Particle Filtering Framework for a Class of Randomized Optimization Algorithms

Enlu Zhou, Michael C. Fu, and Steven I. Marcus

Abstract

We reformulate a deterministic optimization problem as a filtering problem, where the goal is to compute the conditional distribution of the unobserved state given the observation history. We prove that in our formulation the conditional distribution converges asymptotically to a degenerate distribution concentrated on the global optimum. Hence, the goal of searching for the global optimum can be achieved by computing the conditional distribution. Since this computation is often analytically intractable, we approximate it by particle filtering, a class of sequential Monte Carlo methods for filtering, which has proven convergence in "tracking" the conditional distribution. The resultant algorithmic framework unifies some randomized optimization algorithms and provides new insights into their connection.

I. Introduction

Global optimization problems arise in many areas of importance and can be extremely difficult to solve, due to the presence of multiple local optimal solutions and the lack of structural properties such as differentiability and convexity. In a general setting, there is little problem-specific knowledge that can be exploited in searching for improved solutions, and it is often the case that the objective function can only be assessed through "black-box" evaluation, which returns the function value for a specified candidate solution. Many randomized algorithms

E. Zhou is with the H. Milton Stewart School of Industrial & Systems Engineering, Georgia Institute of Technology, GA 30332 USA (e-mail: enlu.zhou@isye.gatech.edu).

M. C. Fu is with the Robert H. Smith School of Business, University of Maryland, College Park, MD 20742 USA (e-mail: mfu@umd.edu).

S. I. Marcus is with the Department of Electrical and Computer Engineering, University of Maryland, College Park, MD 20742 USA (e-mail: marcus@umd.edu).

This work was supported in part by the National Science Foundation (NSF) under Grants CNS-0926194, CMMI-0856256, EECS-0901543, CMMI-1130273, and by the Air Force Office of Scientific Research (AFOSR) under Grant FA9550-10-1-0340.

have been proposed for such optimization problems, such as genetic algorithms [6], simulated annealing [18], random search algorithms [19], the nested partitions method [17], the estimation of distribution algorithms in evolutionary computing [11], cross-entropy method [10], model reference adaptive search [7], and sequential Monte Carlo simulated annealing [21].

To tackle such global optimization problems, our main idea is to reformulate the optimization problem as a filtering problem, which is then solved by particle filtering. The idea comes from viewing the optimal solution as the unobserved state of a dynamic system and sample function values as noisy observations of the optimal function value (hence noisy observations of the unobserved state). The goal of filtering is to compute the *filtering distribution*, which is the conditional distribution of the unobserved state given the observation history. With an appropriate choice of the dynamic system model for the associated optimization problem, we prove that the filtering distribution converges asymptotically to a degenerate distribution concentrated on the optimal solution. Therefore, the task of searching for the optimal solution can be carried out through the procedure of estimating the filtering distribution sequentially. Since the filtering distribution does not have a closed-form expression in general, we apply particle filtering methods [4], a class of sequential Monte Carlo methods, to approximate the filtering distribution. Particle filtering has proven convergence to the filtering distribution under various conditions [3, 13], and has also shown good performance in practice.

Our approach results in a framework that includes many randomized optimization algorithms that are known as model-based optimization methods [24]. The key idea of these algorithms is to iteratively repeat two steps: 1) generate candidate solutions from a sampling distribution over the solution space; 2) update the sampling distribution using the candidate solutions. The hope is that the sampling distribution becomes more and more concentrated on the promising regions of the solution space and eventually converges to a degenerate distribution on the optimal solution. Therefore, the design of these randomized optimization algorithms involves two issues: (i) how to choose the sequence of sampling distributions; and (ii) how to sample from this sequence of distributions. In the particle filtering framework, the sequence of sampling distributions is the sequence of filtering distributions specified by the dynamic system model, and the sampling method is the sequential Monte Carlo (particle filtering) method. Some recent randomized optimization algorithms, including the estimation of distribution algorithms (EDAs), the cross-entropy (CE) method, and model reference adaptive search (MRAS), fall into our particle filtering framework, and interesting insights between them are revealed.

Some previous works have explored the connection between filtering and optimization, such as [12], [8], and our own preliminary conference version [22]. In particular, [12] and [8] use Kalman filter and particle filtering, respectively, to guide the movement of the particles in particle swarm optimization (PSO); however, while these algorithms improve on many other PSO algorithms empirically, they lack a convergence guarantee. From the sampling perspective, the Sequential Monte Carlo (SMC) sampler, which is essentially the sampling technique underlying particle filtering, can be used for optimization [14, 13]. However, to apply the SMC sampler requires artificially constructing a sequence of distributions with a sophisticated choice of forward and backward transition kernels. Our idea is distinct in that it explicitly reformulates an optimization problem as a filtering problem, which naturally leads to a desired sequence of filtering distributions that has proven convergence to the global optimal solution.

II. Filtering for Optimization

The filtering problem involves the estimation of a state that is not directly observed in a dynamic system from the noisy observations of the system. The filtering density can be computed recursively when each new observation arrives sequentially in time. Filtering bears great similarity to the process of searching for the optimum using a randomized optimization algorithm: the optimum, which can be viewed as the unobserved state of a system, is recursively estimated through the function values of the candidate solutions that are generated according to some randomized mechanism. The function values can be viewed as noisy observations of the optimal function value. Intuitively, the more function values observed, the more information is obtained about the unobserved state. The hope is that the filtering density will eventually converge to a Dirac delta function concentrated on the true value of the unobserved state, i.e., the optimum. We formalize this intuition in the rest of this section.

