Stochastic optimization on continuous domains with finite-time guarantees by Markov chain Monte Carlo methods

A. Lecchini-Visintini[*], J. Lygeros[†] and J. Maciejowski[‡]

Abstract

We introduce bounds on the finite-time performance of Markov chain Monte Carlo (MCMC) algorithms in solving global stochastic optimization problems defined over continuous domains. It is shown that MCMC algorithms with finite-time guarantees can be developed with a proper choice of the target distribution and by studying their convergence in total variation norm. This work is inspired by the concept of finite-time learning with known accuracy and confidence developed in statistical learning theory.

I. Introduction

Simulated annealing is a general method for approaching the solution of a global optimization problem [1]. Simulated annealing can be implemented on continuous domains using the general family of Markov chain Monte Carlo (MCMC) methods [2]. In this paper, we introduce rigorous guarantees on the finite-time performance of simulated annealing on continuous domains. We will show that it is possible to derive MCMC algorithms to implement simulated annealing which can find an approximate solution to the problem of optimizing a function of continuous variables, within a specified tolerance and with an arbitrarily high level of confidence after a known finite number of steps.

The background of our work is twofold. On the one hand, our notion of 'approximate domain optimizer' is inspired by the definition of 'probably approximate near minimum' introduced by Vidyasagar in [3], [4]. In the control field, the work of Vidyasagar [3], [4] has been seminal in the development of the so-called randomized approach. Inspired by statistical learning theory,

[*] Department of Engineering, University of Leicester, alv1@leicester.ac.uk.

[†] Automatic Control Laboratory, ETH Zurich, lygeros@control.ee.ethz.ch.

[‡] Department of Engineering, University of Cambridge jmm@eng.cam.ac.uk.

this approach is characterized by the construction of algorithms which make use of independent sampling in order to find probabilistic approximate solutions to difficult control system design applications see e.g. [5], [6]. In our work, the definition of approximate domain optimizer will be essential in establishing rigorous guarantees on the finite-time performance of simulated annealing. On the other hand, we show that our rigorous finite-time guarantees can be achieved by the wider class of algorithms based on MCMC sampling and we ground our results on the theory of convergence, with quantitative bounds on the distance to the target distribution, of the Metropolis-Hastings algorithm and MCMC methods [7]–[10]. In addition, we demonstrate how, under some quite weak regularity conditions, our definition of approximate domain optimizer can be related to the standard notion of approximate optimization considered in the stochastic programming literature [11], [12]. This link provides theoretical support for the use of simulated annealing and MCMC optimization algorithms, which have been proposed, for example, in [13]–[15], for solving stochastic programming problems.

In this paper, beyond the presentation of a simple example, we will not develop any ready-to-use optimization algorithm. Our results enable one to study the computational complexity of MCMC algorithms for stochastic optimization. However, the application of these results to create new efficient algorithms goes beyond the scope of the paper.

The appendix contains the technical proof of the main result. The reader is referred to the extended version of this note [16] for all other proofs. Some of the results of this paper were included in preliminary conference contributions [17], [18].

II. Approximate optimizers

Consider an optimization criterion $U : \Theta \to \mathbb{R}$, with $\Theta \subseteq \mathbb{R}^n$, and let

$$U^* := \sup_{\theta \in \Theta} U(\theta). \tag{1}$$

The following will be a standing assumption for all our results.

Assumption 1: Θ has finite Lebesgue measure. U is well defined point-wise, measurable, and bounded between 0 and 1 (i.e. $U(\theta) \in [0,\, 1]\ \forall \theta \in \Theta$).

For some results another assumption will be needed.

Assumption 2: Θ is compact. U is Lipschitz continuous.

We use L to denote the Lipschitz constant of U, i.e. $\forall \theta_1, \theta_2 \in \Theta,\ |U(\theta_1) - U(\theta_2)| \leq L\|\theta_1 - \theta_2\|$. Assumption 2 implies the existence of a global optimizer, i.e. under Assumption 2, we have $\Theta^* := \{\theta \in \Theta \mid U(\theta) = U^*\} \neq \emptyset$.

If, given an element θ in Θ, the value $U(\theta)$ can be computed directly, we say that U is a deterministic criterion. In problems involving random variables, the value $U(\theta)$ can be the expected value

$$U(\theta) = \int g(x, \theta) P_{\boldsymbol{x}}(dx; \theta) \tag{2}$$

of some function g which depends on both the optimization variable θ, and on some random variable $\boldsymbol{x}$ with probability distribution $P_{\boldsymbol{x}}(\cdot; \theta)$ (which may itself depend on θ). In such problems it is usually not possible to compute $U(\theta)$ directly. In this cases one must perform stochastic simulations and construct a Monte Carlo estimate of $U(\theta)$. The results of this paper apply in the same way to the optimization of both deterministic and expected-value criteria.

We introduce two different definitions of approximate solution to the optimization problem (1). The first is the definition of approximate domain optimizer. It will be essential in establishing finite-time guarantees on the performance of MCMC methods.

Definition 1: Let $\epsilon \geq 0$ and $\alpha \in [0, 1]$ be given numbers. Then θ is an approximate domain optimizer of U with value imprecision ϵ and residual domain α if

$$\lambda(\{\theta' \in \Theta : U(\theta') > U(\theta) + \epsilon\}) \leq \alpha \, \lambda(\Theta) \tag{3}$$

where λ denotes the Lebesgue measure.

That is, the function U takes values strictly greater than $U(\theta) + \epsilon$ only on a subset of values of θ no larger than an α portion of the optimization domain. If both α and ϵ are equal to zero then $U(\theta)$ coincides with the essential supremum of U [19]. We will use

$$\Theta(\epsilon, \alpha) := \{\theta \in \Theta \mid \lambda(\{\theta' \in \Theta \mid U(\theta') > U(\theta) + \epsilon\}) \leq \alpha\lambda(\Theta)\}$$

to denote the set of approximate domain optimizers with value imprecision ϵ and residual domain α.

Vidyasagar introduced in [3], [4] the similar definition of 'probably approximate near minimum' in order to obtain rigorous finite-time guarantees in the optimization of expected value criteria based on independent sampling of the optimization domain.

The following is a more common notion of approximate optimizer.

Definition 2: Let $\epsilon \geq 0$ be a given number. Then θ is an an approximate value optimizer of U with imprecision ϵ if $U(\theta') \leq U(\theta) + \epsilon$ for all $\theta' \in \Theta$.

This notion is commonly used in the stochastic programming literature [11], [12] and provides a direct bound on U^*: $\theta \in \Theta$ is an approximate value optimizer with imprecision $\epsilon > 0$ if and only if $U^* \leq U(\theta) + \epsilon$. We will use

$$\Theta^*(\epsilon) := \{\theta \in \Theta \mid \forall \theta' \in \Theta,\ U(\theta') \leq U(\theta) + \epsilon\}$$

to denote the set of approximate value optimizers with imprecision ϵ.

