Markov Chain Monte Carlo Models, Gibbs Sampling, & Metropolis Algorithm for High-Dimensionality Complex Stochastic Problems: Applications in Network and Computer Security

Yogesh Malhotra, PhD

www.yogeshmalhotra.com

Griffiss Cyberspace, Global Risk Management Network, LLC
306 Market St., Griffiss Air Force Base, Rome, NY 13441, U.S.A.
www.FinRM.org

May 8, 2014

Recognized as one of top-10 computing algorithms with underlying research among top-3 mathematics papers, the impact of Markov Chain Monte Carlo Models across diverse fields including computer science, physics, statistics, finance, economics, and engineering is evident. A paper on its history written by top statisticians notes that: "the development of this methodology has not only changed our solutions to problems, but has changed the way we think about problems." Markov Chain Monte Carlo Models methods originally conceptualized in 1940s at the Los Alamos National Lab during World War II led to the Metropolis algorithm. Markov Chain Monte Carlo Models resulted from the research by the same group of research scientists as those working on the atomic bomb including Stanislaw Ulam and John von Neumann at Los Alamos who around the same time had also created Monte Carlo (MC) methods. John von Neumann was using Monte Carlo to study thermonuclear and fission problems in the late 1940s after the first computer, ENIAC, was developed.

Markov Chain Monte Carlo Models, Gibbs Sampling, & Metropolis Algorithm for High-Dimensionality Complex Stochastic Problems: Applications in Network and Computer Security

Abstract

Markov chain Monte Carlo (MCMC) methods have an important role in solving high-dimensionality stochastic problems characterized by computational complexity. Given their critical importance, there is need for network and security risk management research to relate the MCMC quantitative methodological concerns with network and security risk applications. This article contributes to that research stream. The core quantitative methodological focus of the article is on Monte Carlo Models and MCMC Algorithms, Gibbs Sampling and Metropolis-Hastings Algorithm. Network and security risk management application focus is on how MCMC methods help solve previously unsolvable problems in computational statistical modeling of cryptography, cryptanalytics, and penetration testing; intrusion detection & prevention and anomaly detection; and, privacy in anonymity systems and social networks. Future quantitative methods applied research and development in MCMC and computational statistical computing to address systemic risk and model risk management is recommended.

1. Introduction: Markov chain Monte Carlo (MCMC) Methods

Markov chain Monte Carlo (MCMC) is widely used for solving complex problems related to probability distribution integration and combinatorial optimization (Beichl & Sullivan 2000). It is perhaps the only known general quantitative method that can find approximate solutions to complex problems in polynomial time in some contexts (Jerrum & Sinclair 1996). MCMC methods such as Gibbs sampling (Geman & Geman 1984) and Metropolis-Hastings algorithm (Metropolis et al. 1953, Hastings 1970) have influenced multiple fields of research and practice including computer science, physics, statistics, finance, economics, and engineering (Beichl & Sullivan 2000, Eraker 2001, Gilks et al. 1996). Beichl and Sullivan (2000) describe Metropolis-Hastings algorithm of which Gibbs Sampling is a special case as one of 'top 10 algorithms' in computing and 'the most successful and influential of Monte Carlo method': "Today, topics related to this algorithm constitute an entire field of computational science supported by a deep

theory and having applications ranging from physical simulations to the foundations of computational complexity."

MCMC algorithms have an increasingly important and growing role in network and computer security and cybersecurity, analysis of adversary attacks, penetration testing, and information assurance research and practices. Our current review of research establishes increasing relevance of MCMC, Gibbs Sampling, and Metropolis Algorithm in network and computer security contexts spanning cryptography and cryptanalytic password attacks and authentication analysis (e.g. Chen & Rosenthal 2012, Diaconis 2009, Hanawal & Sundaresan 2010, Muramatsu et al. 2006, Furon et al. 2012, Matsui et al. 2004), signature and anomaly based network intrusion detection and prevention systems (e.g. Scott 1999, 2001, 2004; Zhao & Nygard 2010, Ihler et al. 2006, Jyothsna et al. 2011, Shi & Mei-Feng 2012), and analyzing potential vulnerabilities in anonymity based systems such as Tor network based on onion-routing protocol and other 'social networks' (e.g. Danezis and Troncoso 2009, Troncoso and Danezis 2009).

There also seems growing interest among the broader network and computing security researcher and practitioner communities to develop better grasp of sophisticated quantitative methods such as Bayesian inference and MCMC methods. An example of such interest is evident in the community dialog on cryptography and encryption: 'Schneier on Security' blog on the topic 'TSA Uses Monte Carlo Simulations to Weigh Airplane Risks'.[1] In response to a debate among the readers on his blog about Monte Carlo methods, the renowned cryptography and encryption expert Bruce Schneier acknowledged his own interest in knowing more about Monte Carlo methods. Despite tomes of research published on the topic over last 60 or so years, palpable interest among mainstream researchers and practitioners is understandable. Most research published on Monte Carlo and MCMC methods has grown out of mathematical physicists', mathematicians', and statisticians' academic research characterized by understandable disciplinary formalism and diverse notational styles. Hence, there seems to be a critical need for research to bridge theory and practice by spanning disciplinary formalism of mathematicians and statisticians with applied concerns of network and computer security researchers. The current article with quantitative

[1] https://www.schneier.com/blog/archives/2007/06/tsa_uses_monte.html

methods focus in the context of network and computer security contributes to that research stream aiming to further advance research and practice in MCMC methods.

After the above introduction outlining increasingly important role of MCMC in network and computer security, the remaining sections of the discussion proceed as follows. Section 2 provides an overview of how these sophisticated quantitative methods came to be known as a 'revolution' and 'quantum leap' in statistical computing. Section 3 on Markov chain Monte Carlo Models and MCMC Algorithms forms the core focus of this article with its quantitative methods focus for readers new to these methods. It develops a technical introduction to Markov chain Monte Carlo Models and MCMC Algorithms, Gibbs Sampling and Metropolis-Hastings Algorithm based upon analysis and synthesis of research. Section 4 on applications of MCMC methods in network and computer security provides an overview of examples from network and computer security research. Readers interested in MCMC methods can relate to specific instances from their own research or practice and may consider applying those methods in their own work. Section 5 concludes with a discussion of key benefits of MCMC methods and algorithms in network and computer security and underscores the need for future research on systemic risk management issues including model risk management.