Consider the global optimization problem:

$$x^* = \arg\max_{x \in \mathcal{X}} H(x), \tag{1}$$

where the solution space $\mathcal{X}$ is a nonempty compact set in $\mathbb{R}^n$, and $H : \mathcal{X} \to \mathbb{R}$ is a deterministic function that is bounded, i.e., $\exists H_{lb} > -\infty,\ H_{ub} < \infty$ s.t. $H_{lb} \leq H(x) \leq H_{ub}$, $\forall x \in \mathcal{X}$. We assume that (1) has a unique global optimal solution, i.e., $\exists x^* \in \mathcal{X}$ s.t. $H(x) < H(x^*)$, $\forall x \neq x^*$, $x \in \mathcal{X}$. We will use the shorthand notation H^* to denote $H(x^*)$.

A filtering problem involves a state-space model:

$$\begin{aligned} X_k &= f(X_{k-1}, U_k), \quad k = 1, 2, \ldots, \\ Y_k &= h(X_k, V_k), \quad k = 0, 1, \ldots, \end{aligned} \tag{2}$$

where for any k, $X_k \in \mathbb{R}^{n_x}$ is the unobserved state , $Y_k \in \mathbb{R}^{n_y}$ is the observation, $U_k \in \mathbb{R}^{n_u}$ is the random system noise, and $V_k \in \mathbb{R}^{n_v}$ is the random observation noise. The goal is to compute at each time k the filtering distribution of the current state X_k conditional on the past observations $\{Y_0 = y_0, \ldots, Y_k = y_k\}$, simply denoted as $Y_{0:k} = y_{0:k}$. Throughout the paper we assume the filtering distribution admits a density b_k. Let $\mathcal{F}$ denote the σ-field on the Borel sets of $\mathbb{R}^{n_x}$. Then b_k satisfies

$$P(X_k \in A | Y_{0:k} = y_{0:k}) = \int_A b_k(x) dx, \quad \forall A \in \mathcal{F}.$$

We also define a one-step prediction density $b_{k|k-1}$ that satisifies

$$P(X_k \in A | Y_{0:k-1} = y_{0:k-1}) = \int_A b_{k|k-1}(x) dx, \quad \forall A \in \mathcal{F}.$$

The optimization problem (1) can be viewed as a filtering problem by choosing a state-space model of the following form:

$$X_k = X_{k-1} + U_k, \quad k = 1, 2, \ldots, \tag{3}$$

$$Y_k = H(X_k) - V_k, \quad k = 1, 2, \ldots, \tag{4}$$

where $X_k \in \mathbb{R}^n$, $Y_k \in \mathbb{R}$, and $\{U_k, k = 1, 2, \ldots\}$ and $\{V_k, k = 1, 2, \ldots\}$ are two independent sequences of random variables of appropriate dimensions. The sequences $\{U_k\}$ and $\{V_k\}$ are independent of each other and are also independent of X_0. We assume a prior density b_0 on the unobserved X_0. The transition kernel of (3) (i.e., the distribution of U_k) is denoted by $K_k(\cdot|x_{k-1})$, which satisfies $P(X_k \in A | X_{k-1} = x_{k-1}) = \int_A K_k(x|x_{k-1}) dx$, $\forall A \in \mathcal{F}$. The sequence $\{V_k\}$ is i.i.d. with probability density function (p.d.f.) $\varphi(\cdot)$, which satisfies the following condition:

(C) The p.d.f. $\varphi(\cdot)$ has support on $[0, H_{ub} - H_{lb}]$, and is positive, strictly increasing, and continuous on its support.

Using Bayes' rule, the filtering density of the state-space model above can be shown to evolve as:

$$\begin{aligned} b_k(x) &= \frac{\varphi(H(x) - y_k) b_{k|k-1}(x)}{\int \varphi(H(z) - y_k) b_{k|k-1}(z) dz} \\ &= \frac{\varphi(H(x) - y_k) \int K_k(x|x_{k-1}) b_{k-1}(x_{k-1}) dx_{k-1}}{\int \varphi(H(z) - y_k)(\int K_k(z|x_{k-1}) b_{k-1}(x_{k-1}) dx_{k-1}) dz}. \end{aligned} \tag{5}$$

The intuition of (3)-(4) and (5) and their connection with optimization can be seen easily in the simple case in which $U_k = 0$ for all k: the unobserved state $\{X_k\}$ is constant with the underlying value being the optimum x^*, which needs to be estimated; the observations $\{y_k\}$, which are the function values of the candidate solutions generated in a randomized optimization algorithm, are noisy observations of the optimal function value H^*; the filtering density b_k is our belief about the optimum x^* at iteration k based on the function values $\{y_0, y_1, \ldots, y_k\}$. Eqn. (5) implies that the belief b_k is tuned towards the more promising area where $H(x)$ is greater than y_k, since $\varphi(H(x) - y_k)$ is positive if $H(x) \geq y_k$ and is zero otherwise. Hence, different choices of φ, the p.d.f. of V_k, in fact lead to different sample selection or weighting schemes in the algorithm. In the more general case when U_k's are nonzero, $\{X_k\}$ is a perturbed process around the optimum x^*, and eventually settles down at x^* when the noise U_k gradually dampens down to zero. Thus, U_k brings in more randomness to the optimization algorithm. To show the formal results, we first introduce the following assumptions.