It is easy to see that for all ϵ if $\Theta^* \neq \emptyset$ then $\Theta^* \subseteq \Theta^*(\epsilon)$. Notice that $\Theta^*(\epsilon)$ does not coincide with $\Theta(\epsilon, 0)$. For all ϵ and all α, if $\Theta^*(\epsilon) \neq \emptyset$ then $\Theta^*(\epsilon) \subseteq \Theta(\epsilon, \alpha)$. Conversely, given an approximate domain optimizer it is in general not possible to draw any conclusions about the approximate value optimizers. A relation between domain and value approximate optimality can, however, be established under Assumption 2.

Theorem 1: Let Assumption 2 hold. Let θ be an approximate domain optimizer with value imprecision ϵ and residual domain α. Then, θ is also an approximate value optimizer with imprecision

$$\epsilon + \frac{L}{\sqrt{\pi}} \left[\frac{n}{2}\Gamma\left(\frac{n}{2}\right)\right]^{\frac{1}{n}} [\alpha\lambda(\Theta)]^{\frac{1}{n}}$$

where Γ denotes the gamma function.

The result allows us to select the value of α in such a way that an approximate domain optimizer with value imprecision ϵ and residual domain α is also an approximate value optimizer with imprecision 2ϵ. To do this, we need to select α so that $\frac{L}{\sqrt{\pi}} \left[\frac{n}{2}\Gamma\left(\frac{n}{2}\right)\right]^{\frac{1}{n}} [\alpha\lambda(\Theta)]^{\frac{1}{n}} \leq \epsilon$ hence

$$\alpha \leq \frac{\left[\frac{\epsilon\sqrt{\pi}}{L}\right]^n}{\lambda(\Theta)\left[\frac{n}{2}\Gamma\left(\frac{n}{2}\right)\right]}. \tag{4}$$

To illustrate the above inequality consider the case where the domain Θ is contained in an n-dimensional ball of radius R. Notice that under Assumption 2 the existence of such an R is guaranteed. In this case $\lambda(\Theta) = \frac{2\pi^{\frac{n}{2}}}{n\Gamma(\frac{n}{2})}R^n$. Therefore (4) becomes

$$\alpha \leq \left(\frac{1}{L}\frac{\epsilon}{R}\right)^n. \tag{5}$$

Note that, as n increases, α has to decrease to zero rapidly to ensure the required imprecision of the approximate value optimizer. In this case, α needs to decrease to zero as ϵ^n.

III. Optimization with MCMC: finite time guarantees

In simulated annealing, a random search based on the Metropolis-Hastings algorithm is carried out, such that the distribution of the elements of the domain visited during the search converges to an equilibrium distribution concentrated around the global optimizers.

Here we adopt equilibrium distributions defined by densities proportional to $[U(\theta)+\delta]^J$, where J and δ are strictly positive parameters. We use

$$\pi(d\theta; J, \delta) \propto [U(\theta) + \delta]^J \lambda(d\theta) \tag{6}$$

to denote this equilibrium distribution. The presence of δ is a technical condition required in the proof of our main result and will be discussed later on in this section. In our setting, the so-called 'zero-temperature' distribution is the limiting distribution $\pi(\,\cdot\,; J, \delta)$ for $J \to \infty$ denoted by π_∞. It can be shown that under some technical conditions, π_∞ is a uniform distribution on the set Θ^* of the global maximizers of U [20].

In Fig. 1, we illustrate two algorithms which implement Markov transition kernels with equilibrium distributions $\pi(\,\cdot\,; J, \delta)$. Algorithm I is the classical Metropolis-Hastings algorithm for the case in which U is a deterministic criterion [2]. Algorithm II is a suitably modified version of the Metropolis-Hastings algorithm for the case in which U is an expected-value criterion in the form of (2). This latter algorithm was devised by Müller [13], [15] and Doucet et al. [14].

In the simulated annealing scheme, one would simulate an inhomogeneous chain in which the equilibrium distributions tends to the zero-temperature distribution according to a suitably chosen 'cooling schedule' [21]–[27]. Usually, in an optimization problem defined over continuous variables, the set of global optimizers Θ^* has zero Lebesgue measure (e.g. a set of isolated points). This implies that, in general, convergence to the zero-temperature distribution on continuous domains can only be obtained in the weak sense, see [21, Theorem 3.3]. Notice that this is not the case for a finite domain, where the set of global optimizers is of non-null measure with respect to the reference counting measure [28]–[31].

Weak convergence to π_∞ implies that, asymptotically, $\boldsymbol{\theta}_k$ hits the set of approximate value optimizers $\Theta^*(\epsilon)$, for any $\epsilon > 0$, with probability one [21]–[25]. In recent works, bounds on the expected number of iterations before hitting $\Theta^*(\epsilon)$ [26], or on $P_{\boldsymbol{\theta}_k}(\Theta^*(\epsilon))$ [27], have been obtained. In [27], under some technical conditions, it is proven that $\forall \epsilon > 0$ there is a number C_ϵ such that $P_{\boldsymbol{\theta}_k}(\{\theta \in \Theta \mid U(\theta) \leq U^* - \epsilon\}) \leq C_\epsilon k^{-\frac{1}{3}}(1 + \log k)$ at each step k. In general, the expressions in these bounds cannot be computed. For example, in the bound reported here, C_ϵ is not known in advance. Hence, existing bounds can be used to asses the asymptotic rate of convergence as $k \to \infty$, i.e. as the number of steps grows to infinity, but not as stopping criteria.

Here we show that finite-time guarantees for stochastic optimization by MCMC methods on continuous domains can be obtained by selecting a distribution $\pi(\,\cdot\,; J, \delta)$ with a finite J as the target distribution in place of the zero-temperature distribution π_∞. Our definition of approximate

Algorithm I : MCMC for deterministic criteria

0 Assume that the current state of the chain is $\boldsymbol{\theta}_k$.

1 Generate a proposed state $\tilde{\boldsymbol{\theta}}_{k+1}$ according to $q_{\tilde{\boldsymbol{\theta}}}(\theta|\boldsymbol{\theta}_k)$.

2 Calculate the acceptance probability

$$\rho = \min\left\{ \frac{q_{\tilde{\boldsymbol{\theta}}}(\boldsymbol{\theta}_k|\tilde{\boldsymbol{\theta}}_{k+1})}{q_{\tilde{\boldsymbol{\theta}}}(\tilde{\boldsymbol{\theta}}_{k+1}|\boldsymbol{\theta}_k)} \frac{[U(\tilde{\boldsymbol{\theta}}_{k+1})+\delta]^J}{[U(\boldsymbol{\theta}_k)+\delta]^J}, 1 \right\} .$$

3 With probability ρ, accept the proposed state and set $\boldsymbol{\theta}_{k+1} = \tilde{\boldsymbol{\theta}}_{k+1}$. Otherwise leave the current state unchanged, i.e. set $\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k$.

Algorithm II : MCMC for expected-value criteria

0 Assume that the current state of the chain is $[\,\boldsymbol{\theta}_k, \{\boldsymbol{x}_k^{(j)}|j=1,\dots,J\}\,]$ where $\{\boldsymbol{x}_k^{(j)}|j=1,\dots,J\}$ are J independent extractions generated according to $P_{\boldsymbol{x}}(dx;\boldsymbol{\theta}_k)$.