2. MCMC: A Revolutionary Quantum Leap in Statistical Computing

A paper on the history of MCMC interestingly observes about MCMC that (Robert & Cassella 2011a, 2011b emphasis added): "the development of this methodology has *not only changed our solutions to problems, but has changed the way we think about problems*." The MCMC methods originally conceptualized in 1940s at the Los Alamos National Lab during World War II led to the Metropolis algorithm, the first key MCMC algorithm in the early 1950s (Metropolis et al. 1953). The MCMC was the result of research by the same group of research scientists as those working on the atomic bomb including Stanislaw Ulam and John von Neumann at Los Alamos who around the same time had also created Monte Carlo (MC) methods (Eckhardt 1987). John von Neumann was using MC to study thermonuclear and fission problems in the late 1940s after the first computer, ENIAC, was developed. For high-dimensionality numerical problems, MC methods, though more efficient than conventional numerical methods, may require sampling from high-dimensionality probability distributions often making them

infeasible and inefficient in practice given computational complexity (Hastings 1970). Affected problems in combinatorics, data mining, machine learning, numerical analysis, and sampling show exponential increase in multi-dimensional space with increased high-dimensionality. Resulting sparseness of data is problematic as data needs grow exponentially with increased dimensionality for doing tests of statistical significance. To solve such problems, Hastings (1970), followed by Peskun (1973, 1981), generalized the Metropolis algorithm as a statistical simulation method for overcoming the 'curse of dimensionality'. In particular, as Bayesian inference based on posterior distributions with many parameters compounds the curse of dimensionality, MCMC has a particularly important role in advancing simulation-based Bayesian inference.

MCMC represents a 'quantum leap' in computational statistics (Robert & Cassella 2011) that shifts the emphasis from "closed form" solutions to improved numerical algorithms for solving "real" applied problems where "exact" now means "simulated." Since late 1980's, MCMC has become an all-pervasive method in statistical computation especially for Bayesian inference and for analyzing complex stochastic systems (Green 2014). The power of MCMC particularly in the context of Bayesian inference, besides other areas of computational statistics, results from two key flexibilities it affords for modeling and inference. First, MCMC allows the analyst to be closer to the reality of the process generating the data in terms of analysis as being well-suited for models based upon sparse data. It thus liberates the modeling process from constraints related to the curse of dimensionality. Second, on a related note, it also liberates the modeling process from dimensionality related constraints that earlier limited features of the target distribution to be modeled. The 'revolutionary' Gelfand and Smith paper (1990), one of top-three most cited papers in mathematics in last 20 years (Holmes 2008), is considered as "the genuine starting point for an intensive use of MCMC methods by the mainstream statistical community." Given necessary computing power and statistical computing algorithms such as the Gibbs sampler and the Metropolis–Hastings algorithm, it represented a 'paradigm shift' of interest in Bayesian methods, statistical computing, algorithms and stochastic processes (Robert & Cassella 2011). MCMC is an instance of revolutionary statistical computing methods enabled by computing advances that dramatically increase our ability to solve highly complex problems using statistical

inference across multiple domains. MCMC models enable us to make statistical inferences that were infeasible just a few years ago (Tsay 2010).

The above introductory overview of MCMC developed a perspective of how and why MCMC methods and algorithms came to be known as a 'revolution' and 'quantum leap' in statistical computing. The following section develops a technical introduction to the Markov chain Monte Carlo Models and the MCMC Algorithms, Gibbs Sampling and Metropolis-Hastings Algorithm based upon analysis and synthesis of prior research.

3. Markov chain Monte Carlo Models and MCMC Algorithms

3.1 Markov Process, Monte Carlo, and Markov chain Monte Carlo Models

The Metropolis algorithm is an example of a MCMC process (Kruschke 2010). To understand MCMC, we need to recognize what is a Markov chain as well as what is a Monte Carlo process. *Random walk* is a mathematical formalization of a succession of *random* steps as in steps taken by the proverbial drunk which have equal probability of going to each of the available next steps. For a given step or position, the probabilities of transition or *transition probabilities* to any next step depend only on the current step and next steps and are independent of prior events and steps.

A *Markov chain* is a succession of random steps [from one state to another] characterized by the Markov property of being 'memory-less.' It is memory-less in the sense that each next random step has no memory of, i.e., is totally independent of, all prior states [as well as prior sequence of steps and events] except for the current state from which it moves to the next state. Such a process characterized by the Markov property is called a *Markov process*.

Monte Carlo simulation is a simulation based upon repeated sampling of a lot of random input values from a distribution of inputs to assess the properties of the target outputs distribution by generating representative random values. Hence, it is a quantitative method of translating uncertainties in input variables as represented by their probability distributions to uncertainties of outcome variables represented by their probability distributions. Resulting quantified probabilities of specific outcomes form a probability distribution of predicted outcomes resulting from propagation or translation of input uncertainties into outcome uncertainties (GoldSim 2014). The Metropolis algorithm is a specific type of a Monte Carlo process (Kruschke 2010).

Bayesian forecasting with MCMC methods is a natural way to consider parameter and model uncertainty in forecasting (Tsay 2010). Considering above concepts, in statistical terms, it is useful to think of a stochastic process $\{X_t\}$ where each observed data X_t assumes a value in the parameter space Θ (Tsay 2005, 2010). The *Markov process* $\{X_t\}$ with memory-less property is one for which given value of X_t, values of X_h, $h > t$, do not depend on values of X_s, $s < t$. Such a Markov process $\{X_t\}$ has the following conditional distribution function (Tsay 2005, 2010):

$$P(X_h \mid X_{s,}\, s \leq t) = P(X_h \mid X_t),\ h > t$$

For a discrete time stochastic process $\{X_t\}$, the above property will become:

$$P(X_h \mid X_{t,}\, X_{t-1},\ldots) = P(X_h \mid X_t),\ h > t$$

Expressed differently, the stochastic process $X = \{X_0, X_1, X_2,\ldots X_T\}$ is a *Markov process* because for all $t = 0, 1,\ldots, T-1$, $f(X_{t+1} \mid x_{t,}\, x_{t-1},\ldots, x_o) = f(X_{t+1} \mid x_t)$, i.e., a sequence X_0, X_1,... of random elements of some set is a *Markov chain* if the conditional distribution of X_{t+1} given $x_{t,}\, x_{t-1},\ldots, x_o$ depends on x_t only. The set in which X_t assumes values is called the *state space* of the Markov chain (Geyer 2011). Further, a *stochastic process* X is a random variable $X(t, \omega)$, a function of both time t and state ω, for any $\omega \in \Omega$. For stochastic process $X = \{X_0, X_1, X_2,\ldots X_T\}$, if the change process of X is given by: $C_1 = X_1 - X_0$, $C_2 = X_2 - X_1,\ldots, C_T = X_T - X_{T-1}$, then the stochastic process X is called a *martingale* if $E(C_{t+1} \mid x_{t,}\, x_{t-1},\ldots, x_o) = 0$, or equivalently,

