Pure Random Search in Global Optimization

Georgiana Diana Ojoc

October 15, 2019

Abstract

This paper documents the heuristic method of *pure random search*, in comparison with the deterministic method of *exhaustive search*. Four benchmark functions with variable number of arguments are analysed, namely Ackley Function, Michalewics Function, Rastrigin Function and Rosenbrock Function.

1 Motivation

Global optimization is a branch of applied mathematics and numerical analysis that attempts to find the global minimum or maximum of a function or a set of functions on a given set. It is usually described as a minimization problem because the maximization of the real-valued function $g(x)$ is obviously equivalent to the minimization of the function $f(x) := (-1) \cdot g(x)$.

Random search is a family of numerical optimization methods that do not require the gradient of the problem to be optimized, and can hence be used on functions that are not continuous or differentiable. Such optimization methods are also known as *black-box* methods.

This method can return a reasonable approximation of the optimal solution within a reasonable time under low problem dimensionality, although it does not scale well with problem size, such as the number of dimensions.

Exhaustive search, also known as brute-force search, is a very general problem-solving technique and algorithmic paradigm that consists of systematically enumerating all possible candidates for the solution and checking whether each candidate satisfies the problem's statement.

While a brute-force search is simple to implement, its cost is proportional to the number of candidate solutions, which in many practical problems tends to grow very quickly as the size of the problem increases, concept known as *combinatorial explosion*. Therefore, brute-force search is typically used when the problem size is limited, but it is also useful as a baseline method when benchmarking other algorithms or *metaheuristics*.

2 Method

Considering the global optimization problem $min_{x \in S} f(x)$, *pure random search* generates a sequence of independent, identically distributed points in the feasible region S. Usually, the points are generated according to a uniform distribution, however, any probability distribution can be used. When a stopping criterion is met, the best point of the sequence generated thus far is used as an approximation to the optimal solution.

Pure random search is stated more formally below.

Pure Random Search

Step 0. Initialize $X_0 \in S$ according to probability measure δ on S.
Set iteration index $k = 0$. Set $Y_{min} = Y_0 = f(X_0)$.

Step 1. Generate X_{k+1} according to probability measure δ on S.
Set $Y_{k+1} = f(X_{k+1})$. Update the best point so far, $Y_{min} = min\{Y_{min}, Y_{k+1}\}$.

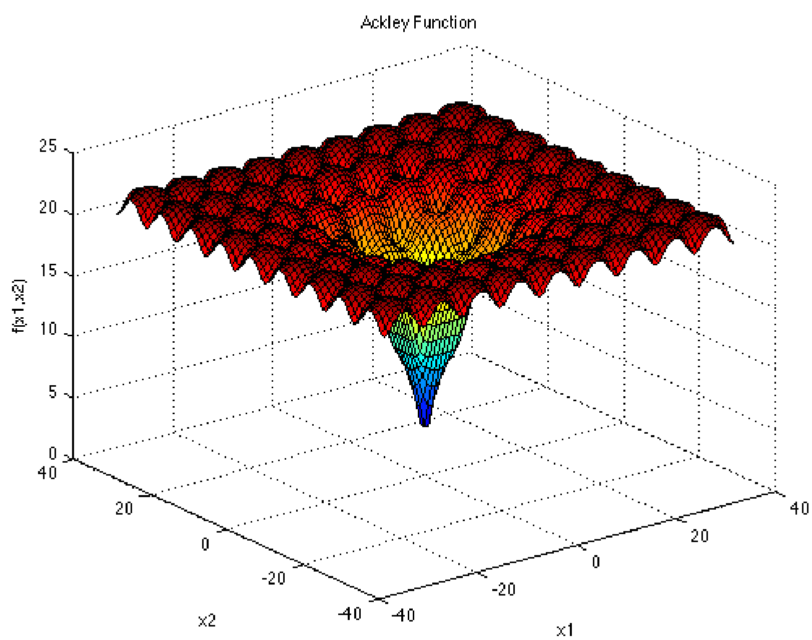
Step 2. If a stopping criterion is met, stop.
Otherwise, increment k and return to *Step 1.*

3 Results

The elapsed time is measured in seconds. The heuristic algorithm generated X_k 10 million times and it has been run 32 times for 2, 5 and 20 dimensions. The deterministic algorithm used a precision of maximum 3 decimals for the incremental value in the search domain of the function and it has been run one time for 2 and 3 dimensions, but for precision 3 and 3 dimensions, the elapsed time was exceeding 10 minutes.

