Simulated annealing: a review and a new scheme

Thomas Guilmeau, Emilie Chouzenoux, Víctor Elvira

SIMULATED ANNEALING: A REVIEW AND A NEW SCHEME

Thomas Guilmeau[†]*, Emilie Chouzenoux*[†]*, and Víctor Elvira*[*]

† Université Paris-Saclay, CentraleSupélec, Inria, CVN, Gif-sur-Yvette, France
∗ School of Mathematics, University of Edinburgh, United Kingdom

ABSTRACT

Finding the global minimum of a nonconvex optimization problem is a notoriously hard task appearing in numerous applications, from signal processing to machine learning. Simulated annealing (SA) is a family of stochastic optimization methods where an artificial temperature controls the exploration of the search space while preserving convergence to the global minima. SA is efficient, easy to implement, and theoretically sound, but suffers from a slow convergence rate. The purpose of this work is two-fold. First, we provide a comprehensive overview on SA and its accelerated variants. Second, we propose a novel SA scheme called *curious simulated annealing*, combining the assets of two recent acceleration strategies. Theoretical guarantees of this algorithm are provided. Its performance with respect to existing methods is illustrated on practical examples.

1. INTRODUCTION

Optimization is at the core of many problems in signal processing and machine learning, e.g., in signal restoration, supervised learning, dictionary learning, or image segmentation, to name a few. In those problems, nonconvexity can arise from sparsity penalty [1], low-rank prior [2], non-linear observation model [3], non-linear regressor [4], blind thus multi-linear problem structure [5], or discrete variables [6].

Nonconvex optimization problems can present many local minima, which can "trap" the algorithms iterates and be of poor quality with respect to the problem at hand. Thus, global optimization methods must be sought for, to escape local minima and thus finding the global solution. To do so, stochasticity is often a key ingredient. The stochastic optimization algorithms builds a sequence of random variables converging to the global minima. In this paper, we focus on an important family of stochastic methods for global optimization, called simulated annealing (SA), relying on the key concept of *annealing*, a concept in physics describing the cooling of a solid until reaching the configuration of minimal energy.

Note that SA is strongly related to the family of methods known as graduated nonconvexity (GNC) or continuation methods, that rely on a deterministic annealing procedure. Even if in some cases, these methods have been shown to beat SA [7], still few theoretical results support GNC [8]. More theoretically sounded deterministic approaches for global optimization are branch and bound (B&B) and particle swarm optimization (PSO) methods. In B&B, the original problem is split into subproblems (i.e., branching) associated to different locations of the space, for which the computation of a lower bound (i.e., bounding) gives information to create new branches [9, 10]. B&B methods share connexions with the stochastic approximation simulated annealing from [11], also relying on a splitting of the space. In PSO, a population of particles exchange information in order to find the optimum of the objective [12]. Population-based implementations of SA are actually strongly related to PSO [13].

The contribution of this paper is twofold. First, we review SA-based global optimization strategies in an unifying manner. We focused on the historical SA [14], fast simulated annealing (FSA) [15] and sequential Monte Carlo simulated annealing (SMC-SA) [16], which we retained for being highly generic methods with solid convergence guarantees. Then, building upon FSA and SMC-SA, a new scheme, called curious SA (CSA), is proposed. We show that CSA inherits from the best convergence guarantees of its ancestors SMC-SA and FSA. We also illustrate its performance on numerical examples. The rest of the paper is organized as follows. Section 2 formulates the problem and introduces notations. Then, Section 3 presents SA, FSA, and SMC-SA, along with their theoretical guarantees. In Section 4, we introduce our algorithm CSA and its convergence theorem. Algorithms performance are compared on two challenging nonconvex problems in Section 5. Finally, conclusions are drawn in Section 6.

2. PROBLEM STATEMENT

We consider the optimization problem given by

$$\text{minimize}_{x\in\mathcal{X}} f(x), \tag{1}$$

where $\mathcal{X}$ is a non-empty subset of $\mathbb{R}^d$. Although this is not always necessary [17], $\mathcal{X}$ is assumed compact. The objective function f is supposed to be defined on $\mathcal{X}$, continuous and thus bounded from below and above on $\mathcal{X}$. Further, we

This work was supported by the European Research Council Starting Grant MAJORIS ERC-2019-STG-850925 and by the *Agence Nationale de la Recherche* of France under PISCES project (ANR-17-CE40-0031-01).

suppose, without lack of generality, that $f(x) \geq 0$ for every $x \in \mathcal{X}$, and that the set $S_* := \{x \in \mathcal{X},\ f(x) = 0\}$ is non-empty with null Lebesgue measure. For $\epsilon > 0$, we also denote $S_\epsilon := \{x \in \mathcal{X},\ f(x) \leq \epsilon\}$ the lower level sets of f.