$E(X_{t+1} \mid x_{t,}\, x_{t-1},\ldots, x_o) = x_t$ for all $t = 0, 1,\ldots, T-1$. The stochastic process X with the same change process is called a *random walk* if $C_1, C_2,\ldots, C_T$ are independent and identically distributed (i.i.d.) with $E(|C_t|) < \infty$ for all $t = 0, 1,\ldots, T$.

Following from (1) and (2) above, if A is a subset of parameter space Θ, the *transition probability function* of the above Markov process will be expressed as follows to connote the transition of X_i from time t to time h (Tsay 2005, 2010).

$$P_t(\theta, h, A) = P(X_h \in A \mid X_t = \theta),\ h > t$$

The above Markov process is said to have a *stationary distribution* if the transition probability depends upon incremental change in time, $h - t$, but not on specific time t. Hence, a Markov chain has *stationary transition probabilities* if the

conditional distribution of X_{t+1} given x_t does not depend on t: this is the primary type of Markov chain of interest for MCMC models (Geyer 2011). A Markov model whose elements follow a Markov chain with stationary transition matrix is called a *Hidden Markov Model* (HMM). HMM, a mixture model with mixing distribution as a finite state Markov chain, assumes that the distribution of an observed data point depends upon an unobservable or *hidden* state.

To make statistical inference for a parameter vector θ and data X, where $\theta \in \Theta$, the distribution P(θ|X) needs to be determined. To use Markov chain simulation for doing so, we need to simulate a Markov process on parameter space Θ, which converges to a stationary distribution P(θ|X). The key to finding such convergence is to use a Markov chain with stationary distribution pre-specified as P(θ|X) and run it until it results in approximate convergence of distribution of current values with the stationary transition distribution (Tsay 2005, 2010). For some transition probability distribution, the initial distribution is said to be *stationary* or *equilibrium* if the Markov chain specified by it and the transition probability distribution is stationary. This can be re-stated as (Geyer 2011): 'the transition probability distribution preserves the initial distribution'. The result will be the determination of many Markov chains that have the desired property noted above. Such Markov chain simulation methods used for determining the stationary distribution P(θ|X) are known as MCMC methods as they make combined use of Markov chain processes and Monte Carlo simulations.

Monte Carlo approach developed at Los Alamos prior to MCMC was devised as a method for using random number generation for computing complex integrals (Walsh 2004). A complex integral such as $\int_a^b h(x)dx$ is expressed as product of function $f(x)$ and probability density function $p(x)$ is: $\int_a^b f(x)\, p(x)dx$. That product expressed as the expectation of $f(x)$ over density $p(x)$, $E_{p(x)}\,[f(x)]$, can be approximated as the average of the summation of function $f(x)$ over a large number of random variables $x_1,\ldots, x_n$ from density $p(x)$. Mathematically,

$$\int_a^b h(x)dx = \int_a^b f(x)\, p(x)dx = E_{p(x)}\,[f(x)] \approx \frac{1}{n}\sum_{i=1}^{n} f(x_i)$$

The above method known as *Monte Carlo integration* is used in Bayesian inference to approximate posterior or marginal posterior distributions. Extending above computation to a conditional function such as $f(y|x)$ results in an analogous simplification of the integral expression in the context of Bayesian inference:

$$I(y) = \int f(y|x)p(x)dx \approx \frac{1}{n}\sum_{i=1}^{n} f(y|x_i)$$

3.2 Gibbs Sampling Algorithm

Influenced by the 'landmark paper' (Robert & Cassella 2011a, 2011b) of Geman and Geman (1984) that developed Gibbs Sampling, Gelfand and Smith (1990) advanced Gibbs Sampling into perhaps the most popular MCMC method (Tsay 2010). Gibbs sampler is the MCMC technique used for generating random variables from a marginal distribution indirectly without the need for calculating the distribution density (Casella & George, 1992). The key advantage of Gibbs sampling is in decomposing high-dimensional estimation problems such as in complex stochastic models into lower-dimensional simpler and more manageable form problems using full conditional distributions of the parameters (Scollink 1996). An extreme example of its use is in the solution of a complex multivariate stochastic model with N parameters (i.e., N-dimensions) using N univariate (i.e., one-dimensional) conditional distributions. When parameters are highly correlated, it may be advisable to use joint draws as it may not be efficient to reduce Gibbs draws into univariate problems (Kruschke 2010, Tsay 2010).

Consistent with Walsh (2004), Tsay (2010) explains Gibbs sampling in the context of estimation of parameters so that the fitted model can be used for making inference. Consider three parameters θ_1, θ_2, and θ_3 for a collection of observed data X and M as the contemplated model to be fitted. Here the word *parameter* is used very generally. For instance, in MCMC framework, a parameter may denote a missing data point or an unobservable latent or "true" variable underlying the observed variable. Following Casella & George (1992) and Scollink (1996), assume that the three conditional distributions of any parameter θ_i given the others are available for θ_1, θ_2, and θ_3 but the likelihood function of the model cannot be analytically or numerically

computed. (In statistics, *likelihood function* (or likelihood) of a set of parameter values θ_i, given observed data X_i, is the probability of those observed variables given the respective parameter values, i.e., $\mathcal{L}(\theta|X) = P(X|\theta)$.) Statistically, $f_1(\theta_1|\theta_2, \theta_3, X, M)$, $f_2(\theta_2|\theta_3, \theta_1, X, M)$, $f_3(\theta_3|\theta_1, \theta_2, X, M)$ denote the three conditional distributions for θ_1, θ_2, and θ_3. Generally, $f_i(\theta_i|\theta_{j\neq i}, X, M)$ represents the conditional distribution of parameter θ_i given the other two parameters θ_j and θ_k, the data X, and the model M. In practice, the exact form of the conditional probability distribution function doesn't need to be known; we should be however able to draw a random number from each of the relevant conditional distributions.

