Inference and computing with decomposable graphs

Peter Green[1] Alun Thomas[2]

[1]School of Mathematics
University of Bristol

[2]Genetic Epidemiology
University of Utah

6 September 2011 / Bayes 250

Outline

Graphical models

The conditional independence graph of a multivariate distribution (for a random vector X, say) tells us much about the structure of the distribution. Recall that $G = (V, E)$ where the vertex set V is the set of indices of the components of X, and there is an (undirected) edge between vertices i and j, written $i \sim j$

$$\text{unless} \quad X_i \perp\!\!\!\perp X_j \mid X_{V \setminus \{i,j\}}$$

Under conditions (positivity is sufficient), global and local Markov properties also hold.

Given i.i.d. observations on X, we are often interested in inferring G, sometimes known as structural learning.

Decomposable graphical models

The case where G is decomposable has been much studied. Decomposability is a graph theory concept with statistical and computational implications.

A graph is complete if every pair of vertices is joined by an edge. A maximal complete subgraph is called a clique. An ordering of the cliques of an undirected graph, $(C_1, C_2, \ldots, C_c)$ is said to be perfect if for each $i = 2, 3, \ldots, c$, there exists $h = h(i)$ such that

$$S_i = C_i \cap \bigcup_{j=1}^{i-1} C_j \subseteq C_h$$

The sets S_i are called separators. If an undirected graph admits a perfect ordering, it is said to be decomposable.

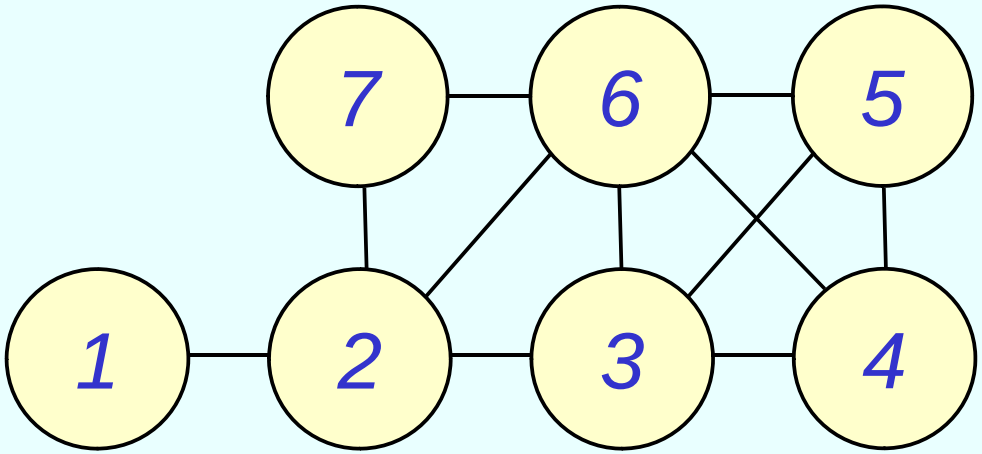
Decomposability: junction trees

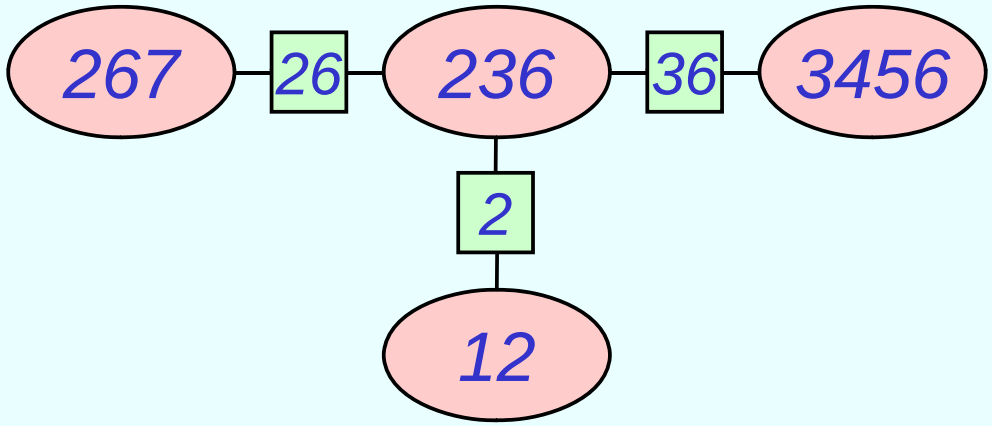
Decomposable graphs are also known as triangulated: a graph is decomposable if and only if it has no chordless k-cycles for $k \geq 4$.

A perfect ordering guides the construction of a junction tree: a graph whose vertices are cliques, and with edges between C_i and $C_{h(i)}$, often labelled with S_i, for $i = 2, 3, \ldots, c$. There may be many perfect orderings, and many junction trees, for a given decomposable graph.

A small decomposable graph

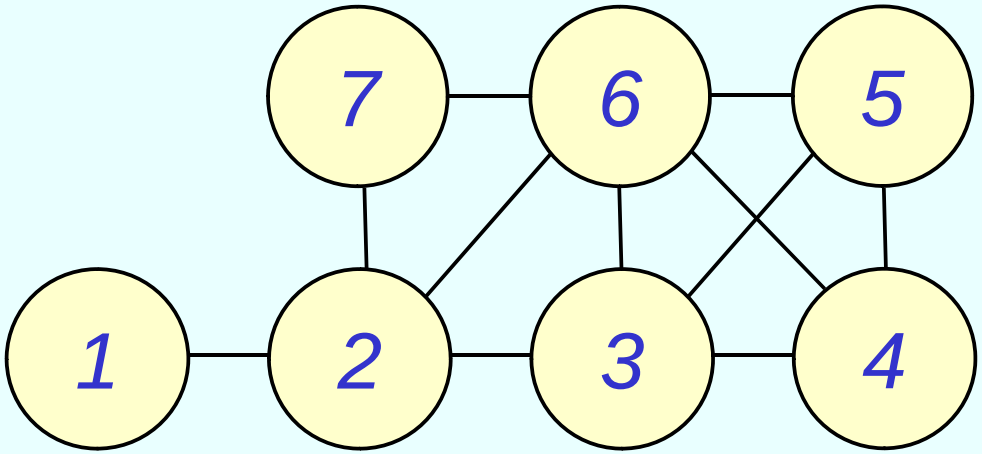
Non-uniqueness of junction tree

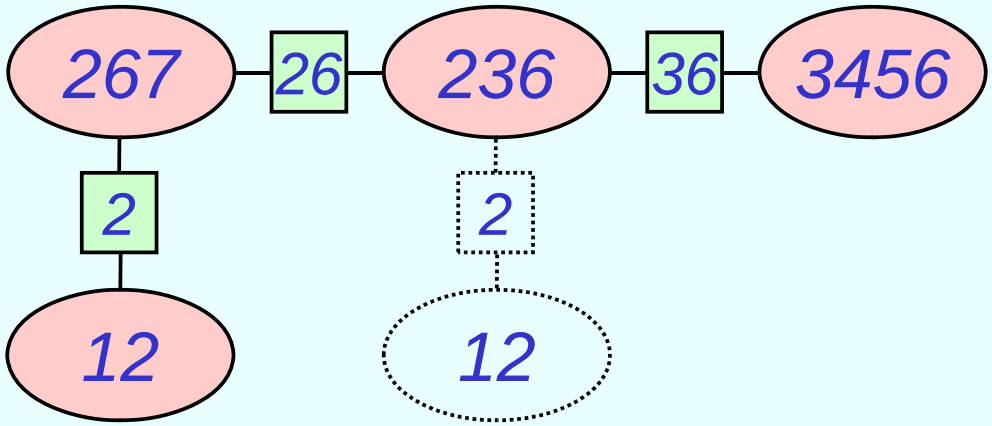




A small decomposable graph

Non-uniqueness of junction tree





Probabilistic significance of decomposability

If the distribution of a random vector X has a decomposable conditional independence graph, then it has a remarkable representation in terms of (often low-dimensional) marginals:

$$p(X) = \frac{\prod_{i=1}^{c} p(X_{C_i})}{\prod_{i=2}^{c} p(X_{S_i})}$$

This is the ultimate generalisation of the fact that for an ordinary Markov chain

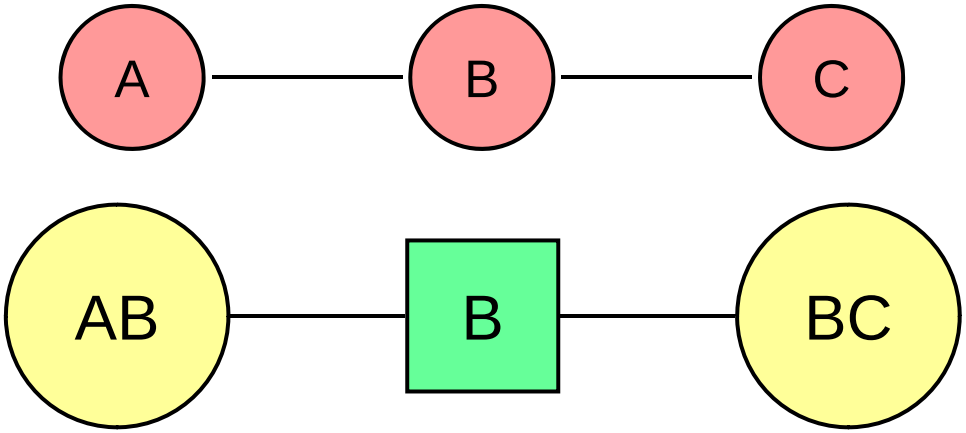
$$p(X) = p(X_0) \prod_{i=1}^{N} p(X_i|X_{i-1}) = \frac{\prod_{i=1}^{N} p(X_{\{i-1,i\}})}{\prod_{i=2}^{N-1} p(X_{i-1})}$$

For a general decomposable graph, the same kind of factorisation follows the edges of the junction tree.

Computational significance of decomposability

There are many consequences for computing with distributions on decomposable graphs, including junction tree algorithms (message passing/probability propagation) for Bayes nets (discrete graphical models).