Throughout this work, the gap between the infimum and the supremum of f is denoted $\Delta f := \sup_{x\in\mathcal{X}} f(x)$. $||\cdot||_\infty$ is the supremum norm and $||\cdot||_2$ is the euclidean norm. The Borel algebra of $\mathcal{X}$ is denoted $\mathcal{B}(\mathcal{X})$. $\mathcal{M}(\mathcal{X})$ is the set of probability measures on $(\mathcal{X}, \mathcal{B}(\mathcal{X}))$. $||\cdot||_{TV}$ is the total variation norm on $\mathcal{M}(\mathcal{X})$. The binary indicator function of a set A is denoted ι_A, and takes the value 1 for $x \in A$ and 0 elsewhere. For $x \in \mathcal{X}$, $A \in \mathcal{B}(\mathcal{X})$, the Dirac measure $\delta_x \in \mathcal{M}(\mathcal{X})$ is such that $\delta_x(A) = 1$ if and only if $x \in A$. We also introduce the function $(\cdot)_+$ such that for every $x \in \mathbb{R}$, $(x)_+ = \max(0, x)$. Further, given a Markov kernel $M : \mathcal{X} \times \mathcal{B}(\mathcal{X}) \to \mathbb{R}^+$, and $m \in \mathcal{M}(\mathcal{X})$, $mM \in \mathcal{M}(\mathcal{X})$ and $mM(dy) = \int_\mathcal{X} m(dx)M(x, dy)$. Readers can refer to [18, Chapter 4] for an introduction to the above measure theory concepts and notations.

3. REVIEW ON SIMULATED ANNEALING METHODS

3.1. Simulated annealing

SA is a widely used global optimization method, that was inspired from statistical physics [19, 20], and that comes with sound theoretical guarantees (see for instance [21, 14]). The Boltzmann distributions play a crucial role in SA. They relate the probability of a state $x \in \mathcal{X}$, its energy $f(x)$ and the temperature $T > 0$ through:

$$\pi_T(x) := \frac{1}{Z_T} \exp\left(-\frac{f(x)}{T}\right). \qquad (2)$$

The Boltzmann distributions concentrate on the set S_* as T goes to 0. In contrast, for T high enough, π_T is easy to sample from. SA exploits this feature, and aims at generating points distributed with density π_T for $T \searrow 0$, thus concentrating on S_*. One of the main feature of SA is the *cooling schedule*, which is a non-negative sequence $\{T_k\}_{k\in\mathbb{N}}$ decreasing to 0. It controls how T goes to 0 and is critical for SA: with a fast decay, iterates $\{x_k\}_{k\in\mathbb{N}}$ could get trapped in local minima, while a slow one may imply slower convergence.

Algorithm 1: SA

```
Initialization with $x_0 \sim \mu_0,\ \mu_0 \in \mathcal{M}(\mathcal{X})$
for $k = 1, ...$ do
    Generate a candidate $y_k \sim G(x_k, dy)$
    Compute the acceptance probability
        $p_k = \exp\left(-\left(\frac{f(y_k) - f(x_k)}{T_k}\right)_+\right)$        (3)
    Set $x_{k+1} = \begin{cases} y_k \text{ with probability } p_k \\ x_k \text{ with probability } 1 - p_k \end{cases}$
end
```

At iteration $k \in \mathbb{N}$, SA uses a symmetric proposal Markov kernel $G(x, dy)$ to generate a candidate point y_k, and then computes the *acceptance probability* p_k. If $f(y_k) \leq f(x_k)$, $x_{k+1} = y_k$ with probability $p_k = 1$. Indeed, Alg. 1 always accepts proposals that improve the objective (in the sense of the resolution of the optimization problem). Otherwise, the acceptance probability decreases to 0 as $T_k \searrow 0$. This allows the algorithm to escape local minima until reaching S_*.