Assumption 1: For all $x \in \{y : H(y) < H(x^*)\}$, the set $\{z \in \mathcal{X} : H(z) \geq H(x)\}$ has strictly positive Lebesgue measure.

Assumption 2: For any $x \in \mathcal{X}$, $\int_A K_k(y|x)dy > 0$ for any $A \in \mathcal{F}$ that contains x and has a positive Lebesgue measure.

Assumption 3: For any fixed $x \in \mathcal{X}$, $\sum_{k=1}^{\infty} |b_{k-1}(x) - b_{k|k-1}(x)| < \infty$.

Assumption 1 ensures that there is always a positive probability to sample any neighborhood of the optimum. This assumption is satisfied by most objective functions such as continuous functions. Assumption 2 ensures that there is a positive transition probability from any point to its neighborhood. It is satisfied for example by Gaussian kernels, which are often used in the algorithms. Assumption 3 is consistent with the intuition that the noise $\{U_k\}$ needs to gradually dampen down to zero so that the perturbed process $\{X_k\}$ eventually settles down at x^*, and furthermore it says that the perturbation has to dampen down fast enough in accordance with the belief. It guides the choice of the kernel: a sufficient condition on the kernel to ensure Assumption 3 is that $K_k(x|y) \leq \max\left\{b_{k-1}(x) + \frac{\alpha_0}{k^\alpha}, \frac{1}{\int_{\mathcal{X}} dx}\right\}$ for all $y \neq x$, where $\alpha_0 > 0$ and $\alpha > 1$. For example, the special case that $K_k(x|y) = \delta(x - y)$ leads to $b_{k-1}(x) = b_{k|k-1}(x)$ for all k and hence satisfies Assumption 3 trivially. This special choice is used in algorithms such as CE and MRAS, which will be seen later.

Lemma 1: If φ satisfies the condition (C), then for an arbitrary and fixed observation sequence

$\{y_0, y_1, \ldots\}$, $\mathbb{E}_{b_k}[H(X)] \geq \mathbb{E}_{b_{k|k-1}}[H(X)]$.

Proof:

$$\begin{aligned} \mathbb{E}_{b_k}[H(X)] &= \frac{\mathbb{E}_{b_{k|k-1}}[H(X)\varphi(H(X) - y_k)]}{\mathbb{E}_{b_{k|k-1}}[\varphi(H(X) - y_k)]} \\ &\geq \mathbb{E}_{b_{k|k-1}}[H(X)] \end{aligned}$$

where the inequality results from the fact that φ is monotonically increasing, which implies $Cov(H(X), \varphi(H(X) - y_{k+1})) \geq 0$. For a proof on the covariance of monotone functions of a random variable, please refer to pp. 207-208 in [15]. ■

Lemma 1 implies that our estimate $\mathbb{E}_{b_k}[H(X)]$ of the optimal function value is improved over $\mathbb{E}_{b_{k|k-1}}[H(X)]$ with the information provided by the observation y_k. Even though the estimate $E_{b_{k-1}}[H(X)]$ will be perturbed to $E_{b_{k|k-1}}[H(X)]$ which might be worse, the perturbation dampens down fast enough such that eventually the estimate converges to the true optimal function value, and shown in the following theorem.

Theorem 1: Under Assumptions 1, 2, and 3, if φ satisfies the condition (C), then for a monotonically increasing observation sequence $\{y_0, y_1, \ldots\}$,

$$\lim_{k\to\infty} \mathbb{E}_{b_k}[H(X)] = H^*.$$

Proof: We first show that $\lim_{k\to\infty} \mathbb{E}_{b_k}[H(X)]$ exists. Define $\Delta_{k-1} \triangleq \mathbb{E}_{b_{k-1}}[H(X)] - \mathbb{E}_{b_{k|k-1}}[H(X)]$. From Lemma 1, we have $\mathbb{E}_{b_k}[H(X)] \geq \mathbb{E}_{b_{k-1}}[H(X)] - \Delta_{k-1}$. Therefore,

$$a_k \triangleq \mathbb{E}_{b_k}[H(X)] + \sum_{i=0}^{k-1} \Delta_i \;\geq\; \mathbb{E}_{b_{k-1}}[H(X)] + \sum_{i=0}^{k-2} \Delta_i = a_{k-1}, \quad \forall k \geq 2.$$

Moreover, $\{a_k\}$ is upper bounded, since for all $k \geq 1$, $a_k \leq H_{ub} + \sum_{i=0}^{k-1} \Delta_i$ and

$$\begin{aligned} \sum_{i=0}^{k-1} \Delta_i &\leq \sum_{i=0}^{k-1} |\Delta_i| \\ &\leq \int_{\mathcal{X}} H(x) \sum_{i=1}^{k} |b_{i-1}(x) - b_{i|i-1}(x)| dx \\ &\leq \int_{\mathcal{X}} H(x) \sum_{i=1}^{\infty} |b_{i-1}(x) - b_{i|i-1}(x)| dx < \infty, \end{aligned}$$

where the last inequality follows from Assumption 3 and the fact that $\mathcal{X}$ is compact. Since $\{a_k\}$ is monotonically increasing and upper bounded, $\lim_{k\to\infty} a_k$ exists. Using the dominated convergence theorem, we conclude that $\sum_{i=0}^{\infty} \Delta_i$ exists and

$$\begin{aligned}\sum_{i=0}^{\infty} \Delta_i &= \sum_{i=1}^{\infty} \int_{\mathcal{X}} H(x)(b_{i-1}(x) - b_{i|i-1}(x))dx \\ &= \int_{\mathcal{X}} H(x) \sum_{i=1}^{\infty} (b_{i-1}(x) - b_{i|i-1}(x))dx < \infty.\end{aligned}$$

Therefore, the limit of the righthand side of $\mathbb{E}_{b_k}[H(X)] = a_k - \sum_{i=0}^{k-1} \Delta_i$ exists, which implies that $\lim_{k\to\infty} \mathbb{E}_{b_k}[H(x)]$ exists.

