

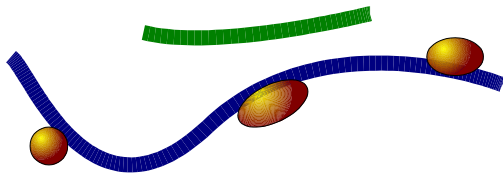
How to reconstruct a large genetic network from n gene perturbations in fewer than n^2 easy steps.

Andreas Wagner (2001) Bioinformatics 17:1183-97.
Contributed to the bioinformatics lecture by Samuel Bandara

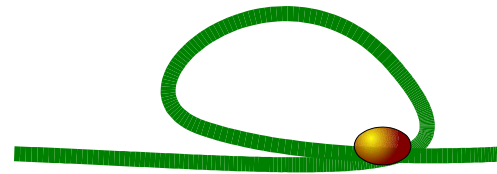
Agenda

- Represent a genetic network by a directed graph $G(V, E)$
- Discover direct & indirect interaction of genes by perturbation
- Reconstruct the most parsimonious graph from results
- Improve reconstruction by considering pos and neg different
- Improve reconstruction by using double-mutant experiments
- Benchmark the algorithm on simulated networks

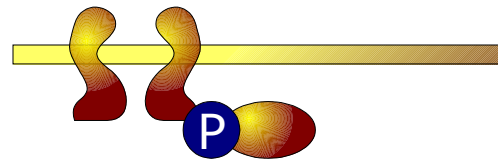
Gene activity manifests itself in various cellular mechanisms



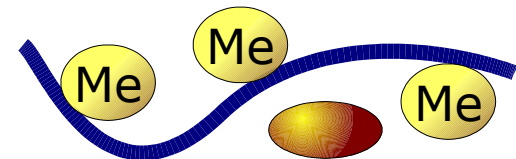
gene expression



alternative splicing



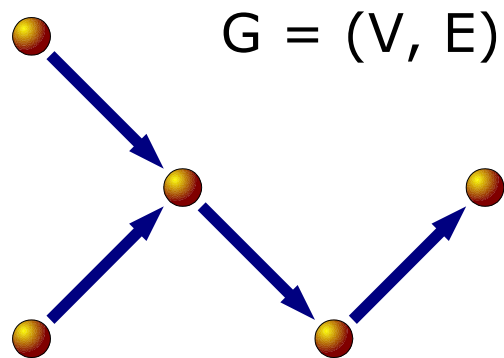
phosphorylation



methylation

- HT techniques for measuring activities are available
- activity of one gene controls the activity of others
- interaction of genes can be discovered by perturbation

Draw the graph of a genetic network to explain the molecular details of life



Various representations

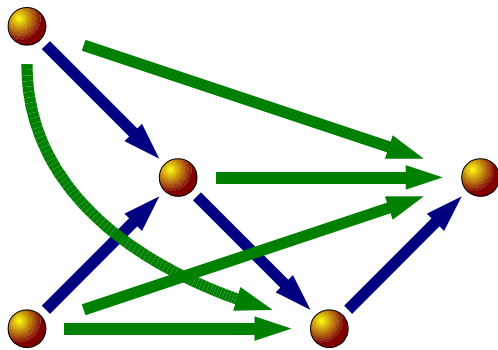
- V : {nodes}, E : {edges}
- adjacency lists
- adjacency matrix
- incidence matrix

- genetic network is represented by a directed graph
- nodes correspond to the genes of the network
- directed edges describe the interaction of genes

Both direct and indirect interaction of genes can be discovered by genetic perturbation

- Fraser *et al.*: RNAi on *C. elegans* chromosome I by feeding bacteria expressing double-stranded RNA
- Hughes *et al.*: 276 deletion mutants, 11 tetracycline responsive mutants, and 13 perturbations by drugs in *S. cerevisiae*
- Spradling *et al.* 1045 strains of *D. melanogaster* containing gene perturbations caused by insertion of transposons
- ...