Each iteration in Alg. 1 consists in fact in one transition of a Markov Chain with stationary distribution π_{T_k} [22]. The associated kernel is the Metropolis-Hastings (MH) kernel defined by

$$P_k(x, dy) := p_k(y, x)G(x, dy) + (1 - r(x))\delta_x(dy), \qquad (4)$$

where $r(x) = \int_\mathcal{X} p_k(y, x)G(x, dy)$. With this kernel, we define $\mu_k(dx) := \mathbb{P}(x_k \in dx) = \mu_{k-1}P_k$, for $k > 1$, and with μ_0 being the initialization distribution.

We can now present a first convergence result, restating Corollary 5.2 and Corollary 5.4 of [14] in a simplified way:

Theorem 1 (Convergence of SA [14])**.** *Under suitable ergodicity hypothesis on G, if there exists* $\xi \in (0, 1)$ *such that*

$$T_k = \frac{(1 + \xi)\Delta f}{\log(k + 2)}, \quad \forall k \in \mathbb{N}, \qquad (5)$$

then $||\mu_k - \pi_k||_{TV} \to 0$ *and*

$$\lim_{k\to+\infty} \mathbb{P}(x_k \in S_\epsilon) = 1, \quad \forall \epsilon > 0. \qquad (6)$$

Theorem 1 states that $\{\mu_k\}_{k\in\mathbb{N}}$ is able to track $\{\pi_k\}_{k\in\mathbb{N}}$ and thus generates iterates converging to S_* provided that the cooling schedule $\{T_k\}_{k\in\mathbb{N}}$ has its inverse decreasing logarithmically.

3.2. Fast Simulated Annealing

The cooling schedule of SA is often considered as being too slow. To circumvent this, FSA [23, 15] generalizes the accept-reject rule used in the MH step in Eq. (3). In FSA, this is generalized to any *acceptance function* q satisfying [15, Hyp. 1], implying that the acceptance probability becomes

$$p_k = q(\rho_k), \text{ where } \rho_k := \left(\frac{f(y_k) - f(x_k)}{T_k}\right)_+. \qquad (7)$$

This generalization of SA allows to use acceptance function q decreasing more slowly than the negative exponential, such as $\rho \longmapsto q(\rho) = \frac{1}{1+\rho}$. For such functions, the acceptance probability p_k will tend to be higher as $T_k \searrow 0$, and convergence can then be established with faster cooling schedules [15]. In particular, let us state here Corollary 3.4 of [15]:

Theorem 2 (Convergence of FSA [15]). *Suppose that f has a finite number of isolated global minima in the interior of $\mathcal{X}$, and that at each of them, f is locally $\mathcal{C}^3$ and its Hessian is positive definite. Assume that we set, for $\gamma \in (0,1]$,*

$$\frac{1}{T_k} = (k+1)^\gamma \log\left((k+1)^\gamma\right). \tag{8}$$

Then, under suitable ergodicity assumptions on G and for the acceptance function $q(\rho) = \frac{1}{1+\rho}$, there exists $C_\epsilon > 0$ such that

$$\mathbb{P}(x_k \in S_\epsilon) \geq 1 - \frac{C_\epsilon}{(k+1)^\gamma}, \quad \forall k \in \mathbb{N}. \tag{9}$$

This result implies (6), while also providing information on the rate of convergence. Contrary to Theorem 1, there is no result on the TV norm convergence of $\{\mu_k - \pi_k\}_{k\in\mathbb{N}}$. Still, FSA allows much faster cooling schedules to be used, which remains a noticeable improvement over SA. In the following, the MH kernels (4) associated to the acceptance function $q(\rho) = \frac{1}{1+\rho}$ will be denoted $P_k^{(F)}$.

3.3. Sequential Monte Carlo simulated annealing

In the SMC-SA algorithm of [16], described in Alg. 2, at iteration $k \in \mathbb{N}$, a population of N_k particles is propagated using the Markov kernels P_k (see (4)), while SA uses only one particle. Also, particles interact through resampling and reweighting, performed at each iteration. This is linked with sequential Monte-Carlo (SMC) methods [18], which aim at sampling from a sequence of distributions. However, in SMC, convergence is often stated in the limit $N \to +\infty$, while the convergence of SMC-SA is stated when $k \to +\infty$, which relates the iterates with the cooling schedule.

