A Monte Carlo Approach to Stability Verification

Brendan Patch
The University of Queensland
University of Amsterdam

Joint work with:
Neil Walton
Michel Mandjes

ANZAPW
April 2015
Example: M/M/1 Queue

Work arrives to the system according to a Poisson process of rate $\lambda$ and has service time $\text{Exp}(\mu)$. 
Example: M/M/1 Queue

Work arrives to the system according to a Poisson process of rate $\lambda$ and has service time $\text{Exp}(1)$.  

\[
\lambda \\
\rightarrow
\]

Number in System

| 3 |
| 2 |
| 1 |

$t$
Work arrives to the system according to a Poisson process of rate $\lambda$ and has service time $\text{Exp}(1)$. 

Example: M/M/1 Queue
Example: M/M/1 Queue

Work arrives to the system according to a Poisson process of rate $\lambda$ and has service time $\text{Exp}(1)$.
Example: M/M/1 Queue

Work arrives to the system according to a Poisson process of rate $\lambda$ and has service time $\text{Exp}(1)$.
Example: M/M/1 Queue

Work arrives to the system according to a Poisson process of rate $\lambda$ and has service time $\text{Exp}(1)$.
Example: M/M/1 Queue

Work arrives to the system according to a Poisson process of rate $\lambda$ and has service time $\text{Exp}(1)$. 

$\lambda$
Example: M/M/1 Queue

Work arrives to the system according to a Poisson process of rate $\lambda$ and has service time $\text{Exp}(1)$.
Sample Paths of the M/M/1 Queue

Figure: $X(t)$ is the number in the queue at time $t$ and $\mu = 1$. 
Sample Paths of the M/M/1 Queue

Figure: $X(t)$ is the number in the queue at time $t$ and $\mu = 1$. 

- $\lambda = 0.5$
- $\lambda = 2$
- $\lambda = 1.05$
Sample Paths of the M/M/1 Queue

Figure: $X(t)$ is the number in the queue at time $t$ and $\mu = 1$. 
Figure: $X(t)$ is the number in the queue at time $t$ and $\mu = 1$. 

Sample Paths of the M/M/1 Queue
Sample Paths of the M/M/1 Queue

Figure: $X(t)$ is the number in the queue at time $t$ and $\mu = 1$. Different $\lambda$ implies different behaviour.
Sample Paths of the M/M/1 Queue

Figure: $X(t)$ is the number in the queue at time $t$ and $\mu = 1$.

- Different $\lambda$ implies different behaviour.
For state space $\mathcal{X} \subset \mathbb{N}^J$ study $f : \mathcal{X} \rightarrow [0, \infty)$.

E.g. $f(x) = \sum_i x(i) = |x|$

Assume $X(\cdot)$ has bounded increments.
Definition of Stability for a Parameter value

The parameter $\lambda$ is **stable** if there exists $\delta > 0$, $\tau_x > 0$, and $\kappa > 0$ such that

$$\mathbb{E} \left[ |X_\lambda(\tau_x)| - |X_\lambda(0)| \bigg| X_\lambda(0) = x \right] \leq -\delta \tau_x$$

for all $x$ such that $|x| \geq \kappa$. 

\[ \bullet X_\lambda(0) \]
Definition of Stability for a Parameter value

The parameter $\lambda$ is **stable** if there exists $\delta > 0$, $\tau_x > 0$, and $\kappa > 0$ such that

$$\mathbb{E} \left[ |X_\lambda(\tau_x)| - |X_\lambda(0)| \mid X_\lambda(0) = x \right] \leq -\delta \tau_x$$

for all $x$ such that $|x| \geq \kappa$. 
Definition of Stability for a Parameter value

The parameter $\lambda$ is **stable** if there exists $\delta > 0$, $\tau_x > 0$, and $\kappa > 0$ such that

$$\mathbb{E} \left[ |X_\lambda(\tau_x)| - |X_\lambda(0)| \middle| X_\lambda(0) = x \right] \leq -\delta \tau_x$$

for all $x$ such that $|x| \geq \kappa$. 
The parameter $\lambda$ is **stable** if there exists $\delta > 0$, $\tau_x > 0$, and $\kappa > 0$ such that

$$\mathbb{E} \left[ |X_\lambda(\tau_x)| - |X_\lambda(0)| \mid X_\lambda(0) = x \right] \leq -\delta \tau_x$$

for all $x$ such that $|x| \geq \kappa$. 