Assume notation $\theta_{a,b}$ wherein a= probability distribution, and, b = specific numeric order or sequence of the draw from that distribution. Then, the computational logic of one iteration of the Gibbs sampling algorithm given arbitrary initial values for θ_2 and θ_3 being $\theta_{2,0}$ and $\theta_{3,0}$ is listed below (Tsay 2010).

a. Draw a random sample from $f_1(\theta_1|\ \theta_{2,0}, \theta_{3,0}, X, M)$ denoting random draw as $\theta_{1,1}$.
b. Draw a random sample from $f_2(\theta_2|\ \theta_{3,0}, \theta_{1,1}, X, M)$ denoting random draw as $\theta_{2,1}$.
c. Draw a random sample from $f_3(\theta_3|\ \theta_{1,1}, \theta_{2,1}, X, M)$ denoting random draw as $\theta_{3,1}$.

At end of the first iteration of draws from each distribution, the parameters $\theta_{1,0}$, $\theta_{2,0}$, and $\theta_{3,0}$ become $\theta_{1,1}$, $\theta_{2,1}$, and $\theta_{3,1}$. Using the updated parameters as input, the second iteration results in updated parameters as $\theta_{1,2}$, $\theta_{2,2}$, and $\theta_{3,2}$. Repetition of the iteration m times will yield the following sequence of random draws: $(\theta_{1,1}, \theta_{2,1}, \theta_{3,1}), \ldots, (\theta_{1,m}, \theta_{2,m}, \theta_{3,m})$. By taking large enough *m*, i.e., simulating a large enough sample, the m^{th} draw, $(\theta_{1,m}, \theta_{2,m}, \theta_{3,m})$ [under some weak regularity conditions requiring prior Gibbs iteration's traversal of full parameter space (Tsay 2010)] is approximately equivalent to a random draw from the joint probability distribution of the three parameters, $f(\theta_1, \theta_2, \theta_3|\ X, M)$.

For real application, Tsay (2010) recommends using sufficiently large n and dropping first m random draws (called *burn-in* sample) from the Gibbs iterations yielding the final Gibbs sample: $(\theta_{1,m+1}, \theta_{2,m+1}, \theta_{3,m+1}), \ldots, (\theta_{1,n}, \theta_{2,n}, \theta_{3,n})$. Prior m random draws are dropped to ensure that the final residual sample converges as close as possible to a random sample from the joint distribution $f(\theta_1, \theta_2, \theta_3|\ X, M)$. The final Gibbs sample being close enough to the random sample from the joint distribution can then be used for computation, for example, of point estimate and variance (Tsay 2010, Walsh 2004).

3.3 Metropolis Algorithm

Originally, attempts to integrate very complex functions using random sampling by mathematical physicists' such as Metropolis & Ulam (1949), Metropolis et al. (1953), and, Hastings (1970) led to development of MCMC methods and Metropolis-Hastings algorithm. Those attempts were aimed at resolving the problems inherent in obtaining samples from complex probability distributions while applying Monte Carlo integration.

Consistent with Walsh (2004), Tsay (2010) considers the case of the conditional probability distribution $p(\theta|X) = f(\theta|X)/K$ (where K is the normalizing constant) for which it is infeasible or very time intensive to compute the normalization constant or for which random draws are unavailable. Given an approximation of that distribution for which random draws are feasible, the Metropolis algorithm (Metropolis & Ulam 1949, Metropolis et al. 1953) generates a sequence of random draws from it whose distributions converge to $f(\theta|X)$ as follows (Walsh 2004, Tsay 2010, Kruschke 2010).

a. Start with a random draw of some initial value θ_o: $f(\theta_o|X) > 0$.
b. Given previous draw θ_{t-1} for the t^{th} iteration, draw a candidate sample θ_* from a known distribution and call it *jumping distribution* $J_t(\theta_t|\theta_{t-1})$, also known as *proposal distribution* or *candidate-generating distribution* (Gelman et al. 2003). The jumping distribution denoting the probability of returning value of θ_t given previous value of θ_{t-1} must be symmetric, i.e., $J_t(\theta_i|\theta_j) = J_t(\theta_j|\theta_i)$ for all θ_i, θ_j, and t.
c. Given candidate sample θ_*, calculate the ratio r of the density at the candidate point θ_* and at the current point θ_{t-1}: $r = p(\theta_*|X) / p(\theta_{t-1}|X) = f(\theta_*|X) / f(\theta_{t-1}|X)$. As the ratio $f(\theta_i|X)$ is being computed for the same probability distribution with two different i-values, the normalization constant K cancels out in both the numerator

and the denominator. That is how MCMC Metropolis algorithm resolves the original problem of computing the normalization constant that motivated the discussion.

d. If the jump from θ_{t-1} to θ_* *increases* the conditional posterior density, i.e., $r > 1$, accept the candidate point θ_* as θ_t, i.e., set $\theta_t = \theta_*$ and return to step b. If the jump *decreases* the conditional posterior density, i.e., $r < 1$, accept the candidate and set $\theta_t = \theta_*$ with probability r; else reject it, i.e. set $\theta_t = \theta_{t-1}$, and return to step b.

As per Walsh (2004), the Metropolis algorithm can be summarized in terms of first computing the acceptance probability of candidate as $r = \min [f(\theta_*|X) / f(\theta_{t-1}|X), 1]$ and then accepting a candidate point with probability r called the *probability of the move* to the proposed position, $p_{move} = \min [\mathcal{P}(\theta_{proposed}) / \mathcal{P}(\theta_{current}), 1]$ (Kruschke 2010).

This generates a Markov chain $(\theta_0, \theta_1,..., \theta_k, ...)$, as the transition probabilities from θ_t to θ_{t+1} depend only on θ_t and not on $(\theta_0, ..., \theta_{t-1})$. Following a sufficient burn-in period of say prior m of n steps, the chain approaches its stationary distribution, and then the samples from the vector $(\theta_{m+1,...}, \theta_n)$ are the samples from $p(\theta|X)$.

