Usage of Hill Climbing and Simulated Annealing algorithms in different scenarios

Simionescu Teodor

November 3, 2022

1 Abstract

The deterministic problem of finding the minimum of a function is not a trivial operation when the function is complicated. Therefore, a non-deterministic algorithm can be more efficient and deliver the expected results.

2 Introduction

This report will provide information regarding two different non-deterministic algorithms: Hill Climbing and Simulated Annealing.

There will be different scenarios in which they will be tested. Four functions and for every function, three different dimensions. To have the most accurate answers as possible , a precision of five decimal is used.

All the results are presented and so based on the differences of time, behaviour and accuracy a conclusion will be drawn.

3 Methods

Hill Climbing will be present in three different forms due to the choice of evaluating the answer and Simulated Annealing will be present in one form. All implementation of Hill Climbing and Simulated Annealing are following a set of similar rules in order to find the minimum required. The solution are stored in binary format and so is made an evaluation through a function that converts the binary input into a decimal value.

3.1 Worst-Improvement Hill Climbing

This method iterates through all the neighbours of the candidate and choose as next point the neighbour that is better (is satisfying the give condition) than the current one, but worse than other better candidates. Due to the fact that this algorithm is looking into every neighbour, the time complexity is slightly higher.

3.2 Best-Improvement Hill Climbing

This method iterates through all the neighbours of the candidate and choose as next point only the best neighbour found. As the Worst-Improvement Hill Climbing, it is also time-consuming.

3.3 First-Improvement Hill Climbing

This method iterates through the neighbours of the candidate till it finds another point that is better than the actual candidate. Therefore, this method has a slightly worst accuracy than the one mentioned above.

3.4 Simulated Annealing

The difference between this method and the other ones from above is that Simulated Annealing provides the option to choose a 'bad' neighbour (is NOT satisfying the give condition), based on a temperature that gets decreased with each iteration. The temperature will be at the beginning 100, and will stop when is lower or equal of 10^{-9}. To assure the accuracy, the temperature will be multiplied by 0.98765. In this manner, the halting condition will be approached only after a few thousands changes.

4 Experimental Setup

The functions that will be used are: De Jong, Schwefel, Rastrigin and Michalewicz. Every function named will be tested for the dimensions: 5, 10 and 30. To reduce the possibility of failures, every set of parameters will be executed 30 times.

5 Results

5.1 De Jong

5.1.1 The function

$$f(x) = \sum_{i=1}^{n} x_i^2 \qquad -5.12 \leq x_i \leq 5.12$$

5.1.2 Global minimum

$f(x)$=0, x$_i = 0, i = 1 : n$

5.1.3 Hill Climbing-Best improvement results

Dimension	Min	Max	Mean	MinTime	MaxTime	MeanTime
5	0	0	0	400,63538	434,85856	417,74697
10	0	0	0	2935,23781	2967,95805	2951,59793
30	0	0	0	44135.51652	47488.21365	45811.86509

5.1.4 Hill Climbing-Worst improvement results

Dimension	Min	Max	Mean	MinTime	MaxTime	MeanTime
5	8,29160	4,93869	11,61515	16,89011	17,36301	17,12656
10	19,82334	38,43777	29,13055	61,89519	64,89519	63,39519
30	117,04329	165,37527	141,20928	475,74600	500,12521	487,93560

5.1.5 Hill Climbing-First improvement results

Dimension	Min	Max	Mean	MinTime	MaxTime	MeanTime
5	0	0	0	17,01978	18,64247	17,83113
10	0	0	0	63,31277	67,77253	65,54265
30	0	0	0	500,12521	520,11011	510,11766

5.1.6 Simulated Annealing results

Dimension	Min	Max	Mean	MinTime	MaxTime	MeanTime
5	0	0	0	14,98113	15,29679	15,13896
10	0	0	0	27,89553	28,62525	27,80205
30	0	0	0	70,41288	73,54651	71,97970

5.2 Schwefel

5.2.1 The function

$$f(x) = \sum_{i=1}^{n} -x_i \sin\left(\sqrt{|x_i|}\right) \qquad -500 \leq x_i \leq 500$$

5.2.2 Global minimum

$f(x)$=n·$418,9829, x_i = 420.9687, i = 1 : n$

5.2.3 Hill Climbing-Best improvement results

Dimension	Min	Max	Mean	MinTime	MaxTime	MeanTime
5	-2094,91380	-1942,23995	-2018,57687	44,23412	45,42963	44,83188
10	-4115,65342	-4036,63542	-4076,14442	394,78124	407,70517	401,24321
30	-11764.15532	-11045.16549	-11404.66040	10198.08817	11020.08817	10609.08817

5.2.4 Hill Climbing-Worst improvement results

Dimension	Min	Max	Mean	MinTime	MaxTime	MeanTime
5	-2094,91380	-1942,23995	-2018,57687	44,23412	45,42963	44,83188
10	-2273,72570	-1595,22323	-1934,47446	94,81537	100,51585	97,66561
30	-3689,86682	-2670,34120	-3180,10401	713,72649	747,99547	730,86098