Message passing



	A=0	A=1
B=0	3/4	1/4
B=1	2/3	1/3

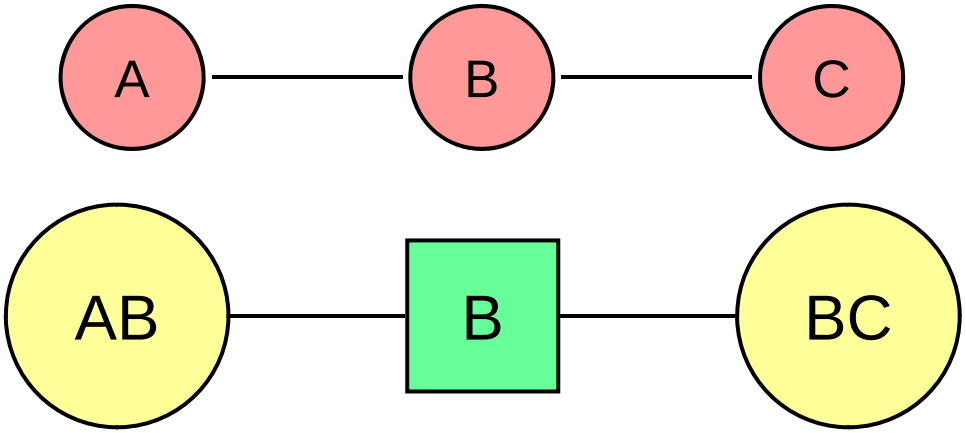
B=0	1
B=1	1

	C=0	C=1
B=0	.3	.1
B=1	.4	.2

	A=0	A=1
B=0	3/4×.4/1	1/4 ×.4/1
B=1	2/3 ×.6/1	1/3 ×.6/1

B=0	.4
B=1	.6

Message passing



	A=0	A=1
B=0	.3	.1
B=1	.4	.2

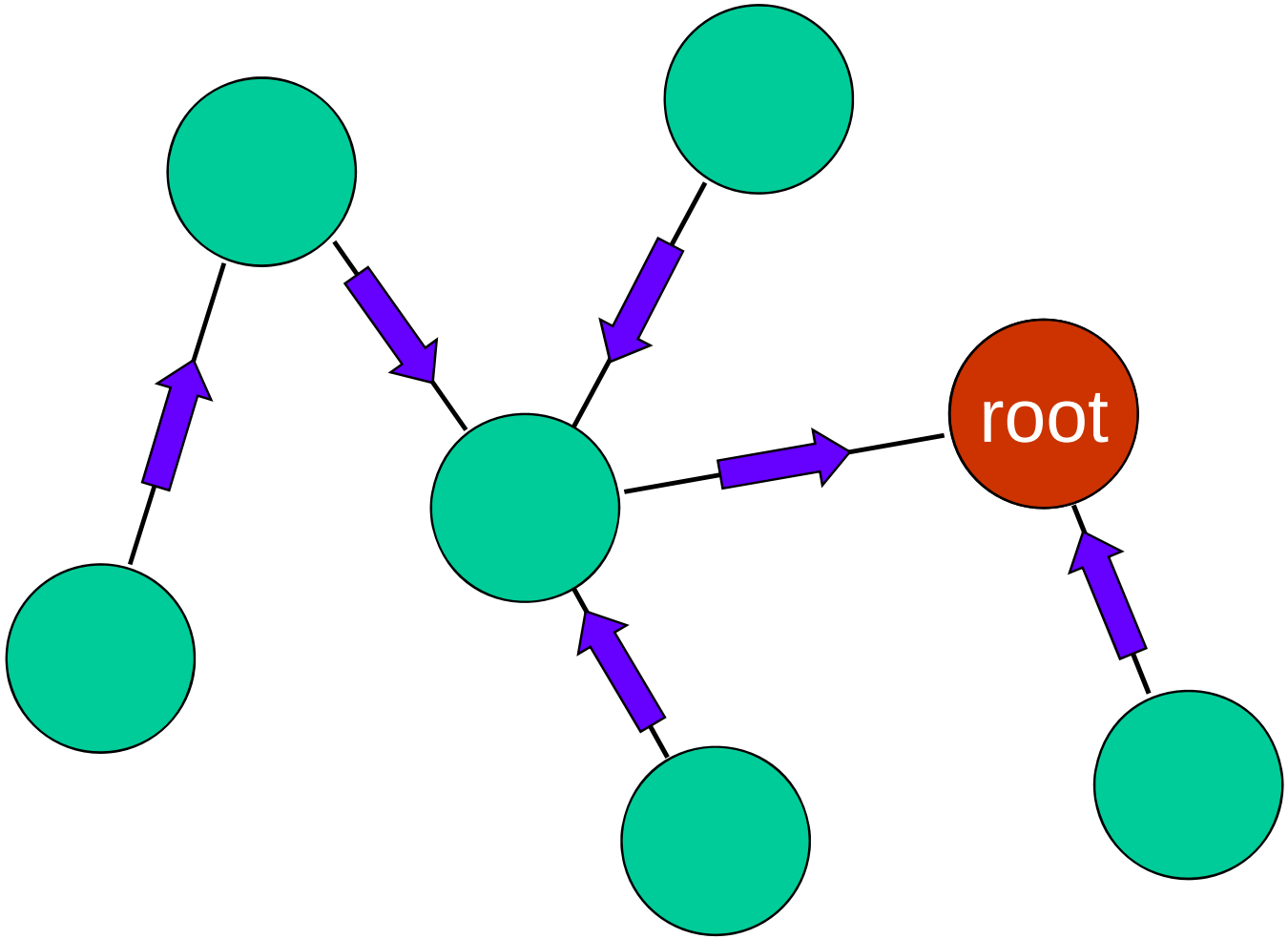
B=0	.4
B=1	.6

	C=0	C=1
B=0	.3	.1
B=1	.4	.2

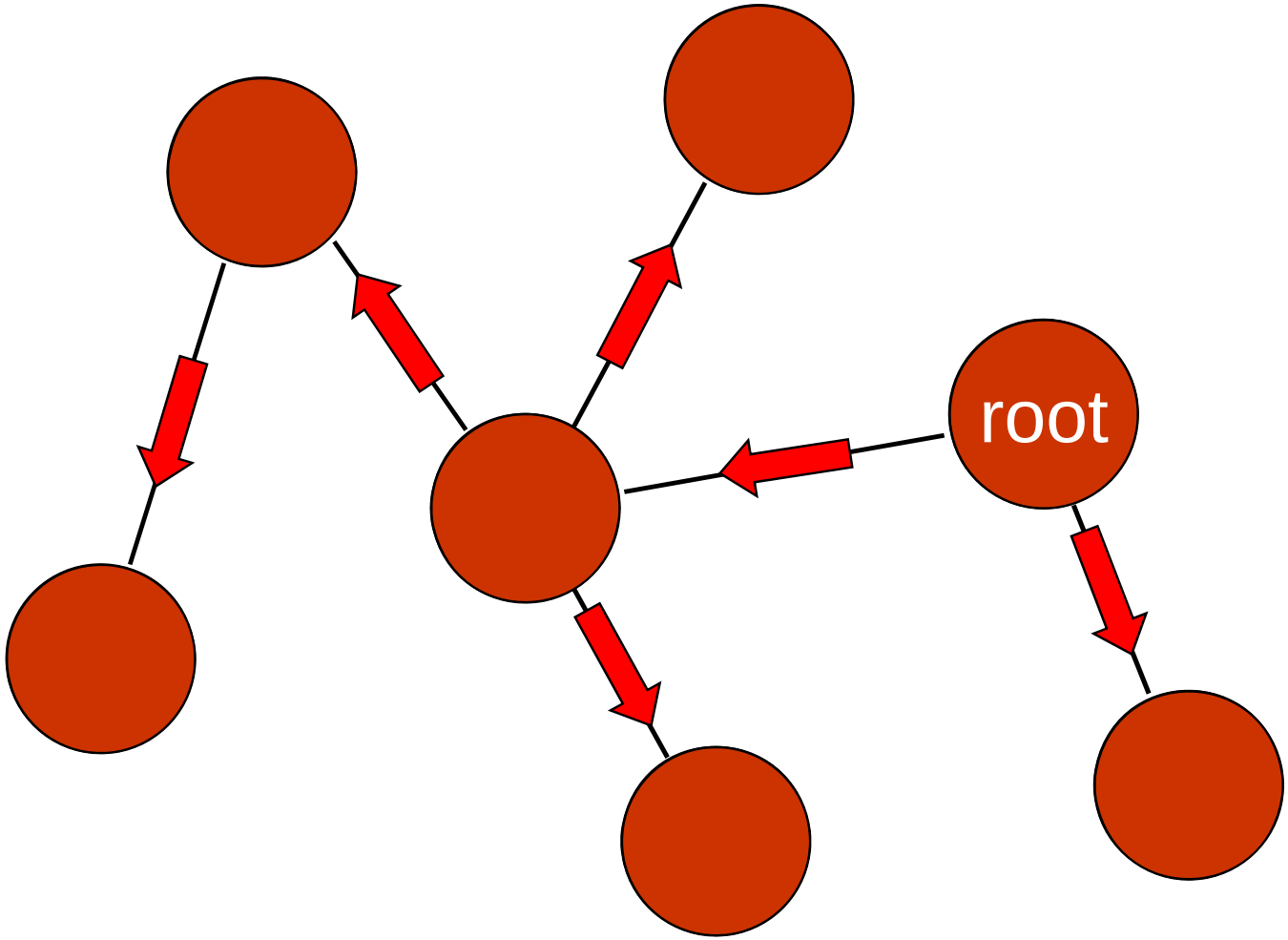
	A=0	A=1
B=0	3/4×.4/1	1/4 ×.4/1
B=1	2/3 ×.6/1	1/3 ×.6/1

B=0	.4
B=1	.6

Scheduling the messages



Scheduling the messages



Statistical significance of decomposability

Maximum likelihood estimates can be computed exactly for contingency tables and multivariate Gaussian distributions on decomposable graphs, and there are exact tests for conditional independence. Some of this theory extends to mixed data models based on CG distributions.

In Bayesian modelling, the ideas of hyper Markov modelling allow the construction of prior distributions respecting the graphical structure, which in turn supports the adoption of priors that are guaranteed to be consistent across models.

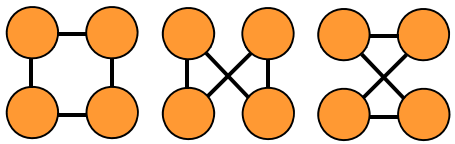
The clique–separator factorisation yields dramatic speed-ups in computing MCMC updates in structural learning, and in simulation and posterior analysis of fitted models.

How restrictive is decomposability?

How many graphs are decomposable?

There are $2^{\binom{v}{2}}$ graphs altogether on v vertices.

For $v \leq 3$ vertices, all are decomposable
for 4 vertices, $61/64$
for 6, $\approx 80\%$
for 16, $\approx 45\%$.

The 3 non-decomposable 4-vertex graphs: 

Bayesian graphical model determination

Given n i.i.d. samples $\mathbf{X} = (X_1, X_2, \ldots, X_n)$ from a multivariate distribution on $\mathcal{R}^v$ parameterised by the graph G and parameters ψ, a typical formulation takes the form

$$p(G, \psi, \mathbf{X}) = p(G)p(\psi|G)p(\mathbf{X}|G, \psi)$$

and we perform joint structural/quantitative learning by computing the posterior $p(G, \psi|\mathbf{X}) \propto p(G, \psi, \mathbf{X})$.

Decomposable G: see Giudici & G (1999) (Gaussian case) and by Giudici, G & Tarantola (2000) (contingency table case). These follow the important work of Dawid & Lauritzen (1993) on hyper-Markov laws that encode parameter priors $p(\psi|G)$ that are consistent across G.

Bayesian graphical model determination

Given n i.i.d. samples $\mathbf{X} = (X_1, X_2, \ldots, X_n)$ from a multivariate distribution on $\mathcal{R}^v$ parameterised by the graph G and parameters ψ, a typical formulation takes the form

$$p(G, \psi, \mathbf{X}) = p(G)p(\psi|G)p(\mathbf{X}|G, \psi)$$

and we perform joint structural/quantitative learning by computing the posterior $p(G, \psi|\mathbf{X}) \propto p(G, \psi, \mathbf{X})$.

Decomposable G: see Giudici & G (1999) (Gaussian case) and by Giudici, G & Tarantola (2000) (contingency table case). These follow the important work of Dawid & Lauritzen (1993) on hyper-Markov laws that encode parameter priors $p(\psi|G)$ that are consistent across G.

Bayesian graphical model determination

Given n i.i.d. samples $\mathbf{X} = (X_1, X_2, \ldots, X_n)$ from a multivariate distribution on $\mathcal{R}^v$ parameterised by the graph G and parameters ψ, a typical formulation takes the form

$$p(G, \psi, \mathbf{X}) = p(G)p(\psi|G)p(\mathbf{X}|G, \psi)$$

and we perform joint structural/quantitative learning by computing the posterior $p(G, \psi|\mathbf{X}) \propto p(G, \psi, \mathbf{X})$.

General G: Earlier and later work, by Dellaportas & Forster and others – but use non-hierarchical non-necessarily-consistent formulations. See also Jones *et al*, *Stat. Sci.*, 2005.

Bayesian graphical model determination

The Giudici & G work on decomposable graphical gaussian model determination considers the joint posterior $p(G, \psi|\mathbf{X})$. In the gaussian case $\mathbf{X} \sim N_v(\mu, \Sigma)$, the graph G is encoded in the pattern of zeroes in the concentration (inverse variance) matrix:

$$(\Sigma^{-1})_{ij} = 0 \Leftrightarrow X_i \perp\!\!\!\perp X_j \mid X_{V\setminus\{i,j\}}$$

The model places a hyper inverse Wishart prior on Σ^{-1}, in various versions, and exploits ideas of covariance selection and positive definite matrix completion.

Bayesian graphical model determination

In MCMC sampling using single-edge moves, a junction tree representation of the current G permits both

- cheap pre-testing that the proposed new graph G' is decomposable
- fast local updating of the graph from G to G' when the move passes the Metropolis–Hastings acceptance test

Pre-tests for maintaining decomposability

Frydenberg & Lauritzen: Let G and G' be decomposable graphs on the same vertex set, with G' formed from G by the addition of exactly one edge. Then this edge must be contained in exactly one clique of G.

Giudici & Green: If a and b are non-adjacent vertices in a decomposable graph G, then the graph G' formed from G by connecting (a, b) is decomposable if and only if either

1. a and b are in different connected components, or
2. they are in the same component of G, and there exist cliques $a \cup R$ and $b \cup T$, for which $S = R \cap T$ is a separator on the path between $a \cup R$ and $b \cup T$ in a junction forest representing G.

Pre-tests for maintaining decomposability

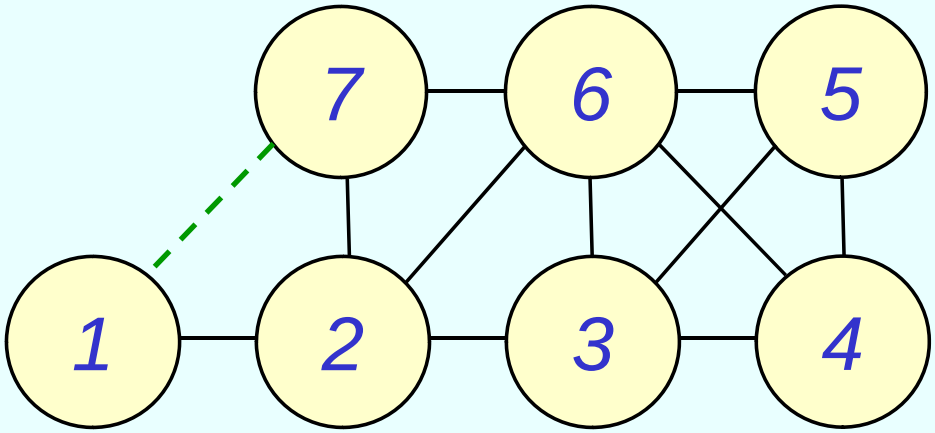
Frydenberg & Lauritzen: Let G and G' be decomposable graphs on the same vertex set, with G' formed from G by the addition of exactly one edge. Then this edge must be contained in exactly one clique of G.