3.4 Metropolis-Hastings Algorithm

Hastings algorithm (Hastings 1970, Tsay 2010) based upon generalization of the Metropolis algorithm uses an arbitrary transition probability function $J_t(\theta_i|\theta_j) = \Pr(\theta_i \rightarrow \theta_j)$. Correspondingly, it calculates the ratio r of the density at the candidate point θ_* and at the current point θ_{t-1}: $r = (f(\theta_*|X) / J_t(\theta_*|\theta_{t-1})) / (f(\theta_{t-1}|X) / J_t(\theta_{t-1}|\theta_*))$

$$= (f(\theta_*|X) \; J_t(\theta_{t-1}|\theta_*)) \; / (f(\theta_{t-1}|X) \; J_t(\theta_*|\theta_{t-1}))$$

It also sets acceptance probability of the candidate point (Hastings 1970, Walsh 2004):

$r = \min [(f(\theta_*|X) / J_t(\theta_*|\theta_{t-1})) / (f(\theta_{t-1}|X) / J_t(\theta_{t-1}|\theta_*)), 1]$

$= \min [(f(\theta_*|X) \; J_t(\theta_{t-1}|\theta_*)) / (f(\theta_{t-1}|X) \; J_t(\theta_*|\theta_{t-1})), 1]$

As apparent, the Hastings algorithm represents a more general case of the Metropolis algorithm: when jump density is symmetric, i.e., $J_t(\theta_i|\theta_j) = J_t(\theta_j|\theta_i)$, it reduces to the

original Metropolis algorithm. The modified algorithm is known as the *Metropolis-Hastings algorithm* which is very general and broadly applicable. A caveat about the Metropolis-Hastings algorithm is that the algorithm's convergence to a solution is contingent upon the availability of a fine-tuned proposal distribution. Otherwise, if the proposal distribution is too narrow or too broad, greater proportion of the proposed jumps will be rejected or the move will be restricted to a narrow localized parameter space. Gibbs sampling, in contrast, is more forgiving as it does not require 'artful tuning' of a proposal distribution (Kruschke 2010).

The above discussion on Monte Carlo Models and MCMC Algorithms, Gibbs Sampling and Metropolis-Hastings Algorithm developed the core quantitative methodological focus of this article. Next section develops an understanding of how network and computer security research and practice represent increasingly important domains for application of above research methods based upon MCMC algorithms and Bayesian inference. The following discussion also provides specific examples from three key domains of network and computer security research wherein solutions to complex high-dimensional stochastic problems relied upon creative applications of MCMC.

4. MCMC Models in Computer & Network Security Contexts

The following review of research establishes increasing importance of MCMC, Gibbs Sampling, and Metropolis Algorithm in three key contexts of network and computer security research and practice. Related discussion on application of MCMC methods in network and computer security highlights selective examples from network and computer security research. The discussion is illustrative given the methodological focus of the article. Focus is on demonstrating through specific examples how MCMC methods and MCMC algorithms are applied in practice in the given contexts. The three specific contexts of network and computer security research that are the focus of the following research methods discussion on MCMC are listed below.

(1) Cryptography, Cryptanalytics & Penetration Testing
(e.g. Chen & Rosenthal 2012, Diaconis 2009, Hanawal & Sundaresan 2010, Muramatsu et al. 2006, Furon et al. 2012, Matsui et al. 2004),

(2) Intrusion Detection & Prevention and Anomaly Detection

(e.g. Scott 1999, 2001, 2004; Zhao & Nygard 2010, Ihler et al. 2006, Jyothsna et al. 2011, Shi & Mei-Feng 2012), and,

(3) Privacy in Anonymity Systems and Social Networks

(e.g. Danezis and Troncoso 2009, Troncoso and Danezis 2009).

4.1 Cryptography, Cryptanalytics & Penetration Testing

Author's interest in MCMC for network and computing security was motivated in course of applied R&D on quantitative risk management models for global banking and finance model risk, market risk and operational risk management (Malhotra 2014a, 2014b, 2014c, 2014d, 2014e). His prior research focused on analyzing vulnerabilities in the mainstream global encryption standards and cryptographic protocols based on mathematical and algebraic analysis of cryptanalytic algorithms such as algebraic number sieves (Malhotra 2013a, 2013b, 2013c, 2013d). While analyzing the mathematical and statistical foundations of computing and network encryption schemes, his interest focused on computational and statistical foundations of cryptography and cryptanalysis using cryptanalytic tools. In that process he found some very interesting research in MCMC methods by statisticians and mathematicians advancing quantitative methods research on cryptography and cryptanalysis. Three such examples of cryptography and cryptanalysis-related MCMC research are outlined below.

An interesting research stream in this applied area is focused on decrypting and attacking ciphers underlying network and computing encryption mechanisms (Chen & Rosenthal 2010, Diaconis 2009). Research pioneering integrated use of cryptography and MCMC algorithms by Chen & Rosenthal (2010) advances MCMC for decryption of substitution ciphers, transposition ciphers, and substitution-plus-transposition ciphers. Based on the frequency analysis of combinations of characters such as bi-grams and tri-grams, their research has delved into in-depth statistical analysis for optimization of such decryption attacks. They analyzed the transitions of consecutive text symbols in bi-grams to develop a matrix of such transitions then used it for computing the probability of the respective transitions. MCMC algorithms were used for searching the probability maximizing functions given the high-dimensionality of the search space of such functions. Their analysis has examined diverse combinations of variables such as MCMC iterations, scaling parameter, cipher text amount, number of repetitions, and, swap vs.

slide vs. block-slide moves. They report success rates of up to 70% and above with transposition key lengths up to 40.

Diaconis (2009) motivates MCMC application in the context of cryptography and cryptanalysis and provides an analytical treatment of the Metropolis algorithm and related theorems. Advancing upon Diaconis (2009), Hanawal and Sundaresan (2010) develop an empirical study in which they generate randomized passwords using MCMC and the Metropolis algorithm. They show how a high-dimensional problem characterized by a distribution with difficult to compute normalizing constant can be reframed using the Metropolis algorithm after which the solution is no longer hindered by the need for the normalizing constant. Related MCMC enabled credential authentication and decoding research includes examples such as dynamic signature verification (Muramatsu et al. 2006), decoding fingerprints (Furon et al. 2012), and face recognition (Matsui et al. 2004). Above examples illustrate use of MCMC methods such as Metropolis algorithm in solving difficult to compute or otherwise infeasible high-dimensionality problems in cryptography, cryptanalysis, and penetration testing.

