Function Minimization: Hill Climbing and Simulated Annealing Strategies

Buzdea Stefan
Pricop Tudor-Andrei

October 19, 2023

1 Abstract

In this paper, we conduct a comprehensive exploration and comparison of two heuristic optimization techniques: Simulated Annealing and Hill Climbing, with a primary focus on their iterated versions, and various algorithmic variants such as the first improvement, best improvement, and worst improvement strategies. Our study centers on addressing the global minimum optimization problem for functions across multiple dimensions (5, 10, and 30 dimensions). We thoroughly examine the practical performance of these methods, providing valuable insights into execution times and result quality. This investigation offers a clear understanding of the strengths and weaknesses of each variant, aiding in method selection.

In conclusion, we find that Simulated Annealing, though slightly slower, offers more accurate results compared to Hill Climbing. Moreover, the simplicity of the first improvement variant delivers faster outcomes while maintaining a close proximity to the best improvement variant.

2 Introduction

This study aims to investigate two methods for locating global minima or maxima of functions with any number of variables. We will compare established heuristic optimization techniques: Simulated Annealing and Hill Climbing, with a focus on their iterated versions. The Hill Climbing algorithm will encompass three variants: first improvement, best improvement, and worst improvement, while for the Simulated Annealing algorithm, we will utilize the best improvement variant. These methods will be put to the test across four benchmark functions, namely De Jong 1, Schwefel's, Rastrigin's, and Michalewicz's. To ensure a comprehensive analysis, we will explore these functions in 5, 10, and 30 dimensions, implementing our experiments in Python.

Through rigorous experimentation, we will report the best minimum values across multiple runs, the average minimum values, and the average execution time for each run of each method and variant. By contrasting these outcomes, our objective is to shed light on significant differences and gain a deeper understanding of the contributing factors.

This research not only aims to determine the more effective method for locating global minima but also delves into the practical implications of these findings. In conclusion, we will summarize the key insights and their relevance in the domain of optimization problems.

3 Methods

In this study, we have employed two heuristic optimization techniques to seek the global minimum of functions with varying numbers of variables. Our aim was to implement each of the three improvement variants in a straightforward manner, avoiding additional optimizations to provide a clear understanding of their implications.

For both algorithms, we represented solutions using binary arrays and real number arrays. For the binary representation, we divided the function's domain into 10^{ε} equally sized subintervals to achieve the desired precision, and then calculated the necessary number of bits to represent each interval, accounting for the function's dimensionality. To decode the solutions into real numbers for function evaluation, we utilized a simple conversion algorithm, followed by scaling the number into the function's domain.

Now, let's delve into the operational aspects of each algorithm. Initially, both approaches begin with a random bitstring. Subsequently, we generate neighboring solutions by flipping individual bits, selecting the one with the greatest improvement based on the chosen improvement method, and updating the best solution encountered so far.

Distinguishing characteristics emerge when comparing Hill Climbing and Simulated Annealing algorithms. In Hill Climbing, if the selected neighbor does not yield a better solution, it signifies convergence to a local minimum, prompting the initiation of another iteration. In contrast, Simulated Annealing introduces an element of exploration by allowing a chosen neighbor that is not superior to the current solution to be retained. This stochastic feature can lead to moments where the algorithm accepts a point with a higher objective value, facilitating the exploration of a broader solution space.

Regarding the three improvement variants, each serves a unique purpose. The best improvement method exhaustively examines all neighbors to identify the largest improvement, the worst improvement similarly evaluates all neighbors but seeks the smallest improvement, and the first improvement selects a neighbor at random. In all cases, we prioritize selecting neighbors with a lower objective value. These comprehensive experiments enable us to conduct meaningful comparisons between the two optimization methods and the three improvement variants. The insights gained through these experiments shed light on the distinctions and outcomes, aligning with the research objectives outlined in the study.

4 Functions

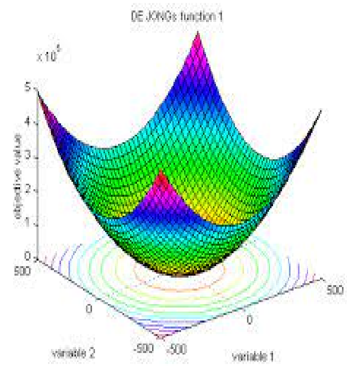

Figure 1: De Jong's function: $f_n(x) = \sum_{i=1}^{n} ix_i^2$
$-5.12 \leq x_i \leq -5.12$
global minimum: $f(x) = 0$
http://www.geatbx.com/docu/fcnindex-01.html [2]