Ackley Function

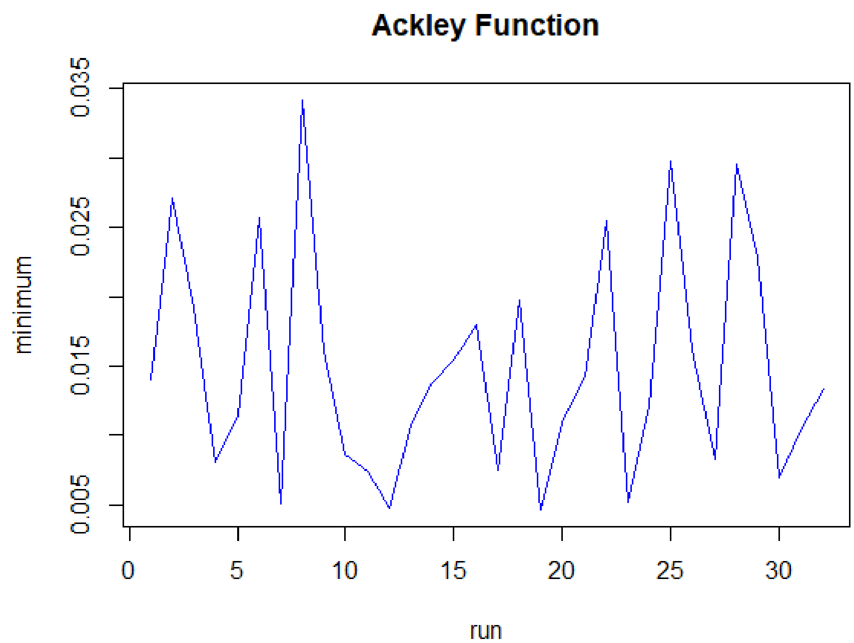
- Number of variables: n variables
- Definition: $f(\mathbf{x}) = f(x_1, ..., x_n) = 20 + e - 20e^{-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}} - e^{\frac{1}{n}\sum_{i=1}^{n} cos(2\pi x_i)}$
- Search domain: $-15 \leqslant x_i \leqslant 30,\ i = 1, 2, ..., n$
- Global minimum: $x^* = (0, 0, ..., 0),\ f(x^*) = 0$
- Function graph: for $n = 2$



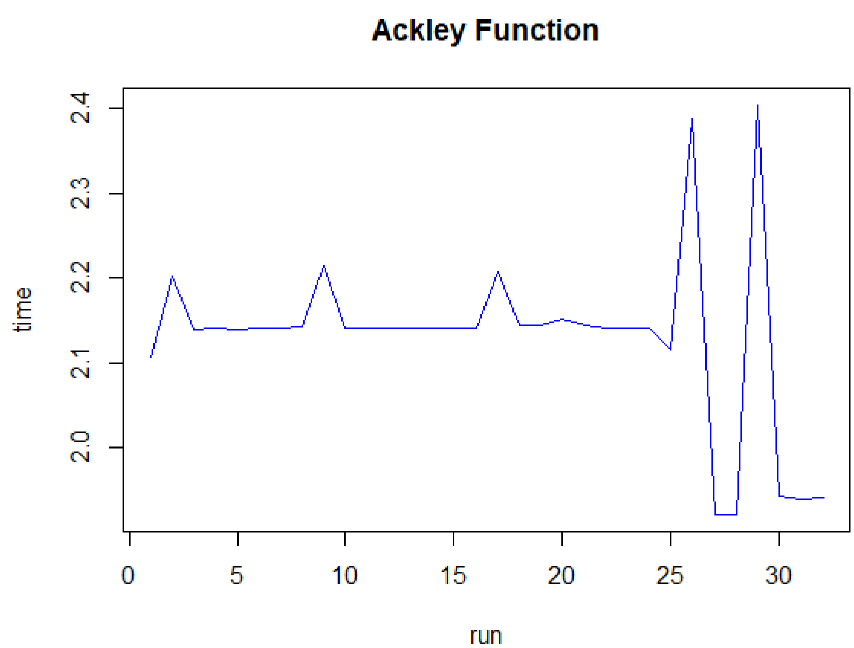
2 dimensions	**Lowest**	**Highest**	**Mean**	**Median**	**Standard deviation**
Minimum	0.004681	0.034145	0.014899	0.013587	0.008184
Time	1.921920	2.404063	2.129708	2.141146	0.107491

5 dimensions	**Lowest**	**Highest**	**Mean**	**Median**	**Standard deviation**
Minimum	2.240291	4.023564	3.299587	3.394518	0.411717
Time	4.104451	4.559666	4.434038	4.537758	0.190649

