Efficient Markov Chain Monte Carlo methods for Reliability Estimation and Uncertainty Quantification in Complex Systems and Networks

Konstantin Zuev

University of Southern California

http://www-bcf.usc.edu/~kzuev

11 March, 2013
Institute for Risk & Uncertainty
University of Liverpool
My Research: A Big Picture

Main Research Areas:

- Reliability Engineering
- Uncertainty Quantification
- Critical Infrastructure Networks
My Research: A Big Picture

Main Research Areas:
- Reliability Engineering
- Uncertainty Quantification
- Critical Infrastructure Networks

Methods & Concepts:
- Markov Chain Monte Carlo
- Computational Bayesian Inference
- Network Science
My Research: A Big Picture

Main Research Areas:
- Reliability Engineering
- Uncertainty Quantification
- Critical Infrastructure Networks

Methods & Concepts:
- Markov Chain Monte Carlo
- Computational Bayesian Inference
- Network Science
My Research: A Big Picture

Main Research Areas:
- Reliability Engineering
- Uncertainty Quantification
- Critical Infrastructure Networks

Methods & Concepts:
- Markov Chain Monte Carlo
- Computational Bayesian Inference
- Network Science
MCMC Revolution


...asking about applications of Markov Chain Monte Carlo (MCMC) is a little like asking about applications of the quadratic formula... you can take any area of science, from hard to social, and find a burgeoning MCMC literature specifically tailored to that area.
MCMC Revolution


...asking about applications of Markov Chain Monte Carlo (MCMC) is a little like asking about applications of the quadratic formula... you can take any area of science, from hard to social, and find a burgeoning MCMC literature specifically tailored to that area.

- Originated in Statistical Physics: sampling from the Boltzman distribution.
MCMC Revolution


...asking about applications of Markov Chain Monte Carlo (MCMC) is a little like asking about applications of the quadratic formula... you can take any area of science, from hard to social, and find a burgeoning MCMC literature specifically tailored to that area.

- Originated in Statistical Physics: sampling from the Boltzman distribution.
- A key computational tool in Bayesian Statistics: sampling from the posterior.
MCMC Revolution


...asking about applications of Markov Chain Monte Carlo (MCMC) is a little like asking about applications of the quadratic formula... you can take any area of science, from hard to social, and find a burgeoning MCMC literature specifically tailored to that area.

- Originated in Statistical Physics: sampling from the Boltzmann distribution.
- A key computational tool in Bayesian Statistics: sampling from the posterior.
- Extensively used in Computer Science, Biochemistry, Economics, Finance.
MCMC Revolution


...asking about applications of Markov Chain Monte Carlo (MCMC) is a little like asking about applications of the quadratic formula...
you can take any area of science, from hard to social, and find a burgeoning MCMC literature specifically tailored to that area.

- Originated in **Statistical Physics**: sampling from the Boltzmann distribution.
- A key computational tool in **Bayesian Statistics**: sampling from the posterior.
- Extensively used in **Computer Science, Biochemistry, Economics, Finance**.

MCMC can be efficiently used for **Reliability Estimation and Uncertainty Quantification**
MCMC Revolution


...asking about applications of Markov Chain Monte Carlo (MCMC) is a little like asking about applications of the quadratic formula... you can take any area of science, from hard to social, and find a burgeoning MCMC literature specifically tailored to that area.

- Originated in Statistical Physics: sampling from the Boltzman distribution.
- A key computational tool in Bayesian Statistics: sampling from the posterior.
- Extensively used in Computer Science, Biochemistry, Economics, Finance.

MCMC can be efficiently used for Reliability Estimation and Uncertainty Quantification.
MCMC Revolution


...asking about applications of Markov Chain Monte Carlo (MCMC) is a little like asking about applications of the quadratic formula... you can take any area of science, from hard to social, and find a burgeoning MCMC literature specifically tailored to that area.

- Originated in Statistical Physics: sampling from the Boltzmann distribution.
- A key computational tool in Bayesian Statistics: sampling from the posterior.
- Extensively used in Computer Science, Biochemistry, Economics, Finance.

MCMC can be efficiently used for Reliability Estimation and Uncertainty Quantification
Problem:

To estimate $\int_X h(x) \pi(x) \, dx$

$X \subseteq \mathbb{R}^d$

$h$:

$X \to \mathbb{R}$ function of interest

$\pi(x)$:

"target" PDF on $X$

"Easy" Cases:

$d$ is small ($d = 1, 2, 3$) $\Rightarrow$ numerical integration

$\pi(x)$ is easy to sample from $\Rightarrow$ Monte Carlo method

Typically in Applications:

$d$ is large ($d \sim 10^3$)

$\pi(x)$ is known only up to a scaling constant, $\pi(x) \propto f(x)$

MCMC:

a powerful simulation method that efficiently solves this problem

Konstantin Zuev (USC)
MCMC in a Nutshell: What is It For?

Problem: To estimate

\[ \mathbb{E}_\pi[h(x)] = \int_X h(x)\pi(x)dx \]

- \( X \subseteq \mathbb{R}^d \) parameter space
- \( h : X \rightarrow \mathbb{R} \) function of interest
- \( \pi(x) \) "target" PDF on \( X \)
MCMC in a Nutshell: What is It For?

Problem: To estimate

$$\mathbb{E}_\pi[h(x)] = \int_{\mathcal{X}} h(x)\pi(x)dx$$

- $\mathcal{X} \subseteq \mathbb{R}^d$ parameter space
- $h : \mathcal{X} \to \mathbb{R}$ function of interest
- $\pi(x)$ “target” PDF on $\mathcal{X}$

“Easy” Cases:

- $d$ is small ($d = 1, 2, 3$) $\Rightarrow$ numerical integration
- $\pi(x)$ is easy to sample from $\Rightarrow$ Monte Carlo method
MCMC in a Nutshell: What is It For?

Problem: To estimate

\[ \mathbb{E}_{\pi}[h(x)] = \int_{\mathcal{X}} h(x)\pi(x)dx \]

- \( \mathcal{X} \subseteq \mathbb{R}^d \) parameter space
- \( h : \mathcal{X} \rightarrow \mathbb{R} \) function of interest
- \( \pi(x) \) “target” PDF on \( \mathcal{X} \)

“Easy” Cases:
- \( d \) is small \( (d = 1, 2, 3) \) \( \Rightarrow \) numerical integration
- \( \pi(x) \) is easy to sample from \( \Rightarrow \) Monte Carlo method

Typically in Applications:
- \( d \) is large \( (d \sim 10^3) \)
- \( \pi(x) \) is known only up to a scaling constant, \( \pi(x) \propto f(x) \)
MCMC in a Nutshell: What is It For?

Problem: To estimate

$$\mathbb{E}_\pi[h(x)] = \int_{\mathcal{X}} h(x)\pi(x)dx$$

- $\mathcal{X} \subseteq \mathbb{R}^d$ parameter space
- $h : \mathcal{X} \rightarrow \mathbb{R}$ function of interest
- $\pi(x)$ “target” PDF on $\mathcal{X}$

“Easy” Cases:
- $d$ is small ($d = 1, 2, 3$) $\Rightarrow$ numerical integration
- $\pi(x)$ is easy to sample from $\Rightarrow$ Monte Carlo method

Typically in Applications:
- $d$ is large ($d \sim 10^3$)
- $\pi(x)$ is known only up to a scaling constant, $\pi(x) \propto f(x)$

MCMC: a powerful simulation method that efficiently solves this problem
MCMC in a Nutshell: How does It Work?

Monte Carlo method

\[ \pi[h(x)] = \int_X h(x) \pi(x) \, dx \approx \frac{1}{N} \sum_{i=1}^{N} h(x_i), \quad x_i \sim \pi(x) \]

How to obtain samples from \( \pi(x) \) \( \propto f(x) \)?

MCMC obtains samples from \( \pi(x) \) by generating a Markov Chain.
MCMC in a Nutshell: How does It Work?

- Monte Carlo method

\[
\mathbb{E}_\pi[h(x)] = \int_X h(x)\pi(x)dx
\]
MCMC in a Nutshell: How does It Work?

- Monte Carlo method

\[ E_\pi[h(x)] = \int h(x) \pi(x) dx \approx \frac{1}{N} \sum_{i=1}^{N} h(x_i), \quad x_i \sim \pi(x) \]
MCMC in a Nutshell: How does It Work?

- Monte Carlo method

\[
\mathbb{E}_\pi[h(x)] = \int_X h(x)\pi(x)dx \approx \frac{1}{N} \sum_{i=1}^{N} h(x_i), \quad x_i \sim \pi(x)
\]
MCMC in a Nutshell: How does It Work?

- Monte Carlo method

\[ \mathbb{E}_\pi[h(x)] = \int_X h(x) \pi(x) \, dx \approx \frac{1}{N} \sum_{i=1}^{N} h(x_i), \quad x_i \sim \pi(x) \]

- How to obtain samples from \( \pi(x) \propto f(x) \)?
MCMC in a Nutshell: How does It Work?

- Monte Carlo method

\[
\mathbb{E}_\pi [h(x)] = \int_x h(x) \pi(x) dx \approx \frac{1}{N} \sum_{i=1}^{N} h(x_i), \quad x_i \sim \pi(x)
\]

- How to obtain samples from \( \pi(x) \propto f(x) \)?

**MCMC** obtains samples from \( \pi(x) \) by generating a **Markov Chain**:

\[
\int_x h(x) \pi(x) dx \approx \frac{1}{N - N_0} \sum_{i=N_0+1}^{N} h(x_i)
\]
Why is MCMC useful in Engineering?
Why is MCMC useful in Engineering?

