Sampling from a very large set defined by a complex system of linear constraints

Martin Hazelton
m.hazelton@massey.ac.nz
Massey University, New Zealand

Presented at the Second Wellington Workshop on Probability and Mathematical Statistics, 30 November – 1 December 2010

Setting the Scene

- Given network $\mathcal{G} = (\mathcal{N}, \mathcal{A})$:
  - $\mathcal{N}$ set of nodes.
  - $\mathcal{A}$ set of directed arcs.

Setting the Scene

- Given network $\mathcal{G} = (\mathcal{N}, \mathcal{A})$:
  - $\mathcal{N}$ set of nodes.
  - $\mathcal{A}$ set of directed arcs.

- $\mathcal{R}_{ij}$ is set of paths (routes) from $i \in \mathcal{N}$ to $j \in \mathcal{N}$.
  - $\mathcal{R} = \bigcup_{i,j} \mathcal{R}_{ij}$ set of all paths.

Setting the Scene

- Given network $\mathcal{G} = (\mathcal{N}, \mathcal{A})$:
  - $\mathcal{N}$ set of nodes.
  - $\mathcal{A}$ set of directed arcs.
- $\mathcal{R}_{ij}$ is set of paths (routes) from $i \in \mathcal{N}$ to $j \in \mathcal{N}$.
  - $\mathcal{R} = \bigcup_{i,j} \mathcal{R}_{ij}$ set of all paths.
- $\{Y_j : j \in \mathcal{R}\}$ set of (path specific) independent Poisson r.v.s
  - $\mathbb{E}[Y_j] = \lambda_j$ assumed known $\forall j \in \mathcal{R}$.

Setting the Scene

- Given network $\mathcal{G} = (\mathcal{N}, \mathcal{A})$:
  - $\mathcal{N}$ set of nodes.
  - $\mathcal{A}$ set of directed arcs.
- $\mathcal{R}_{ij}$ is set of paths (routes) from $i \in \mathcal{N}$ to $j \in \mathcal{N}$.
  - $\mathcal{R} = \bigcup_{i,j} \mathcal{R}_{ij}$ set of all paths.
- $\{Y_j : j \in \mathcal{R}\}$ set of (path specific) independent Poisson r.v.s
  - $\mathbb{E}[Y_j] = \lambda_j$ assumed known $\forall j \in \mathcal{R}$.
- $X_i = \sum_{j \in \mathcal{R}} a_{ij} Y_j$ implied arc r.v.s

$$a_{ij} = \begin{cases} 1 & \text{if arc } i \text{ part of path } j \\ 0 & \text{otherwise.} \end{cases}$$

Extra Notation

- $\boldsymbol{Y} = (Y_1, \ldots, Y_r)^\mathsf{T}$.
  - r is number of paths.
- $\boldsymbol{X} = (X_1, \ldots, X_c)^\mathsf{T}$.
  - c is number of arcs.
- $A = [a_{ij}]$ is routing matrix.
- Fundamental routing equation

$$\boldsymbol{X} = A\boldsymbol{Y}$$

The Problem(s)

Main Problem

Sample $Y|X = x$.

The Problem(s)

Main Problem

Sample $\boldsymbol{Y}|\boldsymbol{X} = \boldsymbol{x}$.

Variants

1. Relax Poisson assumption: $\{Y_j : j \in \mathcal{R}\}$ general r.v.s on $\mathbb{Z}^+$.

The Problem(s)

Main Problem

Sample $\boldsymbol{Y} | \boldsymbol{X} = \boldsymbol{x}$.

Variants

1. Relax Poisson assumption: $\{Y_j : j \in \mathcal{R}\}$ general r.v.s on $\mathbb{Z}^+$.
   - Straightforward in principle by acceptance-rejection sampling.

The Problem(s)

Main Problem

Sample $\boldsymbol{Y}|\boldsymbol{X} = \boldsymbol{x}$.

Variants

1. Relax Poisson assumption: $\{Y_j : j \in \mathcal{R}\}$ general r.v.s on $\mathbb{Z}^+$.
   - Straightforward in principle by acceptance-rejection sampling.
2. Condition only on subset of arc counts: $\{X_i : i \in \mathcal{A}_0\}$ for $\mathcal{A}_0 \subseteq \mathcal{A}$.

The Problem(s)

Main Problem

Sample $\boldsymbol{Y}|\boldsymbol{X} = \boldsymbol{x}$.