**Definition of Stability for a Parameter value**
The parameter \( \lambda \) is **UNSTABLE** if there exists \( \delta > 0 \), \( \tau_x > 0 \), and \( \kappa > 0 \) such that

\[
\mathbb{E} \left[ |X_{\lambda}(\tau_x)| - |X_{\lambda}(0)| \, \mid \, X_{\lambda}(0) = x \right] \geq \delta \tau_x
\]

for all \( x \) such that \( |x| \geq \kappa \).
• The parameter set $\mathcal{L}$ is unstable if there is a subset $\tilde{\mathcal{L}} \subset \mathcal{L}$ of positive Lebesgue measure such that all $\lambda \in \tilde{\mathcal{L}}$ are unstable.

• The parameter set is stable if all $\lambda \in \mathcal{L}$ are stable.

• Not stable is not necessarily unstable!
Stability of M/M/1 Queue

• For $x > \tau$

\[
\mathbb{E} \left[ X(\tau) - X(0) \mid X(0) = x \right] = (\lambda - \mu) \tau,
\]

and the jumps are bounded. Hence $\lambda > \mu$ is an unstable parameter set under our definition.
If a parameter set contains a measurable amount of the red region then it is unstable.
Stability of M/M/1 Queue

If a parameter set contains a measurable amount of the red region then it is unstable.

• Is $\mathcal{L} = \{(\lambda, \mu) : \lambda \in [0, 1], \mu \in [2, 3]\}$ unstable?
Stability of M/M/1 Queue

If a parameter set contains a measurable amount of the red region then it is unstable.

\[ \lambda, \mu \]

Is \( L = \{ (\lambda, \mu) : \lambda \in [0, 1], \mu \in [2, 3] \} \) unstable? NO, it is stable, \( \lambda < \mu \) for all \( (\lambda, \mu) \in L \).
Stability of M/M/1 Queue

If a parameter set contains a measurable amount of the red region then it is unstable.

- Is $L = \{(\lambda, \mu) : \lambda \in [0, 1], \mu \in [2, 2]\}$ unstable? NO, it is stable, $\lambda < \mu$ for all $(\lambda, \mu) \in L$.
- Is $L = \{(\lambda, \mu) : \lambda \in [1, 2], \mu \in [0.5, 2]\}$ unstable?
Stability of M/M/1 Queue

If a parameter set contains a measurable amount of the red region then it is unstable.

- Is $\mathcal{L} = \{ (\lambda, \mu) : \lambda \in [0, 1], \mu \in [2, 2] \}$ unstable? NO, it is stable, $\lambda < \mu$ for all $(\lambda, \mu) \in \mathcal{L}$.
- Is $\mathcal{L} = \{ (\lambda, \mu) : \lambda \in [1, 2], \mu \in [0.5, 2] \}$ unstable? YES. $\lambda > \mu$ for a measurable amount of $(\lambda, \mu) \in \mathcal{L}$. 
Sometimes an analytical result is more difficult.

- E.g. Dai 1995
- Three parameters, FCFS
A goal of my PhD is to develop algorithms that:

- Give the stability region of complex queueing networks.
- Are substantially more efficient than brute force.
- Are proven to work.
A goal of my PhD is to develop algorithms that:

- Give the stability region of complex queueing networks.
- Are substantially more efficient than brute force.
- Are proven to work.

Let’s see what we have so far...
Not all unstable parameters “equally” unstable

- Find most “unstable” $\lambda \in \mathcal{L}$ then test for stability.
- Optimisation problem.
- Simulated annealing?
M/M/1 Instability

- Find most "unstable" $\lambda \in \mathcal{L}$ then test for stability.
- Optimisation problem.
- Simulated annealing?
Algorithm: \( S_k = (Y_k, \Lambda_k) \) \( k=1,2,... \) attains values in \( (X, L) \).

1. Given \( S_k = (x, \lambda) \) we sample \( \gamma \) uniformly from \( L \).
2. Let \( X_\gamma(t) \) run until \( \tau_x \), with \( X_\gamma(0) = x \). Set \( y = X_\gamma(\tau_x) \).
3. If \( |y| > |x| \) then set \( S_{k+1} = (y, \gamma) \); otherwise, set

\[
S_{k+1} = \begin{cases} (y, \gamma) & \text{with probability } e^{\eta(|y| - |x|)}, \\ (x, \lambda) & \text{otherwise}. \end{cases}
\]
Stability Verification Algorithm

Algorithm: $S_k = (Y_k, \Lambda_k)_{k=1,2,\ldots}$ attains values in $(\mathcal{X}, \mathcal{L})$.