1 Propose a new state $[\,\tilde{\boldsymbol{\theta}}_{k+1}, \{\tilde{\boldsymbol{x}}_{k+1}^{(j)}|j=1,\dots,J\}\,]$ where $\tilde{\boldsymbol{\theta}}_{k+1}$ is generated according to $q_{\tilde{\boldsymbol{\theta}}}(\theta|\boldsymbol{\theta}_k)$ and $\{\tilde{\boldsymbol{x}}_{k+1}^{(j)}|j=1,\dots,J\}$ are J independent extractions generated according to $P_{\boldsymbol{x}}(dx;\tilde{\boldsymbol{\theta}}_{k+1})$.

2 Calculate the acceptance probability

$$\rho = \min\left\{ \frac{q_{\tilde{\boldsymbol{\theta}}}(\boldsymbol{\theta}_k|\tilde{\boldsymbol{\theta}}_{k+1})}{q_{\tilde{\boldsymbol{\theta}}}(\tilde{\boldsymbol{\theta}}_{k+1}|\boldsymbol{\theta}_k)} \frac{\displaystyle\prod_{j=1}^{J}[g(\tilde{\boldsymbol{x}}_{k+1}^{(j)},\tilde{\boldsymbol{\theta}}_{k+1})+\delta]}{\displaystyle\prod_{j=1}^{J}[g(\boldsymbol{x}_{k}^{(j)},\boldsymbol{\theta}_{k})+\delta]}, 1 \right\}$$

3 With probability ρ, accept the proposed state and set $\boldsymbol{\theta}_{k+1} = \tilde{\boldsymbol{\theta}}_{k+1}$ and $\{\boldsymbol{x}_{k+1}^{(j)} = \tilde{\boldsymbol{x}}_{k+1}^{(j)}|j=1,\dots,J\}$. Otherwise leave the current state unchanged, i.e. set $\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k$ and $\{\boldsymbol{x}_{k+1}^{(j)} = \boldsymbol{x}_{k}^{(j)}|j=1,\dots,J\}$.

Fig. 1. The basic iterations of the Metropolis-Hastings algorithm with equilibrium distributions $\pi(\cdot;J,\delta)$ for the maximization of deterministic and expected-value criteria. In both algorithms, $q_{\tilde{\boldsymbol{\theta}}}(\cdot|\boldsymbol{\theta}_k)$ is the density of the 'proposal distribution'.

domain optimizer given in Section II is essential for establishing this result. The definition of approximate domain optimizers carries an important property, which holds regardless of what the criterion U is: if ϵ and α have non-zero values then the set of approximate global optimizers $\Theta(\epsilon,\alpha)$ always has non-zero Lebesgue measure. The following theorem establishes a lower bound on the measure of the set $\Theta(\epsilon,\alpha)$ with respect to a distribution $\pi(\cdot;J,\delta)$ with finite J. It is important to stress that the result holds universally for *any* optimization criterion U on a bounded domain. The only minor requirement is that U takes values in $[0,\,1]$.

Theorem 2: Let Assumption 1 hold. Let $\Theta(\epsilon,\alpha)$ be the set of approximate domain optimizers of U with value imprecision ϵ and residual domain α. Let $J \geq 1$ and $\delta > 0$, and consider the distribution $\pi(d\theta;J,\delta) \propto [U(\theta)+\delta]^J\lambda(d\theta)$. Then, for any $\alpha \in (0,\,1]$ and $\epsilon \in [0,\,1]$, the following inequality holds

$$\pi(\Theta(\epsilon,\alpha);J,\delta) \geq \frac{1}{1+\left[\dfrac{1+\delta}{\epsilon+1+\delta}\right]^J \left[\dfrac{1}{\alpha}\dfrac{1+\delta}{\epsilon+\delta}-1\right]\dfrac{1+\delta}{\delta}} . \tag{7}$$

Notice that, for given non-zero values of ϵ, α, and δ the right-hand side of (7) can be made arbitrarily close to 1 by choice of J. The importance of the choice of a target distribution $\pi(\,\cdot\,; J, \delta)$ with a finite J is that the total variation distance $\|P_{\boldsymbol{\theta}_k} - \pi(\,\cdot\,; J, \delta)\|_{\text{TV}}$ between the distribution of the state of the chain $P_{\boldsymbol{\theta}_k}$ and the target distribution $\pi(\,\cdot\,; J, \delta)$ is a meaningful quantity [10]. Convergence of the Metropolis-Hastings algorithm and MCMC methods in total variation distance is a well studied problem. The theory provides simple conditions under which one derives upper bounds on $\|P_{\boldsymbol{\theta}_k} - \pi(\,\cdot\,; J, \delta)\|_{\text{TV}}$ that decrease to zero as $k \to \infty$ [7]–[10]. It is then appropriate to introduce the following finite-time result.

Proposition 3: Let Assumption 1 hold. Let $\boldsymbol{\theta}_k$ with distribution $P_{\boldsymbol{\theta}_k}$ be the state of the chain of an MCMC algorithm with target distribution $\pi(\,\cdot\,; J, \delta)$. For given $\alpha \in (0,\,1]$, $\epsilon \in (0,\,1]$ and $\sigma \in (0,\,1)$, if

$$J \geq \frac{1+\epsilon+\delta}{\epsilon}\left[\log\frac{\sigma}{1-\sigma} + \log\frac{1}{\alpha} + 2\log\frac{1+\delta}{\delta}\right] \tag{8}$$

then,

$$P_{\boldsymbol{\theta}_k}(\Theta(\epsilon, \alpha); J, \delta) \geq \sigma - \|P_{\boldsymbol{\theta}_k} - \pi(\,\cdot\,; J, \delta)\|_{\text{TV}}\,.$$

In other words, the statement "$\boldsymbol{\theta}_k$ is an approximate domain optimizer of U with value imprecision ϵ and residual domain α" can be made with confidence $\sigma - \|P_{\boldsymbol{\theta}_k} - \pi(\,\cdot\,; J, \delta)\|_{\text{TV}}$.

Inequality (8) is derived from inequality (7) Then, the result follows directly from the definition of the total variation distance [10].

If the optimization criterion is Lipschitz continuous, Theorem 2 can be used together with Theorem 1 to derive a lower bound on the measure of the set of approximate value optimizers with a given imprecision. An example of such a bound is the following.