Main Research Areas:
- Reliability Engineering
- Uncertainty Quantification
- Critical Infrastructure Networks

Methods & Concepts:
- Markov chain Monte Carlo
- Computational Bayesian inference
- Network Science
Why is MCMC useful in Engineering?

Main Research Areas:
- Reliability Engineering
- Uncertainty Quantification
- Critical Infrastructure Networks

Methods & Concepts:
- Markov chain Monte Carlo
- Computational Bayesian inference
- Network Science
System Reliability Problem

Reliability Problem: To estimate the probability of failure $p_F$

\[ p_F = \mathbb{P}(x \in F) = \int_{\mathbb{R}^d} \pi(x) I_F(x) dx \]

Notation:
- $x \in \mathbb{R}^d$ represents the uncertain excitation of a system
- $x$ is a random vector with joint PDF $\pi(x)$
- $F \subset \mathbb{R}^d$ is a failure domain (unacceptable system performance)
- $g(x)$ is a performance function (loss function)
- $b^\star$ is a critical threshold for performance
- $I_F(x) = 1$ if $x \in F$ and $I_F(x) = 0$ if $x \not\in F$
System Reliability Problem

Reliability Problem: To estimate the probability of failure $p_F$

$$p_F = \Pr(x \in F) = \int_{\mathbb{R}^d} \pi(x) I_F(x) dx$$

Notation:
- $x \in \mathbb{R}^d$ represents the uncertain excitation of a system
  - $x$ is a random vector with joint PDF $\pi(x)$
- $F \subset \mathbb{R}^d$ is a failure domain (unacceptable system performance)
  $$F = \{x : g(x) \geq b^*\}$$
- $g(x)$ is a performance function (loss function)
- $b^*$ is a critical threshold for performance
- $I_F(x) = 1$ if $x \in F$ and $I_F(x) = 0$ if $x \notin F$
Why is Reliability Problem Computationally Challenging?

\[ p_F = \int_{\mathbb{R}^d} \pi(x) I_F(x) dx, \quad F = \{ x : g(x) \geq b^* \} \]
Why is Reliability Problem Computationally Challenging?

\[
p_F = \int_{\mathbb{R}^d} \pi(x) I_F(x) \, dx, \quad F = \{ x : g(x) \geq b^* \}
\]

Typically in Applications:
- The relationship between \( x \) and \( I_F(x) \) is not explicitly known
Why is Reliability Problem Computationally Challenging?

\[ p_F = \int_{\mathbb{R}^d} \pi(x) I_F(x) \, dx, \quad F = \{ x : g(x) \geq b^* \} \]

Typically in Applications:

- The relationship between \( x \) and \( I_F(x) \) is not explicitly known
- We can compute \( I_F(x) \) for any \( x \), but this computation is expensive
Why is Reliability Problem Computationally Challenging?

\[ p_F = \int_{\mathbb{R}^d} \pi(x) I_F(x) dx, \quad F = \{x : g(x) \geq b^*\} \]

Typically in Applications:

- The relationship between \( x \) and \( I_F(x) \) is not explicitly known
- We can compute \( I_F(x) \) for any \( x \), but this computation is expensive
- The probability of failure \( p_F \) is very small, \( p_F \sim 10^{-2} - 10^{-9} \)
Why is Reliability Problem Computationally Challenging?

\[ p_F = \int_{\mathbb{R}^d} \pi(x) I_F(x) dx, \quad F = \{ x : g(x) \geq b^* \} \]

Typically in Applications:
- The relationship between \( x \) and \( I_F(x) \) is not explicitly known
- We can compute \( I_F(x) \) for any \( x \), but this computation is expensive
- The probability of failure \( p_F \) is very small, \( p_F \sim 10^{-2} - 10^{-9} \)
- The dimension \( d \) is very large, \( d \sim 10^3 \)
Why is Reliability Problem Computationally Challenging?

\[ p_F = \int_{\mathbb{R}^d} \pi(x)I_F(x)dx, \quad F = \{x : g(x) \geq b^*\} \]

Typically in Applications:
- The relationship between \( x \) and \( I_F(x) \) is not explicitly known
- We can compute \( I_F(x) \) for any \( x \), but this computation is expensive
- The probability of failure \( p_F \) is very small, \( p_F \sim 10^{-2} - 10^{-9} \)
- The dimension \( d \) is very large, \( d \sim 10^3 \)

Consequences:
- Numerical integration is computationally infeasible
- Monte Carlo method is too expensive
Why is Reliability Problem Computationally Challenging?

\[ p_F = \int_{\mathbb{R}^d} \pi(x) I_F(x) dx, \quad F = \{ x : g(x) \geq b^* \} \]

Typically in Applications:
- The relationship between \( x \) and \( I_F(x) \) is not explicitly known
- We can compute \( I_F(x) \) for any \( x \), but this computation is expensive
- The probability of failure \( p_F \) is very small, \( p_F \sim 10^{-2} - 10^{-9} \)
- The dimension \( d \) is very large, \( d \sim 10^3 \)

Consequences:
- Numerical integration is computationally infeasible
- Monte Carlo method is too expensive

Idea: To use advanced simulation methods based on MCMC
Subset Simulation for Estimation of Small $\rho_F = \mathbb{P}(x \in F)$
Subset Simulation for Estimation of Small $\rho_F = \mathbb{P}(x \in F)$
Subset Simulation for Estimation of Small $\rho_F = \mathbb{P}(x \in F)$

$\mathbb{R}^d = F_0 \supset F_1 \supset \ldots \supset F_m = F$
Subset Simulation for Estimation of Small $\rho_F = \mathbb{P}(x \in F)$

\[
\mathbb{R}^d = F_0 \supset F_1 \supset \ldots \supset F_m = F
\]

\[
F = \{x : g(x) \geq b^*\}
\]
Subset Simulation for Estimation of Small $\rho_F = \mathbb{P}(x \in F)$

\[ \mathbb{R}^d = F_0 \supset F_1 \supset \ldots \supset F_m = F \]

\[ F = \{ x : g(x) \geq b^* \} \]

\[ F_k = \{ x : g(x) \geq b_k^* \} \]

\[ b_1^* < b_2^* < \ldots < b_m^* = b^* \]
Subset Simulation for Estimation of Small $\rho_F = \mathbb{P}(x \in F)$

$\mathbb{R}^d = F_0 \supset F_1 \supset \ldots \supset F_m = F$

$F = \{x : g(x) \geq b^*\}$

$F_k = \{x : g(x) \geq b_k^*\}$

$b_1^* < b_2^* < \ldots < b_m^* = b^*$

\[
\rho_F = \prod_{k=0}^{m-1} P(F_{k+1}|F_k)
\]
Subset Simulation for Estimation of Small $p_F = \mathbb{P}(x \in F)$

$$P(F_{k+1}|F_k) \approx \frac{1}{N} \sum_{i=1}^{N} I_{F_{k+1}}(x_k^{(i)})$$

$$x_k^{(i)} \sim \pi(x|F_k) = \frac{\pi(x)I_{F_k}(x)}{P(F_k)}$$

$\mathbb{R}^d = F_0 \supset F_1 \supset \ldots \supset F_m = F$

$F = \{x : g(x) \geq b^*\}$

$F_k = \{x : g(x) \geq b_k^*\}$

$b_1^* < b_2^* < \ldots < b_m^* = b^*$

$$p_F = \prod_{k=0}^{m-1} P(F_{k+1}|F_k)$$
Subset Simulation for Estimation of Small $\rho_F = \mathbb{P}(x \in F)$

$$P(F_{k+1}|F_k) \approx \frac{1}{N} \sum_{i=1}^{N} I_{F_{k+1}}(x_k^{(i)})$$

$$x_k^{(i)} \sim \pi(x|F_k) = \frac{\pi(x)I_{F_k}(x)}{P(F_k)}$$

- How to sample from $\pi(x|F_k)$?

\[ \mathbb{R}^d = F_0 \supset F_1 \supset \ldots \supset F_m = F \]

\[ F = \{x : g(x) \geq b^* \} \]

\[ F_k = \{x : g(x) \geq b_k^* \} \]

\[ b_1^* < b_2^* < \ldots < b_m^* = b^* \]

\[ \rho_F = \prod_{k=0}^{m-1} P(F_{k+1}|F_k) \]
Subset Simulation for Estimation of Small $\rho_F = \mathbb{P}(x \in F)$

$$\mathbb{R}^d = F_0 \supset F_1 \supset \ldots \supset F_m = F$$

$$F = \{ x : g(x) \geq b^* \}$$

$$F_k = \{ x : g(x) \geq b_k^* \}$$

$$b_1^* < b_2^* < \ldots < b_m^* = b^*$$

$$\rho_F = \prod_{k=0}^{m-1} P(F_{k+1}|F_k)$$

$$P(F_{k+1}|F_k) \approx \frac{1}{N} \sum_{i=1}^{N} I_{F_{k+1}}(x_k^{(i)})$$

$$x_k^{(i)} \sim \pi(x|F_k) = \frac{\pi(x) I_{F_k}(x)}{P(F_k)}$$

- How to sample from $\pi(x|F_k)$?
- Use an MCMC algorithm
Sampling from $\pi(x|F_k)$
Sampling from $\pi(x|F_k)$

The original Metropolis-Hastings algorithm
Sampling from $\pi(x|F_k)$

The original Metropolis-Hastings algorithm suffers from the curse of dimensionality.
Sampling from $\pi(x|F_k)$

The original Metropolis-Hastings algorithm suffers from the curse of dimensionality

**Modified Metropolis-Hastings algorithm:** $x^{(n)} \leadsto x^{(n+1)}$
Sampling from $\pi(x|F_k)$

The original Metropolis-Hastings algorithm suffers from the curse of dimensionality

Modified Metropolis-Hastings algorithm: $x^{(n)} \sim x^{(n+1)}$

- Generate a candidate state $y$
Sampling from $\pi(x|F_k)$

The original Metropolis-Hastings algorithm suffers from the curse of dimensionality

**Modified Metropolis-Hastings algorithm:** $x^{(n)} \leadsto x^{(n+1)}$

- Generate a candidate state $y$
  - For each coordinate $j = 1 \ldots d$:
    - Generate $\hat{y}_j \sim f_j(\cdot|x_j^{(n)})$
    - Compute the acceptance probability
      \[
      a_j = \min\left\{1, \frac{\pi_j(\hat{y}_j)}{\pi_j(x_j^{(n)})}\right\}
      \]
    - Accept/Reject $\hat{y}_j$
      \[
      y_j = \begin{cases} 
        \hat{y}_j, & \text{with prob } a_j \\
        x_j^{(n)}, & \text{with prob } 1 - a_j 
      \end{cases}
      \]
Sampling from $\pi(x|F_k)$

The original Metropolis-Hastings algorithm suffers from the curse of dimensionality.