1. Given $S_k = (x, \lambda)$ we sample $\gamma$ uniformly from $\mathcal{L}$.
2. Let $X_{\gamma}(t)$ run until $\tau_x$, with $X_{\gamma}(0) = x$. Set $y = X_{\gamma}(\tau_x)$.
3. If $|y| > |x|$ then set $S_{k+1} = (y, \gamma)$; otherwise, set

$$S_{k+1} = \begin{cases} 
(y, \gamma) & \text{with probability } e^{\eta(|y| - |x|)}, \\
(x, \lambda) & \text{otherwise.}
\end{cases}$$
Algorithm: $S_k = (Y_k, \Lambda_k)_{k=1,2,...}$ attains values in $(\mathcal{X}, \mathcal{L})$.

1. Given $S_k = (x, \lambda)$ we sample $\gamma$ uniformly from $\mathcal{L}$.
2. Let $X_{\gamma}(t)$ run until $\tau_x$, with $X_{\gamma}(0) = x$. Set $y = X_{\gamma}(\tau_x)$.
3. If $|y| > |x|$ then set $S_{k+1} = (y, \gamma)$; otherwise, set $S_{k+1} = \begin{cases} (y, \gamma) & \text{with probability } e^{\eta(|y|-|x|)}, \\ (x, \lambda) & \text{otherwise}. \end{cases}$
Stability Verification Algorithm

Algorithm: \( S_k = (Y_k, \Lambda_k)_{k=1,2,...} \) attains values in \((\mathcal{X}, \mathcal{L})\).

1. Given \( S_k = (x, \lambda) \) we sample \( \gamma \) uniformly from \( \mathcal{L} \).
2. Let \( X_\gamma(t) \) run until \( \tau_x \), with \( X_\gamma(0) = x \). Set \( y = X_\gamma(\tau_x) \).
3. If \( |y| > |x| \) then set \( S_{k+1} = (y, \gamma) \); otherwise, set

\[
S_{k+1} = \begin{cases} 
(y, \gamma) & \text{with probability } e^{\eta(|y| - |x|)}, \\
(x, \lambda) & \text{otherwise}.
\end{cases}
\]
Stability Verification Algorithm

Algorithm: \( S_k = (Y_k, \Lambda_k)_{k=1,2,...} \) attains values in \((\mathcal{X}, \mathcal{L})\).

1. Given \( S_k = (x, \lambda) \) we sample \( \gamma \) uniformly from \( \mathcal{L} \).
2. Let \( X_\gamma(t) \) run until \( \tau_x \), with \( X_\gamma(0) = x \). Set \( y = X_\gamma(\tau_x) \).
3. If \( |y| > |x| \) then set \( S_{k+1} = (y, \gamma) \); otherwise, set
   \[
   S_{k+1} = \begin{cases} 
   (y, \gamma) & \text{with probability } e^{\eta(|y| - |x|)}, \\
   (x, \lambda) & \text{otherwise.}
   \end{cases}
   \]
Stability Verification Algorithm

Algorithm: \( S_k = (Y_k, \Lambda_k)_{k=1,2,...} \) attains values in \((\mathcal{X}, \mathcal{L})\).

1. Given \( S_k = (x, \lambda) \) we sample \( \gamma \) uniformly from \( \mathcal{L} \).
2. Let \( X_\gamma(t) \) run until \( \tau_x \), with \( X_\gamma(0) = x \). Set \( y = X_\gamma(\tau_x) \).
3. If \( |y| > |x| \) then set \( S_{k+1} = (y, \gamma) \); otherwise, set

\[
S_{k+1} = \begin{cases} 
(y, \gamma) & \text{with probability } e^{\eta(|y| - |x|)}, \\
(x, \lambda) & \text{otherwise.}
\end{cases}
\]
Stability Verification Algorithm

Algorithm: $S_k = (Y_k, \Lambda_k)_{k=1,2,...}$ attains values in $(X, L)$.

1. Given $S_k = (x, \lambda)$ we sample $\gamma$ uniformly from $L$.
2. Let $X_\gamma(t)$ run until $\tau_x$, with $X_\gamma(0) = x$. Set $y = X_\gamma(\tau_x)$.
3. If $|y| > |x|$ then set $S_{k+1} = (y, \gamma)$; otherwise, set

   $S_{k+1} = \begin{cases} (y, \gamma) & \text{with probability} \quad e^{\eta(|y| - |x|)}, \\ (x, \lambda) & \text{otherwise.} \end{cases}$
Stability Verification Algorithm

Algorithm: $S_k = (Y_k, \Lambda_k)_{k=1,2,...}$ attains values in $(X, L)$.

1. Given $S_k = (x, \lambda)$ we sample $\gamma$ uniformly from $L$.
2. Let $X_\gamma(t)$ run until $\tau_x$, with $X_\gamma(0) = x$. Set $y = X_\gamma(\tau_x)$.
3. If $|y| > |x|$ then set $S_{k+1} = (y, \gamma)$; otherwise, set

$$S_{k+1} = \begin{cases} (y, \gamma) & \text{with probability } e^{\eta(|y| - |x|)}, \\ (x, \lambda) & \text{otherwise.} \end{cases}$$
Algorithm: $S_k = (Y_k, \Lambda_k)_{k=1,2,...}$ attains values in $(X, L)$.