Proposition 4: Let Assumption 1 and 2 hold. In addition, assume that Θ is contained in an n-dimensional ball of radius R. Let $\boldsymbol{\theta}_k$ with distribution $P_{\boldsymbol{\theta}_k}$ be the state of the chain of an MCMC algorithm with target distribution $\pi(\,\cdot\,; J, \delta)$. For given $\epsilon \in (0,\,1]$ and $\sigma \in (0,\,1)$, if

$$J \geq \frac{1+\epsilon+\delta}{\epsilon}\left[\log\frac{\sigma}{1-\sigma} + n\log\left(\frac{LR}{\epsilon}\right) + 2\log\frac{1+\delta}{\delta}\right] \tag{9}$$

then

$$P_{\boldsymbol{\theta}_k}(\Theta^*(2\epsilon); J, \delta) \geq \sigma - \|P_{\boldsymbol{\theta}_k} - \pi(\,\cdot\,; J, \delta)\|_{\text{TV}}\,.$$

In other words, the statement "$\boldsymbol{\theta}_k$ is an approximate value optimizer of U with value imprecision 2ϵ" can be made with confidence $\sigma - \|P_{\boldsymbol{\theta}_k} - \pi(\,\cdot\,; J, \delta)\|_{\text{TV}}$.

The proof follows by substituting α with the right-hand side of (5) in (8) and from the definition of the total variation distance.

Finally, Theorem 2 provides a criterion for selecting the parameter δ in $\pi(\,\cdot\,;J,\delta)$. For given ϵ and α, there exists an optimal choice of δ which minimizes the value of J required to ensure $\pi(\Theta(\epsilon,\alpha);J,\delta)\geq\sigma$. The advantage of choosing the smallest J, consistent with the required σ, is computational. The exponent J coincides with the number of Monte Carlo simulations of random variable $\boldsymbol{x}$ which must be done at each step in Algorithm II. The smallest J reduces also the peakedness of $\pi(\cdot;J,\delta)$. In turn, reducing the peakedness of $\pi(\cdot;J,\delta)$ will decrease the number of steps required to achieve the desired reduction of $\|P_{\boldsymbol{\theta}_k}-\pi(\,\cdot\,;J,\delta)\|_{\text{TV}}$.

The optimal choice of δ is specified by the following result.

Proposition 5: For fixed $\epsilon>0$, $\alpha>0$, and $\sigma\in(0.5,\,1)$, the function

$$f(\delta)=\frac{1+\epsilon+\delta}{\epsilon}\left[\log\frac{\sigma}{1-\sigma}+\log\frac{1}{\alpha}+2\log\frac{1+\delta}{\delta}\right]\,,$$

i.e. the right hand side of inequality (8), is convex in δ and attains its global minimum at the unique solution (for δ) of the equation

$$\log\frac{1+\delta}{\delta}+\log\frac{\sqrt{\sigma}}{\sqrt{1-\sigma}}+\log\frac{1}{\sqrt{\alpha}}=\frac{1+\epsilon+\delta}{\delta(1+\delta)}.$$

For example, if $\epsilon=0.01$, $\alpha=0.01$ and $\sigma=0.99$, then one obtains $\delta=0.15$ and $J=1540$. Notice that the result of Proposition 5 holds also for inequality (9) provided that α in the statement of Proposition 5 is replaced by the right hand side of (5).

IV. Convergence

In this section we illustrate the statement of Propositions 3 and 4. We base the discussion on the simplest available result on the convergence of MCMC methods in total variation distance, taken from [9]. In this case, the proposal distribution, denoted by its density $q_{\tilde{\boldsymbol{\theta}}}(\theta|\boldsymbol{\theta}_k)$ in Algorithms I and II, is independent of the current state $\boldsymbol{\theta}_k$.

Theorem 6 ([9]): Let $P_{\boldsymbol{\theta}_k}$ be the distribution of the state of the chain in the Metropolis-Hastings algorithm with an independent proposal distribution. Let π denote the target distribution. Let p and q denote respectively the density of π and the density of the proposal distribution and assume that $p(\theta)>0,\,\forall\theta\in\Theta$ and $q(\theta)>0,\,\forall\theta\in\Theta$. If there exists M such that $p(\theta)\leq Mq(\theta)\,,\ \forall\theta\in\Theta$, then

$$\|\pi-P_{\boldsymbol{\theta}_k}\|_{\text{TV}}\leq\left(1-\frac{1}{M}\right)^k. \tag{10}$$

Proof: See [9, Theorem 2.1], or [2, Theorem7.8].

Here, we chose $q_{\tilde{\boldsymbol{\theta}}}$ as the uniform distribution over Θ. Sampling using an independent uniform proposal distribution is a naïve strategy in an MCMC approach and cannot be expected to perform efficiently [2]. However, it allows us to present some simple illustrative examples where convergence bounds can be derived with a few basic steps.

In some cases the naïve strategy can produce approximate domain optimizers very efficiently. One such case occurs under the assumption that the optimization criterion $U(\theta)$ has a 'flat top', i.e. the set of global optimizers Θ^* has non-zero Lebesgue measure. The same assumption has been used in [21, Theorem 4.2] to obtain the strong convergence of simulated annealing on a continuous domain. In this case, the application of Theorem 6 provides the following result.

Proposition 7: Let the notation and assumptions of Proposition 3 hold. In particular, assume that $\boldsymbol{\theta}_k$ is the state of the chain of the Metropolis-Hastings algorithm with independent uniform proposal distribution. In addition, given $\rho \in (0,\,1)$, let $\sigma = (1+\gamma)\rho$ for some $\gamma \in (0,\,\frac{1-\rho}{\rho})$. Let Θ^* be the set of global optimizer of U and assume that $\lambda(\Theta^*) \geq \beta\lambda(\Theta)$ for some $\beta \in (0,1)$. If

$$k \geq \frac{\log \gamma\rho}{\log(1-\beta)} \tag{11}$$

then $P_{\boldsymbol{\theta}_k}(\Theta(\epsilon,\alpha); J, \delta) \geq \rho$.

In (11), it is convenient to choose $\gamma \approx \frac{1-\rho}{\rho}$. Hence, the number of iterations grows approximately as $-\log(1-\rho) = \log(\frac{1}{1-\rho})$ and $-\frac{1}{\log(1-\beta)}$ and is independent of ϵ and α. In Algorithm II the total number of required samples of $\boldsymbol{x}$ is given by the number of iterations multiplied by J. In this case, it can be shown that a nearly optimal choice is $\gamma = \frac{1}{2}\frac{1-\rho}{\rho}$. Hence, using (8) for the case of approximate domain optimization, we obtain that the required samples of $\boldsymbol{x}$ grow as $\frac{1}{\epsilon}$, $\log\frac{1}{\alpha}$, and approximately as $(\log\frac{1}{1-\rho})^2$. Instead, using (9) for the case of approximate value optimization, we obtain that the required samples of $\boldsymbol{x}$ grow as $\frac{1}{\epsilon}\log\frac{1}{\epsilon}$, $(\log\frac{1}{1-\rho})^2$, $\log LR$ and n.

If the 'flat top' condition is not met it can be easily seen that the use of a uniform proposal distribution can lead to an exponential number of iterations. In the general case, by applying Theorem 6 we obtain the following result.