Variants

1. Relax Poisson assumption: $\{Y_j : j \in \mathcal{R}\}$ general r.v.s on $\mathbb{Z}^+$.
   - Straightforward in principle by acceptance-rejection sampling.
2. Condition only on subset of arc counts: $\{X_i : i \in \mathcal{A}_0\}$ for $\mathcal{A}_0 \subseteq \mathcal{A}$.
3. Constraint $\mathcal{G}$: e.g.
   - $\mathcal{G}$ a linear network;
   - $\mathcal{G}$ a directed acyclic graph.

The Problem(s)

Main Problem

Sample $\boldsymbol{Y}|\boldsymbol{X} = \boldsymbol{x}$.

Variants

1. Relax Poisson assumption: $\{Y_j : j \in \mathcal{R}\}$ general r.v.s on $\mathbb{Z}^+$.
   - Straightforward in principle by acceptance-rejection sampling.
2. Condition only on subset of arc counts: $\{X_i : i \in \mathcal{A}_0\}$ for $\mathcal{A}_0 \subseteq \mathcal{A}$.
3. Constraint $\mathcal{G}$: e.g.
   - $\mathcal{G}$ a linear network;
   - $\mathcal{G}$ a directed acyclic graph.

The Problem(s)

Main Problem

Sample $\boldsymbol{Y}|\boldsymbol{X} = \boldsymbol{x}$.

Variants

1. Relax Poisson assumption: $\{Y_j : j \in \mathcal{R}\}$ general r.v.s on $\mathbb{Z}^+$.
   - Straightforward in principle by acceptance-rejection sampling.
2. Condition only on subset of arc counts: $\{X_i : i \in \mathcal{A}_0\}$ for $\mathcal{A}_0 \subseteq \mathcal{A}$.
3. Constraint $\mathcal{G}$: e.g.
   - $\mathcal{G}$ a linear network;
   - $\mathcal{G}$ a directed acyclic graph.
4. Sampling a contingency table with fixed margins is special case.

The Problem(s)

Main Problem

Sample $\boldsymbol{Y}|\boldsymbol{X} = \boldsymbol{x}$.

Variants

1. Relax Poisson assumption: $\{Y_j : j \in \mathcal{R}\}$ general r.v.s on $\mathbb{Z}^+$.
   - Straightforward in principle by acceptance-rejection sampling.
2. Condition only on subset of arc counts: $\{X_i : i \in \mathcal{A}_0\}$ for $\mathcal{A}_0 \subseteq \mathcal{A}$.
3. Constraint $\mathcal{G}$: e.g.
   - $\mathcal{G}$ a linear network;
   - $\mathcal{G}$ a directed acyclic graph.
4. Sampling a contingency table with fixed margins is special case.

The Solution in Theory

$$\mathbb{P}(\boldsymbol{Y} = \boldsymbol{y} | \boldsymbol{X} = \boldsymbol{x}) = p(\boldsymbol{y}|\boldsymbol{x}) = \frac{p(\boldsymbol{y})p(\boldsymbol{x}|\boldsymbol{y})}{p(\boldsymbol{x})}$$

The Solution in Theory

$$\mathbb{P}(\boldsymbol{Y} = \boldsymbol{y} | \boldsymbol{X} = \boldsymbol{x}) = p(\boldsymbol{y}|\boldsymbol{x}) = \frac{p(\boldsymbol{y})p(\boldsymbol{x}|\boldsymbol{y})}{p(\boldsymbol{x})}$$

$$p(\boldsymbol{y}) = \frac{e^{-\sum_j \lambda_j} \prod_j \lambda_j^{y_j}}{\prod_j y_j!} \quad \text{(no problem)}$$

The Solution in Theory

$$\mathbb{P}(\boldsymbol{Y} = \boldsymbol{y} | \boldsymbol{X} = \boldsymbol{x}) = p(\boldsymbol{y}|\boldsymbol{x}) = \frac{p(\boldsymbol{y})p(\boldsymbol{x}|\boldsymbol{y})}{p(\boldsymbol{x})}$$

$$p(\boldsymbol{y}) = \frac{e^{-\sum_j \lambda_j} \prod_j \lambda_j^{y_j}}{\prod_j y_j!} \quad \text{(no problem)}$$

$$p(\boldsymbol{x}|\boldsymbol{y}) = \mathbf{1}_{\{\boldsymbol{x}=A\boldsymbol{y}\}} \quad \text{(big problem)}$$