1. Given $S_k = (x, \lambda)$ we sample $\gamma$ uniformly from $L$.
2. Let $X_\gamma(t)$ run until $\tau_x$, with $X_\gamma(0) = x$. Set $y = X_\gamma(\tau_x)$.
3. If $|y| > |x|$ then set $S_{k+1} = (y, \gamma)$; otherwise, set

$$S_{k+1} = \begin{cases} (y, \gamma) & \text{with probability } e^{\eta(|y| - |x|)} , \\ (x, \lambda) & \text{otherwise.} \end{cases}$$
Stability Verification Algorithm

Algorithm: $S_k = (Y_k, \Lambda_k)_{k=1,2,...}$ attains values in $(X, L)$.

1. Given $S_k = (x, \lambda)$ we sample $\gamma$ uniformly from $L$.
2. Let $X_\gamma(t)$ run until $\tau_x$, with $X_\gamma(0) = x$. Set $y = X_\gamma(\tau_x)$.
3. If $|y| > |x|$ then set $S_{k+1} = (y, \gamma)$; otherwise, set

$$S_{k+1} = \begin{cases} (y, \gamma) & \text{with probability } e^{\eta(|y| - |x|)}, \\ (x, \lambda) & \text{otherwise.} \end{cases}$$
Stability Verification Algorithm

Algorithm:  \( S_k = (Y_k, \Lambda_k) \) \( k=1,2,\ldots \) attains values in \((X, \mathcal{L})\).

1. Given \( S_k = (x, \lambda) \) we sample \( \gamma \) uniformly from \( \mathcal{L} \).
2. Let \( X_\gamma(t) \) run until \( \tau_x \), with \( X_\gamma(0) = x \). Set \( y = X_\gamma(\tau_x) \).
3. If \( |y| > |x| \) then set \( S_{k+1} = (y, \gamma) \); otherwise, set

\[
S_{k+1} = \begin{cases} 
(y, \gamma) & \text{with probability } e^{\eta(|y| - |x|)}, \\
(x, \lambda) & \text{otherwise.}
\end{cases}
\]
Stability Verification Algorithm

Algorithm: \( S_k = (Y_k, \Lambda_k)_{k=1,2,...} \) attains values in \((\mathcal{X}, \mathcal{L})\).

1. Given \( S_k = (x, \lambda) \) we sample \( \gamma \) uniformly from \( \mathcal{L} \).
2. Let \( X_\gamma(t) \) run until \( \tau_x \), with \( X_\gamma(0) = x \). Set \( y = X_\gamma(\tau_x) \).
3. If \(|y| > |x|\) then set \( S_{k+1} = (y, \gamma) \); otherwise, set

\[
S_{k+1} = \begin{cases} 
(y, \gamma) & \text{with probability } e^{\eta(|y| - |x|)}, \\
(x, \lambda) & \text{otherwise.}
\end{cases}
\]
Stability Verification Algorithm

- Asymptotically (or sooner) accept only unstable samples (if they are there).
- Detect any unstable drift.
• Asymptotically (or sooner) accept only unstable samples (if they are there).
• Detect any unstable drift.
Stability Verification Algorithm

- Asymptotically (or sooner) accept only unstable samples (if they are there).
- Detect any unstable drift.
• Asymptotically (or sooner) accept only unstable samples (if they are there).
• Detect any unstable drift.
Stability Verification Algorithm

- Asymptotically (or sooner) accept only unstable samples (if they are there).
- Detect any unstable drift.
• Asymptotically (or sooner) accept only unstable samples (if they are there).
• Detect any unstable drift.
Theorem 1

If the set $\mathcal{L}$ is stable then

$$
\liminf_{k \to \infty} \frac{|Y_k|}{k} = 0.
$$

Theorem 2

If the set $\mathcal{L}$ is unstable then

$$
\liminf_{k \to \infty} \frac{|Y_k|}{k} > 0.
$$
Example

$\lambda = 1$

$\mu_1$

$\mu_3$

$\mu_1$

$\mu_1$

$\mu_1$

$\mu_2$
Figure: Unstable $\mathcal{L}$?
Figure: Stable $\mathcal{L}$?
• Hypothesis test for stability
• Adapting algorithm to estimate unstable region
• More interesting test cases and real world tools.
• Other Monte Carlo methods: Cross Entropy, Particle Filters, etc.

To be continued...