Perturbation experiments do not discriminate between direct and indirect relationships



- $Adj(i)$: the adjacency set
{ nodes adjacent to node i }
- $Acc(i)$: the accessibility set
{ nodes accessible from i }
- clearly, $Adj(i) \subseteq Acc(i)$

find most parsimonious graph that explains observation

- contains the least number of edges
- observation explained if accessibility maintained

$$\min(H | \bar{H} = \bar{G}) \Leftrightarrow \min(H | G \subseteq \bar{H})$$

Prove that shortcut-free graph explaining the observations is most parsimonious and unique

- THEOREM 1: There is exactly one graph G_{pars} that explains the observations and that is more parsimonious than any other.
 - LEMMA 1: For any observation, there exists an explaining graph that is free of shortcuts.
 - LEMMA 2: The edges of a shortcut-free graph are a subset of the edges of any other graph explaining the observations.
 - COROLLARY 1: The shortcut-free graph explaining the observations is unique and is the most parsimonious one.
- for the proof to hold, the observed graph is considered acyclic

10011010

00101110

11001101

Prove that deleting shortcuts yields most parsimonious graph explaining observations

- THEOREM 2: The set of edges in the most parsimonious graph explaining the observations is the set of observations after removal of those edges that are short-cuts of a longer path.

$$\forall i \in V(G_{\text{pars}}) \text{ Adj}(i) = \text{Acc}(i) \setminus \bigcup_{j \in \text{Acc}(i)} \text{Acc}(j)$$

- PROOF PART 1: Edges of G_{pars} are a subset of the accessibility after removal of those edges that are shortcuts: $\text{Adj}(i) \subseteq \text{RHS}$
- PROOF PART 2: The accessibility after removal of those edges that are shortcuts, is a subset of G_{pars} : $\text{RHS} \subseteq \text{Adj}(i)$
- $\text{Adj}(i) \subseteq \text{RHS} \wedge \text{RHS} \subseteq \text{Adj}(i) \Leftrightarrow \text{Adj}(i) = \text{RHS}$

10011010

00101110

11001101

The accessibility graph is pruned recursively with particular prevention from revisiting nodes

PRUNE_ACC(i)

for all nodes j in Acc(i)

if Acc(j) = empty then declare j as visited

else call PRUNE_ACC(j)

for all nodes j in Acc(i)

for all nodes k in Adj(j)

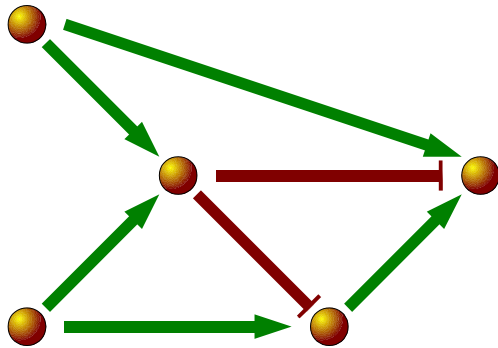
if k in Acc(i) then delete k from Adj(i)

declare node i as visited

for all nodes i of G

if node not visited yet then call PRUNE_ACC(i)

Considering whether regulation is positive or negative improves quality of reconstruction



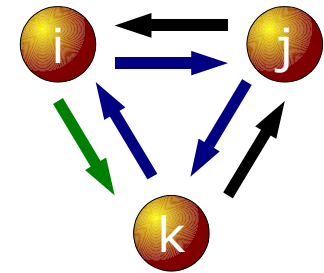
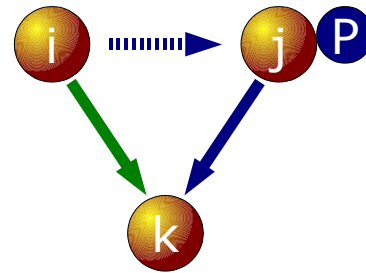
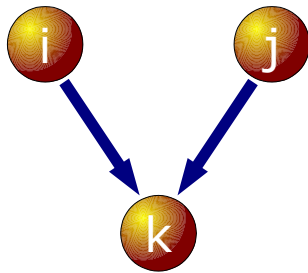
- **pos**: +1 in adjacency matrix
- **neg**: -1 in adjacency matrix
- sign of products along paths indicate redundancy if equal

for all nodes j in $\text{Acc}(i)$

for all nodes k in $\text{Adj}(j)$

if k in $\text{Acc}(i)$ and $\text{Acc}(i, k) = \text{Acc}(i, j) * \text{Acc}(j, k)$ then
delete k from $\text{Adj}(i)$