Algorithm 2: SMC-SA

```
Initialize the algorithm $x_k^{(n)} \sim \mu_0$ for $1 \leq n \leq N_0$;
for $k = 1, \ldots$ do
    Compute the self-normalized weights
      $w_k^{(n)} \propto \frac{\pi_k}{\pi_{k-1}}(x_{k-1}^{(n)})$
    Resample $\{\tilde{x}_k^{(n)}\}_{n=1}^{N_k}$ from $\{x_{k-1}^{(n)}, w_k^{(n)}\}_{n=1}^{N_k}$
    Generate $\{x_k^{(n)}\}_{n=1}^{N_k}$ propagating the points
      $\{\tilde{x}_k^{(n)}\}_{n=1}^{N_k}$ with the MH kernel $P_k(x, dy)$
end
```

Let us now state the main theorem of [16] (Theorem 2 and Corollary 2.1 in [16]):

Theorem 3 (Convergence of SMC-SA [16]). *Consider $\mu_k(dx) = \frac{1}{N_k}\sum_{n=1}^{N_k} \delta_{x_k^{(n)}}(dx)$ and $\mathcal{F}_k$ the history of all past samples until iteration k of Alg. 2. Then, if the cooling schedule is logarithmic and the sequence $\{N_k\}_{k\in\mathbb{N}}$ increases fast enough, under ergodicity hypothesis on G, there exists a sequence $\{c_k\}_{k\in\mathbb{N}} \searrow 0$, such that for any bounded function ϕ,*

$$\mathbb{E}\left[|\mu_k(\phi) - \pi_k(\phi)|\,|\,\mathcal{F}_{k-1}\right] \leq c_k ||\phi||_\infty. \tag{10}$$

The above result states that μ_k and π_k are getting closer in a sense that is weaker than in the TV norm sense of Theorem 1, with an equivalent cooling schedule. The assumption on the growing number of particles N_k is made explicit in [16] but is difficult to satisfy in practice since it requires to know c_{k-1} and Δf. Also, [16] shows that the reweighting and the resampling in SMC-SA improve upon parallel SA runs provided the distributions μ_k and π_k are close enough.

4. PROPOSED CURIOUS SA

4.1. Proposed algorithm

FSA and SMC-SA improve upon SA in different directions. Combining them is a natural idea, schematized below, that we propose to explore in this section.

$$\begin{tikzcd} \text{SA} \arrow[r] \arrow[d] & \text{SMC-SA} \arrow[d] \\ \text{FSA} \arrow[r] & \textbf{Curious SA} \end{tikzcd}$$

The proposed combination yields a novel algorithm, which we call *curious simulated annealing* (CSA), described in Alg. 3. The change of acceptance function, reminiscent from FSA, should allow better state space exploration, while the weighting and resampling steps inherent to SMC-SA should allow meaningful exchange between particles.

Algorithm 3: CSA

```
Initialize the algorithm $x_k^{(n)} \sim \mu_0$ for $1 \leq n \leq N$;
for $k = 1, \ldots$ do
    Compute the self-normalized weights
      $w_k^{(n)} \propto \frac{\pi_k}{\pi_{k-1}}(x_{k-1}^{(n)})$
    Resample $\{\tilde{x}_k^{(n)}\}_{n=1}^{N}$ from $\{x_{k-1}^{(n)}, w_k^{(n)}\}_{n=1}^{N}$
    Generate $\{x_k^{(n)}\}_{n=1}^{N}$ propagating the points
      $\{\tilde{x}_k^{(n)}\}_{n=1}^{N}$ with the MH kernel $P_k^{(F)}(x, dy)$
end
```

4.2. Convergence analysis

We state below our convergence theorem, for the proposed CSA method, along with a schematic proof of it.

Theorem 4 (Convergence of CSA). *Denote $\mu_k(dx) = \frac{1}{N}\sum_{n=1}^{N} \delta_{x_k^{(n)}}(dx)$ and $\mathcal{F}_k$ the history of all past samples until iteration k. Then, under the hypothesis of Theorem 2, for any $\epsilon > 0$, there exists $C_\epsilon > 0$ such that*

$$\mathbb{E}[\mu_k(S_\epsilon)\,|\,\mathcal{F}_{k-1}] \geq 1 - \frac{C_\epsilon}{(k+1)^\gamma}, \quad \forall k \in \mathbb{N}. \tag{11}$$

Moreover, if $\{x_k\}_{k\in\mathbb{N}}$ is a sequence generated by CSA, then

$$\mathbb{E}[\mu_k(S_\epsilon)\,|\,\mathcal{F}_{k-1}] \geq \mathbb{P}(x_k \in S_\epsilon), \quad \forall k \in \mathbb{N}. \qquad (12)$$

In Theorem 4, Eq. (11) is akin to (6) or (9). In contrast with Theorem 3, no assumption is made anymore on the number of particles. CSA benefits from the fast cooling schedule of FSA, which was not the case of SMC-SA. Furthermore, Eq. (12) shows that the additional steps of CSA are beneficial and improve upon FSA performance.