4.2 Intrusion Detection & Prevention and Anomaly Detection

Intrusion detection and intrusion prevention is another network and computer security research and practice area that has benefited from applications of MCMC methods and algorithms (e.g. Scott 1999, 2001, 2004; Zhao & Nygard 2010, Ihler et al. 2006). Many such models depend upon anomaly detection wherein behavior of traffic generated by the customers is distinguished from that of the attackers based upon distinct probability distributions. Scott (1999, 2001) distinguished customers' traffic as a Poisson process from the traffic from the two-state continuous time Markov process generated by attackers breaking into and exiting the accounts. The presence of the attacker also generates additional traffic as an independent second Poisson process. Given all processes as homogeneous, account traffic data is represented as discrete time Hidden Markov Models in which case the hidden state indicates intrusion or attack status and presence of absence of the attacker (Scott 2001, 2004). These studies used the MCMC algorithm Gibbs sampler for sampling each state in the hidden Markov chain given most recent draws of nearest neighbors. MCMC is critical for solution of such high-dimensionality problems as the likelihood function for the HMM quickly becomes infeasible to compute even for small values of hidden chain's size of the state space.

Using a Bayesian approach to learning and inference for time series data, Ihler et al. (2006) use a similar Hidden Markov-Poisson model for an adaptive anomaly detection algorithm. Their study determined the time complexity of each MCMC iteration as $O(T)$, linear in the length of the time series, and the series shows rapid convergence. Given the high false-positive rate of anomaly detection intrusion systems, Shi & Mei-Feng (2012) show how several research studies using HMM benefited from MCMC and related methods for analyzing intrusion detection systems. Their study uses HMM given high-dimensionality resulting from number of states, the classic problem for which MCMC was devised as a solution as discussed earlier.

Similarly, Zhao & Nygard (2010) use the Metropolis-Hastings algorithm to infer the distribution of intruders in a wireless network from limited local information used by a fuzzy logic algorithm to assess the impact of the intruders on a monitored point. A comprehensive review of MCMC and other related Bayesian inference and machine learning models for anomaly based intrusion detection and prevention systems is available in Jyothsna et al. (2011). They review key distinctions between statistical models (such as: threshold model, Markov process model, statistical moments model, multivariate model, time series model), cognition models (such as finite state machine, description scripts, and adept systems), cognition based techniques (such as boosted decision tree, support vector machine, artificial neural network), machine learning based detection techniques, kernel based online anomaly detection, and detection models based on computer immunology and models based on user intention. Use of MCMC also optimizes use of computational processing power needed by the wireless base station which then only needs to query and process network packets to the specific nodes identified by the conditional distribution. Above examples illustrate the use of MCMC methods such as Gibbs sampling and Metropolis-Hastings algorithm in solving difficult to compute or otherwise infeasible high-dimensionality problems in intrusion detection and prevention and anomaly detection such as for telecom networks.

4.3 Privacy in Anonymity Systems and Social Networks

Anonymity systems such as Tor network based on onion-routing allow two parties to exchange information without disclosing their network identifiers to each other or to any other third party. The engineering principle underlying such communications is that

the messages entering and leaving the network should be *cryptographically unlinkable* (Danezis and Troncoso 2009, Troncoso and Danezis 2009). Privacy of such networks which ensure that anonymity is safeguarded is prized across commercial, social, and government and military communications. The application of MCMC algorithms in case of such anonymity systems is to make it statistically and computationally feasible to infer who is talking with who based upon network traffic patterns of messages.

Anonymity is measured as the uncertainty that the adversary has about who is conversing with who by using information theoretic measures of entropy (Danezis and Troncoso 2009). The key limitations of those measures include measuring anonymity of a single message and *not* the systems as a whole, and, most seriously, statistical infeasibility of computing relevant probability distributions. Contribution of the Danezis and Troncoso (2009) at Microsoft Research is to address the 'hard problem' of calculating the probability distributions over senders or receivers of messages. To solve the above statistical computational problems, they demonstrate the use of probabilistic modeling and Bayesian inference, which despite their power, are handicapped by considerable computational complexity that often makes it not possible to compute the probability distributions. It is in this specific context that MCMC sampling algorithms, including Metropolis-Hastings (MH) algorithm and Gibbs sampler, come to the rescue for extracting samples that provide approximations of relevant probability distributions from observations of 'rather complex systems' (Danezis and Troncoso 2009).

Above review of MCMC methods and algorithms advancing research and practice in network and computer security and cybersecurity, analysis of adversary attacks, penetration testing, and information assurance establishes their increasingly important and growing role. Given the methodological scope and focus of the discussion, only specific examples and contexts within network and computer security research could be addressed. The concluding discussion further highlights some of the broader implications of this stream of research with both methodological and applied recommendations.

5. Conclusion

Markov chain Monte Carlo (MCMC) may be described as a widely used set of general quantitative methods to find approximate solutions to complex problems in polynomial time. Recognized as one of top-10 computing algorithms with underlying research among top-3 mathematics papers, its impact across diverse fields including computer science, physics, statistics, finance, economics, and engineering is evident. The article focuses on highlighting the increasingly important and critical role of MCMC algorithms in network and computer security research and practice. The greatest impact of MCMC methods and algorithms is probably in case of problems where outputs lack interpretability because of high-dimensionality and complex interactions in inputs. Several of the network and computer security contexts highlighted in the discussion reviewed such problems and their resolution using MCMC. Our review established increasing importance of MCMC, Gibbs Sampling, and Metropolis Algorithm in modeling cryptography and cryptanalytic password attacks and authentication analysis; signature and anomaly based network intrusion detection and prevention systems; and analyzing potential vulnerabilities in anonymity based systems such as Tor network based on onion-routing protocol and other 'social networks'.

