Journal of Computational Physics 231 (2012) 2002–2014

Contents lists available at SciVerse ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Stochastic global optimization as a filtering problem

Panos Stinis

Department of Mathematics, University of Minnesota, Minneapolis, MN 55455, United States

ARTICLE INFO

Article history:
Received 20 April 2011
Received in revised form 6 October 2011
Accepted 15 November 2011
Available online 26 November 2011

Keywords:
Particle filtering
Stochastic global optimization
Exponential densities
Parametric density estimation

ABSTRACT

We present a reformulation of stochastic global optimization as a filtering problem. The motivation behind this reformulation comes from the fact that for many optimization problems we cannot evaluate exactly the objective function to be optimized. Similarly, we may not be able to evaluate exactly the functions involved in iterative optimization algorithms. For example, we may only have access to noisy measurements of the functions or statistical estimates provided through Monte Carlo sampling. This makes iterative optimization algorithms behave like stochastic maps. Naive global optimization amounts to evolving a collection of realizations of this stochastic map and picking the realization with the best properties. This motivates the use of filtering techniques to allow focusing on realizations that are more promising than others. In particular, we present a filtering reformulation of global optimization in terms of a special case of sequential importance sampling methods called particle filters. The increasing popularity of particle filters is based on the simplicity of their implementation and their flexibility. We utilize the flexibility of particle filters to construct a stochastic global optimization algorithm which can converge to the optimal solution appreciably faster than naive global optimization. Several examples of parametric exponential density estimation are provided to demonstrate the efficiency of the approach.

1. Introduction

Optimization problems are ubiquitous in science and engineering ranging from space craft trajectory design to DNA studies and computer science [9,15,16]. As a result of the vast number of optimization applications and their related intricacies, the construction of optimization algorithms continues to be a subject of intense research. The main problem in optimization is to locate, usually through an iterative procedure, the optimal values (minima or maxima) of an objective function which depends on a number of parameters. The related problem of estimating the values of the parameters for which the objective function attains its optimal value is of equal importance in applications.

For many optimization problems the function to be optimized is a function of several variables and can have multiple local extrema. Usually, optimization algorithms are guaranteed to find a local extremum when started from an arbitrary initial condition. In order to reach a global extremum we should use an optimization algorithm with multiple starting values and choose the best solution. This is called naive global optimization. In naive global optimization each solution starting from a different initial condition evolves independently of the others. Intuitively, we expect that a global optimization algorithm should benefit from the interaction of the different solutions. There is a category of global (deterministic and stochastic) optimization algorithms (e.g. branch and bound, tabu methods, genetic algorithms, ant colony optimization, swarm optimization, hill climbing) [6,9,18] which allow for the different solutions to interact with the purpose of allocating more resources in areas of the parameter space which seem more promising in an optimization sense.

E-mail address: stinis@math.umn.edu

0021-9991/$ - see front matter © 2011 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2011.11.019

In the current work, we propose an algorithm that belongs to this category. For many optimization problems we cannot evaluate exactly the objective function to be optimized. Similarly, we may not be able to evaluate exactly the functions involved in iterative optimization algorithms. For example, we may only have access to noisy measurements of the functions or statistical estimates provided through Monte Carlo sampling. This makes iterative optimization algorithms behave like stochastic maps. The corresponding global optimization problem also becomes stochastic. The algorithm we present here is based on the reformulation of the stochastic global optimization problem as a filtering problem. In particular, we present a filtering reformulation of stochastic global optimization in terms of a special case of sequential importance sampling methods called particle filters [8,14]. In this setting, the desired interaction of the solutions starting from different initial conditions is effected through the filtering step (see Section 3 for more details).