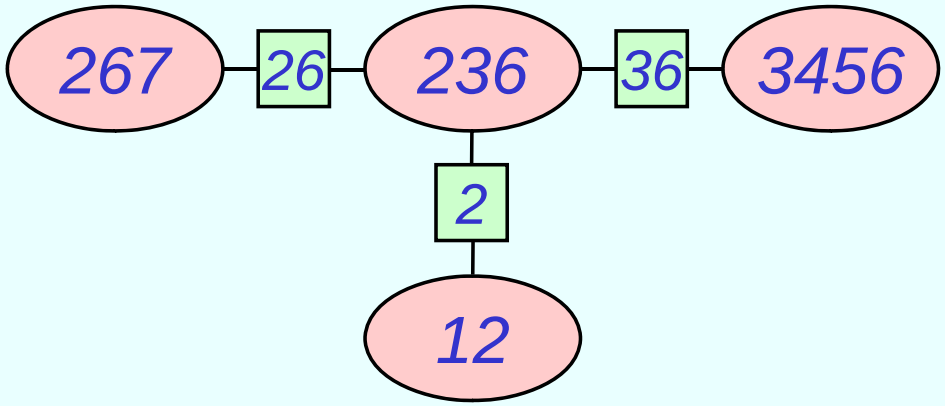
Giudici & Green: If a and b are non-adjacent vertices in a decomposable graph G, then the graph G' formed from G by connecting (a, b) is decomposable if and only if either

1. a and b are in different connected components, or
2. they are in the same component of G, and there exist cliques $a \cup R$ and $b \cup T$, for which $S = R \cap T$ is a separator on the path between $a \cup R$ and $b \cup T$ in a junction forest representing G.

Single edge move

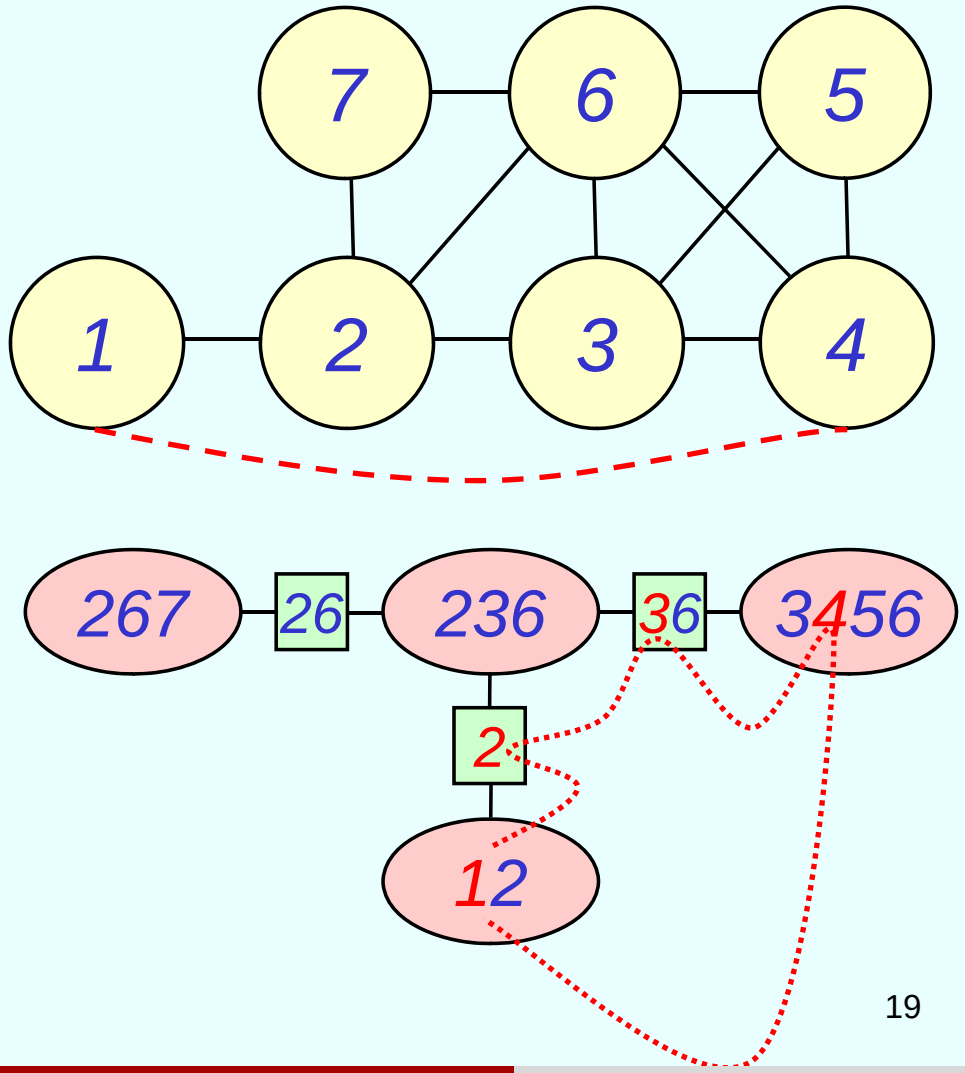
You can add edge (1,7) since $1 \cup R$ and $7 \cup T$ are cliques (with $R=\{2\}$ and $T=\{2,6\}$) and $R \cap T=\{2\}$ is a separator on path between them





Single edge move

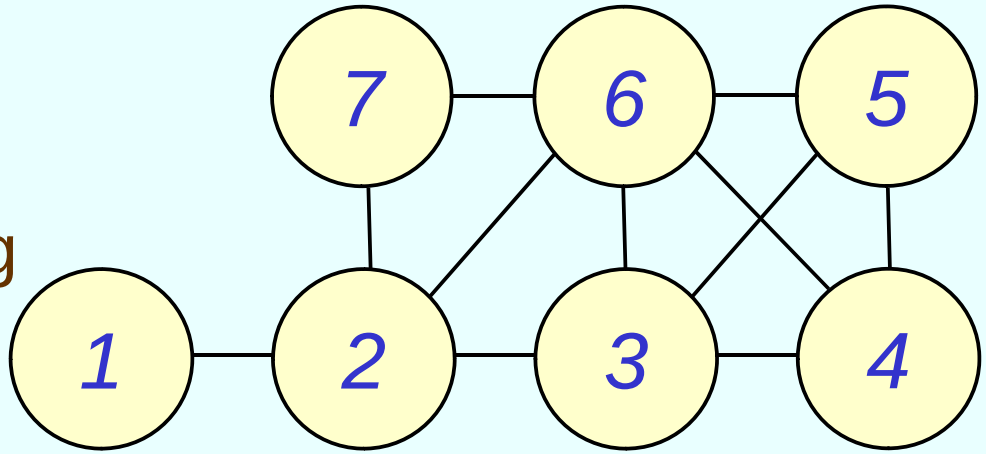
You cannot add edge (1,4) since the only cliques containing 1 and 4 resp. are {1,2} and {3,4,5,6}, and {2}∩{3,5,6} is not a separator on path between them

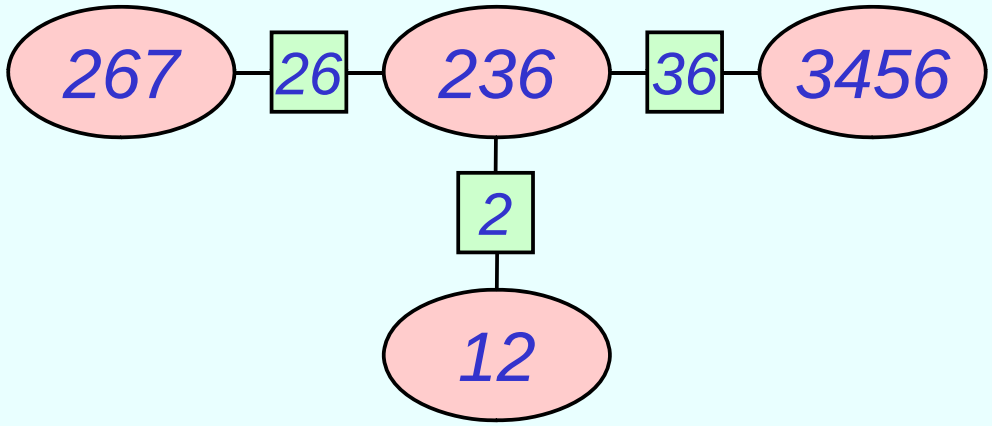


19

Single edge move

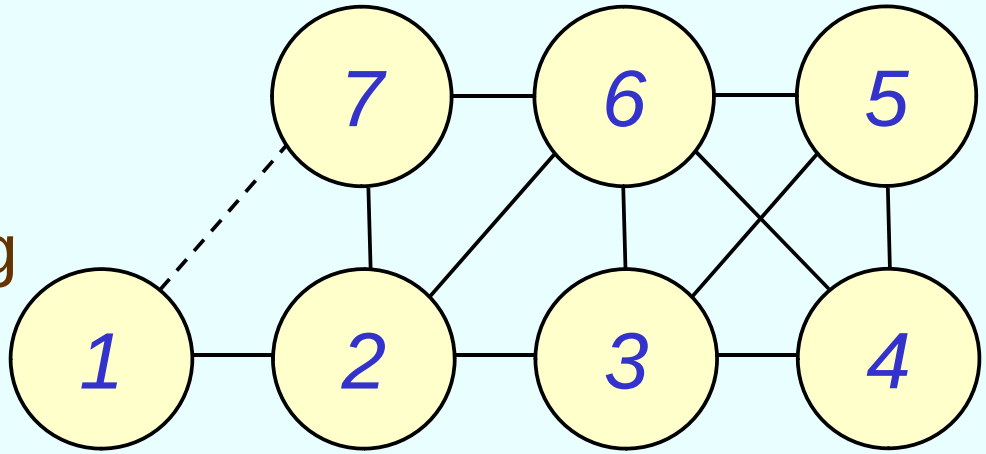
Once the test is complete, actually committing to adding or deleting the edge is little work



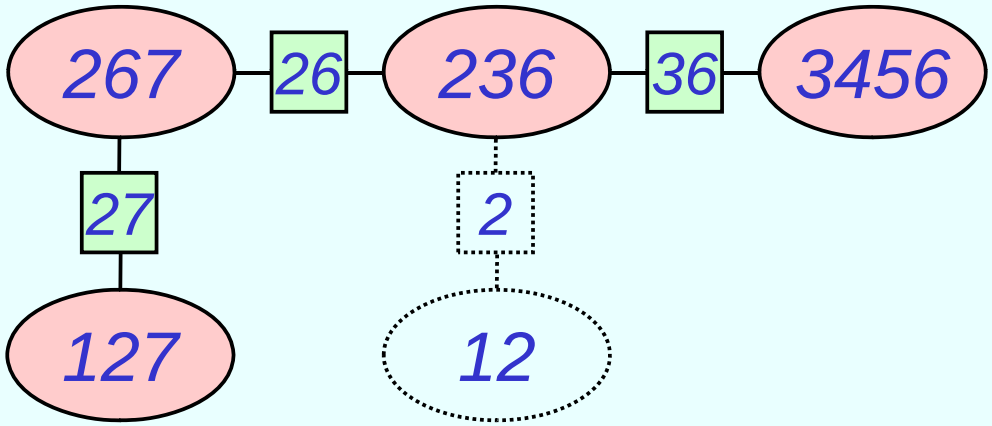


Single edge move

Once the test is complete, actually committing to adding or deleting the edge is little work

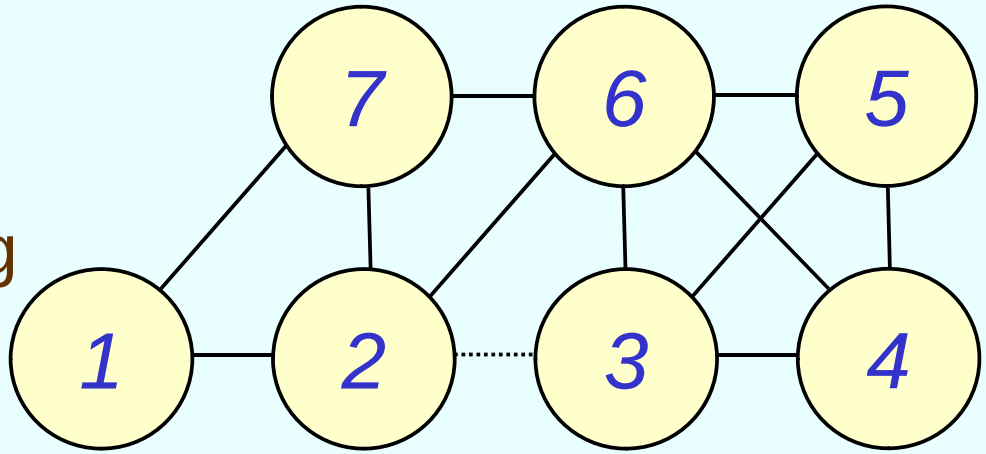


It makes only a (relatively) local change to the junction tree

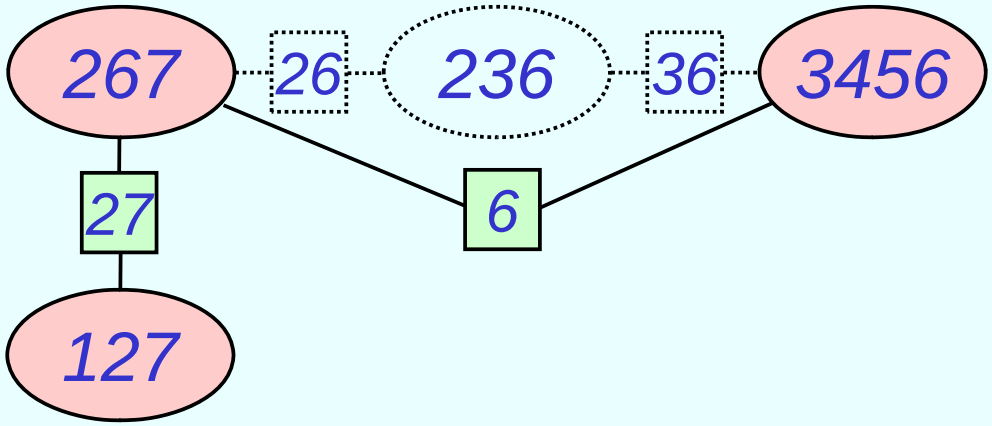


Single edge move

Once the test is complete, actually committing to adding or deleting the edge is little work