Modified Metropolis-Hastings algorithm: $x^{(n)} \sim x^{(n+1)}$

- Generate a candidate state $y$

  For each coordinate $j = 1 \ldots d$:
  - Generate $\hat{y}_j \sim f_j(\cdot|x_j^{(n)})$
  - Compute the acceptance probability
    \[
    a_j = \min\left\{1, \frac{\pi_j(\hat{y}_j)}{\pi_j(x_j^{(n)})}\right\}
    \]
  - Accept/Reject $\hat{y}_j$
    \[
    y_j = \begin{cases} 
    \hat{y}_j, & \text{with prob } a_j \\
    x_j^{(n)}, & \text{with prob } 1 - a_j
    \end{cases}
    \]

- Accept/Reject $y$

  \[
  x^{(n+1)} = \begin{cases} 
  y, & \text{if } y \in F_k \\
  x^{(n)}, & \text{if } y \notin F_k
  \end{cases}
  \]

Konstantin Zuev (USC)
Statistical Properties and Efficiency of Subset Simulation

\[ \hat{p}_F = \prod_{k=0}^{m-1} \left( \frac{1}{N} \sum_{i=1}^{N} I_{F_{k+1}}(x_k^{(i)}) \right), \quad x_k^{(i)} \sim \pi(x|F_k) \]
Statistical Properties and Efficiency of Subset Simulation

SS estimator: \( \hat{p}_F = \prod_{k=0}^{m-1} \left( \frac{1}{N} \sum_{i=1}^{N} I_{F_{k+1}}(x^{(i)}_k) \right), \quad x^{(i)}_k \sim \pi(x|F_k) \)

Statistical properties:

- \( \hat{p}_F \) is asymptotically unbiased and bias is \( O(1/N) \)
- \( \hat{p}_F \) is consistent and its coefficient of variation \( \delta = O(1/\sqrt{N}) \)
Statistical Properties and Efficiency of Subset Simulation

**SS estimator:**

\[ \hat{p}_F = \prod_{k=0}^{m-1} \left( \frac{1}{N} \sum_{i=1}^{N} I_{F_{k+1}}(x^{(i)}_k) \right), \quad x^{(i)}_k \sim \pi(x|F_k) \]

**Statistical properties:**

- \( \hat{p}_F \) is asymptotically unbiased and bias is \( O(1/N) \)
- \( \hat{p}_F \) is consistent and its coefficient of variation \( \delta = O(1/\sqrt{N}) \)

**Efficiency:**

What total number of samples is required to achieve a given accuracy in \( \hat{p}_F \)?

Standard Monte Carlo:

\[ N_T \propto \frac{1}{p_F} \]

Subset Simulation:

\[ N_T \propto |\log p_F| \]

Subset Simulation is very efficient when estimating small probabilities.
Statistical Properties and Efficiency of Subset Simulation

**SS estimator:**

\[ \hat{p}_F = \prod_{k=0}^{m-1} \left( \frac{1}{N} \sum_{i=1}^{N} I_{F_{k+1}}(x_k^{(i)}) \right), \quad x_k^{(i)} \sim \pi(x|F_k) \]

**Statistical properties:**

- \( \hat{p}_F \) is asymptotically unbiased and bias is \( O(1/N) \)
- \( \hat{p}_F \) is consistent and its coefficient of variation \( \delta = O(1/\sqrt{N}) \)

**Efficiency:**

What total number of samples is required to achieve a given accuracy in \( \hat{p}_F \)?

- Standard Monte Carlo: \( N_T \propto 1/p_F \)
- Subset Simulation: \( N_T \propto |\log p_F|^r \), where \( r \leq 3 \)
Statistical Properties and Efficiency of Subset Simulation

SS estimator:

$$\hat{p}_F = \prod_{k=0}^{m-1} \left( \frac{1}{N} \sum_{i=1}^{N} I_{F_k+1}(x_k^{(i)}) \right), \quad x_k^{(i)} \sim \pi(x|F_k)$$

Statistical properties:

- $\hat{p}_F$ is asymptotically unbiased and bias is $O(1/N)$
- $\hat{p}_F$ is consistent and its coefficient of variation $\delta = O(1/\sqrt{N})$

Efficiency:

What total number of samples is required to achieve a given accuracy in $\hat{p}_F$?

- Standard Monte Carlo: $N_T \propto 1/p_F$
- Subset Simulation: $N_T \propto |\log p_F|^r$, where $r \leq 3$

Subset Simulation is very efficient when estimating small probabilities
Reliability of Critical Infrastructure Networks

Main Research Areas:
- Reliability Engineering
- Uncertainty Quantification
- Critical Infrastructure Networks

Methods & Concepts:
- Markov chain Monte Carlo
- Computational Bayesian inference
- Network Science
Reliability of Critical Infrastructure Networks

Main Research Areas:
- Reliability Engineering
- Uncertainty Quantification
- Critical Infrastructure Networks

Methods & Concepts:
- Markov chain Monte Carlo
- Computational Bayesian inference
- Network Science
Network Reliability Problem

Network topology is represented by a graph $G = (V, E)$

- $V = \{v_1, \ldots, v_n\}$ set of all nodes
- $E = \{e_1, \ldots, e_m\}$ set of all links

A network state is $s = (s_1, \ldots, s_m)$, where

- $s_i = 1$ if link $e_i$ is fully operational
- $0 < s_i < 1$ if link $e_i$ is partially operational
- $s_i = 0$ if link $e_i$ is completely failed

The network state space is $S = \{(s_1, \ldots, s_m) : 0 \leq s_i \leq 1\} = [0, 1]^m$

Let $\pi(s)$ be a probability distribution on $S$, $s \sim \pi(s)$

Let $\mu : S \rightarrow \mathbb{R}$ be a performance function (utility function)

The failure domain is $F = \{s : \mu(s) < \mu^\star\} \subset S$

Network Reliability Problem: To estimate the probability of failure $p_F = P(s \in F) = \int_S \pi(s) I_F(s) \, ds$
Network Reliability Problem

- **Network topology** is represented by a graph \( G = (V, E) \)
  - \( V = \{v_1, \ldots, v_n\} \) set of all nodes
  - \( E = \{e_1, \ldots, e_m\} \) set of all links
Network Reliability Problem

- **Network topology** is represented by a graph \( G = (V, E) \)
  - \( V = \{v_1, \ldots, v_n\} \) set of all nodes
  - \( E = \{e_1, \ldots, e_m\} \) set of all links
- A network state is \( s = (s_1, \ldots, s_m) \), where \( 0 \leq s_i \leq 1 \)
Network Reliability Problem

- Network topology is represented by a graph $G = (V, E)$
  - $V = \{v_1, \ldots, v_n\}$ set of all nodes
  - $E = \{e_1, \ldots, e_m\}$ set of all links
- A network state is $s = (s_1, \ldots, s_m)$, where $0 \leq s_i \leq 1$
  - $s_i = 1$ if link $e_i$ is fully operational

The network state space is $S = \{(s_1, \ldots, s_m) : 0 \leq s_i \leq 1\} = [0, 1]^m$

Let $\pi(s)$ be a probability distribution on $S$, $s \sim \pi(s)$

Let $\mu : S \to \mathbb{R}$ be a performance function (utility function)

The failure domain is $F = \{s : \mu(s) < \mu^\star\} \subset S$

Network Reliability Problem: To estimate the probability of failure $p_F = P(s \in F) = \int_S \pi(s) I_F(s) \, ds$
Network Reliability Problem

- **Network topology** is represented by a graph $G = (V, E)$
  - $V = \{v_1, \ldots, v_n\}$ set of all nodes
  - $E = \{e_1, \ldots, e_m\}$ set of all links

- **A network state** is $s = (s_1, \ldots, s_m)$, where $0 \leq s_i \leq 1$
  - $s_i = 1$ if link $e_i$ is fully operational
  - $0 < s_i < 1$ if link $e_i$ is partially operational

---

Optimization Problem:

To estimate the probability of failure $p_F$:

$$p_F = P(s \in F) = \int_S \pi(s) I_F(s) \, ds$$
Network Reliability Problem

- **Network topology** is represented by a graph \( G = (V, E) \)
  - \( V = \{v_1, \ldots, v_n\} \) set of all nodes
  - \( E = \{e_1, \ldots, e_m\} \) set of all links
- **A network state** is \( s = (s_1, \ldots, s_m) \), where \( 0 \leq s_i \leq 1 \)
  - \( s_i = 1 \) if link \( e_i \) is fully operational
  - \( 0 < s_i < 1 \) if link \( e_i \) is partially operational
  - \( s_i = 0 \) if link \( e_i \) is completely failed
Network Reliability Problem