However, the generic particle filter reformulation of stochastic global optimization will *not* necessarily lead to an algorithm that converges faster than the naive global optimization algorithm. The generic particle filter reformulation contains a resampling step. The purpose of the resampling step is to produce more copies (offspring) of a good, in an optimization sense, particle. The offspring particles inherit *all* the properties of the parent particle and in particular, the controlling parameters which appear in the underlying iterative optimization algorithm. However, nothing prevents us from assigning *different* values of the controlling parameters to the different offspring of the same particle. After all, the controlling parameters are adaptive and should be chosen so as to accelerate convergence. The resampling step of the particle filter allows us to *batch* the offspring of a particle according to their ancestry. Then, we can assign different values of the controlling parameters to the offspring within a batch. For the specific case of the Levenberg–Marquardt algorithm, we utilize this (see Section 4.2.1) and construct a modified particle filter algorithm which converges appreciably faster than the naive global optimization algorithm and the generic particle filter reformulation. We provide several examples of parametric exponential density estimation of varying difficulty to demonstrate the efficiency of the approach. More ideas of how to construct modified particle filter algorithms are provided in the discussion section (see Section 6).

The paper is organized as follows. Section 2 discusses local and global optimization problems. In Section 3 we present the reformulation of the stochastic global optimization problem as a filtering problem and in particular a reformulation which uses particle filters. In Section 4 we apply the particle filter reformulation to the problem of estimating the parameters of an exponential density. Section 5 contains numerical results for several examples to illustrate the efficiency of the proposed approach. Finally, in Section 6 we provide a discussion and some directions for future work.

2. Stochastic local and global optimization

Assume that we are given a scalar objective function $H(x)$ depending on d parameters $x_1,\ldots,x_d$. The purpose of the optimization problem is to compute the optimal (maximal or minimal) value of $H(x)$, say $\mathrm{op}(H(x))$, as well as the optimal parameter vector $\hat{x} = (\hat{x}_1,\ldots,\hat{x}_d)$ for which $H(\hat{x}) = \mathrm{op}(H(x))$. For the sake of clarity and without loss of generality, let us assume that we are interested in obtaining the minimum value of the objective function $H(x)$.

A generic optimization algorithm attempts to locate the optimal parameter vector in the following way:

(1) Pick an initial approximation $x^{(0)}$.
(2) For $k \geqslant 0$, compute $x^{(k+1)} = f(x^{(k)})$, where $f(x) = (f_1(x),\ldots,f_d(x))$ is a d-dimensional vector-valued function. Different optimization algorithms use a different function $f(x)$. Note that in general f can depend on states $x^{(i)}$ with $i < k$. Here we restrict attention to algorithms which use only the previous state, i.e. $x^{(k)}$.
(3) Evaluate $H(x^{(k+1)})$. If $H(x^{(k+1)})$ satisfies a stopping criterion or $k+1$ is equal to a maximum number of iterations k_{max}, stop. Else, set $k = k+1$ and proceed to Step 2.

For many optimization problems, the function $H(x)$ has multiple local minima. The generic algorithm described above is, usually, able to locate one of the local minima, unless we are very lucky or know a lot about the problem at hand to guide the choice of $x^{(0)}$. An obvious global version of the algorithm given above is as follows:

(1) Pick a collection of initial approximations $x_1^{(0)},\ldots,x_N^{(0)}$.
(2) Run, say for M iterations, the generic optimization algorithm for the different initial conditions $x_1^{(0)},\ldots,x_N^{(0)}$.
(3) Choose $\tilde{x} = \arg\min_{i=1\ldots N} H\left(x_i^{(M)}\right)$.

As stated in the introduction, for many optimization problems we only have access to a random estimate of the value of the objective function H and of the optimization algorithm function f. We will denote those as H_s and f_s respectively. The uncertainty can come from noisy measurements. Also, it can be due to Monte Carlo sampling error for cases where H and/or f involve expectations with respect to a probability density. Consequently, the iterative optimization sequence becomes stochastic. To be more precise, the optimization algorithm is modified as follows:

(1) Pick an initial approximation $x^{(0)}$.
(2) For $k \geqslant 0$, compute $x^{(k+1)} = f_s(x^{(k)})$.