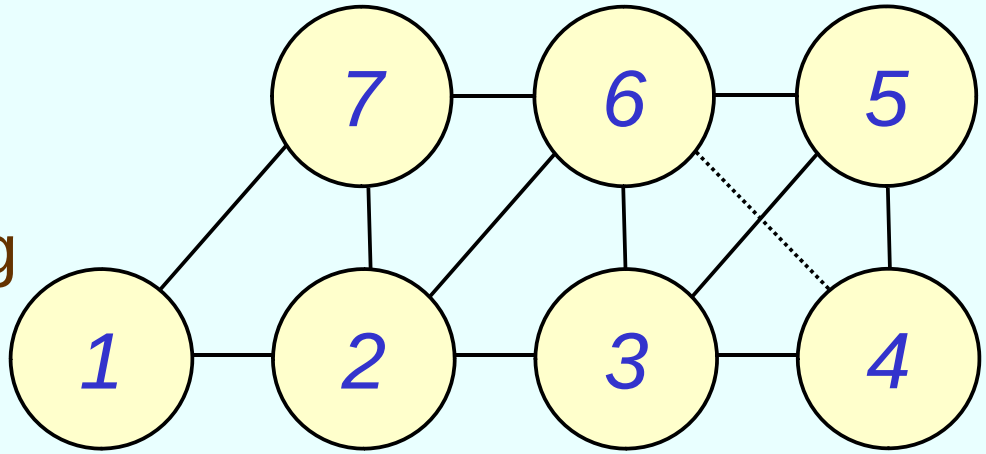


It makes only a (relatively) local change to the junction tree

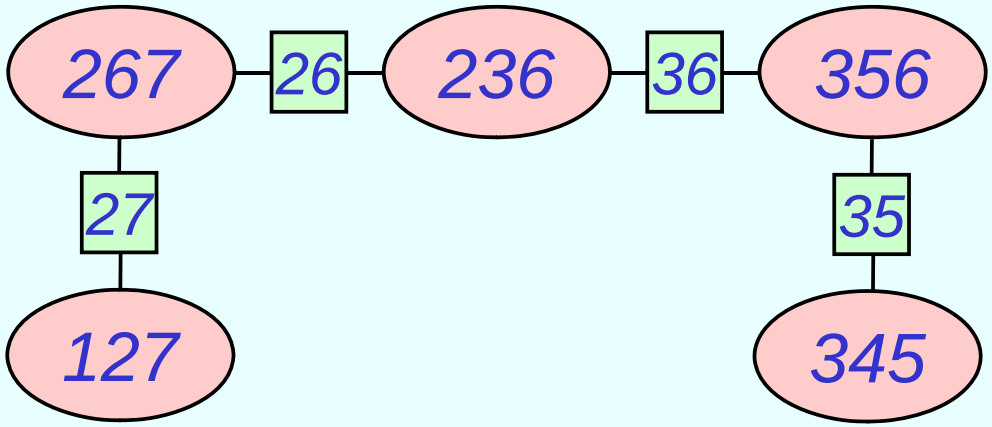


Single edge move

Once the test is complete, actually committing to adding or deleting the edge is little work



It makes only a (relatively) local change to the junction tree



Bayesian graphical model determination

In MCMC sampling using single-edge moves, a junction tree representation of the current G permits both

- cheap pre-testing that the proposed new graph G' is decomposable
- fast local updating of the graph from G to G' when the move passes the Metropolis–Hastings acceptance test

However, the current junction tree may need to be manipulated to a different tree representing the current graph G before the move can be completed.

Using the junction tree as the state

Can we by-pass this manipulation by using directly the junction tree J as part of the model parameterisation, in place of the graph G?

This means augmenting the model so that, conditional on G, the junction tree J is a priori drawn uniformly from among all equivalent junction trees, thus replacing the prior $p(G)$ on decomposable graphs by

$$\widetilde{p}(J) = \frac{p(G(J))}{\mu(G(J))}$$

where $G(J)$ is the decomposable graph determined by J and $\mu(G)$ is the number of equivalent junction trees representing G.

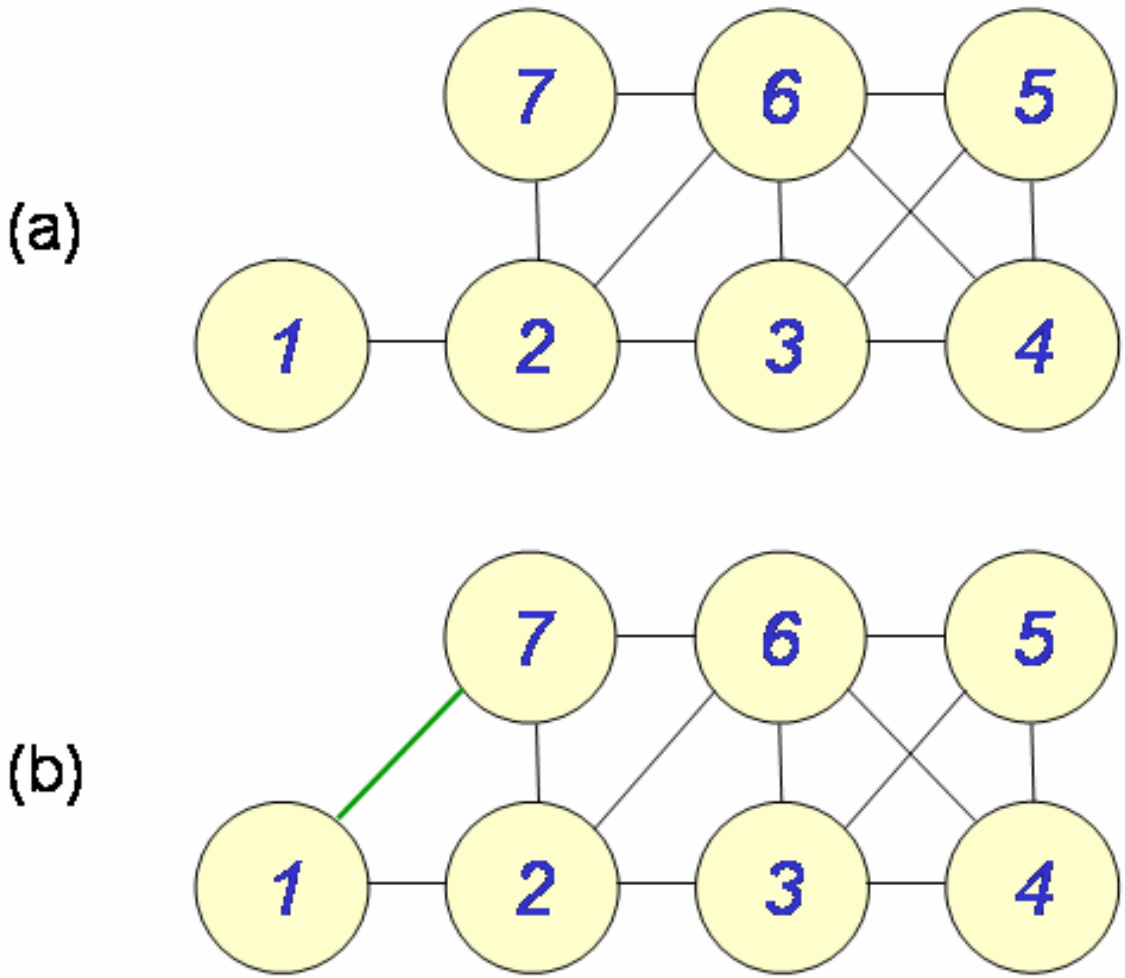
Using the junction tree as the state

Trade-off between

- faster, more restrictive choice of proposed vertex pairs (x, y) specifying edges to be added, and avoidance of the manipulation from one junction tree to another, and
- the space of possible (junction tree) states of the chain being less connected

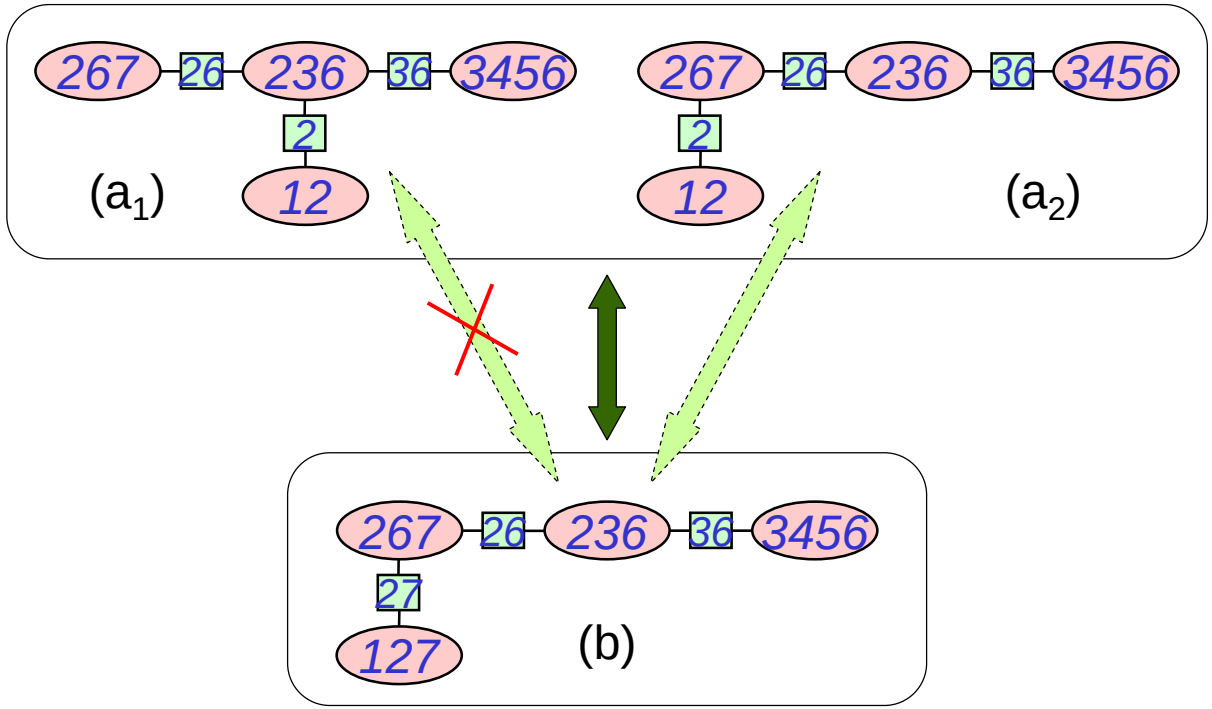
Using the junction tree as the state

Two decomposable graphs differing in only one edge.



Using the junction tree as the state

Whether they are adjacent in the junction tree representation depends on the choice of junction tree.



Using the junction tree as the state

Paraphrasing the conditions for maintaining decomposability:

(C) Connecting x and y by adding an edge (x, y) to G will result in a decomposable graph if and only if x and y are contained in cliques that are adjacent in some junction tree of G.

(D) Disconnecting x and y by removing an edge (x, y) from G will result in a decomposable graph if and only if x and y are contained in exactly one clique.

Our new approach means that we only have to look at the current junction tree in (C).

Using the junction tree as the state

Paraphrasing the conditions for maintaining decomposability:

(C) Connecting x and y by adding an edge (x, y) to G will result in a decomposable graph if and only if x and y are contained in cliques that are adjacent in some junction tree of G.

(D) Disconnecting x and y by removing an edge (x, y) from G will result in a decomposable graph if and only if x and y are contained in exactly one clique.

Our new approach means that we only have to look at the current junction tree in (C).