- Network topology is represented by a graph $G = (V, E)$
  - $V = \{v_1, \ldots, v_n\}$ set of all nodes
  - $E = \{e_1, \ldots, e_m\}$ set of all links
- A network state is $s = (s_1, \ldots, s_m)$, where $0 \leq s_i \leq 1$
  - $s_i = 1$ if link $e_i$ is fully operational
  - $0 < s_i < 1$ if link $e_i$ is partially operational
  - $s_i = 0$ if link $e_i$ is completely failed
- The network state space is $\mathcal{S} = \{(s_1, \ldots, s_m) : 0 \leq s_i \leq 1\}$
Network Reliability Problem

- **Network topology** is represented by a graph $G = (V, E)$
  - $V = \{v_1, \ldots, v_n\}$ set of all nodes
  - $E = \{e_1, \ldots, e_m\}$ set of all links
- A network state is $s = (s_1, \ldots, s_m)$, where $0 \leq s_i \leq 1$
  - $s_i = 1$ if link $e_i$ is fully operational
  - $0 < s_i < 1$ if link $e_i$ is partially operational
  - $s_i = 0$ if link $e_i$ is completely failed
- The network state space is $S = \{(s_1, \ldots, s_m) : 0 \leq s_i \leq 1\} = [0, 1]^m$
Network Reliability Problem

- Network topology is represented by a graph \( G = (V, E) \)
  - \( V = \{v_1, \ldots, v_n\} \) set of all nodes
  - \( E = \{e_1, \ldots, e_m\} \) set of all links
- A network state is \( s = (s_1, \ldots, s_m) \), where \( 0 \leq s_i \leq 1 \)
  - \( s_i = 1 \) if link \( e_i \) is fully operational
  - \( 0 < s_i < 1 \) if link \( e_i \) is partially operational
  - \( s_i = 0 \) if link \( e_i \) is completely failed
- The network state space is \( S = \{(s_1, \ldots, s_m) : 0 \leq s_i \leq 1\} = [0, 1]^m \)
- Let \( \pi(s) \) be a probability distribution on \( S \), \( s \sim \pi(s) \)
Network Reliability Problem

- **Network topology** is represented by a graph $G = (V, E)$
  - $V = \{v_1, \ldots, v_n\}$ set of all nodes
  - $E = \{e_1, \ldots, e_m\}$ set of all links
- **A network state** is $s = (s_1, \ldots, s_m)$, where $0 \leq s_i \leq 1$
  - $s_i = 1$ if link $e_i$ is fully operational
  - $0 < s_i < 1$ if link $e_i$ is partially operational
  - $s_i = 0$ if link $e_i$ is completely failed
- The **network state space** is $S = \{(s_1, \ldots, s_m) : 0 \leq s_i \leq 1\} = [0, 1]^m$
- Let $\pi(s)$ be a probability distribution on $S$, $s \sim \pi(s)$
- Let $\mu : S \to \mathbb{R}$ be a **performance function** (utility function)
Network Reliability Problem

- **Network topology** is represented by a graph $G = (V, E)$
  - $V = \{v_1, \ldots, v_n\}$ set of all nodes
  - $E = \{e_1, \ldots, e_m\}$ set of all links
- A **network state** is $s = (s_1, \ldots, s_m)$, where $0 \leq s_i \leq 1$
  - $s_i = 1$ if link $e_i$ is *fully operational*
  - $0 < s_i < 1$ if link $e_i$ is *partially operational*
  - $s_i = 0$ if link $e_i$ is *completely failed*
- The **network state space** is $S = \{(s_1, \ldots, s_m) : 0 \leq s_i \leq 1\} = [0, 1]^m$
- Let $\pi(s)$ be a probability distribution on $S$, $s \sim \pi(s)$
- Let $\mu : S \to \mathbb{R}$ be a performance function (utility function)
- The **failure domain** is $F = \{s : \mu(s) < \mu^*\} \subset S$
Network Reliability Problem

- **Network topology** is represented by a graph $G = (V, E)$
  - $V = \{v_1, \ldots, v_n\}$ set of all nodes
  - $E = \{e_1, \ldots, e_m\}$ set of all links
- **A network state** is $s = (s_1, \ldots, s_m)$, where $0 \leq s_i \leq 1$
  - $s_i = 1$ if link $e_i$ is fully operational
  - $0 < s_i < 1$ if link $e_i$ is partially operational
  - $s_i = 0$ if link $e_i$ is completely failed
- The **network state space** is $S = \{(s_1, \ldots, s_m) : 0 \leq s_i \leq 1\} = [0, 1]^m$
- Let $\pi(s)$ be a **probability distribution** on $S$, $s \sim \pi(s)$
- Let $\mu : S \to \mathbb{R}$ be a **performance function** (utility function)
- The **failure domain** is $F = \{s : \mu(s) < \mu^*\} \subset S$

Network Reliability Problem: To estimate the **probability of failure** $p_F$

$$p_F = \mathbb{P}(s \in F) = \int_S \pi(s) I_F(s) ds$$
Why is the Network Reliability Problem Challenging?

\[ p_F = \int_S \pi(s) I_F(s) ds, \quad F = \{s : \mu(s) < \mu^*\} \subset [0, 1]^m \]
Why is the Network Reliability Problem Challenging?

\[ p_F = \int_S \pi(s) I_F(s) ds, \quad F = \{ s : \mu(s) < \mu^* \} \subset [0, 1]^m \]

Typically in Applications:
Why is the Network Reliability Problem Challenging?

\[ p_F = \int_S \pi(s)I_F(s)ds, \quad F = \{ s : \mu(s) < \mu^* \} \subset [0, 1]^m \]

Typically in Applications:

- The number of links \( m \) is very large
Why is the Network Reliability Problem Challenging?

\[ p_{\mathcal{F}} = \int_{\mathcal{S}} \pi(s) I_{\mathcal{F}}(s) ds, \quad \mathcal{F} = \{ s : \mu(s) < \mu^* \} \subset [0, 1]^m \]

Typically in Applications:

- The number of links \( m \) is very large

US Western States Power Grid, \( m = 6,594 \)
Why is the Network Reliability Problem Challenging?

\[ p_\mathcal{F} = \int \pi(s) I_\mathcal{F}(s) ds, \quad \mathcal{F} = \{ s : \mu(s) < \mu^* \} \subset [0, 1]^m \]

Typically in Applications:

- The number of links \( m \) is very large

US Western States Power Grid, \( m = 6,594 \)

California Road Network, \( m = 5,533,214 \)
Why is the Network Reliability Problem Challenging?

\[ p_F = \int_S \pi(s) I_F(s) ds, \quad F = \{ s : \mu(s) < \mu^* \} \subset [0, 1]^m \]

Typically in Applications:

- The number of links \( m \) is very large

\[ \mu(s) < \mu^* \]
Why is the Network Reliability Problem Challenging?

\[ p_F = \int_S \pi(s) I_F(s) ds, \quad F = \{ s : \mu(s) < \mu^* \} \subset [0, 1]^m \]

Typically in Applications:

- The number of links \( m \) is very large
- The probability of failure \( p_F \) is very small
Why is the Network Reliability Problem Challenging?

\[ p_F = \int_{S} \pi(s)I_F(s)ds, \quad F = \{s: \mu(s) < \mu^*\} \subset [0, 1]^m \]

Typically in Applications:

- The number of links \( m \) is very large
- The probability of failure \( p_F \) is very small
- The computational effort for evaluating \( \mu(s) \) is significant
Why is the Network Reliability Problem Challenging?

\[ p_F = \int_S \pi(s) I_F(s) ds, \quad F = \{ s : \mu(s) < \mu^* \} \subset [0, 1]^m \]

Typically in Applications:

- The number of links \( m \) is very large
- The probability of failure \( p_F \) is very small
- The computational effort for evaluating \( \mu(s) \) is significant

Consequences:

- Combinatorial exhaustive search methods are not applicable
- Numerical integration is computationally infeasible
- Monte Carlo method is too expensive
Why is the Network Reliability Problem Challenging?

\[ p_F = \int_S \pi(s) I_F(s) ds, \quad F = \{ s : \mu(s) < \mu^* \} \subset [0, 1]^m \]

Typically in Applications:

- The number of links \( m \) is very large
- The probability of failure \( p_F \) is very small
- The computational effort for evaluating \( \mu(s) \) is significant

Consequences:

- Combinatorial exhaustive search methods are not applicable
- Numerical integration is computationally infeasible
- Monte Carlo method is too expensive

Idea: To use Subset Simulation
Example: Maximum Flows in Small-World Networks

Maximum-Flow Problem

A flow on $G$ is $f = \{f_1, \ldots, f_m\}$

- Capacity constraint: $f_i \leq s_i$
- Flow conservation: $f_2 = f_3 + f_5$

The value of a flow is $|f| = \sum_{v \in V} f(a,v) - \sum_{v \in V} f(v,a)$

Max-flow problem: $f_{\text{max}} = \arg \max f |f|$

Maximum-Flow Reliability Problem

Assume capacities are normalized: $0 \leq s_i \leq 1$

For a given $s = (s_1, \ldots, s_m)$, the max-flow performance function:

$\mu_{\text{MF}}(s) = |f_{\text{max}}(s)|$

Let $\pi(s)$ be a probability model for link capacities: $s \sim \pi(s)$

The failure domain: $F = \{s : \mu_{\text{MF}}(s) < \mu^\star\}$

Reliability problem: $p_F = P(s \in F)$
Maximum-Flow Problem

Example: Maximum Flows in Small-World Networks
Example: Maximum Flows in Small-World Networks