Sketch of the proof of Theorem 4: Let $k \in \mathbb{N}$. Starting from μ_{k-1}, Alg. 3 generates three distinct distributions, corresponding respectively to the reweighting, resampling, and propagation steps. Following the proof of Theorem 2 in [16], each step is controlled independantly. The effect of the MH kernels $P_k^{(F)}(x, dy)$ can be controlled using the same technique as in the proof of Theorem 3.3 in [15] (see [15, Eq.3.3] in particular). The control of the reweighting and resampling steps effects is omitted due to a lack of space.

5. NUMERICAL EXPERIMENTS

The numerical experiments have been performed using the Julia language (Version 1.4.2) [24]. Contrary to SA and FSA, SMC-SA and CSA are population-based algorithms so, we chose to compare them against multistart implementations of SA and FSA meaning that N particles follow independantly SA or FSA iterations. This allows to allocate the same computation efforts to each iteration (i.e., temperature value). We studied the record values of the runs in the population after κ iterations, that is $f_\kappa^* := \min\{f(x_k^{(n)}),\ 1 \leq k \leq \kappa,\ 1 \leq n \leq N\}$. In our experiments, we set $N = 250$ and all the particles $\{x_0^{(n)}\}_{n=1}^N$ were initialized with a Gaussian kernel centered at a given point x_0: $x_0^{(n)} \sim \mathcal{N}(dy; x_0, 0.05I)$. SA and SMC-SA have been run with a logarithmic cooling schedule $T_k = \frac{1}{\log(k+1)}$, FSA and CSA with the faster schedule $T_k = \frac{1}{(k+1)\log(k+1)}$. The algorithms used the proposal kernel $G(x, dy) = \mathcal{N}(dy; x, \frac{1}{4}I)$.

5.1. Test Problems

Two test problems have been used, highlighting the performance in two different contexts. Both are designed so that their minimum value is 0.

The objective of Problem (P_1) is the Rosenbrock function in $\mathbb{R}^{10}$, which is ill-conditionned with a unique minimizer that is difficult to find in a large banana-shaped valley:

$$f_1(x) := \sum_{i=1}^{9} 5(x_{i+1} - x_i^2)^2 + (1 - x_i)^2, \quad \forall x \in \mathbb{R}^{10}, \quad (13)$$

minimized at $x_* = (1, 1, ...1)^T$. Problem (P_2) aims at minimizing the Rastrigin function which is highly multimodal with regularly distributed local minima:

$$f_2(x) := 10 + \sum_{i=1}^{10} x_i^2 - \cos(2\pi x_i), \quad \forall x \in \mathbb{R}^{10}, \qquad (14)$$

minimized at $x_* = 0$. We will set $x_0 = 0$ for (P_1) and $x_0 = (1, 1, ..., 1)^T$ for (P_2), respectively.

		SA	FSA	SMC-SA	CSA
(P_1)	$\langle f_{50}^* \rangle$	6.31	6.49	6.41	**4.05**
	σ_{50}^*	0.829	**0.732**	1.15	1.17
	$\langle f_{500}^* \rangle$	3.64	3.72	5.06	**2.19**
	σ_{500}^*	0.761	0.778	1.26	**0.447**
(P_2)	$\langle f_{50}^* \rangle$	3.29	3.36	3.26	**3.23**
	σ_{50}^*	**0.425**	0.453	0.521	0.484
	$\langle f_{500}^* \rangle$	2.52	2.64	2.62	**2.47**
	σ_{500}^*	0.320	**0.304**	0.413	0.502

Table 1: Performances over 50 runs of the algorithms

5.2. Results

For each problem, 50 runs of the algorithms were conducted. We computed the average best record value after $\kappa = 50$ iterations $\langle f_{50}^* \rangle$ and after $\kappa = 500$ iterations $\langle f_{500}^* \rangle$, as well as the corresponding standard deviation σ_{50}^* and σ_{500}^*. This shows the performance of the algorithm with a low time budget and a higher one, as well as the consistency between runs. Table 1 shows the results, with the best ones in bold font.

CSA performs clearly better than all the other algorithms on (P_1), showing the best values for both small and large κ, as well as good consistency among runs. On Problem (P_2), CSA also performs best, although the gap is smaller than previously. The multistart algorithms tend to have the lowest standard deviations. This may be explained by the lack of resampling which lowers the variability of the algorithms.