The Solution in Theory

$$\mathbb{P}(\boldsymbol{Y} = \boldsymbol{y} | \boldsymbol{X} = \boldsymbol{x}) = p(\boldsymbol{y}|\boldsymbol{x}) = \frac{p(\boldsymbol{y})p(\boldsymbol{x}|\boldsymbol{y})}{p(\boldsymbol{x})}$$

$$p(\boldsymbol{y}) = \frac{e^{-\sum_j \lambda_j} \prod_j \lambda_j^{y_j}}{\prod_j y_j!} \quad \text{(no problem)}$$

$$p(\boldsymbol{x}|\boldsymbol{y}) = \mathbf{1}_{\{\boldsymbol{x}=A\boldsymbol{y}\}} \quad \text{(big problem)}$$

$$p(\boldsymbol{x}) = \sum_{\boldsymbol{y}} p(\boldsymbol{y})p(\boldsymbol{x}|\boldsymbol{y}) \quad \text{(another big problem)}$$

Feasible $\boldsymbol{y}$

- Support of $p(\boldsymbol{y}|\boldsymbol{x})$ is 'feasible set' $\mathcal{Y}(\boldsymbol{x}) = \{\boldsymbol{y} : \boldsymbol{x} = A\boldsymbol{y}\}$.
- Normalizing constant $p(\boldsymbol{x})$ requires sum over all elements of $\mathcal{Y}(\boldsymbol{x})$.
- $\mathcal{Y}(\boldsymbol{x})$ too large to enumerate for even moderately large networks.

Feasible $\boldsymbol{y}$

- Support of $p(\boldsymbol{y}|\boldsymbol{x})$ is 'feasible set' $\mathcal{Y}(\boldsymbol{x}) = \{\boldsymbol{y} : \boldsymbol{x} = A\boldsymbol{y}\}$.
- Normalizing constant $p(\boldsymbol{x})$ requires sum over all elements of $\mathcal{Y}(\boldsymbol{x})$.
- $\mathcal{Y}(\boldsymbol{x})$ too large to enumerate for even moderately large networks.

MCMC

- Since normalizing constant unavailable, use MCMC.
- Suppose $\boldsymbol{y}$ current sample from $p(\boldsymbol{y}|\boldsymbol{x})$.
- Sample candidate $\boldsymbol{y}^\dagger$ from proposal distribution q.
- Accept with probability

$$\begin{aligned}\alpha &= \min\left\{1, \frac{p(\boldsymbol{y}^\dagger|\boldsymbol{x})q(\boldsymbol{y})}{p(\boldsymbol{y}|\boldsymbol{x})q(\boldsymbol{y}^\dagger)}\right\} \\ &= \min\left\{1, \frac{p(\boldsymbol{y}^\dagger)p(\boldsymbol{x}|\boldsymbol{y}^\dagger)q(\boldsymbol{y})}{p(\boldsymbol{y})p(\boldsymbol{x}|\boldsymbol{y})q(\boldsymbol{y}^\dagger)}\right\} \\ &= \min\left\{1, \frac{p(\boldsymbol{y}^\dagger)q(\boldsymbol{y})\mathbf{1}_{\{\boldsymbol{x}=A\boldsymbol{y}^\dagger\}}}{p(\boldsymbol{y})q(\boldsymbol{y}^\dagger)}\right\}\end{aligned}$$

Sampling Candidates

- MCMC obviates need to evaluate normalizing constant.
- Still faced with problem of sampling candidates.
- Only candidates in $\mathcal{Y}(\boldsymbol{x}) = \{\boldsymbol{y} : \boldsymbol{x} = A\boldsymbol{y}\}$ may be accepted.
- This set imposes complex set of constraints.
- Difficult to characterise elements of $\mathcal{Y}(\boldsymbol{x})$ in a convenient way for sampling.

A General Method

- Only one suggestion to the general problem in literature to date.
- Tebaldi and West (1998).
- Basic plan:
  - Sample $\boldsymbol{Y}$ element by element.
  - Check for feasibility at each step.