Maximum-Flow Problem

The value of a flow is $|f| = \sum_{v \in V} f(a,v) - \sum_{v \in V} f(v,a)$

Max-flow problem: $f_{max} = \arg \max |f|$

Maximum-Flow Reliability Problem

Assume capacities are normalized: $0 \leq s_i \leq 1$
For a given $s = (s_1, ..., s_m)$, the max-flow performance function: $\mu_{MF}(s) = |f_{max}(s)|$

Let $\pi(s)$ be a probability model for link capacities: $s \sim \pi(s)$
The failure domain: $F = \{s : \mu_{MF}(s) < \mu^{\star}\}$
Reliability problem: $p_F = P(s \in F)$
Example: Maximum Flows in Small-World Networks

Maximum-Flow Problem

A flow on $G$ is $f = \{f_1, \ldots, f_m\}$

- Capacity constraint: $f_i \leq s_i$
- Flow conservation: $f_2 = f_3 + f_5$

The value of a flow is $|f| = \sum_{v \in V} f(a,v) - \sum_{v \in V} f(v,a)$

Max-flow problem: $f_{\text{max}} = \text{arg max} |f|$

Maximum-Flow Reliability Problem

Assume capacities are normalized: $0 \leq s_i \leq 1$

For a given $s = (s_1, \ldots, s_m)$, the max-flow performance function: $\mu_{\text{MF}}(s) = |f_{\text{max}}(s)|$

Let $\pi(s)$ be a probability model for link capacities: $s \sim \pi(s)$

The failure domain: $F = \{s: \mu_{\text{MF}}(s) < \mu^{\star}\}$

Reliability problem: $p_F = P(s \in F)$

Konstantin Zuev (USC)
Example: Maximum Flows in Small-World Networks

Maximum-Flow Problem

A flow on $G$ is $f = \{f_1, \ldots, f_m\}$

- Capacity constraint: $f_i \leq s_i$
- Flow conservation: $f_2 = f_3 + f_5$

The value of a flow is $|f| = \sum_{v \in V} f(a,v) - \sum_{v \in V} f(v,a)$

Max-flow problem: $f_{\text{max}} = \arg \max |f|$

Maximum-Flow Reliability Problem

Assume capacities are normalized: $0 \leq s_i \leq 1$

For a given $s = (s_1, \ldots, s_m)$, the max-flow performance function:

$\mu_{\text{MF}}(s) = |f_{\text{max}}(s)|$

Let $\pi(s)$ be a probability model for link capacities: $s \sim \pi(s)$

The failure domain: $F = \{s: \mu_{\text{MF}}(s) < \mu^\star\}$

Reliability problem: $p_F = P(s \in F)$
A flow on $G$ is $f = \{f_1, \ldots, f_m\}$
Example: Maximum Flows in Small-World Networks

Maximum-Flow Problem

A flow on $G$ is $f = \{f_1, \ldots, f_m\}$

- Capacity constraint: $f_i \leq s_i$

Konstantin Zuev (USC)  MCMC for RE & UQ in Complex Systems & Nets  11 March, 2013 University of Liverpool  15 / 23
Example: Maximum Flows in Small-World Networks

**Maximum-Flow Problem**

A flow on $G$ is $f = \{f_1, \ldots, f_m\}$

- Capacity constraint: $f_i \leq s_i$
- Flow conservation:

![Diagram of a network with labeled edges and nodes]

- Source: $a$
- Sink: $b$
- Edges and capacities:
  - $f_1$ from $a$ to $s_1$
  - $f_2$ from $s_2$ to $a$
  - $f_3$ from $s_3$ to $s_4$
  - $f_4$ from $s_4$ to $b$
  - $f_5$ from $s_5$ to $s_6$
  - $f_6$ from $s_6$ to $b$
  - $f_7$ from $s_7$ to $b$
Example: Maximum Flows in Small-World Networks

**Maximum-Flow Problem**

- A flow on $G$ is $f = \{f_1, \ldots, f_m\}$
  - Capacity constraint: $f_i \leq s_i$
  - Flow conservation: $f_2 = f_3 + f_5$
Maximum-Flow Problem

A flow on $G$ is $f = \{f_1, \ldots, f_m\}$
- Capacity constraint: $f_i \leq s_i$
- Flow conservation: $f_2 = f_3 + f_5$

The value of a flow is

$$|f| = \sum_{v \in V} f(a, v) - \sum_{v \in V} f(v, a)$$
Example: Maximum Flows in Small-World Networks

Maximum-Flow Problem

- **A flow on** $G$ is $f = \{f_1, \ldots, f_m\}$
  - Capacity constraint: $f_i \leq s_i$
  - Flow conservation: $f_2 = f_3 + f_5$

- **The value of a flow** is
  \[ |f| = \sum_{v \in V} f(a, v) - \sum_{v \in V} f(v, a) \]

- **Max-flow problem**:
  \[ f_{\text{max}} = \text{arg max}_f |f| \]
Example: Maximum Flows in Small-World Networks

Maximum-Flow Problem

A flow on $G$ is $f = \{f_1, \ldots, f_m\}$

- Capacity constraint: $f_i \leq s_i$
- Flow conservation: $f_2 = f_3 + f_5$

The value of a flow is

$$|f| = \sum_{v \in V} f(a, v) - \sum_{v \in V} f(v, a)$$

Max-flow problem:

$$f_{max} = \arg \max_f |f|$$

Maximum-Flow Reliability Problem

Assume capacities are normalized:

$0 \leq s_i \leq 1$

For a given $s = (s_1, \ldots, s_m)$, the max-flow performance function:

$$\mu_{MF}(s) = |f_{max}(s)|$$

Let $\pi(s)$ be a probability model for link capacities:

$s \sim \pi(s)$

The failure domain:

$$F = \{s : \mu_{MF}(s) < \mu^{\star}\}$$

Reliability problem:

$$p_F = P(s \in F)$$
A flow on $G$ is $f = \{f_1, \ldots, f_m\}$

- Capacity constraint: $f_i \leq s_i$
- Flow conservation: $f_2 = f_3 + f_5$

The value of a flow is

$$|f| = \sum_{v \in V} f(a, v) - \sum_{v \in V} f(v, a)$$

Max-flow problem:

$$f_{\max} = \arg \max_f |f|$$
**Example: Maximum Flows in Small-World Networks**

**Maximum-Flow Problem**

A flow on $G$ is $f = \{f_1, \ldots, f_m\}$

- Capacity constraint: $f_i \leq s_i$
- Flow conservation: $f_2 = f_3 + f_5$

**Maximum-Flow Reliability Problem**

- Assume capacities are normalized: $0 \leq s_i \leq 1$
- For a given $s = (s_1, \ldots, s_m)$, the max-flow performance function:

$$\mu_{MF}(s) = |f_{max}(s)|$$

The value of a flow is

$$|f| = \sum_{v \in V} f(a, v) - \sum_{v \in V} f(v, a)$$

Max-flow problem:

$$f_{max} = \arg \max_f |f|$$
Maximum-Flow Problem

A flow on $G$ is $f = \{f_1, \ldots, f_m\}$
- Capacity constraint: $f_i \leq s_i$
- Flow conservation: $f_2 = f_3 + f_5$

The value of a flow is
$$|f| = \sum_{v \in V} f(a, v) - \sum_{v \in V} f(v, a)$$

Max-flow problem:
$$f_{\max} = \arg \max_f |f|$$

Maximum-Flow Reliability Problem

- Assume capacities are normalized:
  $$0 \leq s_i \leq 1$$
- For a given $s = (s_1, \ldots, s_m)$, the max-flow performance function:
  $$\mu_{MF}(s) = |f_{\max}(s)|$$
- Let $\pi(s)$ be a probability model for link capacities: $s \sim \pi(s)$
Maximum-Flow Problem

A flow on $G$ is $f = \{f_1, \ldots, f_m\}$

- Capacity constraint: $f_i \leq s_i$
- Flow conservation: $f_2 = f_3 + f_5$

The value of a flow is

$$|f| = \sum_{v \in V} f(a, v) - \sum_{v \in V} f(v, a)$$

Max-flow problem:

$$f_{\text{max}} = \arg \max_f |f|$$

Example: Maximum Flows in Small-World Networks

Maximum-Flow Reliability Problem

- Assume capacities are normalized:
  $$0 \leq s_i \leq 1$$
- For a given $s = (s_1, \ldots, s_m)$, the max-flow performance function:
  $$\mu_{\text{MF}}(s) = |f_{\text{max}}(s)|$$
- Let $\pi(s)$ be a probability model for link capacities: $s \sim \pi(s)$
- The failure domain:
  $$\mathcal{F} = \{s : \mu_{\text{MF}}(s) < \mu^*\}$$
Example: Maximum Flows in Small-World Networks

Maximum-Flow Problem

- A flow on $G$ is $f = \{f_1, \ldots, f_m\}$
  - Capacity constraint: $f_i \leq s_i$
  - Flow conservation: $f_2 = f_3 + f_5$
- The value of a flow is
  $$|f| = \sum_{v \in V} f(a, v) - \sum_{v \in V} f(v, a)$$
- Max-flow problem:
  $$f_{max} = \arg \max_f |f|$$

Maximum-Flow Reliability Problem

- Assume capacities are normalized: $0 \leq s_i \leq 1$
- For a given $s = (s_1, \ldots, s_m)$, the max-flow performance function:
  $$\mu_{MF}(s) = |f_{max}(s)|$$
- Let $\pi(s)$ be a probability model for link capacities: $s \sim \pi(s)$
- The failure domain:
  $$\mathcal{F} = \{s : \mu_{MF}(s) < \mu^*\}$$
- Reliability problem:
  $$p_{\mathcal{F}} = P(s \in \mathcal{F})$$
Small-World Network Models
Small-World Network Models

Stanley Milgram (1933 - 1984)

“Six degrees of separation”

Small-world effect: despite their large size, in most real networks there is a relatively short path between almost any two nodes.
Small-World Network Models

Stanley Milgram (1933 - 1984)
Small-World Network Models

Stanley Milgram (1933 - 1984)
Small-World Network Models

Stanley Milgram (1933 - 1984)

Konstantin Zuev (USC)
MCMC for RE & UQ in Complex Systems & Nets
11 March, 2013 University of Liverpool
Small-World Network Models

Stanley Milgram (1933 - 1984)

“Six degrees of separation”

Small-world effect: despite their large size, in most real networks there is a relatively short path between almost any two nodes.
Small-World Network Models

Small-world effect: despite their large size, in most real networks there is a relatively short path between almost any two nodes.