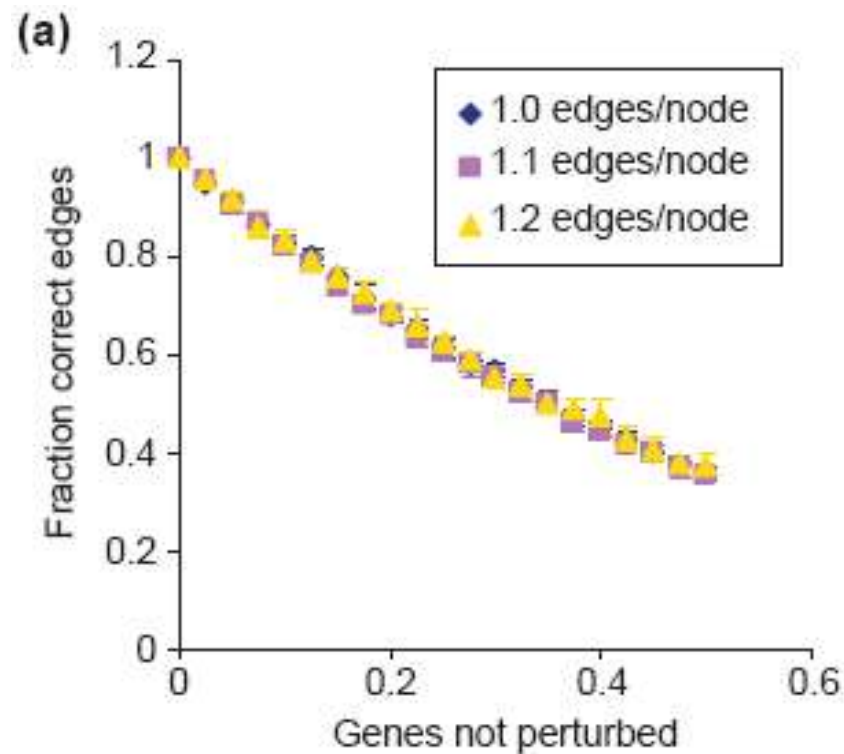
Using data from double-mutant experiments allows for refinement of network structure



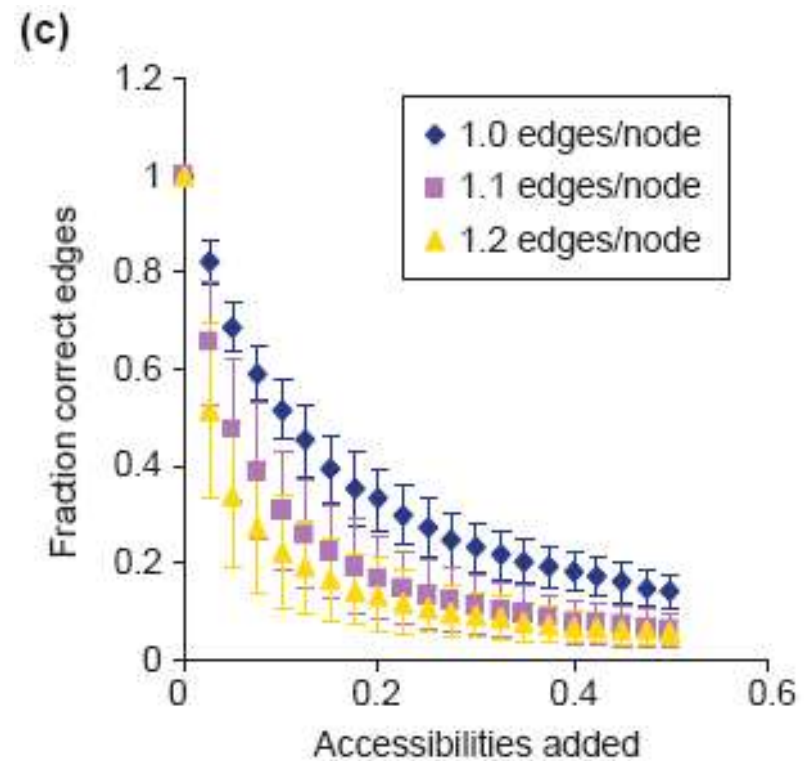
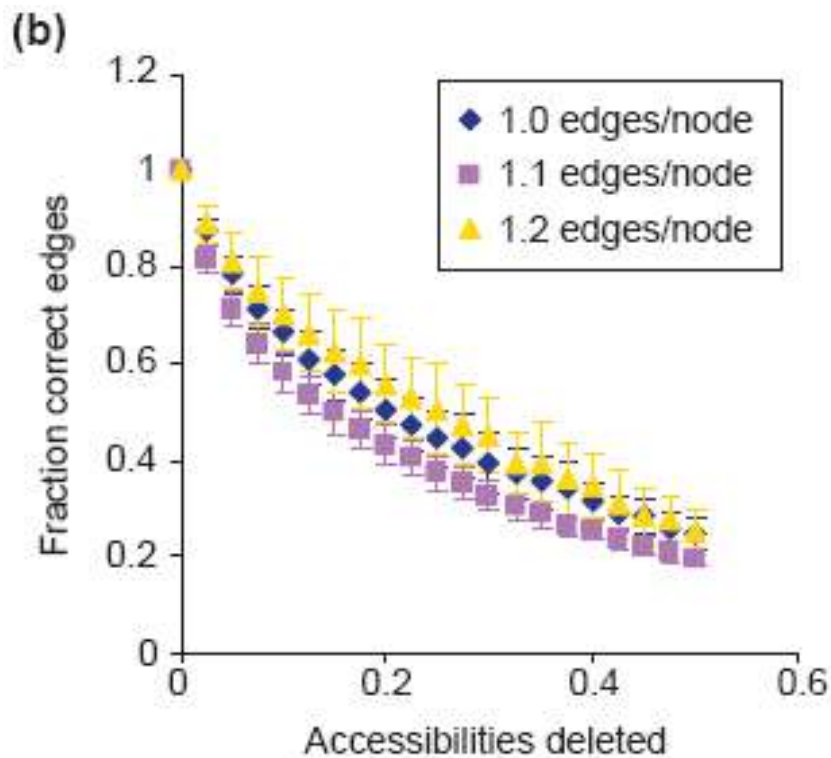
$k \in \text{Acc}_j(i)$, $k \notin \text{Acc}(i)$
→ add k to $\text{Adj}(i)$
→ add k to $\text{Adj}(j)$