Algorithm

1. Reorder columns of A then partition it as $A = [A_1, A_2]$ where A_1 non-singular (square) matrix.

Algorithm

1. Reorder columns of A then partition it as $A = [A_1, A_2]$ where A_1 non-singular (square) matrix.
2. Partition $\boldsymbol{Y}^\top = (\boldsymbol{Y}_{[1]}^\top, \boldsymbol{Y}_{[2]}^\top)$ correspondingly.

Algorithm

1. Reorder columns of A then partition it as $A = [A_1, A_2]$ where A_1 non-singular (square) matrix.
2. Partition $\boldsymbol{Y}^\mathsf{T} = (\boldsymbol{Y}_{[1]}^\mathsf{T}, \boldsymbol{Y}_{[2]}^\mathsf{T})$ correspondingly.
3. Need sample only $\boldsymbol{Y}_{[2]}$ because

$$\boldsymbol{Y}_{[1]} = A_1^{-1}(\boldsymbol{x} - A_2\boldsymbol{Y}_{[2]}).$$

Algorithm

1. Reorder columns of A then partition it as $A = [A_1, A_2]$ where A_1 non-singular (square) matrix.
2. Partition $\boldsymbol{Y}^\mathsf{T} = (\boldsymbol{Y}_{[1]}^\mathsf{T}, \boldsymbol{Y}_{[2]}^\mathsf{T})$ correspondingly.
3. Need sample only $\boldsymbol{Y}_{[2]}$ because

$$\boldsymbol{Y}_{[1]} = A_1^{-1}(\boldsymbol{x} - A_2\boldsymbol{Y}_{[2]}).$$

4. Any $\boldsymbol{Y}_{[2]}$ producing non-negative $\boldsymbol{Y}_{[1]}$ is feasible.

Algorithm

1. Reorder columns of A then partition it as $A = [A_1, A_2]$ where A_1 non-singular (square) matrix.
2. Partition $\boldsymbol{Y}^\mathsf{T} = (\boldsymbol{Y}_{[1]}^\mathsf{T}, \boldsymbol{Y}_{[2]}^\mathsf{T})$ correspondingly.
3. Need sample only $\boldsymbol{Y}_{[2]}$ because

$$\boldsymbol{Y}_{[1]} = A_1^{-1}(\boldsymbol{x} - A_2\boldsymbol{Y}_{[2]}).$$

4. Any $\boldsymbol{Y}_{[2]}$ producing non-negative $\boldsymbol{Y}_{[1]}$ is feasible.
5. Sequantially sample each element of $\boldsymbol{Y}_{[2]}$ conditional on others.

Algorithm

1. Reorder columns of A then partition it as $A = [A_1, A_2]$ where A_1 non-singular (square) matrix.
2. Partition $\boldsymbol{Y}^\mathsf{T} = (\boldsymbol{Y}_{[1]}^\mathsf{T}, \boldsymbol{Y}_{[2]}^\mathsf{T})$ correspondingly.
3. Need sample only $\boldsymbol{Y}_{[2]}$ because

$$\boldsymbol{Y}_{[1]} = A_1^{-1}(\boldsymbol{x} - A_2\boldsymbol{Y}_{[2]}).$$

4. Any $\boldsymbol{Y}_{[2]}$ producing non-negative $\boldsymbol{Y}_{[1]}$ is feasible.
5. Sequantially sample each element of $\boldsymbol{Y}_{[2]}$ conditional on others.
   - Use narrowing sequence of uniform proposal densities at each step.

Solution?

- This method can fail entirely in special circumstances.

Solution?

- This method can fail entirely in special circumstances.

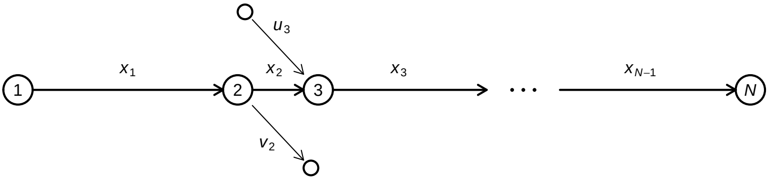
Solution?

- This method can fail entirely in special circumstances.
- When it does work in can mix very slowly.

Solution?

- This method can fail entirely in special circumstances.
- When it does work in can mix very slowly.
- Not computationally feasible for large networks.

A Reliable Solution for Linear Networks



- For a linear network, one route per node pair.
- Build proposal distribution on crude bus travel model.