Next we show that $\lim_{k\to\infty} \mathbb{E}_{b_k}[H(x)]$ is equal to H^*. Since $\{y_k\}$ is monotonically increasing and upper bounded, it has a limit, denoted by $\bar{y} \triangleq \lim_{k\to\infty} y_k$. There are two cases to consider: (i) $\bar{y} = H^*$; (ii) $\bar{y} < H^*$.

(i) For the case $\bar{y} = H^*$, we prove by contradiction. Suppose

$$\lim_{k\to\infty} \mathbb{E}_{b_k}[H(X)] = H_* < H^*. \tag{6}$$

Since $\bar{y} = H^*$, there exists a K such that $y_K > (H_* + H^*)/2$. For all $k \geq K$, $\varphi(H(x) - y_k) = 0$ for any $x \in \{x \in \mathcal{X} : H(x) < (H_* + H^*)/2\}$, and hence,

$$b_k(x) \begin{cases} = 0, & \text{if } x \in \{x \in \mathcal{X} : H(x) < (H_* + H^*)/2\}; \\ \geq 0, & \text{otherwise.} \end{cases}$$

Therefore,

$$\int H(x)b_k(x)dx \geq (H_* + H^*)/2, \quad \forall k \geq K.$$

Taking the limit on both sides gives

$$\lim_{k\to\infty} \mathbb{E}_{b_k}[H(x)] \geq (H_* + H^*)/2 > H_*,$$

which is a contradiction with (6).

(ii) For the case $\bar{y} < H^*$, we first prove that for any $x \in \mathcal{X}_{\bar{y}} \triangleq \{z \in \mathcal{X} : \bar{y} \leq H(z)\}$, $b_k(x) > 0$ and $b_{k|k-1}(x) > 0$ for all k. Assumption 1 guarantees that the set $\mathcal{X}_{\bar{y}}$ has a positive Lebesgue measure. For any fixed $x \in \mathcal{X}_{\bar{y}}$, since $\varphi(H(x) - y_k) > 0$ for all k, $b_{k|k-1}(x) > 0$ implies that $b_k(x) > 0$; on the other hand, if $b_k(x) > 0$, then $b_{k+1|k}(x) = \int K_k(x|x_k)b_k(x_k)dx_k > 0$, which follows from Assumption 2. Therefore, given $b_0(x) > 0$, using the argument above iteratively leads to the conclusion that $b_{k|k-1}(x) > 0$ and $b_k(x) > 0$ for all k. Furthermore,

$$\mathbb{E}_{b_{k+1|k}}[\varphi(H(X) - y_{k+1})] \geq \int_{\mathcal{X}_{\bar{y}}} b_{k+1|k}(x)\varphi(H(x) - y_{k+1})dx > 0.$$

Therefore, by induction $b_k(x)$ can be rewritten as

$$b_k(x) = b_0(x) \prod_{i=0}^{k-1} \left(\frac{b_{i+1|i}(x)}{b_i(x)} \frac{\varphi(H(x) - y_{i+1})}{\mathbb{E}_{b_{i+1|i}}[\varphi(H(X) - y_{i+1})]} \right). \tag{7}$$

Based on this expression of b_k, in the following we will prove by contradiction. Assume $\lim_{k\to\infty} \mathbb{E}_{b_k}[H(X)] = H_* < H^*$. From this assumption, a trivial argument of contradiction leads to

$$B \triangleq \lim_{k\to\infty} \int_{\{x:H(x)\leq H^*\}} b_k(x)dx > 0.$$

We can write

$$\begin{aligned} & \mathbb{E}_{b_k}[\varphi(H(x) - y_{k+1})] \\ = \; & \int_{\{x:H(x)\leq H_*\}} \varphi(H(x) - y_{k+1}) b_k(x)dx + \int_{\{x:H(x)>H_*\}} \varphi(H(x) - y_{k+1}) b_k(x)dx \\ \leq \; & \varphi(H_* - y_{k+1}) \int_{\{x:H(x)\leq H_*\}} b_k(x)dx + \varphi(H^* - y_{k+1}) \int_{\{x:H(x)>H_*\}} b_k(x)dx. \end{aligned}$$

Thanks to the continuity of φ, taking limits on both sides of the inequality above yields

$$\lim_{k\to\infty} \mathbb{E}_{b_k}[\varphi(H(x) - y_{k+1})] \leq \varphi(H_* - \bar{y})B + \varphi(H^* - \bar{y})(1 - B). \tag{8}$$

Assumption 3 implies $\lim_{k\to\infty} |b_k(x) - b_{k+1|k}(x)| = 0$. Applying the bounded convergence theorem gives us

$$\lim_{k\to\infty} \mathbb{E}_{b_k}[\varphi(H(X) - y_{k+1})] = \lim_{k\to\infty} \mathbb{E}_{b_{k+1|k}}[\varphi(H(X) - y_{k+1})]. \tag{9}$$