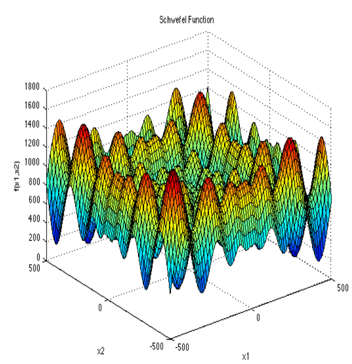

Figure 2: Schwefel's function: $f_n(x) = \sum_{i=1}^{n} -x_i sin(\sqrt{|x_i|})$
$-500 \leq x_i \leq 500$
global minimum: $f_n(x) = -n * 418.9829$
https://www.sfu.ca/ ssurjano/schwef.html [3]

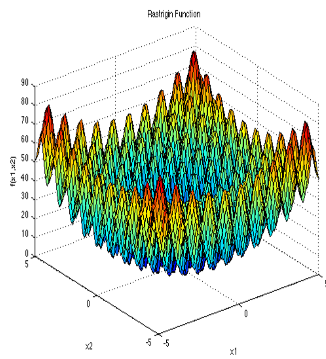

Figure 3: Rastrigin's function: $f_n(x) = 10n + \sum_{i=1}^{n}(x_i^2 - 10cos(2\pi x_i))$
$-5.12 \leq x_i \leq -5.12$
global minimum: $f(x) = 0$
https://www.sfu.ca/ ssurjano/rastr.html [4]

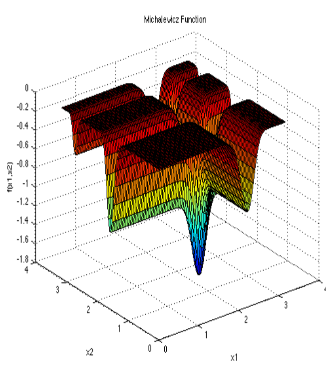

Figure 4: Michalewicz's function: $f_n(x) = -\sum_{i=1}^{n} sin(x_i)(sin\frac{ix_i^2}{\pi})^{20}$
$0 \leq x_i \leq \pi$
global minimum: $f_5(x) = -4.687, f_{10}(x) = -9.66$
https://www.sfu.ca/ ssurjano/michal.html [5]

5 Experimental results

DeJong				
Dimension	Algorithm	Minim	Average	Time
30	HCB	0.0	0.0	805 s
	HCF	0.0	0.0	100 s
	HCW	0.0	0.0	1849 s
	SA	0.0	0.0	999 s
10	HCB	0.0	0.0	421 s
	HCF	0.0	0.0	79 s
	HCW	0.0	0.0	804 s
	SA	0.0	0.0	744 s
5	HCB	0.0	0.0	297 s
	HCF	0.0	0.0	55 s
	HCW	0.0	0.0	746 s
	SA	0.0	0.0	1072 s

Rastrigin				
Dimension	Algorithm	Minim	Average	Time
30	HCB	20.62561	27.04519	661 s
	HCF	30.07828	38.73196	86 s
	HCW	53.07521	60.42784	3943 s
	SA	17.86534	26.27192	1297 s
10	HCB	2.98923	4.46861	303 s
	HCF	4.22577	5.95132	75 s
	HCW	8.44828	9.89533	1534 s
	SA	2.04951	4.12050	1065 s
5	HCB	0.0	0.0	237 s
	HCF	0.0	0.99536	64 s
	HCW	0.99504	1.21035	1258 s
	SA	0.0	0.0	1309 s

Schwefel				
Dimension	Algorithm	Minim	Average	Time
30	HCB	-10865.19414	-10623.35167	2081 s
	HCF	-10543.07683	-10127.72135	397 s
	HCW	-10433.93263	-10068.23198	13135 s
	SA	-11114.98883	-10922.64678	2400 s
10	HCB	-4023.95232	-3872.213	919 s
	HCF	-3998.89712	-3811.251	183 s
	HCW	-3913.63986	-3759.2614	5869 s
	SA	-4109.41635	-4002.11006	1670 s
5	HCB	-2094.91345	-2074.639	710 s
	HCF	-2094.80931	-2063.229	155 s
	HCW	-2041.14675	-2010.24513	4553 s
	SA	-2094.91346	-2094.872	2242 s

Michalewicz				
Dimension	Algorithm	Minim	Average	Time
30	HCB	-26.74368	-25.62685	541 s
	HCF	-25.91867	-25.02531	76 s
	HCW	-22.32989	-20.96834	2164 s
	SA	-27.45123	-27.10534	1003 s
10	HCB	-9.55213	-9.34905	236 s
	HCF	-9.32472	-9.10362	55 s
	HCW	-8.76004	-8.21024	643 s
	SA	-9.61076	-9.38035	881 s
5	HCB	-4.68765	-4.68672	194 s
	HCF	-4.68704	-4.67891	48 s
	HCW	-4.68331	-4.68205	525 s
	SA	-4.68765	-4.68673	1150 s

6 Experimental Description

In our experimental setup, we conducted a comprehensive analysis employing the Hill Climbing and Simulated Annealing algorithms with the aim of identifying the global minima for functions featuring varying dimensions. Our primary objective was to achieve a high level of precision, ensuring accurate results to at least 5 decimal places.