Multiple-edge perturbations

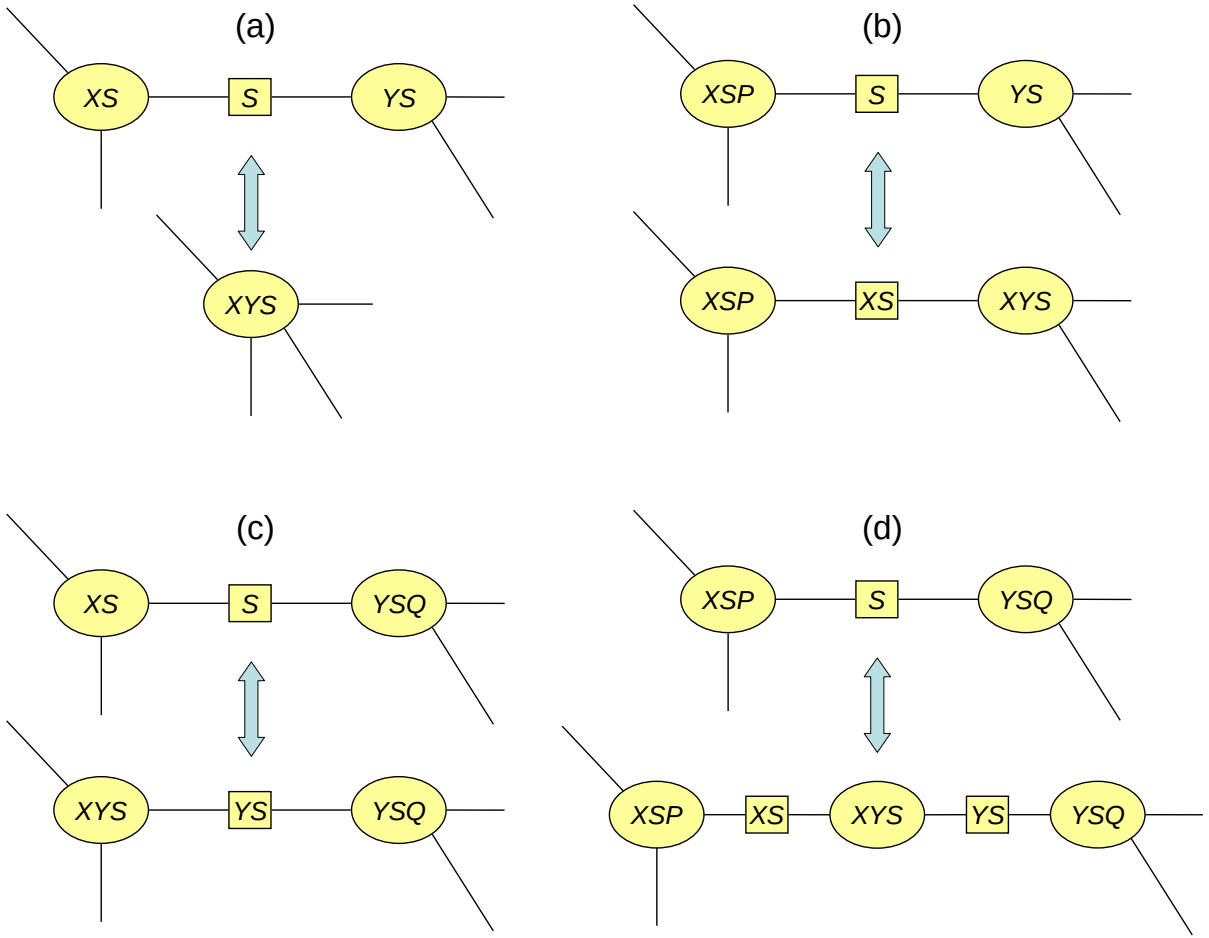
We can make bigger perturbations, without losing ability to pre-test for maintaining decomposability and make local updates.

We say two disjoint non-empty connected sets of vertices X and Y are completely connected if every vertex in X is connected to every vertex in Y. They are completely disconnected if no vertices in X are connected to any vertices in Y.

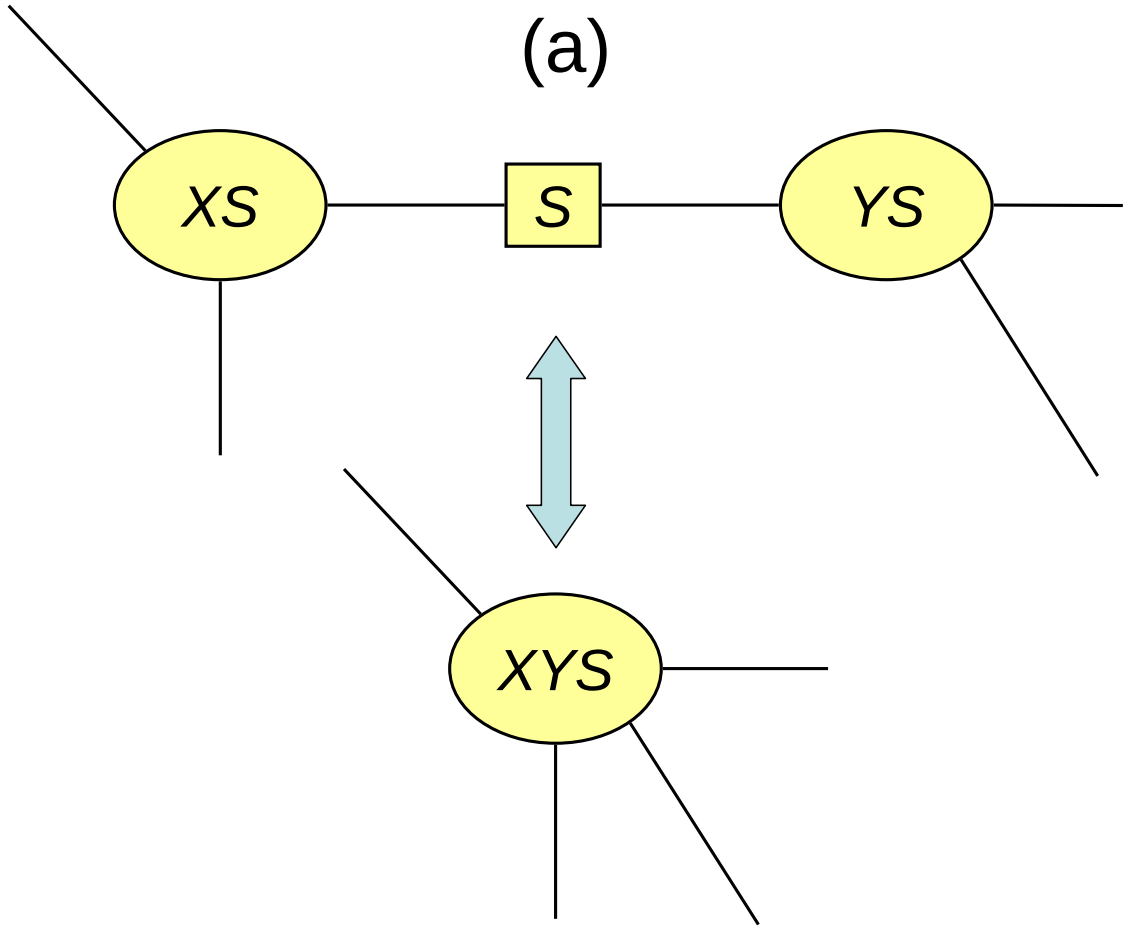
(C) If X and Y are completely disconnected and subsets of cliques that are adjacent in some junction tree, then X and Y can be completely connected, resulting in a new decomposable graph

(D) If X and Y are completely connected, and subsets of exactly one clique, and some other stuff too complicated to fit in here but all checkable locally, then X and Y can be completely disconnected, resulting in a new decomposable graph

Manipulations to junction tree on connecting or disconnecting X and Y.



Manipulations to junction tree on connecting or disconnecting X and Y.



Enumerating junction trees

To use this, we need to know the number of equivalent junction trees for the graph G. We do! It is

$$\mu(G) = \prod_{i=1}^{s} \nu(S_{[i]})$$

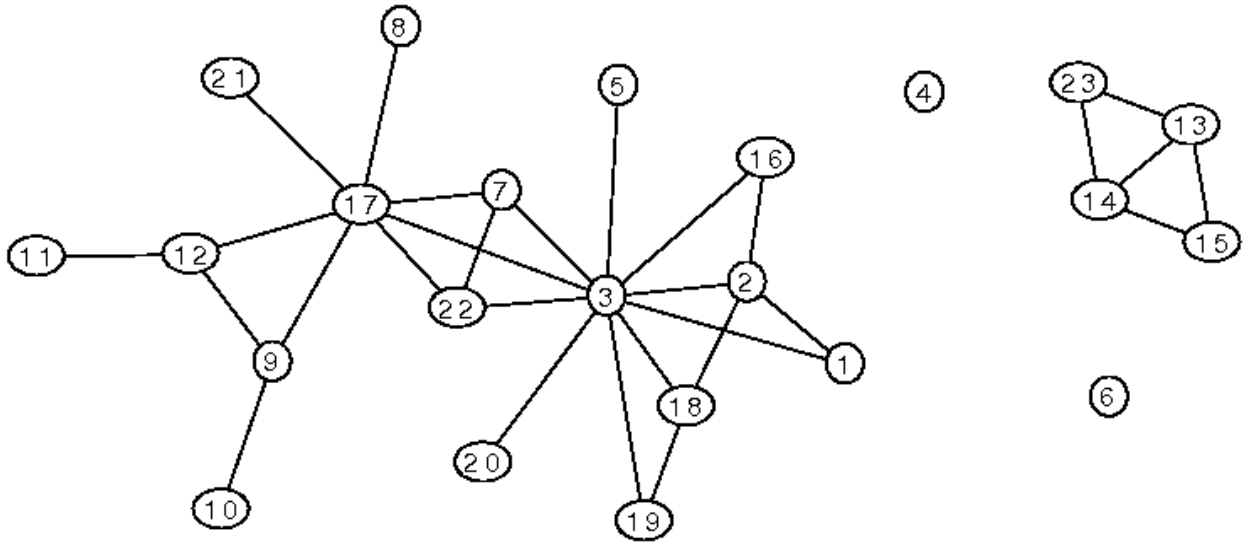
where $S_{[i]}, i = 1, 2, \ldots, s$ are the distinct separators, and

$$\nu(S) = t_S^{m_S - 1} \prod_{j=1}^{m_S+1} f_j.$$

Here t_S is the number of nodes in T_S, the subtree of J induced by the cliques containing S, m_S is the multiplicity of separator S, and $f_j, j = 1, 2, \ldots, m_S + 1$ are the sizes of the components of the forest F_S obtained from T_S by deleting links associated with S.

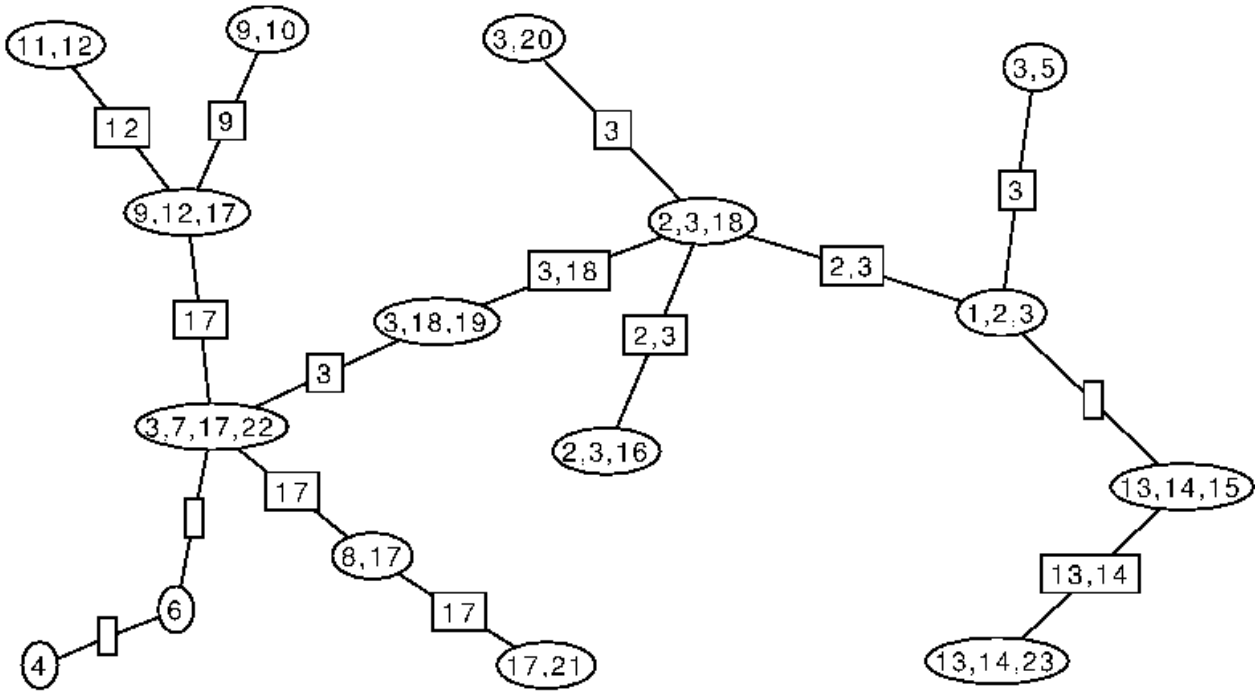
Example of enumerating junction trees

A decomposable graph G containing 23 vertices in 4 disjoint components.