Beyond the focus of the current discussion, there are two key important issues to focus on for future research and development. First is the development of quantitative methods and algorithms for addressing high-dimensional computationally complex problems relevant to emerging paradigms such as big data analysis and quantum computing. We need to recognize that modern statistical paradigms such as Bayesian inference are themselves reliant on computational statistical methods such as MCMC for their prowess. Second is increasing and critical need for recognition and resolution of problems at general systems level where they are sometimes called *systemic* problems. Some of our methodological and applications discussion explicitly or implicitly recognized this systemic concern. Particularly, in the case of computer and network security, the problems across most domains being addressed relate to the broader focus on risk management as well.

Relating above methodological and applied concerns togeher, one focus of future research needs to be on *model risk management*. Solving complex high-dimensional problems with inaccurate models is often punished in any domain, whether it is

computing or (say) investment banking. Model risk management has gained currency in investment banking but is equally important for any domain reliant upon high-dimensional and computationally complex problems such as network and computer security. 'Model risk arises from the potential adverse consequences of making decisions based on incorrect or misused model outputs and reports.' Knowing history of applications of MCMC ranging from chemical-physics to network computing, most readers can perhaps relate to the above concern about model risk. Few may, however, recognize that the above statement is from a top investment bank strategy document.

Whether it is quantum computing or quantitative finance, regardless, it is imperative to ensure that models perform as specified and intended; models are conceptually sound and used appropriately and that model users are aware of the models' strengths and limitations and how these can impact their decisions. That is essentially model risk management. Advancing beyond network computing to investment banking, one may possibly discern that the computational statistical methods and models related concerns impact both fields [and others] as well. Following prior discussion, one may even speculate that Wall Street and Pentagon [among others] may be probably grappling with similar model risk management concerns; albeit probably oblivious of the commonality of systemic problems they may share.

In any case, the approaches to mitigate operating risk associated with the use of models needs to evolve to reflect recent trends in practice. In particular, there are a number of new areas where it is not possible for the "human eye" to necessarily detect material flaws: in the case of models operating over very small time scales, or where outputs lack interpretability due to high-dimensionality and complex interactions in inputs, the periodic inspection of predicted versus realized outcomes is unlikely to be an effective risk mitigate. These situations require a holistic validation framework of the system focused on identifying and mitigating potential failures, taking into account the models' objectives, their implementation including the joint interaction of software and hardware, their response to potential input shocks in real time and the fail-safe mechanisms. The above quote is attributed to a top investment bank as well.

Selected References

1. Beichl, I. & Sullivan, F. The Metropolis Algorithm. Computing in Science and Engineering, 2(1), pp. 65-69, January/February 2000.

2. Casella, G. & George, E. I. (1992). Explaining the Gibbs sampler. The American Statistician, 46 (3), pp. 167-174, August, 1992.

3. Chen, Jian & Rosenthal, Jefrey S. Decrypting Classical Cipher Text Using Markov Chain Monte Carlo. Statistics and Computing, 22(2), pp. 397-413, March 2012.

4. Danezis, George and Carmela Troncoso. The Application of Bayesian Inference to Traffic analysis. Technical Report: MSR-TR-2009-112. Microsoft Research. 18 August 2009.

5. Diaconis, Persi. The Markov Chain Monte Carlo Revolution. Bulletin of the American Mathematical Society, 46(2), pp. 179-205, April 2009.

6. Eraker, B. "MCMC analysis of Diffusion Models with Application to Finance," Journal of Business & Economic Statistics, 19 (2), pp. 177-191, 2001.

7. Eckhardt, Roger. Stan Ulam, John von Neumann, and the Monte Carlo Method. Los Alamos Science, Special Issue, pp. 131-141.

8. Furon, T., A. Guyader, and, F. Cerou. Decoding fingerprints using the Markov Chain Monte Carlo method. IEEE International Workshop on Information Forensics and Security (WIFS). pp. 187-192.2-5 Dec. 2012.

9. Gamerman, Dani & Lopes, Hedibert F. Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference (2nd edn). Chapman & Hall/CRC, Boca Raton, FL, 2006.

10. Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. Bayesian Data Analysis, 2nd ed., Chapman and Hall/CRC, London, 2003.

11. Geman, S. & Geman, D. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 6, pp. 721-741, November 1984.

12. Geyer, Charles J. "Introduction to Markov Chain Monte Carlo." In Handbook of Markov Chain Monte Carlo, Ed. Steve Brooks, Andrew Gelman, Galin L. Jones & Xiao-Li Meng. pp. 3-48. Chapman & Hall / CRC. 2011.

13. Gilks, W., S. Richardson, and D. Spiegelhalter, Markov Chain Monte Carlo in Practice. London, U.K.: Chapman and Hall, 1996.

14. GoldSim. What is the Monte Carlo method? Goldsim Technology Group. 2014. WWW: http://www.goldsim.com/Web/Introduction/Probabilistic/MonteCarlo/

15. Green, P.J. A primer on Markov chain Monte Carlo. University of Bristol. Lecture Notes. 2014.

16. Hanawal, M. K. and R. Sundaresan. Randomised attacks on passwords. Technical Report TR-PME-2010-11, DRDO-IISc Programme on Advanced Research in Mathematical Engineering, IISc, Bangalore, 12 February 2010.

17. Hastings, W. Monte Carlo sampling methods using Markov chains and their application. Biometrika, 57, pp. 97-109, 1970.

18. Holmes, Chris. Markov Chain Monte Carlo and Applied Bayesian Statistics: A Short Course. Oxford Centre for Gene Function. Oxford University. 2008.

19. Hu, Jiankun and Xinghuo Yu, "A Simple and Efficient Hidden Markov Model Scheme for Host-Based Anomaly Intrusion Detection" IEEE Network Journal, 23(1), January/February 2009.

20. Ihler, A., J. Hutchins and P. Smyth, "Adaptive event detection with time-varying Poisson processes" ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), Philadelphia, PA, Aug. 2006.

21. Jerrum, M. and A. Sinclair, "The Markov chain Monte Carlo method: An approach to approximate counting and integration," in Approximations for NP-Hard Problems, D. Hochbaum, Ed. Boston, MA: PWS Publishing, 1996.

22. Jyothsna, V. et al. A Review of Anomaly based Intrusion Detection Systems. International Journal of Computer Applications, 28(7), August 2011.