Overview of Sampler for Linear Networks

- Assume passengers forget origin, so all currently on bus are equally likely to leave at next exit.
- Easy to simulate conditional on numbers boarding/alighting.
- Get proposal probabilities from simple combinatorics.
- Always generates feasible candidate $\boldsymbol{y}$.
- Acceptance probabilities easy to compute.
- For details see Hazelton (2010), *Technometrics* **52**, 221-230.

Overview of Sampler for Linear Networks

- Assume passengers forget origin, so all currently on bus are equally likely to leave at next exit.
- Easy to simulate conditional on numbers boarding/alighting.
- Get proposal probabilities from simple combinatorics.
- Always generates feasible candidate $\boldsymbol{y}$.
- Acceptance probabilities easy to compute.
- For details see Hazelton (2010), *Technometrics* **52**, 221-230.

Extension to Other Ordered Networks

- Linear network method clearly extentable to trees...

Extension to Other Ordered Networks

- Linear network method clearly extentable to trees...
- ... in fact, any directed acyclic graph.

Extension to Other Ordered Networks

- Linear network method clearly extentable to trees...
- ... in fact, any directed acyclic graph.
- Clearly (?) not extendable to 'unordered' networks.

Network Based Transport Models

- Describe flow of vehicles through network.
- Used for a variety of traffic planning and control purposes.
- Come in many flavours:
  - Static or dynamic
  - Equilibrium; mesoscopic; micro-simulation
- Heavily parameterized
- Defined at micro level, but macro properties of interest

Model Parameters

- Travel demand parameters $\boldsymbol{\theta}$ critical.
- Examples:
    - Origin-destination (OD) rates of traffic flow
    - Route choice probabilities
- Relate directly to path flows, $\boldsymbol{Y}$.

Data for Model Fitting

- Widely available data are link (arc) traffic counts $\boldsymbol{x}$.
- Modest measurement error.
- Relate only indirectly to parameters of interest.

MCMC For Taffic Models

- Joint posterior of $\boldsymbol{Y}, \boldsymbol{\theta}$ is

$$
\begin{aligned}
f(\boldsymbol{\theta}, \boldsymbol{Y}|\boldsymbol{x}) &= f(\boldsymbol{\theta}|\boldsymbol{Y}, \boldsymbol{x}) p(\boldsymbol{Y}|\boldsymbol{x}, \boldsymbol{\theta}) \\
&= f(\boldsymbol{\theta}|\boldsymbol{Y}) p(\boldsymbol{Y}|\boldsymbol{x})
\end{aligned}
$$

- Suggests two stage process, switching between sampling $\boldsymbol{Y}$ and $\boldsymbol{\theta}$.
- Sampling $\boldsymbol{\theta}|\boldsymbol{Y}$ typically simple.

MCMC For Taffic Models

- Joint posterior of $\boldsymbol{Y}, \boldsymbol{\theta}$ is

$$\begin{aligned} f(\boldsymbol{\theta}, \boldsymbol{Y} | \boldsymbol{x}) &= f(\boldsymbol{\theta} | \boldsymbol{Y}, \boldsymbol{x}) p(\boldsymbol{Y} | \boldsymbol{x}, \boldsymbol{\theta}) \\ &= f(\boldsymbol{\theta} | \boldsymbol{Y}) p(\boldsymbol{Y} | \boldsymbol{x}) \end{aligned}$$

- Suggests two stage process, switching between sampling $\boldsymbol{Y}$ and $\boldsymbol{\theta}$.
- Sampling $\boldsymbol{\theta} | \boldsymbol{Y}$ typically simple.
- Sampling $\boldsymbol{Y} | \boldsymbol{x}, \boldsymbol{\theta}$ is hard part, as we've seen!

Acknowledgement

Joint work with Katharina Parry

Support from the Royal Society of New Zealand (Marsden fund) gratefully acknowledged.

References and Further Reading

- Copy of these slides
  `http://www.massey.ac.nz/~mhazelto/seminars`

- Hazelton, M.L. (2010). Bayesian inference for network-based modes with a linear inverse structure. *Transportation Research Part B* **44**, 674–685.
- Hazelton, M.L. (2010). Statistical inference for transit system origin-destination matrices. *Technometrics* **52**, 221-230.
- Tebaldi, C. and M. West (1998). Bayesian inference on network traffic using link count data (with discussion). *Journal of the American Statistical Association* **93** 557–576.