

Chapter III

Stochastic

Optimization Algorithms

Pierre Collet

Université du Littoral Côte d'Opale, France

Jean-Philippe Rennard

Grenoble Graduate School of Business, France

ABSTRACT

When looking for a solution, deterministic methods have the enormous advantage that they do find global optima. Unfortunately, they are very CPU intensive, and are useless on untractable NP-hard problems that would require thousands of years for cutting-edge computers to explore. In order to get a result, one needs to revert to stochastic algorithms that sample the search space without exploring it thoroughly. Such algorithms can find very good results, without any guarantee that the global optimum has been reached; but there is often no other choice than using them. This chapter is a short introduction to the main methods used in stochastic optimization.

INTRODUCTION

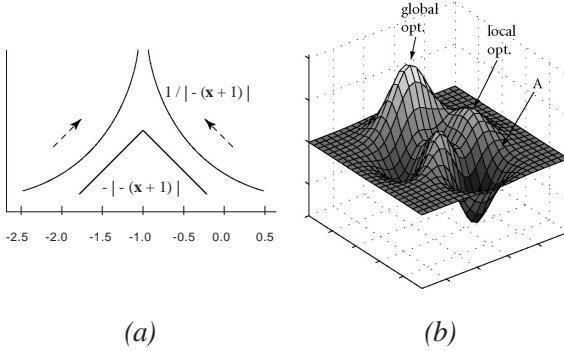
The never-ending search for productivity has made optimization a core concern for engineers. Quick process, low-energy consumption, short and economical supply chains are now key success factors.

Given a space Ω of individual solutions $\omega \in R^n$ and an objective function $f, f(\omega) \rightarrow R$, optimizing is the process of finding the solution ω^* which minimizes (maximizes) f .

For hard problems, optimization is often described as a walk in a *fitness landscape*.

First proposed by biologist S. Wright (1932), fitness landscapes aimed at representing the fitness of a living organism according to the genotype space. While optimizing, *fitness* measures the quality of a solution, and fitness landscapes plot solutions and corresponding goodness (fitness). If one wishes to optimize the function $x+1=0$, then depending on the choice of the error measure, fitness can for example be defined as $-|-(x+1)|$ or as $1/|-(x+1)|$. The optimization process then tries to find the peak of the fitness landscape (see Figure 1(a)).

Figure 1. (a) Fitness landscapes for $x + 1 = 0$. (b) A multimodal fitness landscape.



This example is trivial and the optimum is easy to find. Real problems are often multimodal, meaning that their fitness landscapes contain several local optima (i.e., points whose neighbors all have a lower fitness; see Figure 1(b)). This is particularly true when variables interact with one another (*epistasis*).

Usual analytical methods, like gradient descent, are often unable to find a global optimum, since they are unable to deal with such functions. Moreover, companies mostly deal with combinatorial problems like quadratic assignment, timetabling, or scheduling problems. These problems using discrete states generate non-continuous objective functions that are unreachable through analytical methods.

Stochastic optimization algorithms were designed to deal with highly complex optimization problems. This chapter will first introduce the notion of complexity and then present the main stochastic optimization algorithms.

NP-Complete Problems and Combinatorial Explosion

In December, Santa Claus must prepare the millions of presents he has to distribute for Christmas. Since the capacity of his sleigh is finite, and he prefers to minimize the number of runs, he would like to find the best way to organize the packs. Despite the apparent trivi-

ality of the task, Santa Claus is facing a very hard problem. Its simplest formulation is the *one-dimensional bin packing problem*. Given a list $L = (a_1, a_2, \dots, a_n)$ of items with sizes $0 < s(a_i) \leq 1$, what is the minimum number m of unit-capacity bins B_j such that $\sum_{a_i \in B_j} s(a_i) \leq 1, 1 \leq j \leq m$? This problem is known to be *NP-hard* (Coffman, Garey, & Johnson, 1996).

Various forms of the *bin packing problem* are very common. The transportation industry must optimize truck packing given weight limits, the press has to organize advertisements minimizing the space, and the sheet metal industry must solve the *cutting-stock problem* (how to minimize waste when cutting a metal sheet).

Such problems are very tough because we do not know how to build algorithms that can solve them in *polynomial-time*; they are said to be *intractable problems*. The only algorithms we know for them need an *exponential-time*. Table 1 illustrates the evolution of time algo-

Table 1. Polynomial vs. non-polynomial functions complexity growth

O(N)	N=17	N=18	N=19	N=20
N	17×10^{-9} s	18×10^{-9} s	19×10^{-9} s	20×10^{-9} s
N^2	289×10^{-9} s	324×10^{-9} s	361×10^{-9} s	400×10^{-9} s
N^5	1.4×10^{-3} s	1.8×10^{-3} s	2.4×10^{-3} s	3.2×10^{-3} s
2^N	131×10^{-6} s	262×10^{-6} s	524×10^{-6} s	1×10^{-3} s
5^N	12.7 mn	1 h	5.29 h	26.4h
TSP	2.9 h	2 days	37 days	2 years !
$N!$	4 days	74 days	4 years	77 years !

Considering 10^9 operations per second, evolution of the algorithm time according to its complexity. TSP stands for Traveling Salesman Problem, with complexity $\frac{(N-1)!}{2}$ for N towns (see following sections).

15 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/stochastic-optimization-algorithms/21118

Related Content

Generating Fully Bounded Chaotic Attractors

Zeraoulia Elhadj (2011). *International Journal of Artificial Life Research* (pp. 36-42).

www.irma-international.org/article/generating-fully-bounded-chaotic-attractors/56320

Topic Modeling as a Tool to Gauge Political Sentiments from Twitter Feeds

Debabrata Sarddar, Raktim Kumar Dey, Rajesh Bose and Sandip Roy (2020). *International Journal of Natural Computing Research* (pp. 14-35).

www.irma-international.org/article/topic-modeling-as-a-tool-to-gauge-political-sentiments-from-twitter-feeds/250254

Feature Selection and Recognition of Face by using Hybrid Chaotic PSO-BFO and Appearance-Based Recognition Algorithms

Santosh Kumar and Sanjay Kumar Singh (2015). *International Journal of Natural Computing Research* (pp. 26-53).

www.irma-international.org/article/feature-selection-and-recognition-of-face-by-using-hybrid-chaotic-pso-bfo-and-appearance-based-recognition-algorithms/146883

Blockchain Technology in Healthcare Analytics

S. Karthigai Selvi, R. Siva Shankar and K. Ezhilarasan (2024). *Bio-Inspired Optimization Techniques in Blockchain Systems* (pp. 114-129).

www.irma-international.org/chapter/blockchain-technology-in-healthcare-analytics/338087

The Worst-Case Stabilization Time of a Self-Stabilizing Algorithm under the Weakly Fair Daemon Model

Tetz C. Huang, Ji-Cherng Lin, Chih-Yuan Chen and Cheng-Pin Wang (2010). *International Journal of Artificial Life Research* (pp. 45-52).

www.irma-international.org/article/worst-case-stabilization-time-self/46028