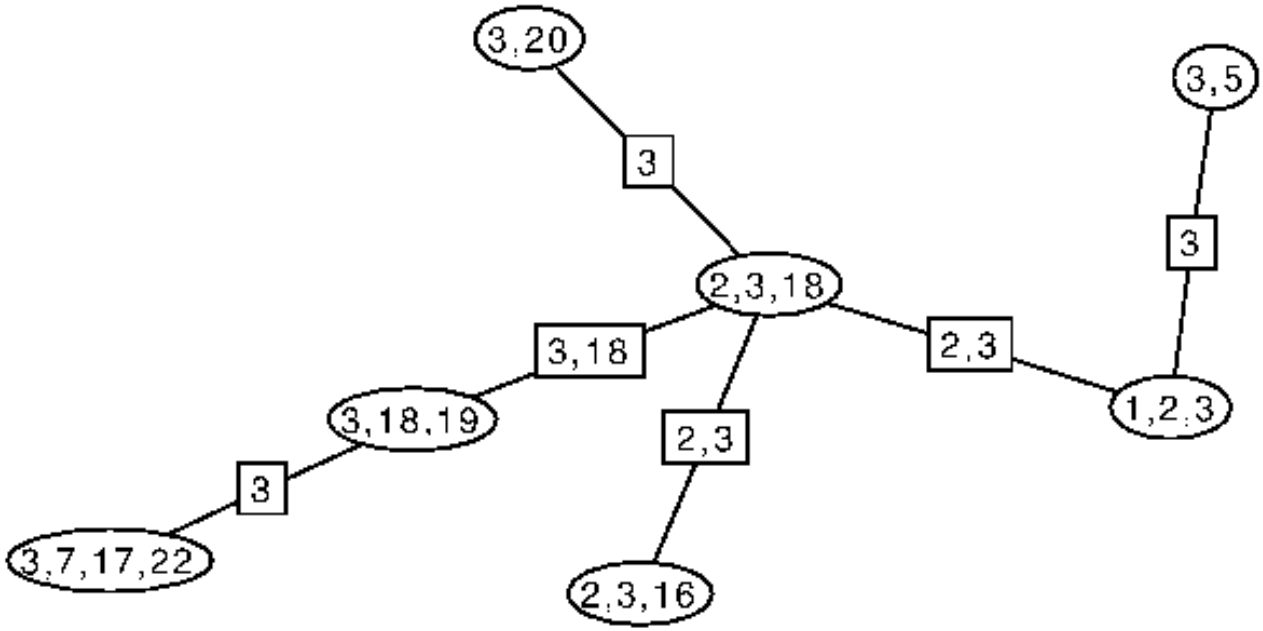
Example of enumerating junction trees

One possible junction tree J for the graph shown before.



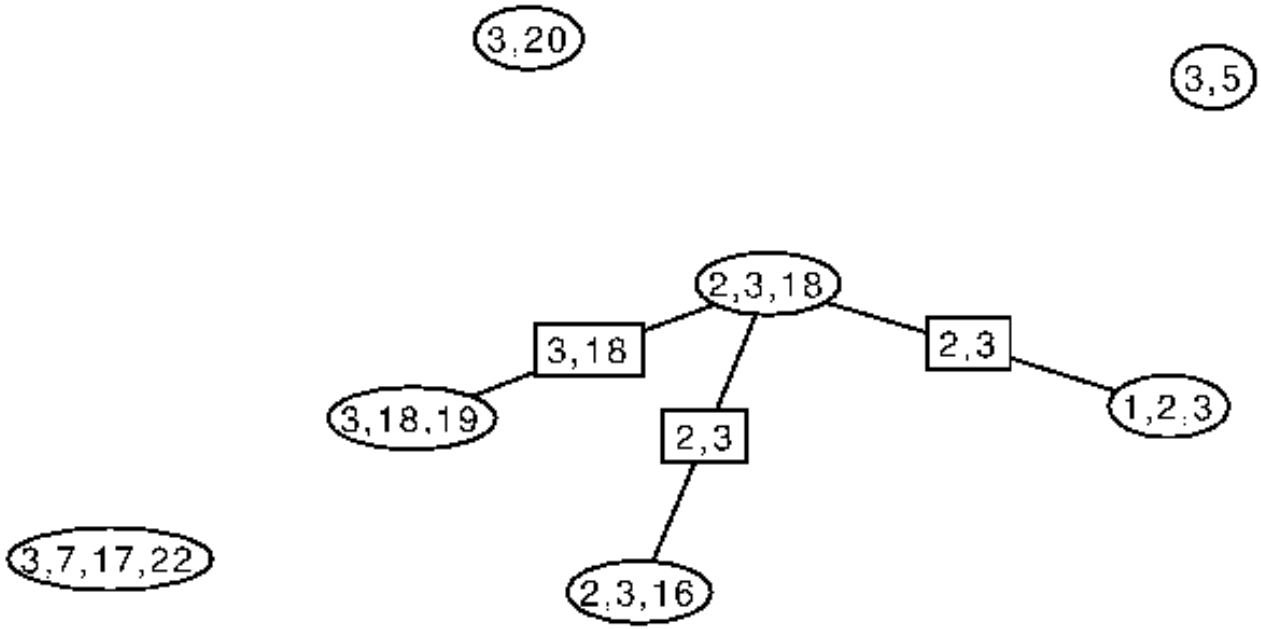
Example of enumerating junction trees

$T_{\{3\}}$, the connected subtree of the junction graph J induced by the cliques that contain the separator $\{3\}$.



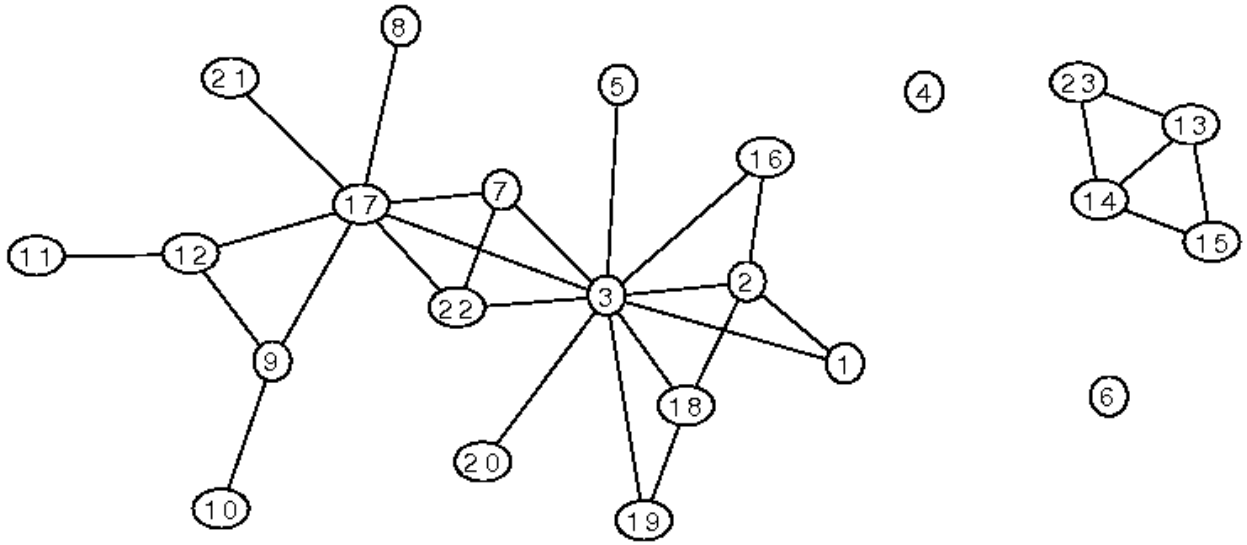
Example of enumerating junction trees

$F_{\{3\}}$, the forest obtained by from the tree $T_{\{3\}}$ by deleting edges associated with the separator $\{3\}$.



Example of enumerating junction trees

The decomposable graph G can be represented by 57,802,752 different junction trees!



Demo

All decomposable graphs on 7 vertices

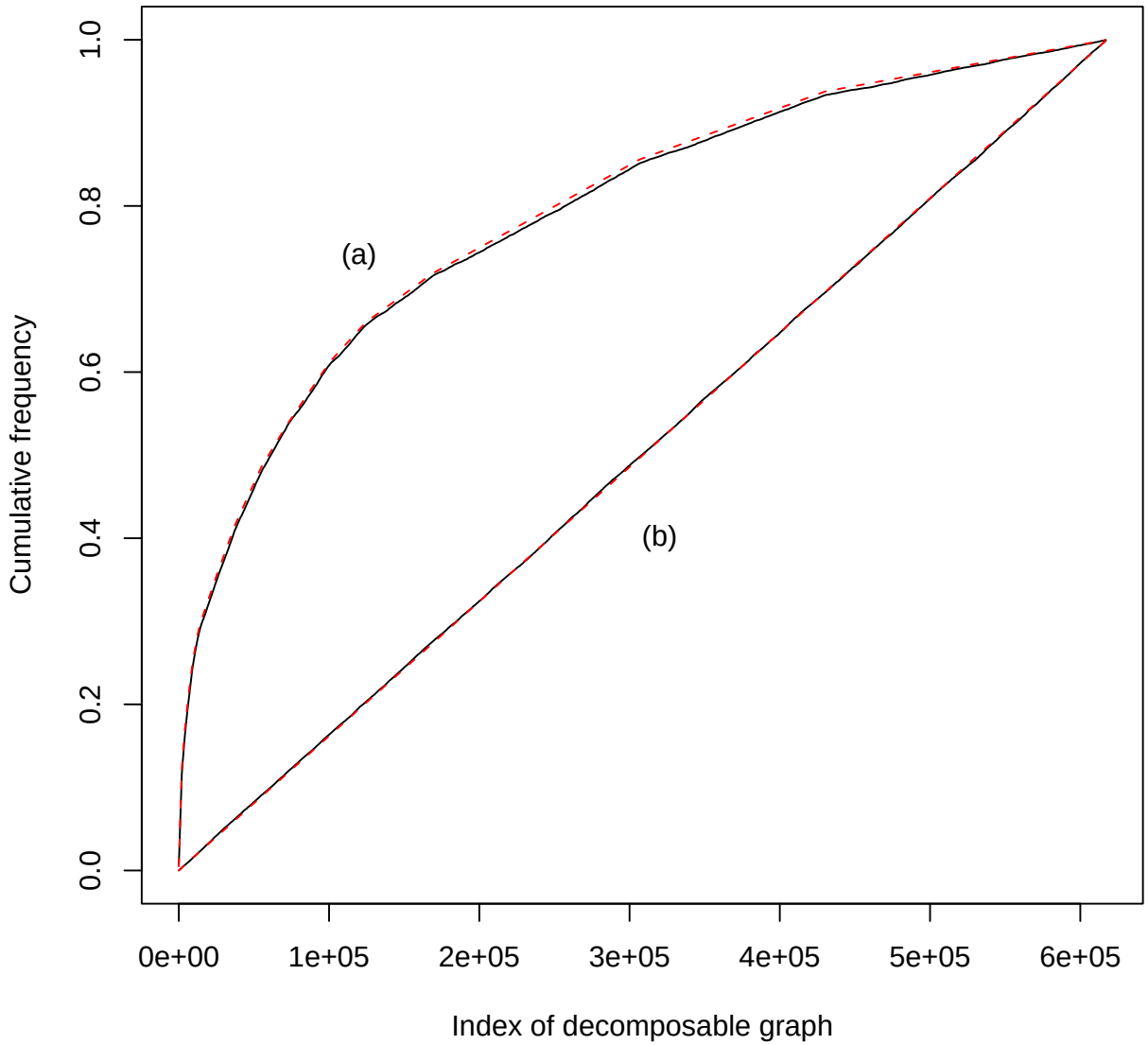
We iterated through all 2,097,152 undirected graphs on 7 labelled vertices and identified the 617,675 decomposable ones. A list of the cliques of each decomposable graph was found and used as an index into a table of counters.

The decomposable graphs were sorted from those with most representations (16,807 for the trivial graph) to least (187,447 have a single junction tree).

To test the uniformity of sampling with the new sampler, we used it to sample both uniformly on decomposable graphs, and uniformly on junction trees: 1,000,000 graphs sampled in each case.

All decomposable graphs on 7 vertices

Comparing theoretical and empirical distributions over graphs, when sampling uniformly (a) on trees, (b) on graphs



A graphical Gaussian intra-class model

Given a decomposable graph G on v vertices labelled $1, 2, \ldots, v$, and real scalar parameters $\sigma^2 > 0$ and ρ, we define a non-negative definite matrix $V = V_G(\sigma^2, \rho)$ by

$$V_{ij} = \begin{cases} \sigma^2 & \text{if } i = j \\ \rho\sigma^2 & \text{if } (i,j) \text{ is an edge in } G, \end{cases}$$

and $(V^{-1})_{ij} = 0$ if (i,j) is not an edge in G.
By Grone et al (1984), since G is decomposable and V restricted to each clique is positive definite, V exists and is unique, in fact the unique completion of the specified entries that is positive definite; it is the variance matrix of a v–variate Gaussian distribution for which G is the conditional independence graph. We call this the graphical Gaussian intra-class model (GGIM).