Proposition 8: Let the notation and assumptions of Proposition 3 hold. In particular, assume that $\boldsymbol{\theta}_k$ is the state of the chain of the Metropolis-Hastings algorithm with independent uniform proposal distribution. In addition, given $\rho \in (0,\,1)$, let $\sigma = (1+\gamma)\rho$ for some $\gamma \in (0,\,\frac{1-\rho}{\rho})$. If $k \geq \left(\frac{1+\delta}{\delta}\right)^J \log\left(\frac{1}{\gamma\rho}\right)$ or, equivalently,

$$k \geq \left[\frac{(1+\gamma)\rho}{1-(1+\gamma)\rho}\frac{1}{\alpha}\left(\frac{1+\delta}{\delta}\right)^2\right]^{\frac{1+\epsilon+\delta}{\epsilon}\log\left(\frac{1+\delta}{\delta}\right)} \log\frac{1}{\gamma\rho} \tag{12}$$

then $P_{\boldsymbol{\theta}_k}(\Theta(\epsilon,\alpha); J, \delta) \geq \rho$.

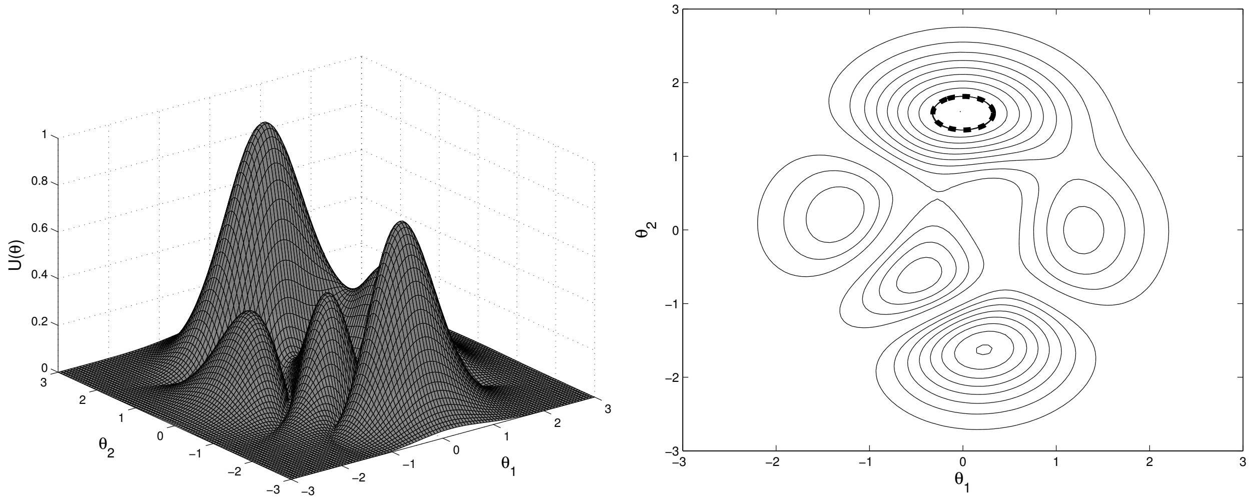

Fig. 2. Function $U(\theta)$ (left panel) and its level sets (right panel). The 0.9 level set is highlighted as a dashed ellipse.

Hence, the number of iterations turns out to be exponential in $\frac{1}{\epsilon}$. Therefore, using Theorem 6 with $q_{\tilde{\boldsymbol{\theta}}}$ as the independent uniform proposal distribution, the only general bounds that we can guarantee are exponential.

V. Numerical example

Let $\theta \in \Theta = [-3,3] \times [-3,3]$ and consider the function $V(\theta) = 3(1-\theta_1)^2 e^{-\theta_1^2-(\theta_2+1)^2} - 10(\frac{\theta_1}{5} - \theta_1^3 - \theta_2^5)e^{-\theta_1^2-\theta_2^2} - \frac{1}{3}e^{-(\theta_1+1)^2-\theta_2^2}$ (the Matlab function `peaks`). We define the function $U : \Theta \to [0,1]$ by $U(\theta) = \frac{|V(\theta)|}{\max_{\theta' \in \Theta} |V(\theta')|}$. The scaling factor $\max_{\theta' \in \Theta} |V(\theta')| = 8.1062$ and a Lipschitz constant of $U(\theta)$, $L = 1.725$, were computed numerically using a grid on Θ. Multiplicative noise was added using the function $g(\boldsymbol{x}, \theta) = (1+\boldsymbol{x})U(\theta)$ where $\boldsymbol{x}$ is normally distributed with mean 0 and variance 0.25. The objective is to maximize the expected value of $g(\boldsymbol{x}, \theta)$ which is indeed equal to $U(\theta)$. The function U and its level sets are shown in Fig. 2. The 0.9 level set, which coincides with $\Theta^*(0.1)$, is highlighted in the figure.

The MCMC Algorithm II of Fig. 1 was applied to this function. The design parameter $\delta = 0.1$ and an independent uniform proposal distribution q were used throughout. To demonstrate the convergence of the algorithm, $2,000$ independent runs of the algorithm, of $10,000$ steps each, were generated. We then computed the fraction of runs that found themselves in $\Theta^*(0.1)$ at different time points; for simplicity we refer to this fraction as the 'success rate'. In all cases the success rate quickly settled to a steady state value, suggesting that the algorithm converged. To demonstrate the bound of Proposition 4 the steady state success rate as a function of the exponent J is reported in Fig. 3; more precisely, the figure shows the decay of 1 minus the

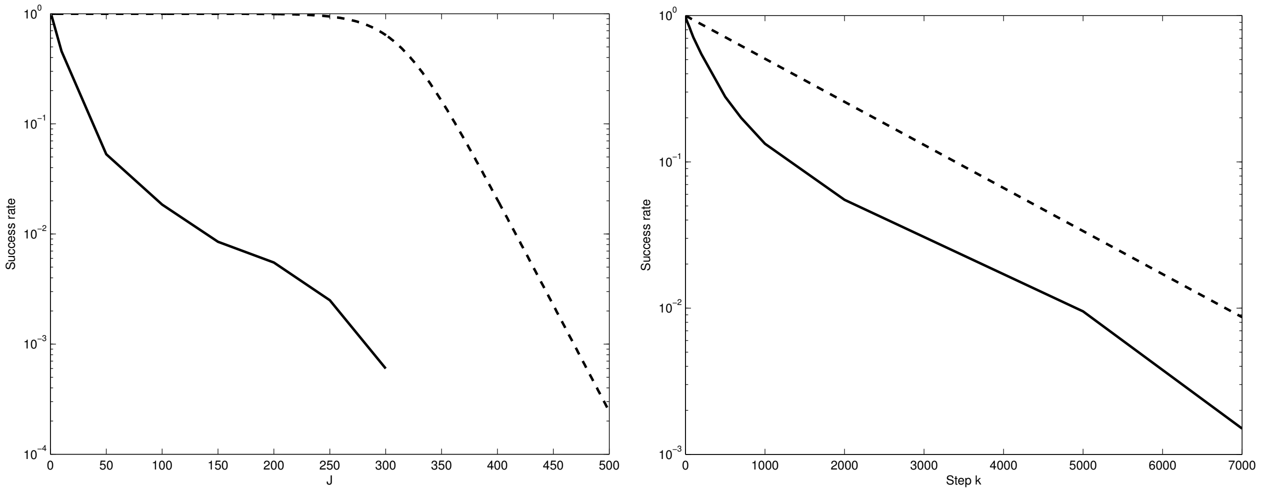

Fig. 3. Left panel: Decay of $(1-$ success rate$)$ as a function of the exponent J. Empirical value (solid) and the bound based on Proposition 4 (dashed). Right panel: Success rate for $J = 100$ at each step (solid) and the bound (10) (dashed). In both panels the plots are in semi logarithmic scales.

steady state success rate as a function of J. The figure also shows the corresponding theoretical bound based on Proposition 4. In the right panel of Fig. 3 we concentrate on the case $J = 100$ and plot the absolute value of the difference between the success rate at different time points and the steady state success rate. According to Theorem 6, one would expect this difference to decay to 0 geometrically at a rate $1 - \frac{1}{M}$. For comparison purposes, the corresponding curve, for the numerically estimated value $M = 1475$, is also plotted.