$k \in \text{Adj}(i)$, $k \notin \text{Acc}_j(i)$, $k \in \text{Acc}(j)$
→ remove k from $\text{Adj}(i)$
→ add j to $\text{Adj}(i)$ if $j \notin \text{Acc}(i)$

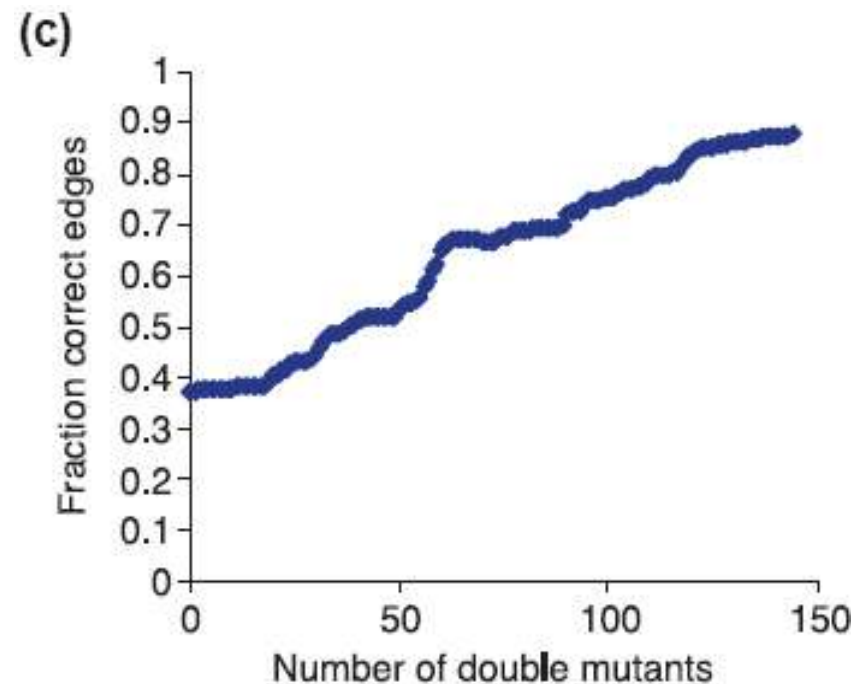
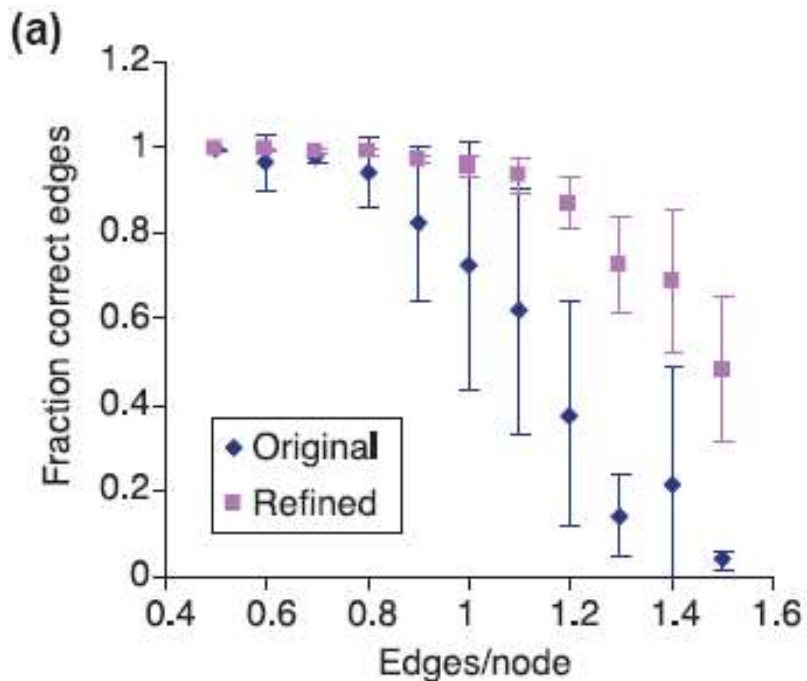
Sensitivity of the algorithm to incomplete data