6. CONCLUSIONS

SA algorithms are widely used in global optimization as they are easy to implement and offer sound theoretical guarantees. In this paper, we have proposed a novel scheme, the *curious simulated annealing* algorithm, which combines the features of two improved SA implementations: FSA and SMC-SA. We have illustrated its efficiency on two global optimization benchmarks. Moreover, we have described a sketch of its theoretical properties that would explain its good performance. Due to lack of space, we leave as a future work the completion of this theoretical analysis.

7. REFERENCES

[1] A. Marmin, M. Castella, J-C. Pesquet, and L. Duval. Sparse signal reconstruction for nonlinear models via piecewise rational optimization. *Signal Processing*, 179:107835:1–107835:13, 2021.

[2] Y. Chi, Y. M. Lu, and Y. Chen. Nonconvex optimization meets low-rank matrix factorization: An overview. *IEEE Transactions on Signal Processing*, 62(20):5239–5269, 2019.

[3] Y. Plan and R. Vershynin. The generalized Lasso with non-linear observations. *IEEE Transactions on Information Theory*, 62(3):1528–1537, 2016.

[4] B. D. Haeffele and R. Vidal. Global optimality in neural network training. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4390–4398, 2017.

[5] M. Wang, Y. Panagakis, P. Snape, and S. Zafeiriou. Learning the multilinear structure of visual data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6053–6061, 2017.

[6] T. F. Chan, S. Esedoglu, and M. Nikolova. Algorithms for finding global minimizers of image segmentation and denoising models. *SIAM Journal of Applied Mathematics*, 66(5):1632–1648, 2006.

[7] A. Blake. Comparison of the efficiency of deterministic and stochastic algorithms for visual reconstruction. *IEEE Transations on Pattern Analysis and Machine Intelligence*, 11(1), 1989.

[8] H. Mobahi and J. W. Fisher III. On the link between gaussian homotopy continuation and convex envelopes. In *Proceedings of the International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR)*, volume 8932, pages 43–56, 01 2015.

[9] T. G. W. Epperly and E. N. Pistikopoulos. A reduced branch and bound algorithm for global optimization. *Journal of Global Optimization*, 11:287–311, 1997.

[10] J. M. Fowkes, N. I. M. Gould, and C. L. Farmer. A branch and bound algorithm for the global optimization of hessian lipschitz continuous functions. *Journal of Global Optimization*, 56:1791–1815, 2013.

[11] F. Liang, Y. Cheng, and G. Lin. Simulated stochastic approximation annealing for global optimization with a square-root cooling schedule. *Journal of the American Statistical Association*, 109, 06 2014.

[12] M. R. Bonyadi and Z. Michalewicz. Particle swarm optimization for single objective continuous space problems: a review. *Evolutionary Computation*, 25(1):1–54, 2017.

[13] E. Onbaşoğlu and L. Özdamar. Parallel simulated annealing algorithms in global optimization. *Journal Of Global Optimization*, 19(1):27–50, 2001.

[14] H. Haario and E. Saksman. Simulated annealing process in general state space. *Advances in Applied Probability*, 23(4):866–893, 1991.

[15] 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 Processes and their Applications*, 119(6):1912–1931, 2009.

[16] E. Zhou and X. Chen. Sequential Monte Carlo simulated annealing. *Journal of Global Optimization*, pages 101–124, 2013.

[17] C. Andrieu, L. A. Breyer, and A. Doucet. Convergence of simulated annealing using Foster-Lyapunov criteria. *Journal of Applied Probability*, 38(4):975–994, 2001.

[18] N. Chopin and O. Papaspilopoulos. *An Introduction to Sequential Monte Carlo*. Springer, 2020.

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

[20] V. Černý. Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of Optimization Theory and Applications*, 45(1):41–51, 1985.

[21] A. Dekkers and E. Aarts. Global optimization and simulated annealing. *Mathematical Programming*, 50(3):367–393, 1991.

[22] G. O. Roberts and J. S. Rosenthal. General state space Markov chain and MCMC algorithms. *Probability Surveys*, 1:20–71, 2004.

[23] G. Gielis and C. Maes. A simple approach to time-inhomogeneous dynamics and applications to (fast) simulated annealing. *Journal of Physics A: Mathematical and General*, 32(29):5389–5407, 1999.

[24] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah. Julia: A fresh approach to numerical computing. *SIAM Review*, 59:65–98, 2017.