VI. Conclusions

In this paper, we have introduced a novel approach for obtaining rigorous finite-time guarantees on the performance of MCMC algorithms in the optimization of functions of continuous variables. In particular we have established the values of the the temperature parameter in the target distribution which allow one to reach a solution, which is within the desired level of approximation with the desired confidence, in a finite number of steps. Our work was motivated by the MCMC algorithm (Algorithm II), introduced in [13]–[15], for solving stochastic optimization problems. Our results enable novel research on the development of efficient MCMC algorithms for the solution of stochastic programming problems with rigorous finite-time guarantees. The extended version of this paper [16] contains some observations on the computational complexity of MCMC algorithm for optimization which can be drawn from our results, and a preliminary comparison with some other approaches to stochastic optimization.

Acknowledgments

Work supported by EPSRC, Grant EP/H021558/1 and EP/C014006/1, and by the European Commission under projects HYGEIA FP6-NEST-4995 and iFly FP6-TREN-037180.

Appendix

Proof of Theorem 2: Let $\bar{\alpha} \in (0,1]$ and $\rho \in (0,1]$ be given numbers. To simplify the notation, let $U_\delta(\theta) := U(\theta) + \delta$ and let π_δ be a normalized measure such that $\pi_\delta(d\theta) \propto U_\delta(\theta)\lambda(d\theta)$, i.e. $\pi_\delta(d\theta) := \pi(d\theta; 1, \delta)$. In the first part of the proof we establish a lower bound on

$$\pi\left(\{\theta \in \Theta \mid \pi_\delta(\{\theta' \in \Theta \mid \rho\, U_\delta(\theta') > U_\delta(\theta)\}) \le \bar{\alpha}\}; J, \delta\right) .$$

Let $y_{\bar{\alpha}} := \inf\{y \mid \pi_\delta(\{\theta \in \Theta \mid U_\delta(\theta) \le y\}) \ge 1 - \bar{\alpha}\}$. To start with we show that the set $\{\theta \in \Theta \mid \pi_\delta(\{\theta' \in \Theta \mid \rho\, U_\delta(\theta') > U_\delta(\theta)\}) \le \bar{\alpha}\}$ coincides with $\{\theta \in \Theta \mid U_\delta(\theta) \ge \rho\, y_{\bar{\alpha}}\}$. Notice that the quantity $\pi_\delta(\{\theta \in \Theta \mid U_\delta(\theta) \le y\})$ is a non decreasing right continuous function of y because it has the form of a distribution function (see e.g. [32, p. 162], see also [4, Lemma 11.1]). Therefore we have $\pi_\delta(\{\theta \in \Theta \mid U_\delta(\theta) \le y_{\bar{\alpha}}\}) \ge 1 - \bar{\alpha}$ and

$$y \ge \rho\, y_{\bar{\alpha}} \;\Rightarrow\; \pi_\delta(\{\theta' \in \Theta \mid \rho\, U_\delta(\theta') \le y\}) \ge 1 - \bar{\alpha} \;\Rightarrow\; \pi_\delta(\{\theta' \in \Theta \mid \rho\, U_\delta(\theta') > y\}) \le \bar{\alpha} \, .$$

Moreover,

$$y < \rho\, y_{\bar{\alpha}} \;\Rightarrow\; \pi_\delta(\{\theta' \in \Theta \mid \rho\, U_\delta(\theta') \le y\}) < 1 - \bar{\alpha} \;\Rightarrow\; \pi_\delta(\{\theta' \in \Theta \mid \rho\, U_\delta(\theta') > y\}) > \bar{\alpha}$$

and taking the contrapositive one obtains $\pi_\delta(\{\theta' \in \Theta \mid \rho\, U_\delta(\theta') > y\}) \le \bar{\alpha} \;\Rightarrow\; y \ge \rho\, y_{\bar{\alpha}}$. Therefore $\{\theta \in \Theta \mid U_\delta(\theta) \ge \rho\, y_{\bar{\alpha}}\} = \{\theta \in \Theta \mid \pi_\delta(\{\theta' \in \Theta \mid \rho\, U_\delta(\theta') > U_\delta(\theta)\}) \le \bar{\alpha}\}$.

We now derive a lower bound on $\pi\left(\{\theta \in \Theta \mid U_\delta(\theta) \ge \rho\, y_{\bar{\alpha}}\}; J, \delta\right)$. Let us introduce the notation $A_{\bar{\alpha}} := \{\theta \in \Theta \mid U_\delta(\theta) < y_{\bar{\alpha}}\}$, $\bar{A}_{\bar{\alpha}} := \{\theta \in \Theta \mid U_\delta(\theta) \ge y_{\bar{\alpha}}\}$, $B_{\bar{\alpha},\rho} := \{\theta \in \Theta \mid U_\delta(\theta) < \rho\, y_{\bar{\alpha}}\}$ and $\bar{B}_{\bar{\alpha},\rho} := \{\theta \in \Theta \mid U_\delta(\theta) \ge \rho\, y_{\bar{\alpha}}\}$. Notice that $B_{\bar{\alpha},\rho} \subseteq A_{\bar{\alpha}}$ and $\bar{A}_{\bar{\alpha}} \subseteq \bar{B}_{\bar{\alpha},\rho}$. The quantity $\pi_\delta(\{\theta \in \Theta \mid U_\delta(\theta) < y\})$ as a function of y is the left continuous version of $\pi_\delta(\{\theta \in \Theta \mid U_\delta(\theta) \le y\})$ [32, p. 162]. Hence, the definition of $y_{\bar{\alpha}}$ implies $\pi_\delta(A_{\bar{\alpha}}) \le 1 - \bar{\alpha}$ and $\pi_\delta(\bar{A}_{\bar{\alpha}}) \ge \bar{\alpha}$. Notice that

$$\begin{aligned}
\pi_\delta(A_{\bar{\alpha}}) \le 1 - \bar{\alpha} \quad &\Rightarrow \quad \frac{\delta\lambda(A_{\bar{\alpha}})}{\left[\int_\Theta U_\delta(\theta)\lambda(d\theta)\right]} \le 1 - \bar{\alpha} \qquad \text{because } U(\theta) \ge 0\ \forall\theta \, , \\
\pi_\delta(\bar{A}_{\bar{\alpha}}) \ge \bar{\alpha} \quad &\Rightarrow \quad \frac{(1+\delta)\lambda(\bar{A}_{\bar{\alpha}})}{\left[\int_\Theta U_\delta(\theta)\lambda(d\theta)\right]} \ge \bar{\alpha} \qquad \text{because } U(\theta) \le 1\ \forall\theta \, .
\end{aligned}$$