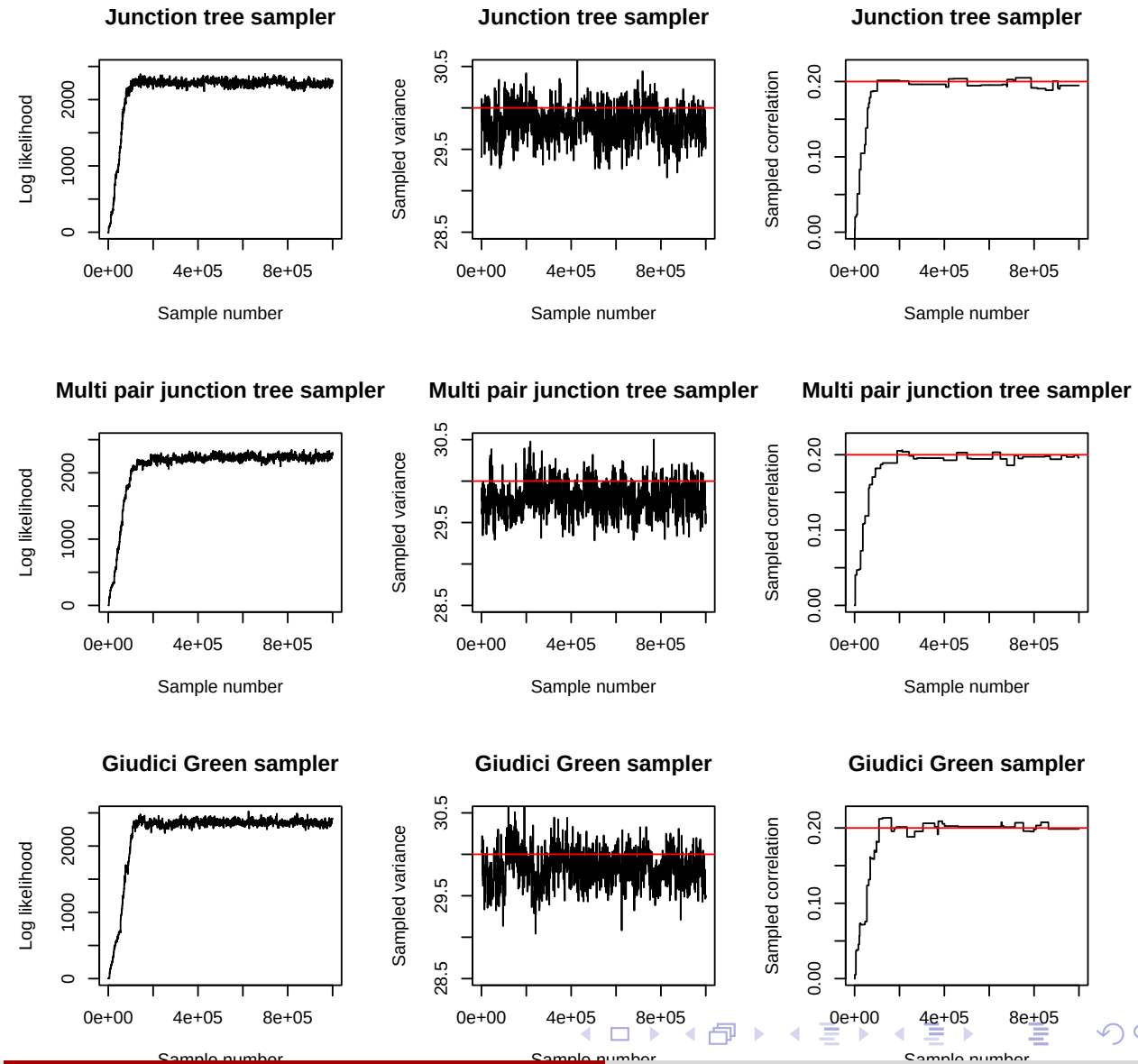
A 50-vertex graphical Gaussian intra-class model

We simulated 1000 GGIM observations on 50 variables with $\sigma^2 = 30$ and $\rho = 0.2$. We used a second order Markov Chain graphical structure, that is, $(V^{-1})_{ij} = 0$ for all i and j such that $|i - j| > 2$.

In each case we started from the initial conditions of $\sigma^2 = 1$, $\rho = 0$ and G set to have no edges indicating complete independence between the 50 variables. We made 1,000,000 Metropolis–Hastings updates with each sampler and output values indicating the state of the chain after ever 100 iterations. The parameters σ^2 and ρ were updated after each 1,000 Metropolis–Hastings steps. For the junction tree samplers we also randomized the junction tree after every 1,000 Metropolis–Hastings steps.

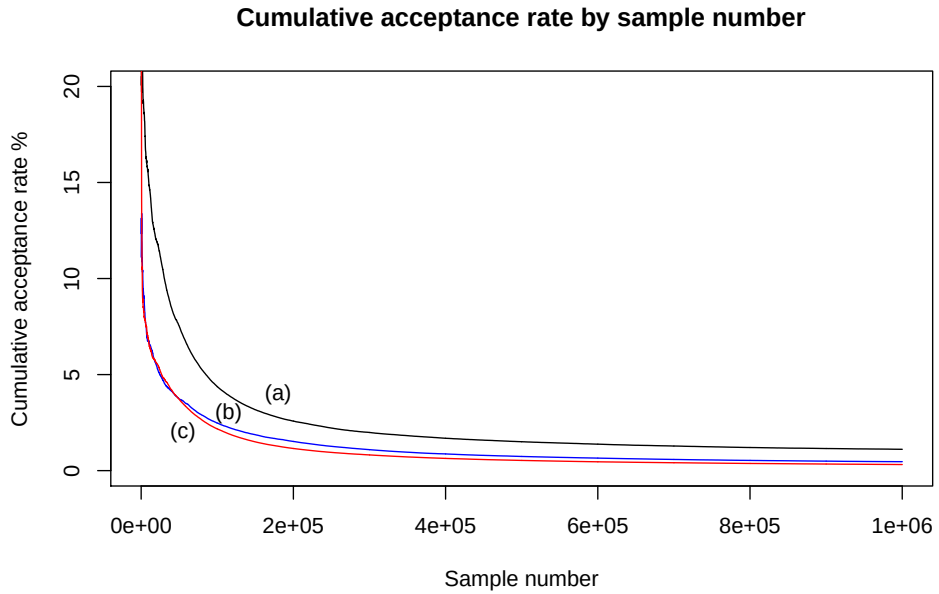
A 50-vertex graphical Gaussian intra-class model

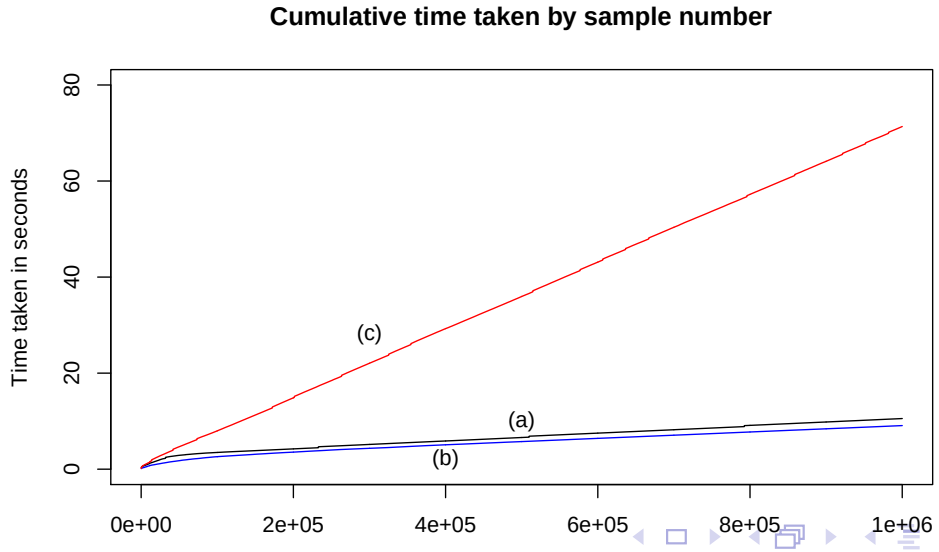
Log likelihoods and parameter estimates for three samplers for the GGIM model, plotted by sample number. The values of the parameters used to generate the data are shown by the red horizontal lines.



A 50-vertex graphical Gaussian intra-class model

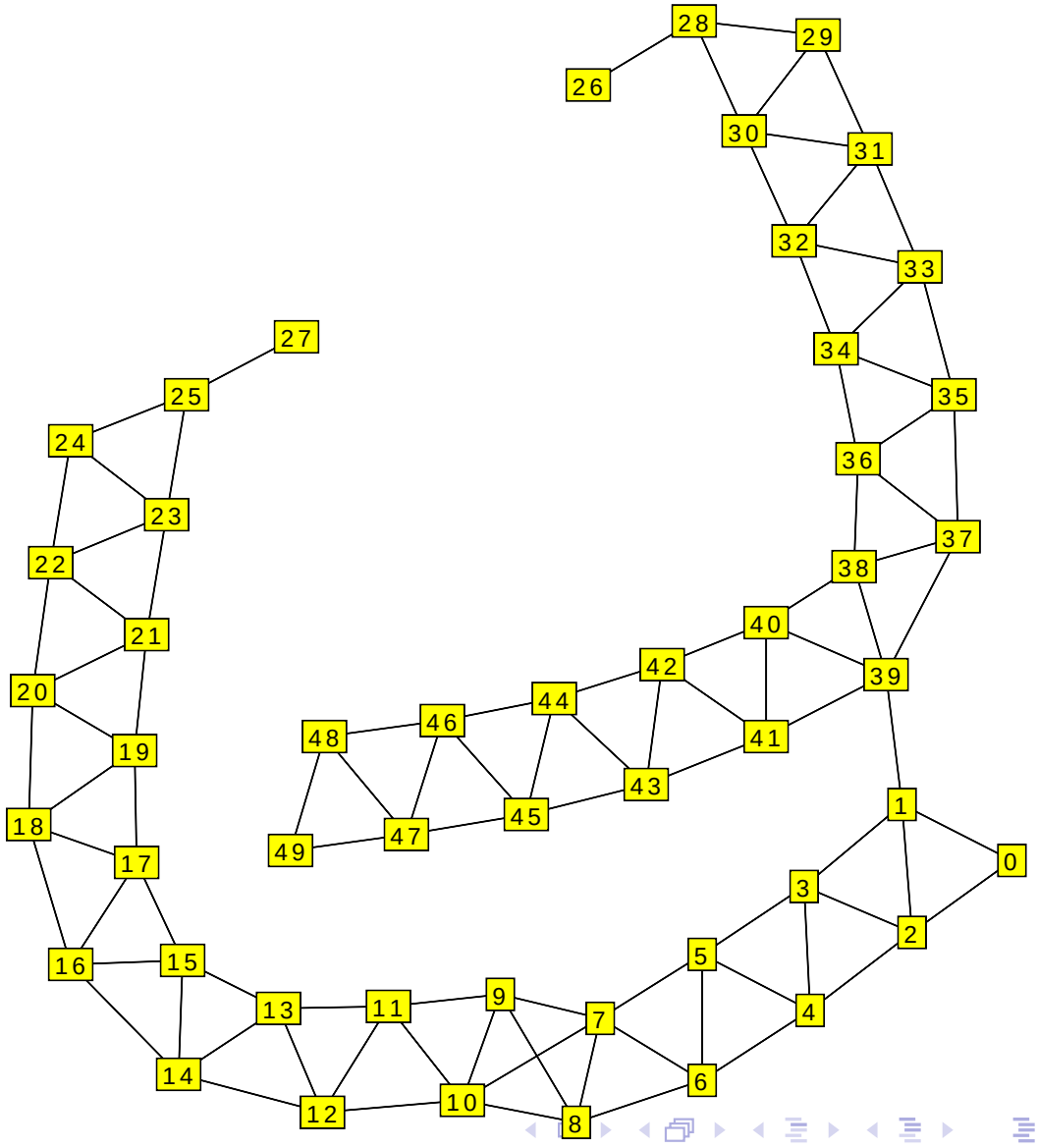
Cumulative acceptance rates and times taken by the three samplers for the GGIM model. In each case the curve (a) is the single edge junction tree sampler, (b) is the multi edge junction tree sampler, and (c) is the Giudici–Green sampler.



