>
<td>34 seconds</td>
<td>141 minutes</td>
<td>82 minutes</td>
<td>20 minutes</td>
</tr>
<tr>
<td>SH3-10</td>
<td>2466</td>
<td>2044</td>
<td>Yes</td>
<td>203 minutes</td>
<td>~ 5 days</td>
<td>~ 5 days</td>
<td>140 minutes</td>
</tr>
<tr>
<td>SH3-10</td>
<td>2466</td>
<td>2043</td>
<td>No</td>
<td>203 minutes</td>
<td>6+ days</td>
<td>6+ days</td>
<td>620 minutes</td>
</tr>
</tbody>
</table>

So clique size is 422. The hardest computations.

32 PEs @ 500MHz. Load balancing is critical. “No” is harder than “yes.”
A Toolchain for Microarray Analysis

- cDNA or mRNA Microarrays
  - Raw Data
  - Normalization
    - Gene Expression Profiles
    - Compute Spearman’s Rank Coefficients
      - Edge-Weighted Graph
      - Filter With Threshold Value
        - Pre-Processing Tools
          - e.g., Graph Separators and Partitioning
        - Clique Extraction
          - e.g., Maximum Clique*, All Maximal Cliques
            *NP-complete
        - Post-Processing Tools
          - e.g., Neighborhood Search, Subgraph Expansion
        - Validation*
          *Putative and Experimental

UT-ORNL  22 June 2005  DIMACS
A Sample Study

• Data acquisition depends on:
  – organism and tissue type
  – independent variable (e.g., time course, life stages)
  – chip technologies/vendors; cDNA vs mRNA
  – normalization methods, coefficient computations

• In this particular study:
  – 32 *Mus musculus* RI strains
  – brain tissue
  – Affymetrix U74Av2 mRNA Arrays
  – MAS5.0 package, Spearman rank order
Computational Experience

- 12,422 probe set IDs (genes, vertices)
- Over 100M edges
- Employed a variety of thresholds
- Many days of highly parallel CPU time
- With the threshold set at 0.5:
  - the maximum clique size is 369
  - density made this a difficult computation
- But we could do it via FPT:
  - contrast with brute force
Zeroing in on Biological Relevance

• Clique versus clustering
• Too low a threshold produces large cliques, which can be hard to evaluate
• Too high a threshold produces small cliques, which can exaggerate noise
• Iterating, we settle on a threshold of 0.85:
  - maximum clique size is 17
  - there are 5227 maximal cliques
Clique Size Distribution

Number of Cliques

Clique Size

 UT-ORNL  22 June 2005  DIMACS
Gene Distribution Across Cliques

Number of Cliques with Gene

Genes (Probe Set IDs)

UT-ORNL 22 June 2005 DIMACS
Sample Genes of Interest

- 93806_at = Veli3 (aka Lin7c)
- In the human ortholog:
  - structural cytoskeletal protein
  - signal transducer
- Important interactions with Cask, Mask1

Sample Cliques of Interest

- Common CREs
- Enriched Ontologies
- Quantitative Trait Loci
Technology Integration

Gene Expression Data Acquisition and Normalization

- Quantitative Linkage Analysis
- Sequence Analysis (SNPs, CREs)
- Data Mining (Gene Set Enrichment, Biological Ontologies)
- Combinatorial Algorithms, Clique Analysis

WebQTL DB
GeneKeyDB

WebQTL.org
GoTreeMachine
GeneNetViz

Computational Tool Modules
Data Infrastructure
Visualization & Validation

UT-ORNL 22 June 2005 DIMACS
Let’s Remember What We’re Trying to Accomplish

• Every cell in an organism has the same DNA.
• It’s the regulatory mechanisms that seem to change from one tissue type to another.
• Clique gives us putative co-regulation, and with it a set of targets for verification.
• The main goal remains the elucidation of gene regulatory networks.
• Now let’s look at the bigger picture.
How do These Cliques Interact?

• Motivation:
  – capture the notion of regulatory networks interacting with other regulatory networks

• The clique intersection graph:
  – each clique is represented by a vertex
  – if the intersection of a pair of cliques is nonempty, then an edge is added to connect their corresponding vertices
The Clique Intersection Graph for this Data Set

15-cliques in green
16-cliques in black
17-cliques in red
Dealing with Noisy Data

• Clique is the “gold standard.”
• But the data is seldom without errors.
• What we really want are very dense subgraphs.
• It’s straightforward to use neighborhoods, but on real data:
  – 1-neighborhoods produce edge densities of only around 16%.
  – 2-neighborhoods produce edge densities of only around 6%.
Dealing with Noisy Data

Paraclique:
• Clique gloms onto highly connected vertices.
• 280-clique is transformed into a 466-paraclique.
• Edge density is north of 96%.
• Lift and separate.
Paracliques and QTL

There’s a high probability that somewhere in here is a polymorphism controlling this trait.

Seven Quantitative Trait Loci

Transcript abundance can be the phenotype!
Paracliques and QTL
Currently Working On

- More and larger *M. musculus* mRNA arrays
- New *H. sapiens* mRNA arrays (>19k genes)
- The eukaryotic “usual suspects”
- Prokaryotes
  - *Rhodopseudomomas palustris*
    - three operons for nitrogen fixation
  - *Shewanella oneidensis*
    - Metallic precipitates
  - *Synechococcus elongatus*
    - Heatshock cyanobacteria
Ongoing Work, Threshold Setting

- WAG, Kentucky Windage
- Maximum Clique Size
- Scrutinize Key Gene Correlates
- Use Functional Similarity

![Graph showing Functional Similarity via GO and Distance via Spearman Rank with an inflection point.]

Graph Algorithms Research Laboratory – University of Tennessee

22 June 2005
Ongoing Implementation Efforts

• Sample codes released:
  - Clustal XP (Phylogeny)
  - CAMDA (Disease Screening)
• Building out to clique variants
• Establishing a Portal at ORNL
• Porting codes to SGI Altix, Cray X1
• ORNL National Leadership Class Facility
• Most importantly: professional interaction
Collaborators and Acknowledgments

UT:
  Naima Moustaid-Moussa
  Greg Peterson
  Arnold Saxton

ORNL:
  Stefan Kirov
  Frank Larimer
  Nagiza Samatova
  Jay Snoddy
  Brynn Voy
  Bing Zhang

UTHSC:
  Elissa Chesler
  Ivan Gerling
  Rob Williams

International:
  Mike Fellows
  Henning Fernau
  Rolf Niedermeier
  Fran Rosamond

Postdocs:
  Faisal Abu-Khzam (CS)
  Nicole Baldwin (µBio)
  Henry Suters (Math)

Students:
  John Eblen
  Xinxia Peng
  Chris Symons
  Yun Zhang
  Jon Scharff
  Josh Steadmon