To ensure the reliability of our findings, we adopted an iterative approach for both algorithms, executing a substantial number of iterations, typically in the thousands. The number of iterations was inversely proportional to the dimensionality of the functions, allowing us to strike a balance between accuracy and computational efficiency. In each case, we meticulously recorded crucial metrics, including the best value, average value, and execution times. To obtain reliable average values, we conducted a minimum of 15 runs for each scenario.

For both algorithms and across all dimensions (5, 10, and 30), we employed a step size (ε) of 10^{-3}. This choice facilitated efficient traversal of the function's domain, increasing the number of neighboring points, ultimately enhancing the accuracy of our programs.

7 Comparison and Interpretation

In the comparative analysis of our study, we discern substantial disparities both in terms of solution quality and computational time between the two heuristic algorithms, as well as among the various improvement variants (i.e., first improvement, best improvement, and worst improvement).

Let us begin by comparing the performance of the methods that yield the most favorable solutions, specifically, Hill Climbing and Simulated Annealing, both implemented with the best improvement variant. It is evident that Simulated Annealing consistently outperforms Hill Climbing in terms of solution quality, producing both superior minimum and average solutions. However, it is worth noting that Simulated Annealing exhibits significantly longer execution times. Consequently, when considering accuracy, Simulated Annealing stands out as the most proficient approach, albeit at the expense of increased computation time.

A noteworthy observation is the extensive computational time required by Hill Climbing when employing the worst improvement variant. While it is possible that our implementation may not be optimal, the results unequivocally demonstrate that this variant offers the poorest trade-off between solution quality and computational efficiency.

An interesting finding pertains to the Hill Climbing algorithm employing the first improvement strategy. This approach not only offers the shortest execution time but also delivers solutions closely approximating those obtained with the best improvement variant. Consequently, the first improvement method emerges as a highly practical algorithm, requiring considerably less time compared to its counterparts.

Turning our attention to the actual solution values, we can ascertain that our experiment yielded results conforming to known thresholds for 30-dimensional versions of the functions. Specifically, for the Rastrigin function, solutions with values below 30 were achieved, while the Michalewicz function produced solutions below 25. The Schwefel function's solutions fell below 10,000, and for De Jong 1, a minimum value of 0.0 was obtained. With these outcomes, we can confidently assert that our experimentation with the algorithms and their variants was indeed successful.

8 Conclusion

In conclusion, our comparative study of Hill Climbing and Simulated Annealing, encompassing three distinct improvement variants, highlights the significant impact of algorithm choice on solution quality and computational efficiency. Simulated Annealing excels in accuracy, albeit at the expense of longer execution times, while Hill Climbing's first improvement variant offers a pragmatic balance. Our results align with established thresholds for 30-dimensional functions, affirming the success of our experimental endeavor.

References

[1] Basseur, M. Goeffon, A. *On the Efficiency of Worst Improvement for Climbing NK-Landscapes.* Universite d'Angers [France].

[2] Pohlheim, H. *Example Functions (single and multi-objective functions) 2 Parametric Optimization.* GEATbx, 1994 - 2006.

[3] Bingham, D. Surjanovic, S. *Test Functions and Datasets: Schwefel Function.* Simon Fraser University [Canada] : Virtual Library of Simulation Experiments, 2013.

[4] Bingham, D. Surjanovic, S. *Test Functions and Datasets: Rastrigin Function.* Simon Fraser University [Canada] : Virtual Library of Simulation Experiments, 2013.

[5] Bingham, D. Surjanovic, S. *Test Functions and Datasets: Michalewicz Function.* Simon Fraser University [Canada] : Virtual Library of Simulation Experiments, 2013.

[6] Croitoru, E. *Teaching: Genetic Algorithms.* Romania [Iasi] : Computer Science UAIC.

[7] *Simulated Annealing Explained.* Baeldung.com, 2023.

[8] Rawat, U. *Introduction to Hill Climbing — Artificial Intelligence.* geeksforgeeks.org, 2023.

[9] Pozrikidis, C. *Boundary Integral And Singularity Methods For Linearized Viscous Flow.* Cambridge [England] : Cambridge University Press, 1992. Print.