Using (8) and (9), we have

$$\begin{aligned} & \lim_{i\to\infty} \frac{\varphi(H(x) - y_{i+1})}{\mathbb{E}_{b_{i+1|i}}[\varphi(H(X) - y_{i+1})]} \\ = \; & \frac{\lim_{i\to\infty} \varphi(H(x) - y_{i+1})}{\lim_{i\to\infty} \mathbb{E}_{b_{i+1|i}}[\varphi(H(X) - y_{i+1})]} \\ = \; & \frac{\varphi(H(x) - \bar{y})}{\lim_{i\to\infty} \mathbb{E}_{b_i}[\varphi(H(X) - y_{i+1})]} \\ \geq \; & \frac{\varphi(H(x) - \bar{y})}{\varphi(H_* - \bar{y})B + \varphi(H^* - \bar{y})(1 - B)}. \end{aligned} \tag{10}$$

Since $B > 0$ and φ is strictly increasing, it follows that

$$\varphi(H_* - \bar{y})B + \varphi(H^* - \bar{y})(1 - B) < \varphi(H^* - \bar{y}).$$

Hence, by Assumption 1, the set $\mathcal{B} \triangleq \{x \in \mathcal{X}_{\bar{y}} : \varphi(H_* - \bar{y})B + \varphi(H^* - \bar{y})(1 - B) < \varphi(H(x) - \bar{y})\}$ has a positive Lebesgue measure. For any fixed $x \in \mathcal{B}$, following (10) we have

$$\lim_{i \to \infty} \frac{\varphi(H(x) - y_{i+1})}{\mathbb{E}_{b_{i+1|i}}[\varphi(H(X) - y_{i+1})]} > 1. \tag{11}$$

From Assumption 3, we can easily show that

$$\frac{b_{i+1|i}(x)}{b_i(x)} \to 1 \text{ as } i \to \infty, \quad \forall x \in \mathcal{B}. \tag{12}$$

Using (11) and (12), we conclude from (7) that

$$\lim_{k \to \infty} b_k(x) = \infty, \quad \forall x \in \mathcal{B}.$$

Thus, by Fatou's Lemma, we have

$$\liminf \int_{\mathcal{X}} b_k(x)dx \geq \liminf \int_{\mathcal{B}} b_k(x)dx \geq \int_{\mathcal{B}} \liminf b_k(x)dx = \infty,$$

which is a contradiction with $\int_{\mathcal{X}} b_k(x)dx = 1$. Hence, it follows that $\lim_{k \to \infty} \mathbb{E}_{b_k}[H(X)] = H^*$.

■

Remark 1: Applying Markov inequality, we have for any $\epsilon > 0$, $P\{H^* - H(X_k) > \epsilon\} \leq \frac{H^* - E_{b_k}[H(X)]}{\epsilon} \to 0$ as $k \to \infty$, where X_k follows the distribution b_k. Hence, we conclude that $H(X_k) \to H^*$ in probability.

Remark 2: Since any bounded continuous real function ψ with $x^* = \max_{x \in \mathcal{X}} \psi(x)$ satisfies the conditions imposed on H, we can follow the same approach above to show that $\int_{\mathcal{X}} \psi(x) b_k(x)dx \to \psi(x^*)$ as $k \to \infty$, which implies that $\{b_k\}$ converges to the Dirac delta function concentrated on x^*.

III. Particle Filtering for Optimization

Now that we have reformulated an optimization problem as a filtering problem, estimation of the optimum is equivalent to estimation of the filtering density b_k, which evolves according to (5). A filtering problem in general does not have a closed-form solution except in some rare cases, such as a linear Gaussian system for which the Kalman filter is the optimal filter. The intractability is mainly due to the fact that the integral in the recursive equation of the filtering density, such as eqn. (5), is infinite dimensional. Particle filtering provides a computationally viable way for approximate filtering. It is a class of Monte-Carlo-based methods that have shown good performance in many nonlinear/non-Gaussian systems. Hence, we will rely on the particle filtering methods to develop an algorithmic framework for solving optimization problems.

Particle filtering approximates b_k using a finite number of particles/samples and mimicking the evolution of the filtering density through the propagation of particles. More specifically, particle filtering approximates b_k by a p.d.f.

$$\hat{b}_k(x) = \sum_{i=1}^{N} w_k^i \delta(x - x_k^i), \tag{13}$$

where δ denotes the Dirac delta function, $\{x_k^i\}_{i=1}^N$ are the support points, and $\{w_k^i\}_{i=1}^N$ are the associated probabilities/weights. A detailed derivation can be found in [1] that shows how to draw samples $\{x_k^i\}_{i=1}^N$ and calculate their corresponding weights $\{w_k^i\}_{i=1}^N$ such that $\hat{b}_k$ tracks b_k "closely" with a proven convergence. Below we present a framework for optimization that is based on the simplest version of particle filtering, which is sometimes also called sequential importance sampling resampling (SISR) or bootstrap particle filter.

