

Stochastic Optimization: a Review

Dimitris Fouskakis¹ and David Draper²

¹*Department of Mathematical Sciences, University of Bath, Claverton Down, Bath BA2 7AY, UK. Email: df@maths.bath.ac.uk* ²*Department of Applied Mathematics and Statistics, Baskin School of Engineering, University of California, 1156 High Street, Santa Cruz CA 95064, USA. Email: draper@ams.ucsc.edu; web <http://www.ams.ucsc.edu/~draper>.*

Summary

We review three leading stochastic optimization methods—simulated annealing, genetic algorithms, and tabu search. In each case we analyze the method, give the exact algorithm, detail advantages and disadvantages, and summarize the literature on optimal values of the inputs. As a motivating example we describe the solution—using Bayesian decision theory, via maximization of expected utility—of a variable selection problem in generalized linear models, which arises in the cost-effective construction of a patient sickness-at-admission scale as part of an effort to measure quality of hospital care.

Key words: Bayesian decision theory; Genetic algorithms; Global optimization; Heuristic methods; Hybrid algorithms; Local search; Maximization of expected utility; Simulated annealing; Variable selection; Tabu search.

1 Introduction

In the past 50 years, since the development of digital computers, many investigators have studied the problem of numerically optimizing an objective function. One approach is *stochastic optimization*, in which the search for the optimal solution involves randomness in some constructive way. If \mathcal{S} denotes the (finite) set of all possible solutions, the task we consider is to maximize or minimize the *objective function* $f: \mathcal{S} \rightarrow \mathbb{R}$. In the case of maximization, on which we focus here, the problem is to find a *configuration* $x_{opt} \in \mathcal{S}$ which satisfies

$$f(x_{opt}) \geq f(x) \quad \text{for all } x \in \mathcal{S}. \quad (1)$$

All of the optimization methods we consider have the character of a discrete-time search chain, in which an initial member x_0 of \mathcal{S} is chosen by some means and becomes the *current configuration* x_t at time $t = 0$, and the algorithm then iteratively repeats the process of deciding on a *move* from x_t to a *proposed configuration* x_{t+1} . Many of the algorithms we examine rely on a *neighborhood* structure in deciding where to move next; this requires a rule, often based on a measure of distance, which uniquely identifies all of the neighbors of a given current configuration.

It is easy to see that as the dimension of \mathcal{S} increases the harder the task becomes, and more time is needed to find the optimal, or at least a near-optimal, configuration. Another difficulty in this problem is that it is common for the objective function to have many *local optima*. An algorithm like the well-known *local search* (LS; e.g., Aarts & Korst, 1989; Algorithm 1), which only accepts moves with higher values of the objective function than the previous move, will not perform well in this situation, since it is likely that the search will get stuck in a local optimum.

Algorithm 1: Local Search (LS)

```

Begin;
  Choose a random configuration  $x_0$ ;
  Set  $x := x_0$ ;
  Repeat:
    Generate a new configuration  $y$  from the neighborhood of  $x$ ;
    If  $f(y) \geq f(x)$  then  $x := y$ ;
  Until  $f(y) < f(x)$  for all  $y$  in the neighborhood of  $x$ ;
End.

```

The disadvantages of LS algorithms can be formulated as follows:

- By definition, such algorithms terminate in a local maximum, and there is generally no information as to the amount by which this local maximum falls short of a global maximum;
- The obtained local maximum depends on the initial configuration, for the choice of which generally no guidelines are available; and
- It is typically not possible to give an upper bound for the computation time.

To avoid some of these disadvantages, a number of potential improvements are possible:

- Execution of the algorithm for a large number of initial configurations, say M , at the cost of an increase in computation time; for $M \rightarrow \infty$, in the case in which the number of elements of \mathcal{S} is finite, such an algorithm finds a global maximum with probability 1, if only because a global maximum is encountered as an initial configuration with probability 1 as $M \rightarrow \infty$;
- Use of information gained from previous runs of the algorithm to improve the choice of an initial configuration for the next run;
- Introduction of a more complex move-generation mechanism, in order to be able to “jump away from” the local maxima found by the simple approach to generating moves. To choose this more complex move-generation mechanism properly requires detailed knowledge of the problem itself; and
- Acceptance of moves which correspond to a decrease in the objective function in a limited way, in the hope that this will eventually lead to a higher local maximum.

In this paper we review three leading stochastic optimization methods (plus several variations on them)—*simulated annealing* (SA), *genetic algorithms* (GA), and *tabu search* (TS)—each of which (because of its use of ideas like the ones above) often manages to avoid the disadvantages of LS algorithms. In Section 2 below we give details of a complicated optimization problem which serves to illustrate the central features of each of these three methods. Sections 3–5 are devoted to the three algorithms in turn; in each case we analyze the method, give the exact algorithm, detail advantages and disadvantages, and summarize the literature on optimal values of the inputs. In Section 6 we present some results on the performance of SA, GA, and TS in the optimization problem described in Section 2, and Section 7 concludes with a comparative discussion.

For recent contrasting and complimentary reviews of leading stochastic optimization methods in the literatures of other fields, see, e.g., Dasgupta & Michalewicz (1997), Fourer (1998), Gray *et al.* (1997), Neumaier (2002), Pintér (1996), Reeves (1999), Smith (1999), and Törn *et al.* (1999). Aarts & Lenstra (1997) offers a more extensive discussion of the methods examined here, and Reidys & Stadler (2002) presents a closer look at the theory underlying these methods.