“Six degrees of separation”
Small-World Network Models

**Small-world effect:** despite their large size, in most real networks there is a relatively short path between almost any two nodes.

“Six degrees of separation”
Small-World Network Models

Small-World Ring model \( \otimes (n,k) \)

Realization of \( \otimes (16,3) \)

Small-World Torus model \( \odot (n,k) \)

Realization of \( \odot (4,1) \)

Componentwise:

\[ \otimes (n^2, k+2) = \odot (n,k) \]

Topologically:

\[ \otimes (n^2, k+2) \neq \odot (n,k) \]

\[ \odot (n,k) \]

has more regular links,

\[ \otimes (n^2, k+2) \]

has more random shortcuts

**Question:**

What model, \( \otimes (n^2, k+2) \) or \( \odot (n,k) \), produces a more reliable network?
Small-World Network Models

- Small-World Ring model $\otimes(n, k)$

Realization of $\otimes(16, 3)$

Small-World Torus model $\triangleright(4, 1)$

Componentwise:

$\otimes(n^2, k + 2) = \triangleright(n, k)$

Topologically:

$\otimes(n^2, k + 2) \neq \triangleright(n, k)$

$\triangleright(n, k)$ has more regular links, $\otimes(n^2, k + 2)$ has more random shortcuts.

Question: What model, $\otimes(n^2, k + 2)$ or $\triangleright(n, k)$, produces a more reliable network?
Small-World Network Models

- Small-World Ring model $\otimes(n, k)$

- Small-World Torus model $\odot(n, k)$

Componentwise: $\otimes(n^2, k+2) = \odot(n, k)$

Topologically: $\otimes(n^2, k+2) \neq \odot(n, k)$

$\odot(n, k)$ has more regular links,
$\otimes(n^2, k+2)$ has more random shortcuts

Question: What model, $\otimes(n^2, k+2)$ or $\odot(n, k)$, produces a more reliable network?
Small-World Network Models

- Small-World Ring model $\otimes(n, k)$

Componentwise:
$$\otimes(n^2, k + 2) = \oplus(n, k)$$

Topologically:
$$\otimes(n^2, k + 2) \neq \oplus(n, k) \oplus(n, k)$$

has more regular links,
$$\otimes(n^2, k + 2)$$ has more random shortcuts

Question: What model, $\otimes(n^2, k + 2)$ or $\oplus(n, k)$, produces a more reliable network?
Small-World Network Models

- Small-World Ring model $\otimes(n, k)$

Componentwise: $\otimes(n^2, k + 2) = \otimes(n, k)$

Topologically: $\otimes(n^2, k + 2) \neq \otimes(n, k)$

$\otimes(n, k)$ has more random shortcuts.
Small-World Network Models

- Small-World Ring model $\otimes(n, k)$

Realization of $\otimes(16, 3)$

Small-World Torus model $\odot(n, k)$

Realization of $\odot(4, 1)$

Componentwise:

$\otimes(n^2, k+2) = \odot(n, k)$

Topologically:

$\otimes(n^2, k+2) \neq \odot(n, k)$

$\odot(n, k)$ has more regular links, $\otimes(n^2, k+2)$ has more random shortcuts.

Question: What model, $\otimes(n^2, k+2)$ or $\odot(n, k)$, produces a more reliable network?
Small-World Network Models

- Small-World Ring model $\otimes(n, k)$

![Small-World Ring model diagram]

Question: What model, $\otimes(n, k)$ or $\oslash(n, k)$, produces a more reliable network?
Small-World Network Models

- Small-World Ring model $\otimes(n, k)$

Realization of $\otimes(16, 3)$

Componentwise:

$\otimes(n^2, k + 2) = \otimes(n, k)$

Topologically:

$\otimes(n^2, k + 2) \neq \otimes(n, k) \otimes(n, k)$ has more regular links, $\otimes(n^2, k + 2)$ has more random shortcuts.

Question: What model, $\otimes(n^2, k + 2)$ or $\otimes(n, k)$, produces a more reliable network?
Small-World Network Models

- Small-World Ring model $\otimes(n, k)$
- Small-World Torus model $\boxtimes(n, k)$

Realization of $\otimes(16, 3)$

Componentwise: $\otimes(n^2, k+2) = \boxtimes(n, k)$

Topologically: $\otimes(n^2, k+2) \neq \boxtimes(n, k)$

has more regular links,
$\otimes(n^2, k+2)$ has more random shortcuts

Question: What model, $\otimes(n^2, k+2)$ or $\boxtimes(n, k)$, produces a more reliable network?
Small-World Network Models

- Small-World Ring model $\otimes(n, k)$
- Small-World Torus model $\boxtimes(n, k)$

Realization of $\otimes(16, 3)$

Realization of $\boxtimes(4, 1)$

Componentwise: $\otimes(n^2, k + 2) = \boxtimes(n, k)$

Topologically: $\otimes(n^2, k + 2) \neq \boxtimes(n, k)$ has more regular links, $\otimes(n^2, k + 2)$ has more random shortcuts.

Question: What model, $\otimes(n^2, k + 2)$ or $\boxtimes(n, k)$, produces a more reliable network?
Small-World Network Models

- Small-World Ring model $\otimes(n, k)$
- Small-World Torus model $\boxtimes(n, k)$

Realization of $\otimes(16, 3)$

Componentwise: $\otimes(n^2, k + 2) = \boxtimes(n, k)$

Topologically: $\otimes(n^2, k + 2) \neq \boxtimes(n, k)$ has more regular links, $\otimes(n^2, k + 2)$ has more random shortcuts

Question: What model, $\otimes(n^2, k + 2)$ or $\boxtimes(n, k)$, produces a more reliable network?
Small-World Network Models

- Small-World Ring model $\otimes(n, k)$
  
  Realization of $\otimes(16, 3)$

- Small-World Torus model $\boxtimes(n, k)$
  
  Topologically: $\otimes(n, 2k + 2) \neq \boxtimes(n, k)$
  
  Kolmogorov-Chaitin theory of $\boxtimes(n, k)$:
  $\otimes(n, k) + 2 \leq \text{size}(\boxtimes(n, k))$
Small-World Network Models

- Small-World Ring model $\otimes(n, k)$

\[ \text{Realization of } \otimes(16, 3) \]

- Small-World Torus model $\boxtimes(n, k)$

\[ \text{Realization of } \boxtimes(4, 1) \]

Componentwise:

\[ \otimes(n^2, k+2) = \boxtimes(n, k) \]

Topologically:

\[ \otimes(n^2, k+2) \neq \boxtimes(n, k) \boxtimes(n, k) \]

has more regular links,

\[ \otimes(n^2, k+2) \]

has more random shortcuts

Question: What model, $\otimes(n^2, k+2)$ or $\boxtimes(n, k)$, produces a more reliable network?
Small-World Network Models

- Small-World Ring model $\otimes(n, k)$
- Small-World Torus model $\boxtimes(n, k)$

Realization of $\otimes(16, 3)$

Componentwise:

$\otimes(n^2, k+2) = \boxtimes(n, k)$

Topologically:

$\otimes(n^2, k+2) \neq \boxtimes(n, k)$

has more regular links,

$\otimes(n^2, k+2)$ has more random shortcuts

Question:

What model, $\otimes(n^2, k+2)$ or $\boxtimes(n, k)$, produces a more reliable network?
Small-World Network Models

- Small-World Ring model $\otimes(n, k)$

  ![Realization of $\otimes(16, 3)$](image1.png)

- Small-World Torus model $\boxtimes(n, k)$

  ![Realization of $\boxtimes(4, 1)$](image2.png)

Question: What model, $\otimes(n, k)$ or $\boxtimes(n, k)$, produces a more reliable network?
Small-World Network Models

- Small-World Ring model $\otimes(n, k)$
- Small-World Torus model $\boxtimes(n, k)$

Realization of $\otimes(16, 3)$

Realization of $\boxtimes(4, 1)$

Componentwise: $\otimes(n^2, k + 2) = \boxtimes(n, k)$
Small-World Network Models

- Small-World Ring model $\otimes(n, k)$

Realization of $\otimes(16, 3)$

Componentwise: $\otimes(n^2, k + 2) = \boxtimes(n, k)$

- Small-World Torus model $\boxtimes(n, k)$

Realization of $\boxtimes(4, 1)$

Topologically: $\otimes(n^2, k + 2) \neq \boxtimes(n, k)$
Small-World Network Models

- Small-World Ring model $\otimes(n, k)$
- Small-World Torus model $\boxtimes(n, k)$

Componentwise: $\otimes(n^2, k + 2) = \boxtimes(n, k)$

Topologically: $\otimes(n^2, k + 2) \neq \boxtimes(n, k)$

$\boxtimes(n, k)$ has more regular links, $\otimes(n^2, k + 2)$ has more random shortcuts
Small-World Network Models

- Small-World Ring model $\otimes(n, k)$
- Small-World Torus model $\boxtimes(n, k)$

Componentwise: $\otimes(n^2, k + 2) = \boxtimes(n, k)$

Topologically: $\otimes(n^2, k + 2) \neq \boxtimes(n, k)$

$\boxtimes(n, k)$ has more regular links, $\otimes(n^2, k + 2)$ has more random shortcuts

Question: What model, $\otimes(n^2, k + 2)$ or $\boxtimes(n, k)$, produces a more reliable network?
Why These Models?
Why These Models?