Algorithm 1: **P**article **F**ilter framework for **O**ptimization (PFO)

- *Step 1: Initialization.* Specify an initial p.d.f./p.m.f. b_0 defined on $\mathcal{X}$, the transition kernels $\{K(\cdot|\cdot)\}$ (i.e., the distributions of $\{U_k\}$), and the p.d.f $\{\varphi(\cdot)\}$ (i.e., the distributions of V_k). Sample $\{\tilde{x}_0^i\}_{i=1}^N$ i.i.d. from b_0. Set $k = 1$.
- *Step 2: Importance Sampling.* For $i = 1, \ldots, N$, draw $x_k^i \sim K_k(x|\tilde{x}_{k-1}^i)$.
- *Step 3: Observation Construction.* Take y_k to be a sample function value $H(x_k^i)$ according to a certain rule. If $k > 1$ and $y_k < y_{k-1}$, then set $y_k := y_{k-1}$.
- *Step 4: Bayes' Updating.* Compute the normalized weights according to
$$w_k^i \propto \varphi(H(x_k^i) - y_k), i = 1, 2, \ldots, N; \quad \sum_{i=1}^N w_k^i = 1.$$
- *Step 5: Resampling.* Draw i.i.d. samples $\{\tilde{x}_k^i\}_{i=1}^N$ from $\hat{b}_k(x) = \sum_{i=1}^N w_k^i \delta(x - x_k^i)$ using sampling with replacement or approximate sampling methods.
- *Step 6: Stopping.* If a stopping criterion is satisfied, then stop; else, $k \leftarrow k + 1$ and go to Step 2.

In the resampling step, several known resampling methods in particle filtering can be used to generate new candidate solutions and can also be easily implemented, such as the density projection method [23] and the resample-move method [5]. The density projection method projects $\hat{b}_k$ to a parameterized family of densities $\{f(\cdot;\theta)\}$ to find the best approximation $f(\cdot;\theta_k)$, and then draws samples $\{\tilde{x}_k^i\}_{i=1}^N$ from $f(\cdot;\theta)$. The resample-move method applies a Markov chain Monte Carlo (MCMC) step to move the particles after they are generated by sampling with replacement. Depending on the resampling methods, the convergence properties of the different instantiations of PFO also differ slightly, but all follow from the existing convergence results of the corresponding particle filters in the literature [23, 2] under suitable assumptions.

The choice of the transition kernel K_k (or in other words, the distribution of U_k) is guided by Assumption 3. A non-zero U_k may be used to alleviate the problem of premature convergence in some randomized optimization algorithms that converge too fast and get stuck in some local optimal solution, because it injects randomness into the algorithm by perturbing the locations of the candidate solutions. Based on this idea we have proposed an improved version of the cross-entropy method and carried out numerical comparison. Details are omitted here due to space limit but can be found in Chapter 5 in [20]. The choice of the p.d.f. φ should balance the trade-off between exploration and exploitation: a more steeply increasing φ assigns more weight to better solutions and hence explores more aggressively around the better solutions, while a more flat φ does the opposite to maintain more exploration over the entire solution space. A choice commonly used in practice is of the form (14), which in the algorithm essentially assigns equal weighs to a certain percentage of elite samples and ignores (i.e., assigns zero weight) to all the other non-elite samples.

The generation of the observations y_k is a key difference between filtering and optimization. In a filtering problem, the observations come from the underlying real system. For example, in a chemical process, the observations could be the measurements of the temperature or pressure that are taken from the process. However, in an optimization problem, there is no such real system, and hence the observation sequence has to be "generated" in a certain way and then is viewed as if it is given. Lemma 1 and Theorem 1 are proved for a fixed observation sequence, together with the convergence results for particle filtering conditioned on a fixed observation sequence [4], implying the convergence of PFO. In fact, the generation of y_k is also a design factor in the algorithm; for example, y_k can be set as a sample quantile of the function values of the candidate solutions generated at iteration k. The setting of a monotonically increasing sequence $\{y_k\}$ in the algorithm can be understood intuitively, since we have some information about the true state, i.e., we know that the true state achieves the largest function value. We exploit this information about the true state by ensuring at each time our estimate is at least as good as the estimate at the previous iteration.

We end this section with a final remark that the particle filtering framework has the potential for guiding the development of new improved algorithms. Besides the various design factors inside PFO that is based on the simplest version of particle filtering, there are many variations of particle filtering that can also be adapted to optimization. In particular, a general particle filter does not have to draw samples according to the transition kernel but rather from an importance density, which can be chosen appropriately to adjust the trade-off between exploitation of promising regions of the solution space and exploration of the entire solution space.

IV. A Unifying Perspective on EDAs, CE, and MRAS

The particle filtering framework PFO provides a unifying perspective on some randomized optimization algorithms: estimation of distribution algorithms (EDAs) [11], the cross entropy (CE) method [16, 10], and model reference adaptive search (MRAS) [7]. EDAs are a class of optimization algorithms based on the key idea of iteratively carrying out the two steps: 1) select elite samples from a pool of samples that are generated from a probability distribution; 2) estimate the probability distribution of selected samples and generate new samples from it. These two steps correspond to the Bayes' updating step and the resampling step in PFO, respectively. The main difficulty in EDAs is to estimate the distribution of the selected samples, which is done by expressing the interaction between the underlying variables of a solution via a probabilistic model. One way used in EDAs is to employ a dynamic Bayesian network (DBN) to represent such a probabilistic model. Put in the context of PFO, the relationship between the components of the state vector X_k is expressed through the use of a DBN, and the joint distribution of the components is b_k. Interestingly, there is a particular particle filter designed especially for DBNs [9], which samples $\{x_k^i\}$ according to the relationship between its components so that the sampling is more efficient.