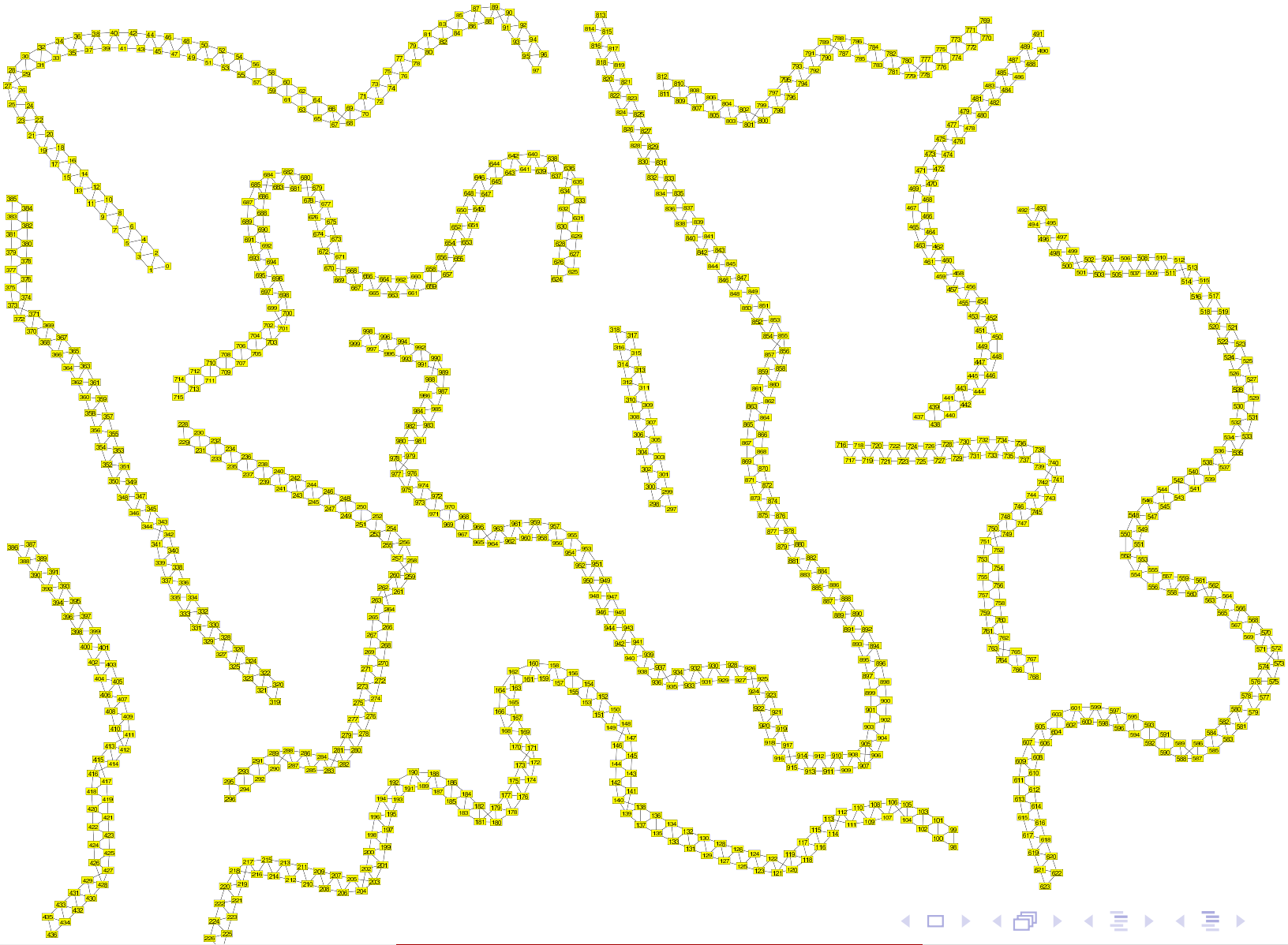


A 50-vertex graphical Gaussian intra-class model

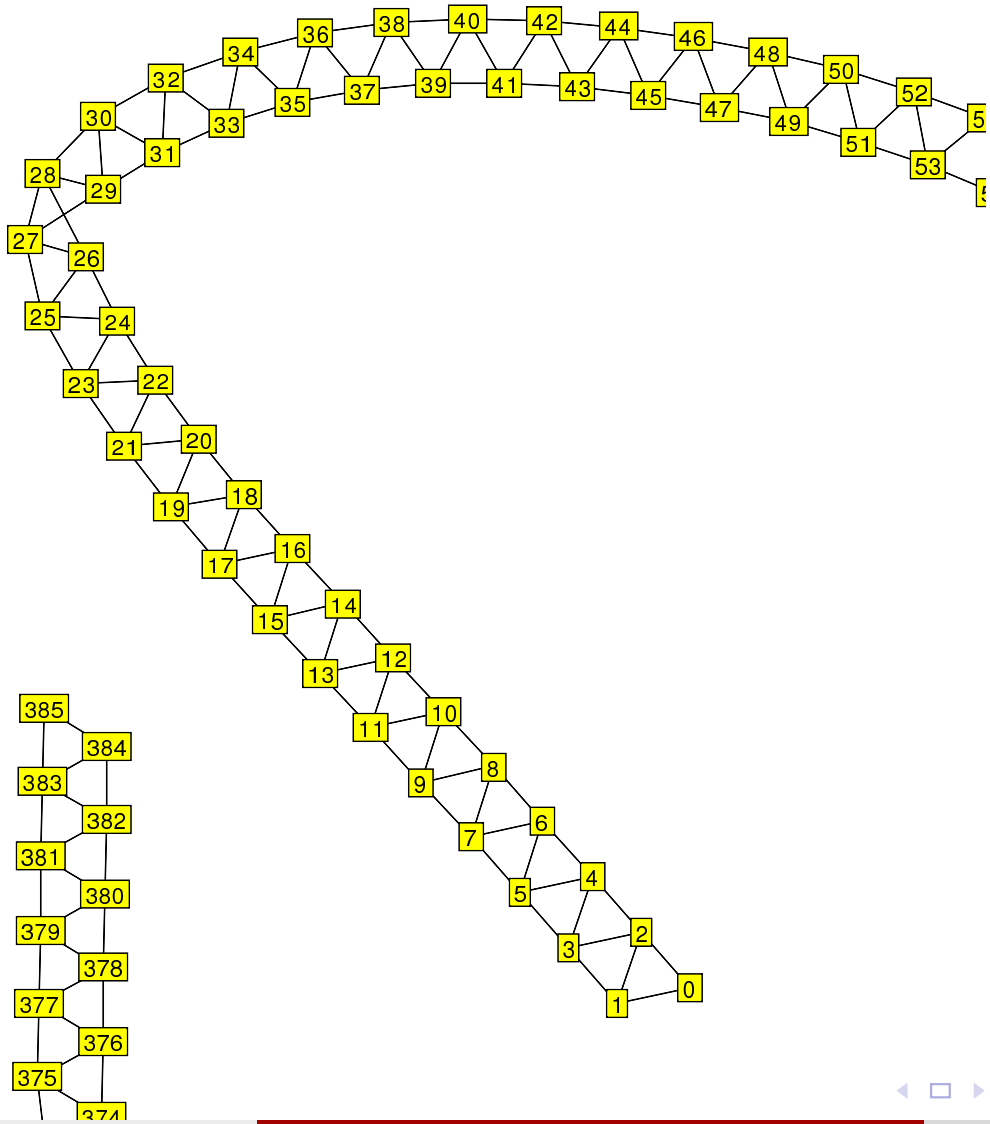
A graph typical of the type sampled early in their runs by all three samplers for the GGIM model. The edge between variables 1 and 39 is spurious, and has to be removed before the correct edges near variables 25 and 26 can be added.



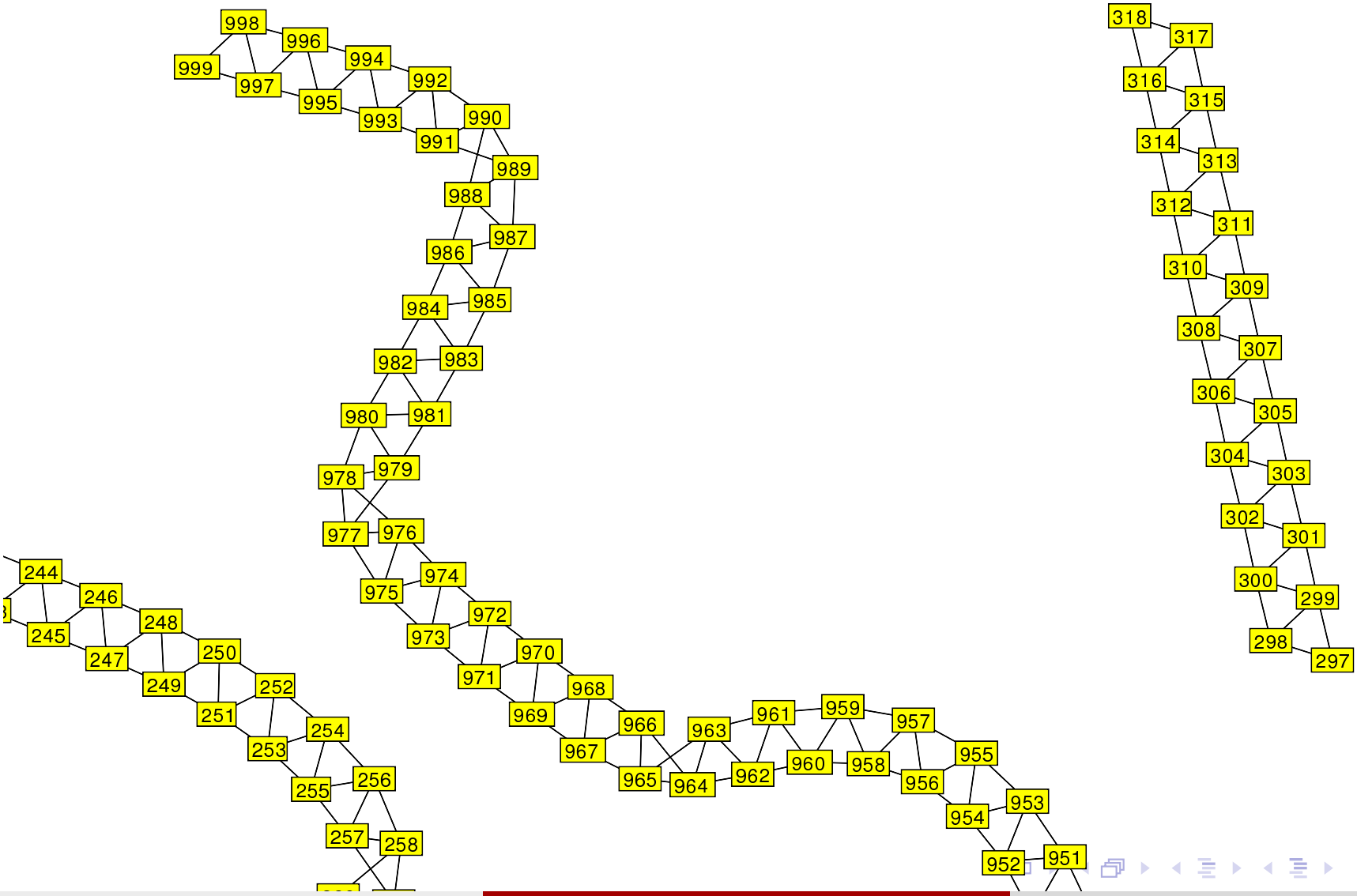
A 1000-vertex graphical Gaussian intra-class model



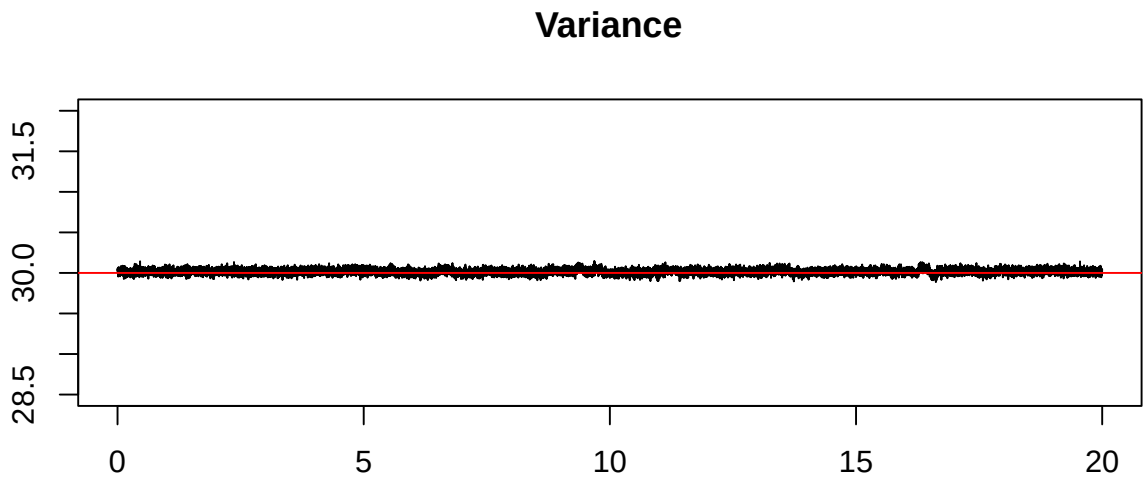
A 1000-vertex graphical Gaussian intra-class model

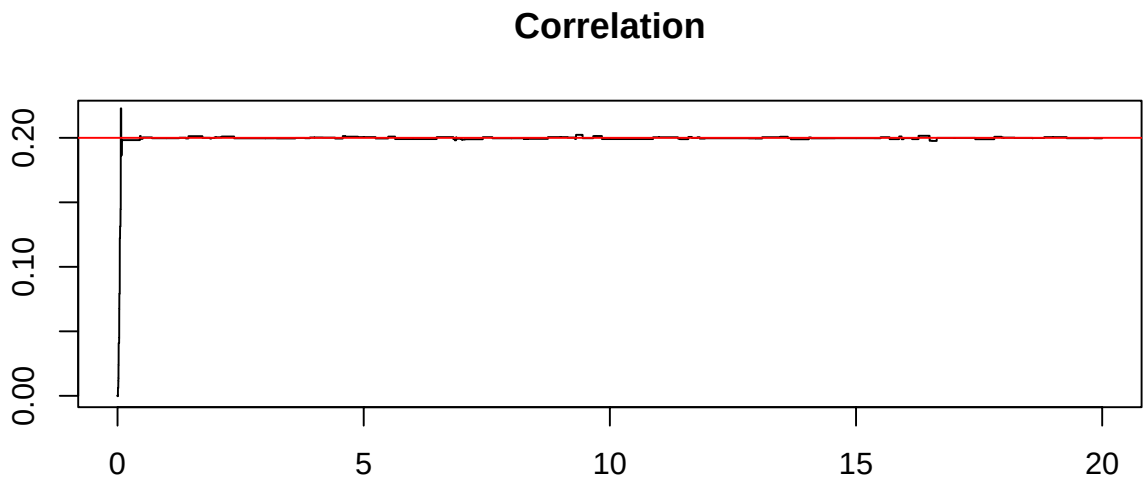


A 1000-vertex graphical Gaussian intra-class model



A 1000-vertex graphical Gaussian intra-class model





- “Enumerating the decomposable neighbours of a decomposable graph under a simple perturbation scheme”, *CSDA*, 2009, by Thomas and Green
- “Enumerating the junction trees of a decomposable graph”, *JCGS*, 2009, by Thomas and Green
- “Sampling decomposable graphs using a Markov chain on junction trees”, *submitted*, 2011, by Green and Thomas
- Webpage: www.stats.bris.ac.uk/∼peter/
- Email: P.J.Green@bristol.ac.uk