- “Realizations” of $\otimes(n, k)$
Why These Models?

- “Realizations” of $\mathfrak{G}(n, k)$
How to Compare Two Network Models?

Given a network realization \( \hat{\otimes} \times \sim \times (n^2, k + 2) \) a source-sink pair \( (a, b) \) the critical threshold \( \mu^* \) we can estimate the failure probability \( p_F(\hat{\otimes}; (a, b); \mu^*) \) using Subset Simulation.

The expected failure probability for a given threshold \( \mu^* \) for the SW-ring model:

\[
\bar{p}_F(\hat{\otimes}, (a, b)) \approx \frac{1}{M} \sum_{i=1}^{M} p_F(\hat{\otimes}_i; (a_i, b_i); \mu^*)
\]

Subset Simulation

Similarly for the SW-torus model:

\[
\bar{p}_F(\hat{\nabla}, (a, b)) \approx \frac{1}{M} \sum_{i=1}^{M} p_F(\hat{\nabla}_i; (a_i, b_i); \mu^*)
\]

Subset Simulation

\( \hat{\nabla}_i \sim \nabla (n, k) (a_i, b_i) \) are chosen uniformly at random
How to Compare Two Network Models?

Given

- a network realization $\hat{\otimes} \sim \otimes(n^2, k + 2)$
- a source-sink pair $(a, b)$
- the critical threshold $\mu^*$
How to Compare Two Network Models?

Given

- a network realization $\hat{\otimes} \sim \otimes(n^2, k + 2)$
- a source-sink pair $(a, b)$
- the critical threshold $\mu^*$

we can estimate the failure probability $p_F(\hat{\otimes}; (a, b); \mu^*)$ using Subset Simulation.
How to Compare Two Network Models?

Given

- a network realization \( \hat{\otimes} \sim \otimes(n^2, k + 2) \)
- a source-sink pair \((a, b)\)
- the critical threshold \(\mu^*\)

we can estimate the failure probability \(p_F(\hat{\otimes}; (a, b); \mu^*)\) using Subset Simulation.

The expected failure probability for a given threshold \(\mu^*\) for the SW-ring model:

\[
\bar{p}_{F,\otimes}(\mu^*) = \mathbb{E}_{\otimes,(a,b)}[p_F(\hat{\otimes}; (a, b); \mu^*)]
\]
How to Compare Two Network Models?

Given

- a network realization \( \hat{\otimes} \sim \otimes(n^2, k + 2) \)
- a source-sink pair \((a, b)\)
- the critical threshold \(\mu^*\)

we can estimate the failure probability \(p_F(\hat{\otimes}; (a, b); \mu^*)\) using Subset Simulation.

The expected failure probability for a given threshold \(\mu^*\) for the SW-ring model:

\[
\bar{p}_{F, \otimes}(\mu^*) = \mathbb{E}_{\otimes, (a, b)}[p_F(\hat{\otimes}; (a, b); \mu^*)] \approx \frac{1}{M} \sum_{i=1}^{M} p_F(\hat{\otimes}_i; (a_i, b_i); \mu^*)
\]

Subset Simulation
How to Compare Two Network Models?

Given

- a network realization \( \hat{\otimes} \sim \otimes(n^2, k + 2) \)
- a source-sink pair \((a, b)\)
- the critical threshold \(\mu^*\)

we can estimate the failure probability \(p_F(\hat{\otimes}; (a, b); \mu^*)\) using Subset Simulation. The expected failure probability for a given threshold \(\mu^*\) for the SW-ring model:

\[
\bar{p}_{F, \otimes}(\mu^*) = \mathbb{E}_{\otimes, (a, b)}[p_F(\hat{\otimes}; (a, b); \mu^*)] \approx \frac{1}{M} \sum_{i=1}^{M} p_F(\hat{\otimes}_i; (a_i, b_i); \mu^*)
\]

Similarly for the SW-torus model:

\[
\bar{p}_{F, \boxtimes}(\mu^*) = \mathbb{E}_{\boxtimes, (a, b)}[p_F(\hat{\boxtimes}; (a, b); \mu^*)] \approx \frac{1}{M} \sum_{i=1}^{M} p_F(\hat{\boxtimes}_i; (a_i, b_i); \mu^*)
\]

- \(\hat{\boxtimes}_i \sim \boxtimes(n, k)\)
- \((a_i, b_i)\) are chosen uniformly at random
How to Compare Two Network Models?

We are interested in the relative behavior of $\bar{p}_{\mathcal{F},\otimes}(\mu^*)$ and $\bar{p}_{\mathcal{F},\boxdot}(\mu^*)$. If we plot $\bar{p}_{\mathcal{F},\otimes}$ vs $\bar{p}_{\mathcal{F},\boxdot}$ treating $\mu^*$ as a parameter, we obtain a curve that lies in the unit square starting at $(0,0)$ and ending at $(1,1)$. We refer to this curve as the relative reliability curve.
How to Compare Two Network Models?

We are interested in the relative behavior of $\tilde{p}_F, \otimes(\mu^*)$ and $\tilde{p}_F, \boxdot(\mu^*)$.

If we plot $\tilde{p}_F, \otimes$ vs $\tilde{p}_F, \boxdot$ treating $\mu^*$ as a parameter, we obtain a curve that
How to Compare Two Network Models?

We are interested in the relative behavior of $\bar{p}_\mathcal{F},\otimes(\mu^*)$ and $\bar{p}_\mathcal{F},\boxdot(\mu^*)$.

If we plot $\bar{p}_\mathcal{F},\otimes$ vs $\bar{p}_\mathcal{F},\boxdot$ treating $\mu^*$ as a parameter, we obtain a curve that

- lies in the unit square
How to Compare Two Network Models?

We are interested in the relative behavior of $\bar{p}_F, \otimes (\mu^*)$ and $\bar{p}_F, \boxdot (\mu^*)$.

If we plot $\bar{p}_F, \otimes$ vs $\bar{p}_F, \boxdot$ treating $\mu^*$ as a parameter, we obtain a curve that

- lies in the unit square
- starts at $(0, 0)$
How to Compare Two Network Models?

We are interested in the relative behavior of $\bar{p}_F, \otimes (\mu^*)$ and $\bar{p}_F, \boxtimes (\mu^*)$.

If we plot $\bar{p}_F, \otimes$ vs $\bar{p}_F, \boxtimes$ treating $\mu^*$ as a parameter, we obtain a curve that

- lies in the unit square
- starts at $(0, 0)$
- ends at $(1, 1)$
How to Compare Two Network Models?

We are interested in the relative behavior of $\bar{p}_F, \otimes (\mu^*)$ and $\bar{p}_F, \bigotimes (\mu^*)$.

If we plot $\bar{p}_F, \otimes$ vs $\bar{p}_F, \bigotimes$ treating $\mu^*$ as a parameter, we obtain a curve that

- lies in the unit square
- starts at $(0, 0)$
- ends at $(1, 1)$

We refer to this curve as the relative reliability curve.
How to Compare Two Network Models?

We are interested in the relative behavior of $\bar{p}_{\mathcal{F}, \otimes}(\mu^*)$ and $\bar{p}_{\mathcal{F}, \boxdot}(\mu^*)$.

If we plot $\bar{p}_{\mathcal{F}, \otimes}$ vs $\bar{p}_{\mathcal{F}, \boxdot}$ treating $\mu^*$ as a parameter, we obtain a curve that

- lies in the unit square
- starts at $(0, 0)$
- ends at $(1, 1)$

We refer to this curve as the relative reliability curve.

Schematic picture:
How to Compare Two Network Models?

We are interested in the relative behavior of $\bar{p}_F, \otimes (\mu^*)$ and $\bar{p}_F, \boxtimes (\mu^*)$.

If we plot $\bar{p}_F, \otimes$ vs $\bar{p}_F, \boxtimes$ treating $\mu^*$ as a parameter, we obtain a curve that

- lies in the unit square
- starts at $(0, 0)$
- ends at $(1, 1)$

We refer to this curve as the relative reliability curve.

Schematic picture:
How to Compare Two Network Models?

We are interested in the relative behavior of $\bar{p}_{\mathcal{F}, \otimes}(\mu^*)$ and $\bar{p}_{\mathcal{F}, \boxdot}(\mu^*)$.

If we plot $\bar{p}_{\mathcal{F}, \otimes}$ vs $\bar{p}_{\mathcal{F}, \boxdot}$ treating $\mu^*$ as a parameter, we obtain a curve that

- lies in the unit square
- starts at $(0, 0)$
- ends at $(1, 1)$

We refer to this curve as the relative reliability curve.