To illustrate CE and MRAS, we make a specific choice of the state-space model (3)-(4). Let the system noise $U_k = 0$ for all k, i.e., $K_k(x|x_{k-1}) = \delta(x - x_{k-1})$. Let the observation noise V_k follow a uniform distribution on $[0, H_{ub} - H_{lb}]$, i.e.,

$$\varphi(u) = \frac{I\{0 \leq u \leq H_{ub} - H_{lb}\}}{H_{ub} - H_{lb}}, \tag{14}$$

where $I\{A\}$ denotes the indicator function: $I\{A\} = 1$ if A is true and $I\{A\} = 0$ otherwise. Since y_k is a sample function value, we always have $H(x) - y_k \leq H_{ub} - H_{lb}$; so the indicator function in (14) reduces to $I\{0 \leq H(x) - y_k\}$ when $u = H(x) - y_k$. Hence, the recursive equation (5) for b_k is simplified to

$$b_k(x) = \frac{I\{H(x) \geq y_k\} b_{k-1}(x)}{\int I\{H(z) \geq y_k\} b_{k-1}(z) dz}. \tag{15}$$

In PFO, the importance sampling step is essentially omitted with $x_k^i = \tilde{x}_{k-1}^i$, and the Bayes' updating step results in

$$w_k^i \propto I\{H(x_k^i) \geq y_k\}, \quad i = 1, \ldots, N. \tag{16}$$

The standard CE method can be viewed as an instantiation of PFO with the choice of the state-space model above and the choice of the density projection method in the resampling step. More specifically, the density projection approach projects $\hat{b}_k$ onto a parameterized family of densities $\{f(\cdot, \theta)\}$ by minimizing the Kullback-Leibler (KL) divergence between $\hat{b}_k$ and $f(\cdot, \theta)$:

$$\begin{aligned} \min_\theta D_{KL}(\hat{b}_k \| f(\cdot, \theta)) &= \int \hat{b}_k(x) \log \frac{\hat{b}_k(x)}{f(x, \theta)} dx \\ &= \mathbb{E}_{\hat{b}_k}[\log \hat{b}_k(X)] - \mathbb{E}_{\hat{b}_k}[\log f(X, \theta)]. \end{aligned}$$

Since the first term does not depend on $f(\cdot, \theta)$, the minimization problem above is equivalent to maximizing the second term in the last line. By plugging into the second term the expression $\hat{b}_k(x) = \sum_{i=1}^N w_i^k \delta(x - x_k^i)$ with w_k^i satisfying (16), the maximization problem is

$$\theta_k = \arg\max_\theta \frac{1}{N} \sum_{i=1}^N I\{H(x_k^i) \geq y_k\} \log f(x_k^i, \theta). \tag{17}$$

Note that (17) is exactly the parameter updating step in the standard CE method. Then new samples $\{\tilde{x}_k^i\}_{i=1}^N$ (or $\{x_{k+1}^i\}_{i=1}^N$) are drawn from $f(\cdot, \theta_k)$ and used to update the parameter again as above.

Compared with EDAs, the standard CE method avoids complicated estimation of the density b_k through the use of density projection without the need to specify the relationships among the components of X_k. However, from a filtering viewpoint, the projection particle filter introduces a projection error that is accumulated over iterations. The reason can be seen by scrutinizing the one-step evolution of the approximate density. Since samples $\{x_k^i\}_{i=1}^N$ are sampled from $f(\cdot, \theta_{k-1})$, the density that the algorithm actually tries to approximate at iteration k is

$$b_k'(x) = \frac{I\{H(x) \geq y_k\} f(x, \theta_{k-1})}{\int I\{H(z) \geq y_k\} f(z, \theta_{k-1}) dz}. \tag{18}$$

Comparing (18) with the expression (15) for b_k, b_{k-1} is replaced by its approximation $f(\cdot, \theta_{k-1})$, which introduces a projection error that is accumulated to the next iteration. This projection error can be corrected by taking $f(\cdot,\theta_{k-1})$ as an importance density and assigning appropriate weights to the samples. Specifically, i.i.d. samples $\{x_k^i\}_{i=1}^N$ are drawn from $f(x,\theta_{k-1})$ to approximate b_k; so according the principle of importance sampling, the weight of each x_k^i should be computed according to

$$\begin{aligned} w_k^i &= \frac{b_k(x_k^i)}{f(x_k^i,\theta_{k-1})} \\ &\propto \frac{I\{H(x_k^i)\geq y_k\}I\{H(x_k^i)\geq y_{k-1}\}\ldots I\{H(x_k^i)\geq y_1\}b_0(x_k^i)}{f(x_k^i,\theta_{k-1})} \\ &\propto \frac{I\{H(x_k^i)\geq y_k\}}{f(x_k^i,\theta_{k-1})}, \end{aligned} \tag{19}$$

where the second line is obtained by applying (15) recursively, and the third line follows from the fact that $\{y_k\}$ is a nondecreasing sequence and the choice that b_0 is a uniform distribution on $\mathcal{X}$. As shown before, projection of $\hat{b}_k(x)$ onto the parameterized family of densities $\{f(\cdot,\theta)\}$ is equivalent to the maximization problem

$$\max_\theta \mathbb{E}_{\hat{b}_k}[\log f(\cdot,\theta)].$$

Since $\hat{b}_k(x)=\sum_{i=1}^N w_k^i(x-x_k^i)$ with w_k^i satisfying (19) now, the maximization problem above can be rewritten as

$$\theta_k=\arg\max_\theta \sum_{i=1}^N \frac{I\{H(x_k^i)\geq y_k\}}{f(x_k^i,\theta_{k-1})}\log f(x_k^i,\theta). \tag{20}$$

Note that (20) is exactly the parameter updating equation in the Monte Carlo version of the MRAS$_0$ algorithm that is presented in [7]. Similarly as in CE, new samples $\{\tilde{x}_k^i\}_{i=1}^N$ (or $\{x_{k+1}^i\}_{i=1}^N$) are drawn from $f(\cdot,\theta_k)$ and used to update the parameter again; the process is repeated iteratively. Therefore, an instantiation of MRAS falls into the particle filtering framework with a slight variation.