5.2.5 Hill Climbing-First improvement results

Dimension	Min	Max	Mean	MinTime	MaxTime	MeanTime
5	-1851,56612	-1562,56481	-1707,06546	29,19583	32,21013	30,70298
10	-2749,53159	-2183,85418	-2466,69289	115,25483	120,64360	117,94921
30	-5675,38541	-3976,70571	-4826,04556	919,53546	928,19173	923,86360

5.2.6 Simulated Annealing results

Dimension	Min	Max	Mean	MinTime	MaxTime	MeanTime
5	-2094,91443	-1857,71565	-1976,31504	14,28973	17,96055	16,12514
10	-4155,48718	-3677,53198	-3916,50958	29,31123	33,06028	31,18576
30	-11555,82857	-10000,00000	-10777,91429	81,18154	85,78127	83,48141

5.3 Rastrigin

5.3.1 The function

$$f(x) = 10 \cdot n + \sum_{i=1}^{n} \left(x_i^2 - 10 \cdot \cos\left(2 \cdot \pi \cdot x_i\right) \right) \qquad -5.12 \leq x_i \leq 5.12$$

5.3.2 Global minimum

$f(x)$ =0, $\mathrm{x}_i = 0, i = 1 : n$

5.3.3 Hill Climbing-Best improvement results

Dimension	Min	Max	Mean	MinTime	MaxTime	MeanTime
5	0	1,00001	0,50001	327,62349	332,42596	330,02473
10	0	3,83764	1,91882	2317,89125	2426,82240	2372,35682
30	20.45686	24.51679	22.48683	40798.98562	43668.15698	42233.57130

5.3.4 Hill Climbing-Worst improvement results

Dimension	Min	Max	Mean	MinTime	MaxTime	MeanTime
5	24,85840	48,19480	36,52660	16,95793	18,18916	17,57355
10	80,22773	111,44372	95,83572	60,58997	64,33755	62,46376
30	343,89544	420,79593	382,34569	458,82411	479,03327	468,92869

5.3.5 Hill Climbing-First improvement results

Dimension	Min	Max	Mean	MinTime	MaxTime	MeanTime
5	4,59179	19,20961	11,90070	19,38373	21,71099	20,54736
10	46,76066	79,55745	63,15906	74,18490	77,55867	75,87179
30	312,19322	383,36991	347,78157	569,17355	588,54418	578,85886

5.3.6 Simulated Annealing results

Dimension	Min	Max	Mean	MinTime	MaxTime	MeanTime
5	0,00000	9,98997	4,99499	14,92356	16,10355	15,51356
10	4,22070	17,12397	10,67234	26,83416	29,33424	28,08420
30	24,07202	54,18718	39,12960	69,25733	73,55635	71,40684

5.4 Michalewicz

5.4.1 The function

$(x) = -\sum_{i=1}^{n} \sin(x_i) \cdot \left(\sin\left(\frac{i \cdot x_i^2}{\pi}\right)\right)^{2 \cdot m} \qquad 0 \leq x_i \leq \pi$

5.4.2 Global minimum

i= 1:n, m=10

$f(x)$= - 4.687, for n = 5

$f(x)$= - 9.66 , for n = 10

$f(x)$= - 29.63 , for n = 30

5.4.3 Hill Climbing-Best improvement results

Dimension	Min	Max	Mean	MinTime	MaxTime	MeanTime
5	-4,68766	-4,68533	0,00000	270,49534	280,30591	275,40063
10	-9,50971	1,14852	-4,18059	1823,43761	1900,39758	1861,91758
30	-27.89463	26.98468	-0.45497	31864.18955	33204.79791	32534.49373

5.4.4 Hill Climbing-Worst improvement results

Dimension	Min	Max	Mean	MinTime	MaxTime	MeanTime
5	-3,12529	-2,03172	-2,57850	16,78945	17,74715	17,26830
10	-4,31495	-3,27956	-3,79726	61,42568	64,55986	62,99277
30	-9,00494	-6,85083	-7,92788	460,51248	485,12319	472,81783

5.4.5 Hill Climbing-First improvement results

Dimension	Min	Max	Mean	MinTime	MaxTime	MeanTime
5	-4,28907	-3,09124	-3,69016	18,23582	19,08209	18,65896
10	-6,09819	-4,38261	-5,24040	67,79175	71,05459	69,42317
30	-11,22863	-11,22863	-11,22863	529,83460	541,93302	535,88381

5.4.6 Simulated Annealing results

Dimension	Min	Max	Mean	MinTime	MaxTime	MeanTime
5	-4,68698	-4,16404	-4,42551	15,01143	16,26427	15,63785
10	-9,58931	-8,53734	-9,06333	29,29870	32,45257	30,87563
30	-28,03831	-25,91540	-26,97686	70,49125	75,33619	72,91372

6 Comparison

The results are showing that **Hill Climbing-Best Improvement** gives the best results against all other methods used to calculate the next point/neighbour, even if it takes a lot of time.

In therms of efficiency the final countdown is between **Hill Climbing-Best Improvement** and **Simulated Annealing**. If the **result** is the important matter, **Hill Climbing-Best Improvement** will serve the best. But when **timing** matters as well, **Simulated Annealing** is the right method to use.

The other two methods, **Hill Climbing-Worst improvement** and **Hill Climbing-First improvement** can also be used successfully in certain contexts.

References

De Jong's function 1
Schwefel's function 7
Rastrigin's function 6
Michalewicz's function 12
Course support