Hence, $\lambda(\bar{A}_{\bar{\alpha}}) > 0$ and

$$\frac{\lambda(A_{\bar{\alpha}})}{\lambda(\bar{A}_{\bar{\alpha}})} \le \frac{1-\bar{\alpha}}{\bar{\alpha}} \frac{1+\delta}{\delta} \, .$$

Notice that $\lambda(\bar{A}_{\bar{\alpha}}) > 0$ implies $\lambda(\bar{B}_{\bar{\alpha},\rho}) > 0$. We obtain

$$\begin{aligned}
\pi\left(\{\theta \in \Theta \mid U_\delta(\theta) \geq \rho\, y_{\bar{\alpha}}\}; J, \delta\right) &= \pi\left(\bar{B}_{\bar{\alpha},\rho}; J, \delta\right) = \frac{\int_{\bar{B}_{\bar{\alpha},\rho}} U_\delta(\theta)^J \lambda(d\theta)}{\int_\Theta U_\delta(\theta)^J \lambda(d\theta)} \\
&= \frac{1}{1 + \dfrac{\int_{B_{\bar{\alpha},\rho}} U_\delta(\theta)^J \lambda(d\theta)}{\int_{\bar{B}_{\bar{\alpha},\rho}} U_\delta(\theta)^J \lambda(d\theta)}} \\
&\geq \frac{1}{1 + \dfrac{\int_{B_{\bar{\alpha},\rho}} U_\delta(\theta)^J \lambda(d\theta)}{\int_{\bar{A}_{\bar{\alpha}}} U_\delta(\theta)^J \lambda(d\theta)}} \\
&\geq \frac{1}{1 + \dfrac{\rho^J y_{\bar{\alpha}}^J}{y_{\bar{\alpha}}^J} \dfrac{\lambda(B_{\bar{\alpha},\rho})}{\lambda(\bar{A}_{\bar{\alpha}})}} \\
&\geq \frac{1}{1 + \rho^J \dfrac{\lambda(A_{\bar{\alpha}})}{\lambda(\bar{A}_{\bar{\alpha}})}} \\
&\geq \frac{1}{1 + \rho^J \dfrac{1-\bar{\alpha}}{\bar{\alpha}} \dfrac{1+\delta}{\delta}} \, .
\end{aligned}$$

Since $\{\theta \in \Theta \mid U_\delta(\theta) \geq \rho\, y_{\bar{\alpha}}\} = \{\theta \in \Theta \mid \pi_\delta(\{\theta' \in \Theta \mid \rho\, U_\delta(\theta') > U_\delta(\theta)\}) \leq \bar{\alpha}\}$ the first part of the proof is complete.

In the second part of the proof we show that the set $\{\theta \in \Theta \mid \pi_\delta(\{\theta' \in \Theta \mid \rho\, U_\delta(\theta') > U_\delta(\theta)\}) \leq \bar{\alpha}\}$ is contained in the set of approximate domain optimizers of U with value imprecision $\tilde{\epsilon} := (\rho^{-1} - 1)(1+\delta)$ and residual domain $\tilde{\alpha} := \frac{1+\delta}{\tilde{\epsilon}+\delta}\, \bar{\alpha}$. Hence, we show that

$$\begin{aligned}
&\{\theta \in \Theta \mid \pi_\delta(\{\theta' \in \Theta \mid \rho\, U_\delta(\theta') > U_\delta(\theta)\}) \leq \bar{\alpha}\} \subseteq \\
&\qquad \{\theta \in \Theta \mid \lambda(\{\theta' \in \Theta \mid U(\theta') > U(\theta) + \tilde{\epsilon}\}) \leq \tilde{\alpha}\, \lambda(\Theta)\} \, .
\end{aligned}$$

We have $U(\theta') > U(\theta) + \tilde{\epsilon} \quad \Leftrightarrow \quad \rho\, U_\delta(\theta') > \rho\, [U_\delta(\theta) + \tilde{\epsilon}] \quad \Rightarrow \quad \rho\, U_\delta(\theta') > U_\delta(\theta)$ which is proven by noticing that $\rho\, [U_\delta(\theta) + \tilde{\epsilon}] \geq U_\delta(\theta) \Leftrightarrow (1-\rho) \geq U(\theta)(1-\rho)$ and $U(\theta) \in [0,\, 1]$. Hence, $\{\theta' \in \Theta \mid \rho\, U_\delta(\theta') > U_\delta(\theta)\} \quad \supseteq \quad \{\theta' \in \Theta \mid U(\theta') > U(\theta) + \tilde{\epsilon}\}$. Therefore,

$$\pi_\delta(\{\theta' \in \Theta \mid \rho\, U_\delta(\theta') > U_\delta(\theta)\}) \leq \bar{\alpha} \quad \Rightarrow \quad \pi_\delta(\{\theta' \in \Theta \mid U(\theta') > U(\theta) + \tilde{\epsilon}\}) \leq \bar{\alpha} \, .$$

Let $Q_{\theta,\tilde{\epsilon}} := \{\theta' \in \Theta \mid U(\theta') > U(\theta) + \tilde{\epsilon}\}$ and notice that

$$\pi_\delta(\{\theta' \in \Theta \mid U(\theta') > U(\theta) + \tilde{\epsilon}\}) = \frac{\displaystyle\int_{Q_{\theta,\tilde{\epsilon}}} U(\theta')\lambda(d\theta') + \delta\lambda(Q_{\theta,\tilde{\epsilon}})}{\displaystyle\int_\Theta U(\theta')\lambda(d\theta') + \delta\lambda(\Theta)} \, .$$

We obtain

$$\begin{aligned}\pi_\delta(\{\theta' \in \Theta \mid U(\theta') > U(\theta) + \tilde\epsilon\}) \le \bar\alpha \;&\Rightarrow\; \tilde\epsilon\,\lambda(Q_{\theta,\tilde\epsilon}) + \delta\lambda(Q_{\theta,\tilde\epsilon}) \le \bar\alpha(1+\delta)\lambda(\Theta)\\ &\Rightarrow\; \lambda(\{\theta' \in \Theta \mid U(\theta') > U(\theta) + \tilde\epsilon\}) \le \tilde\alpha\,\lambda(\Theta)\,.\end{aligned}$$

Hence we can conclude that

$$\pi_\delta(\{\theta' \in \Theta \mid \rho\, U_\delta(\theta') > U_\delta(\theta)\}) \le \bar\alpha \;\Rightarrow\; \lambda(\{\theta' \in \Theta \mid U(\theta') > U(\theta) + \tilde\epsilon\}) \le \tilde\alpha\,\lambda(\Theta)$$

and the second part of the proof is complete.