Remark 3: Note that φ of the form (14) used in CE and MRAS does not exactly satisfy Condition (C) that is used in our theoretical analysis. However, a choice that satisfies Condition (C) is $\varphi(u)=\frac{I\{0\leq u\leq H_{ub}-H_{lb}\}}{(1+e^{-Mu})/(\int_{0\leq z\leq H_{ub}-H_{lb}}1+e^{-Mz}dz)}$, which approaches (14) as M goes to infinity and is a very close approximation to (14) when M is a large positive number.

V. Conclusion

We reformulated a deterministic optimization problem as a filtering problem, for which we proved that the filtering distribution converges asymptotically to a degenerate distribution concentrated on the global optimum. By applying particle filtering methods to this filtering problem, we then proposed a framework for randomized optimization algorithms. This framework provides a unifying perspective on some existing algorithms, including the estimation of distribution algorithms in evolutionary computing, the cross-entropy method, and model reference adaptive search. New insights are obtained to reveal the connection between these methods.

REFERENCES

[1] S. Arulampalam, S. Maskell, N. J. Gordon, and T. Clapp. A tutorial on particle filters for on-line non-linear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, 2002.

[2] N. Chopin. Central limit theorem for sequential Monte Carlo and its applications to Bayesian inference. *The Annals of Statistics*, 32(6):2385 – 2411, 2004.

[3] D. Crisan and A. Doucet. A survey of convergence results on particle filtering methods for practitioners. *IEEE Transaction on Signal Processing*, 50(3):736–746, 2002.

[4] A. Doucet, J. F. G. deFreitas, and N. J. Gordon, editors. *Sequential* M*onte* C*arlo Methods In Practice*. Springer, New York, 2001.

[5] W. Gilks and C. Berzuini. Following a moving target - Monte Carlo inference for dynamic Bayesian models. *Journal of the Royal Statistical Society*, 63(1):127–146, 2001.

[6] D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, 1989.

[7] J. Hu, M. C. Fu, and S. I. Marcus. A model reference adaptive search method for global optimization. *Operations Research*, 55:549–568, 2007.

[8] C. Ji, Y. Zhang, M. Tong, and S. Yang. Particle filter with swarm move for optimization. In *Proceedings of 10th International Conference on Parallel Problem Solving From Nature (PPSN)*, pages 909 – 918, 2008.

[9] D. Koller and U. Lerner. *Sequential* M*onte* C*arlo Methods in Practice*, chapter 10: Sampling in factored dynamic systems, pages 445–464. Statistics for engineering and information science. Springer-Verlag, New York, 2001.

[10] D. P. Kroese, S. Porotsky, and R. Y. Rubinstein. The cross-entropy method for continuous multiextremal optimization. *Methodology and Computing in Applied Probability*, 8:383–407, 2006.

[11] J. A. Lozano, P. Larranaga, I. Inza, and E. Bengoetxea, editors. *Towards a New Evolutionary Computation: Advances on Estimation of Distribution Algorithms*. Springer Verlag, New York, 2006.

[12] C.K. Monson and K.D. Seppi. The Kalman swarm. In *Genetic and Evolutionary Computation Conference (GECCO)*, pages 140 – 150, 2004.

[13] P. Del Moral. F*eynman*-K*ac Formulae: Genealogical and Interacting Particle Systems with Applications*. Springer, New York, 2004.

[14] P. Del Moral, A. Doucet, and T. France. Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society, Series B*, 68:411436, 2006.

[15] S.M. Ross. *Simulation*. Academic Press, 4th edition, 2006.

[16] R. Y. Rubinstein. The cross-entropy method for combinatorial and continuous optimization. *Methodology and Computing in Applied Probability*, 2:127–190, 1999.

[17] L. Shi and S. Ólafsson. Nested partitions method for global optimization. *Operations Research*, 48(3):390 – 407, 2000.

[18] P. J. M. van Laarhoven and E. H. L. Aarts. *Simulated Annealing: Theory and Applications*. Springer, 1987.

[19] Z. B. Zabinsky. *Stochastic Adaptive Search for Global Optimization*. Nonconvex Optimization and Its Applications. Springer, 2003.

[20] E. Zhou. *Particle filtering for Stochastic Control and Global Optimization*. PhD thesis, University of Maryland, College Park, MD, 2009.

[21] E. Zhou and X. Chen. Sequential Monte Carlo simulated annealing. *Journal of Global Optimization*, 2012. Forthcoming. doi:10.1007/s10898-011-9838-3.

[22] E. Zhou, M. C. Fu, and S. I. Marcus. A particle filtering framework for randomized optimization algorithms. In *Proceedings of the 2008 Winter Simulation Conference*, pages 647–654, 2008.

[23] E. Zhou, M. C. Fu, and S. I. Marcus. Solving continuous-state POMDPs via density projection. *IEEE Transactions on Automatic Control*, 55(5):1101 – 1116, 2010.

[24] M. Zlochin, M. Birattari, N. Meuleau, and M. Dorigo. Model-based search for combinatorial optimization: A critical survey. *Annals of Operations Research*, 131:373–395, 2004.