Schematic picture:
Simulation Results

The SW-torus model produces a more reliable network than the SW-ring model as $k$ increases, the relative reliability curve shifts towards the equal reliability line $\bar{p}_{F,\otimes} \approx (\bar{p}_{F,\otimes}, \otimes)$. $\alpha > 1 \Rightarrow$ when both $\bar{p}_{F,\otimes}$ and $\bar{p}_{F,\otimes}$ are small, the SW-torus model produces a substantially more reliable network than the SW-ring model. $(n,k)$ is more efficient than $(n^2,k+2)$.
Simulation Results

The SW-torus model produces a more reliable network than the SW-ring model as $k$ increases, the relative reliability curve shifts towards the equal reliability line $\bar{p}_{F,\otimes} \approx (\bar{p}_{F,\otimes}, \otimes)$. When both $\bar{p}_{F,\otimes}$ and $\bar{p}_{F,\otimes}$ are small, the SW-torus model produces a substantially more reliable network than the SW-ring model.

$\oplus (n,k)$ is more efficient than $\otimes (n^2, k + 2)$.
The SW-torus model produces a more reliable network than the SW-ring model.
1. The SW-torus model produces a more reliable network than the SW-ring model.

2. As $k$ increases, the relative reliability curve shifts towards the equal reliability line.
The SW-torus model produces a more reliable network than the SW-ring model.

As $k$ increases, the relative reliability curve shifts towards the equal reliability line.

$\bar{p}_{\mathcal{F},\bigotimes} \approx (\bar{p}_{\mathcal{F},\otimes})^\alpha, \ \alpha > 1$
Simulation Results

1. The **SW-torus model** produces a more reliable network than the **SW-ring model**

2. As $k$ increases, the relative reliability curve shifts towards the equal reliability line

3. $\bar{p}_F,\boxtimes \approx (\bar{p}_F,\otimes)^\alpha$, $\alpha > 1 \Rightarrow$ when both $\bar{p}_F,\boxtimes$ and $\bar{p}_F,\otimes$ are small, the **SW-torus model** produces a substantially more reliable network than the **SW-ring model**
**Simulation Results**

1. The **SW-torus model** produces a more reliable network than the **SW-ring model**

2. As $k$ increases, the relative reliability curve shifts towards the equal reliability line

3. $\bar{p}_{F,\boxtimes} \approx (\bar{p}_{F,\otimes})^\alpha$, $\alpha > 1$ $\Rightarrow$ when both $\bar{p}_{F,\boxtimes}$ and $\bar{p}_{F,\otimes}$ are small, the **SW-torus model** produces a substantially more reliable network than the **SW-ring model**
Future Research: Directions and Problems

Network Reliability Estimation

▶ How far the Subset Simulation method can be pushed?
| V | ∼ 10^3, 10^4 is ok, what about | V | ∼ 10^5, 10^6? |

▶ Develop a "discrete" analog of Subset Simulation for solving 2-terminal, all-terminal, and other "discrete" network reliability problems.
▶ Apply these methods to real infrastructure networks (Boston highways network, Bursa water distribution network, Kobe water distribution network).

Cascading Failures in Technological Networks

▶ Develop a conceptual framework for analysis of cascading failures. Adopt models of cascades developed for social and economic networks (e.g. the Watts model).
▶ Study the correlation between network reliability and network topology in the presence of cascading failures.

Statistical Analysis of Network Data

▶ Adopt the Bayesian approach for model selection and inferring the model parameters.
▶ Fit the Stochastic Kronecker graph model to real network data (California road network, US air transportation network, US western states power grid).
Future Research: Directions and Problems

1. Network Reliability Estimation
   - How far the Subset Simulation method can be pushed?
     \[ |V| \sim 10^3, 10^4 \text{ is ok, what about } |V| \sim 10^5, 10^6? \]
Future Research: Directions and Problems

1 Network Reliability Estimation

▶ How far the Subset Simulation method can be pushed?
   \[ |V| \sim 10^3, 10^4 \text{ is ok, what about } |V| \sim 10^5, 10^6? \]

▶ Develop a “discrete” analog of Subset Simulation for solving 2-terminal, all-terminal, and other “discrete” network reliability problems.
Future Research: Directions and Problems

1. Network Reliability Estimation

- How far the Subset Simulation method can be pushed?
  \(|V| \sim 10^3, 10^4\) is ok, what about \(|V| \sim 10^5, 10^6|\)?

- Develop a “discrete” analog of Subset Simulation for solving 2-terminal, all-terminal, and other “discrete” network reliability problems.

- Apply these methods to real infrastructure networks (Boston highways network, Bursa water distribution network, Kobe water distribution network).
Future Research: Directions and Problems

1. Network Reliability Estimation
   ▶ How far the Subset Simulation method can be pushed?
     \[ |V| \sim 10^3, 10^4 \text{ is ok, what about } |V| \sim 10^5, 10^6? \]
   ▶ Develop a “discrete” analog of Subset Simulation for solving 2-terminal, all-terminal, and other “discrete” network reliability problems.
   ▶ Apply these methods to real infrastructure networks (Boston highways network, Bursa water distribution network, Kobe water distribution network).

2. Cascading Failures in Technological Networks

3. Statistical Analysis of Network Data
   ▶ Adopt the Bayesian approach for model selection and inferring the model parameters.
   ▶ Fit the Stochastic Kronecker graph model to real network data (California road network, US air transportation network, US western states power grid).
Future Research: Directions and Problems

1. **Network Reliability Estimation**
   - How far the Subset Simulation method can be pushed?
     \[ |V| \sim 10^3, 10^4 \text{ is ok, what about } |V| \sim 10^5, 10^6? \]
   - Develop a “discrete” analog of Subset Simulation for solving 2-terminal, all-terminal, and other “discrete” network reliability problems.
   - Apply these methods to real infrastructure networks (Boston highways network, Bursa water distribution network, Kobe water distribution network).

2. **Cascading Failures in Technological Networks**
   - Develop a conceptual framework for analysis of cascading failures. Adopt models of cascades developed for social and economic networks (e.g. the Watts model).
Future Research: Directions and Problems

1. Network Reliability Estimation
   - How far the Subset Simulation method can be pushed?
     \[ |V| \sim 10^3, 10^4 \text{ is ok, what about } |V| \sim 10^5, 10^6? \]
   - Develop a “discrete” analog of Subset Simulation for solving 2-terminal, all-terminal, and other “discrete” network reliability problems.
   - Apply these methods to real infrastructure networks (Boston highways network, Bursa water distribution network, Kobe water distribution network).

2. Cascading Failures in Technological Networks
   - Develop a conceptual framework for analysis of cascading failures. Adopt models of cascades developed for social and economic networks (e.g. the Watts model).
   - Study the correlation between network reliability and network topology in the presence of cascading failures.
Future Research: Directions and Problems

1. Network Reliability Estimation
   - How far the Subset Simulation method can be pushed?
     \[ |V| \sim 10^3, 10^4 \text{ is ok, what about } |V| \sim 10^5, 10^6? \]
   - Develop a “discrete” analog of Subset Simulation for solving 2-terminal, all-terminal, and other “discrete” network reliability problems.
   - Apply these methods to real infrastructure networks (Boston highways network, Bursa water distribution network, Kobe water distribution network).

2. Cascading Failures in Technological Networks
   - Develop a conceptual framework for analysis of cascading failures. Adopt models of cascades developed for social and economic networks (e.g. the Watts model).
   - Study the correlation between network reliability and network topology in the presence of cascading failures.

3. Statistical Analysis of Network Data
   - Adopt the Bayesian approach for model selection and inferring the model parameters.
   - Fit the Stochastic Kronecker graph model to real network data (California road network, US air transportation network, US western states power grid).
Future Research: Directions and Problems

1. **Network Reliability Estimation**
   - How far the Subset Simulation method can be pushed?
     \[|V| \sim 10^3, 10^4 \text{ is ok, what about } |V| \sim 10^5, 10^6?\]
   - Develop a “discrete” analog of Subset Simulation for solving 2-terminal, all-terminal, and other “discrete” network reliability problems.
   - Apply these methods to real infrastructure networks (Boston highways network, Bursa water distribution network, Kobe water distribution network).

2. **Cascading Failures in Technological Networks**
   - Develop a conceptual framework for analysis of cascading failures. Adopt models of cascades developed for social and economic networks (e.g. the Watts model).
   - Study the correlation between network reliability and network topology in the presence of cascading failures.

3. **Statistical Analysis of Network Data**
   - Adopt the Bayesian approach for model selection and inferring the model parameters.
Future Research: Directions and Problems

1. **Network Reliability Estimation**
   - How far the Subset Simulation method can be pushed?  
     \[ |V| \sim 10^3, 10^4 \text{ is ok, what about } |V| \sim 10^5, 10^6? \]
   - Develop a “discrete” analog of Subset Simulation for solving 2-terminal, all-terminal, and other “discrete” network reliability problems.
   - Apply these methods to real infrastructure networks (Boston highways network, Bursa water distribution network, Kobe water distribution network).

2. **Cascading Failures in Technological Networks**
   - Develop a conceptual framework for analysis of cascading failures. Adopt models of cascades developed for social and economic networks (e.g. the Watts model).
   - Study the correlation between network reliability and network topology in the presence of cascading failures.

3. **Statistical Analysis of Network Data**
   - Adopt the Bayesian approach for model selection and inferring the model parameters.
   - Fit the Stochastic Kronecker graph model to real network data (California road network, US air transportation network, US western states power grid).
Future Conferences

- **ICOSSAR 2013**, 16-20 June, Columbia University, New York, USA
  - Session: “Reliability of Complex Systems and Networks”

- **EURODYN 2014**, 30 June - 2 July, University of Porto, Porto Portugal
  - Mini-Symposium: “Bayesian updating, filtering and inversion for dynamic systems”
  - Organizers: J.L. Beck, A. Taflanidis, K.M. Zuev

- **ASCE-ICVRAM-ISUMA 2014**, 13-16 July, University of Liverpool, UK
  - Organizers: K.M. Zuev, J.L. Beck, E. Zio