23. Kruschke, John. Doing Bayesian Data Analysis: A Tutorial Introduction with R. October 2010. Elsevier, New York.

24. Malhotra, Yogesh. A Risk Management Framework for Penetration Testing of Global Banking & Finance Networks Voice over Internet Protocols. Technical Report. 2014a. Global Risk Management Network, LLC, New York.

25. Malhotra, Yogesh. Analysis of FIX and FAST as Financial Securities Trading and Transactions Messaging Network Protocols. Technical Note. 2014b. Global Risk Management Network, LLC, New York.

26. Malhotra, Yogesh. Analysis of Attack Trees for Mitigating Cybersecurity Attacks on Global Banking & Finance and SCADA Systems. Technical Note. 2014c. Global Risk Management Network, LLC, New York.

27. Malhotra, Yogesh. Bitcoin Protocol: Model of 'Cryptographic Proof' Based Global Crypto-Currency & Electronic Payments System. 2013a. Global Risk Management Network, LLC, New York.

28. Malhotra, Yogesh. Cryptology Beyond Shannon's Information Theory: Preparing for When the 'Enemy Knows the System'. 2013b. Global Risk Management Network, LLC, New York.

29. Malhotra, Yogesh. Future of Bitcoin & Statistical Probabilistic Quantitative Methods: Global Financial Regulation (Interview: Hong Kong Institute of CPAs). Regulatory Compliance Report. 2014d. Global Risk Management Network, LLC, New York.

30. Malhotra, Yogesh. Network Intrusion Detection and Prevention & Active Response: Frameworks, Systems, Methods, Tools & Policies. Technical Notes. 2014e. Global Risk Management Network, LLC, New York.

31. Malhotra, Yogesh. Number Field Sieve Cryptanalysis Algorithms for Most Efficient Prime Factorization on Composites. 2013c. Global Risk Management Network, LLC, New York.

32. Malhotra, Yogesh. Quantum Computing, Quantum Cryptography, Shannon's Entropy and Next Generation Encryption & Decryption. 2013d. Global Risk Management Network, LLC, New York.

33. Matsui, A., S. Clippingdale, F. Uzawa, T. Matsumoto. Bayesian face recognition using a Markov chain Monte Carlo method. Vol.3, pp. 918-921. 23-26 Aug. 2004.

34. Metropolis, N., and S. Ulam. The Monte Carlo Method. Journal of the American Statistical Association, 44, pp. 335-341, 1949.

35. Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., and Teller, E. Equations of state calculations by fast computing machines. Journal of Chemical Physics, 21(6), pp. 1087-1092, 1953.

36. Morral et al. Modeling Terrorism Risk to the Air Transportation System an Independent Assessment of TSA's Risk Management Analysis Tool and Associated Methods. RAND Homeland Security and Defense Center, 2012.

37. Muramatsu, D., M. Kondo, M. Sasaki, S. Tachibana. A Markov chain Monte Carlo algorithm for bayesian dynamic signature verification. IEEE Transactions on Information Forensics and Security, 1(1), pp. 22-34, March 2006.

38. Peskun, P. Optimum Monte Carlo sampling using Markov chains. Biometrika, 60, pp. 607-612, 1973.

39. Peskun, P. Guidelines for choosing the transition matrix in Monte Carlo methods using Markov Chains. Journal of Computational Physics, 40, pp. 327-344, 1981.

40. Robert, Christian & George Casella. A Short History of Markov Chain Monte Carlo: Subjective Recollections from Incomplete Data. Statistical Science, 26(1), pp. 102-115, 2011a.

41. Robert, Christian & George Casella. "A Short History of MCMC: Subjective Recollections from Incomplete Data." In Handbook of Markov Chain Monte Carlo, Ed. Steve Brooks, Andrew Gelman, Galin L. Jones & Xiao-Li Meng. Chapman & Hall / CRC. pp. 49-66. Chapman & Hall / CRC. 2011b.

42. Schneier, B. TSA Uses Monte Carlo Simulations to Weigh Airplane Risks. June 22, 2007.
WWW: https://www.schneier.com/blog/archives/2007/06/tsa_uses_monte.html

43. Scott, S. L. Bayesian Analysis of a Two-State Markov Modulated Poisson Process, Journal of Computational and Graphical Statistics, 8, pp. 662-670, 1999.

44. Scott, Steven L. Bayesian Methods for Hidden Markov Models: Recursive Computing in the 21st Century. 45 (1), pp 69-83, 2004.

45. Scott, Steven L. Detecting Network Intrusion Using a Markov Modulated Non-homogeneous Poisson Process. Journal of the American Statistical Association, 2001.

46. Scollnik, David. An Introduction to Markov Chain Monte Carlo Methods and Their Actuarial Applications. Proceedings of the Casualty Actuarial Society LXXXIII. 114-165, 1996.

47. Shi, Shang-zhe and Sun Mei-feng. Study on HMM Based Anomaly Intrusion Detection Using System Calls. 2nd International Conference on Electronic & Mechanical Engineering and Information Technology (EMEIT-2012), Liaoning, China. 07 Sep - 09 Sep 2012.

48. Trevisan, Luca. Pseudo-randomness and de-randomization. ACM Crossroads 18(3), pp. 27-31, 2012.

49. Troncoso, Carmela and George Danezis. The Bayesian Traffic Analysis of Mix Network. CCS '09 Proceedings of the 16th ACM Conference on Computer and Communications Security. pp. 369-379. November 9–13, 2009, Chicago, Illinois, USA.

50. Tsay, Ruey. S. Analysis of Financial Time Series. 3rd Edition. 2010. Wiley & Sons. New Jersey.

51. Tsay, Ruey. S. Some MCMC Applications in Time Series Analysis. Lecture Notes: Time Series Analysis. 2005. University of Chicago Booth School of Business.

52. Walsh, B. Markov Chain Monte Carlo and Gibbs Sampling, Lecture Notes for EEB 581, version 26 April, 2004. WWW: http://web.mit.edu/~wingated/www/introductions/mcmc-gibbs-intro.pdf.

53. Wigderson, Avi. Randomness, Pseudorandomness, and Derandomization. The Fields Institute for Research in Mathematical Sciences, Fields Notes.

54. Zhao, J. & K. E. Nygard. A Dendritic Cell Inspired Security System in Wireless Sensor Networks. FUTURE COMPUTING 2010: The Second International Conference on Future Computational Technologies and Applications. 2010.