We have shown that given $\bar\alpha \in (0,\, 1]$, $\rho \in (0,\, 1]$, $\tilde\epsilon := (\rho^{-1} - 1)(1+\delta)$ and $\tilde\alpha := \frac{1+\delta}{\tilde\epsilon+\delta}\,\bar\alpha$, then

$$\pi\left(\Theta(\tilde\epsilon, \tilde\alpha); J, \delta\right) \ge \frac{1}{1 + \rho^J \dfrac{1-\bar\alpha}{\bar\alpha}\dfrac{1+\delta}{\delta}} = \frac{1}{1 + \left[\dfrac{1+\delta}{\tilde\epsilon + 1 + \delta}\right]^J \left[\dfrac{1}{\tilde\alpha}\dfrac{1+\delta}{\tilde\epsilon+\delta} - 1\right]\dfrac{1+\delta}{\delta}}\,.$$

Notice that $\tilde\epsilon \in [0,\, 1]$ and $\tilde\alpha \in (0,\, 1]$ are linked through a bijective relation to $\rho \in [\frac{1+\delta}{2+\delta},\, 1]$ and $\bar\alpha \in (0,\, \frac{\tilde\epsilon+\delta}{1+\delta}]$. Hence, the statement of the theorem is eventually obtained by setting the desired $\tilde\epsilon = \epsilon$ and $\tilde\alpha = \alpha$ in the above inequality. ■

References

[1] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by Simulated Annealing. *Science*, 220(4598):671–680, 1983.

[2] C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer-Verlag, New York, second edition, 2004.

[3] M. Vidyasagar. Randomized algorithms for robust controller synthesis using statistical learning theory. *Automatica*, 37(10):1515–1528, 2001.

[4] M. Vidyasagar. *Learning and Generalization: With Application to Neural Networks*. Springer-Verlag, London, second edition, 2003.

[5] R. Tempo, G. Calafiore, and F. Dabbene. *Randomized Algorithms for Analysis and Control of Uncertain Systems*. Springer-Verlag, London, 2005.

[6] T. Alamo, R. Tempo, and E.F. Camacho. A Randomized Strategy for Probabilistic Solutions of Uncertain Feasibility and Optimization Problems. *IEEE Trans. Autom. Control*, 54(11):2545–2559, 2009.

[7] S. P. Meyn and R. L. Tweedie. *Markov Chains and Stochastic Stability*. Springer-Verlag, London, 1993.

[8] J. S. Rosenthal. Minorization Conditions and Convergence Rates for Markov Chain Monte Carlo. *J. Am. Stat. Assoc.*, 90(430):558–566, 1995.

[9] K. L. Mengersen and R. L. Tweedie. Rates of convergence of the Hastings and Metropolis algorithm. *Ann. Stat.*, 24(1):101–121, 1996.

[10] G. O. Roberts and J. S. Rosenthal. General state space Markov chains and MCMC algorithms. *Prob. Surv.*, 1:20–71, 2004.

[11] A. Shapiro. Stochastic programming approach to optimization under uncertainty. *Math. Program., Ser. B*, 112:183–220, 2008.

[12] Y. Nesterov and J.-Ph. Vial. Confidence level solutions for stochastic programming. *Automatica*, 44(6):1559–1568, 2008.

[13] P. Müller. Simulation based optimal design. In J. O. Berger, J. M. Bernardo, A. P. Dawid, and A. F. M. Smith, editors, *Bayesian Statistics 6: proceedings of the Sixth Valencia International Meeting*, pages 459–474. Oxford: Clarendon Press, 1999.

[14] A. Doucet, S.J. Godsill, and P. Robert. Marginal maximum a posteriori estimation using Markov chain simulation. *Statist. Comput.*, 12:77–84, 2002.

[15] P. Müller, B. Sansó, and M. De Iorio. Optimal Bayesian design by Inhomogeneous Markov Chain Simulation. *J. Am. Stat. Assoc.*, 99(467):788–798, 2004.

[16] A. Lecchini-Visintini, J. Lygeros, and J. M. Maciejowski. Stochastic optimization on continuous domains with finite-time guarantees by Markov chain Monte Carlo methods, 2010. Extended version: `arXiv:0906.1055 [math.OC]`.

[17] A. Lecchini-Visintini, J. Lygeros, and J. M. Maciejowski. Simulated annealing: Rigorous finite-time guarantees for optimization on continuous domains. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*. MIT Press, Cambridge, MA, 2008.

[18] A. Lecchini-Visintini, J. Lygeros, and J. M. Maciejowski. On the approximate domain optimization of deterministic and expected value criteria. In *47th IEEE Conference on Decision and Control, Cancun, Mexico*, 2008.

[19] T. Rowland. Essential Supremum. From MathWorld–A Wolfram Web Resource, created by Eric W. Weisstein. http://mathworld.wolfram.com/EssentialSupremum.html.

[20] C.-R. Hwang. Laplace's method revisited: Weak convergence of probability measures. *Ann. Prob.*, 8(6):1177–1182, 1980.

[21] H. Haario and E. Saksman. Simulated annealing process in general state space. *Adv. Appl. Prob.*, 23:866–893, 1991.

[22] S. B. Gelfand and S. K. Mitter. Simulated Annealing Type Algorithms for Multivariate Optimization. *Algorithmica*, 6:419–436, 1991.

[23] C. Tsallis and D. A. Stariolo. Generalized simulated annealing. *Physica A*, 233:395–406, 1996.

[24] M. Locatelli. Simulated Annealing Algorithms for Continuous Global Optimization: Convergence Conditions. *J. Optimiz. Theory App.*, 104(1):121–133, 2000.

[25] C. Andrieu, L.A. Breyer, and A. Doucet. Convergence of simulated annealing using Foster-Lyapunov criteria. *J. App. Prob.*, 38:975–994, 2001.

[26] M. Locatelli. Convergence and first hitting time of simulated annealing algorithms for continuous global optimization. *Math. Meth. Oper. Res.*, 54:171–199, 2001.

[27] S. Rubenthaler, T. Rydén, and M. Wiktorsson. Fast simulated annealing in $\mathbb{R}^d$ with an application to maximum likelihood estimation in state-space models. *Stochastic Process. Appl.*, 119:19121931, 2009.

[28] P. M. J. van Laarhoven and E. H. L. Aarts. *Simulated Annealing: Theory and Applications*. D. Reidel Publishing Company, Dordrecht, Holland, 1987.

[29] D. Mitra, F. Romeo, and A. Sangiovanni-Vincentelli. Convergence and finite-time behavior of simulated annealing. *Adv. Appl. Prob.*, 18:747–771, 1986.

[30] B. Hajek. Cooling schedules for optimal annealing. *Math. Oper. Res.*, 13:311–329, 1988.

[31] J. Hannig, E. K. P. Chong, and S. R. Kulkarni. Relative Frequencies of Generalized Simulated Annealing. *Math. Oper. Res.*, 31(1):199–216, 2006.

[32] B.V. Gnedenko. *Theory of Probability*. Chelsea, New York, fourth edition, 1968.