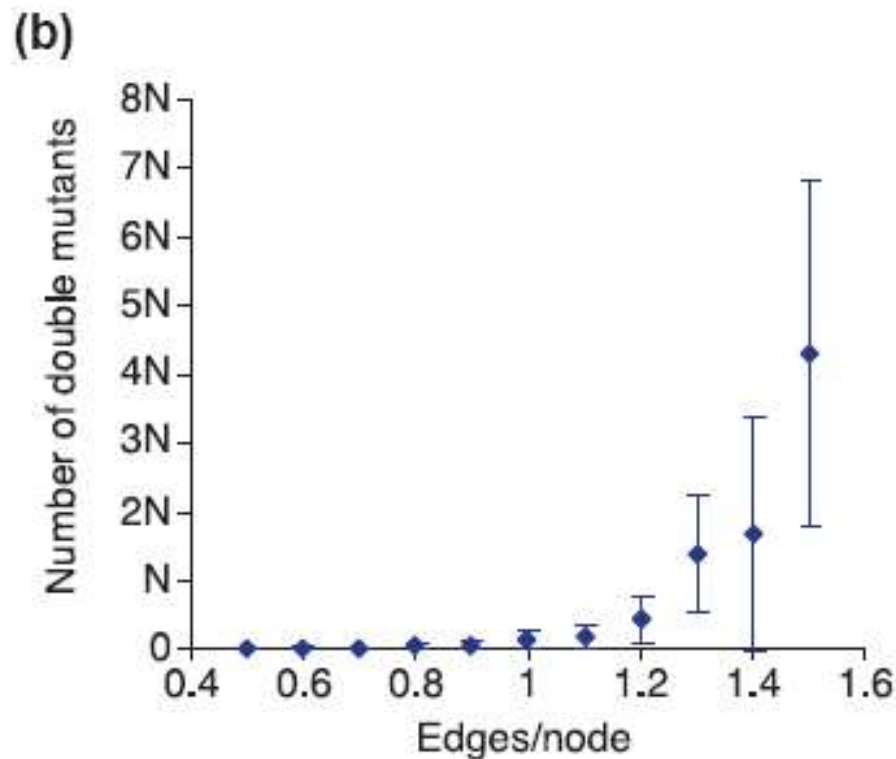
Sensitivity of the algorithm to noisy data



Quality of reconstruction using data obtained from simulated double-mutant experiments



Few double-mutant experiments need to be conducted in order to yield accurate results



Conclusion

- reconstructs the most parsimonious graph explaining the observations made in perturbation experiments
- is much faster than $O(n^2)$ – human genome on desktop
- cyclic interaction is difficult to deal with
- variety of different gene activities limits source of data
- the most parsimonious graph needn't be the real one