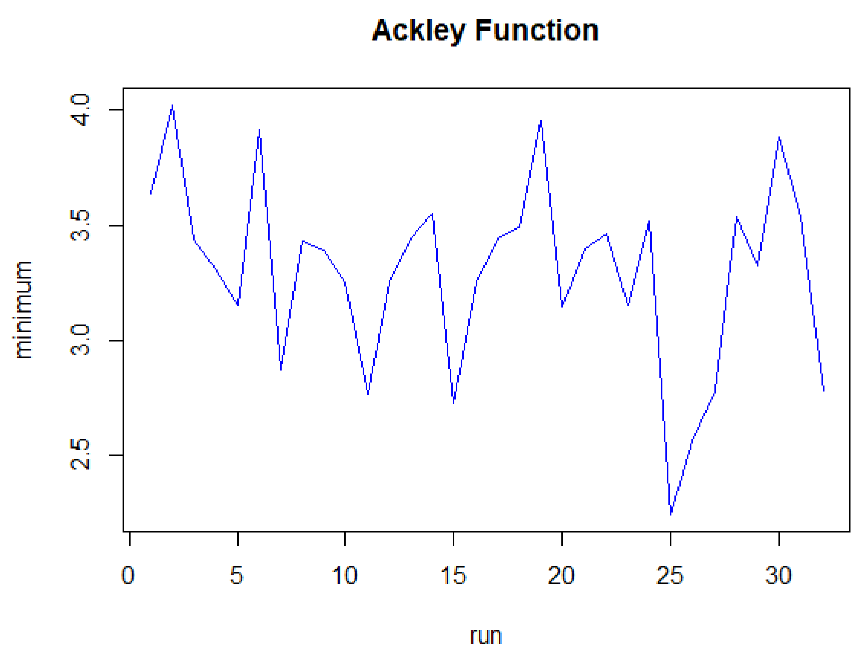
20 dimensions	**Lowest**	**Highest**	**Mean**	**Median**	**Standard deviation**
Minimum	13.241154	15.202898	14.349632	14.450031	0.438106
Time	14.364837	15.999902	15.503589	15.868703	0.665518



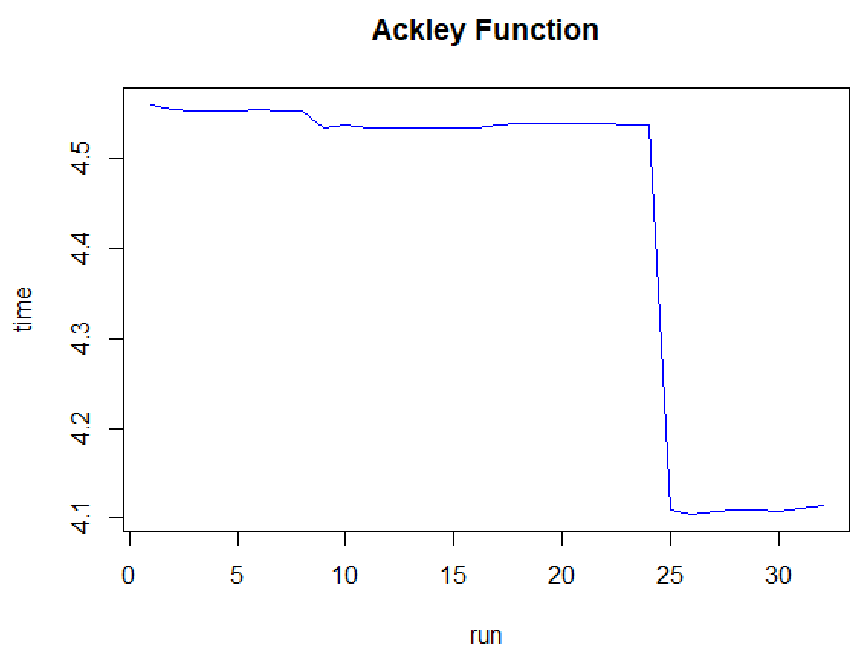
Plot 1. 2 dimensions



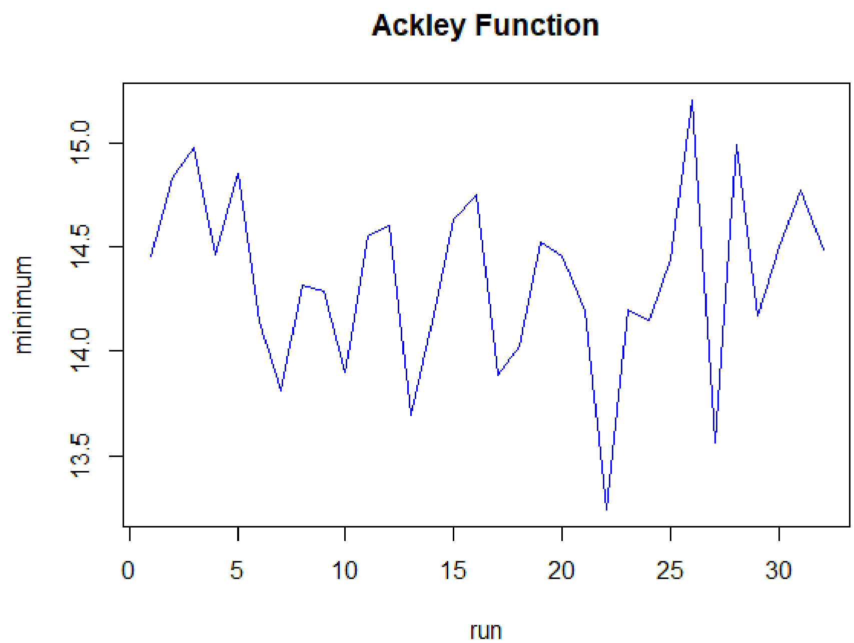
Plot 2. 2 dimensions



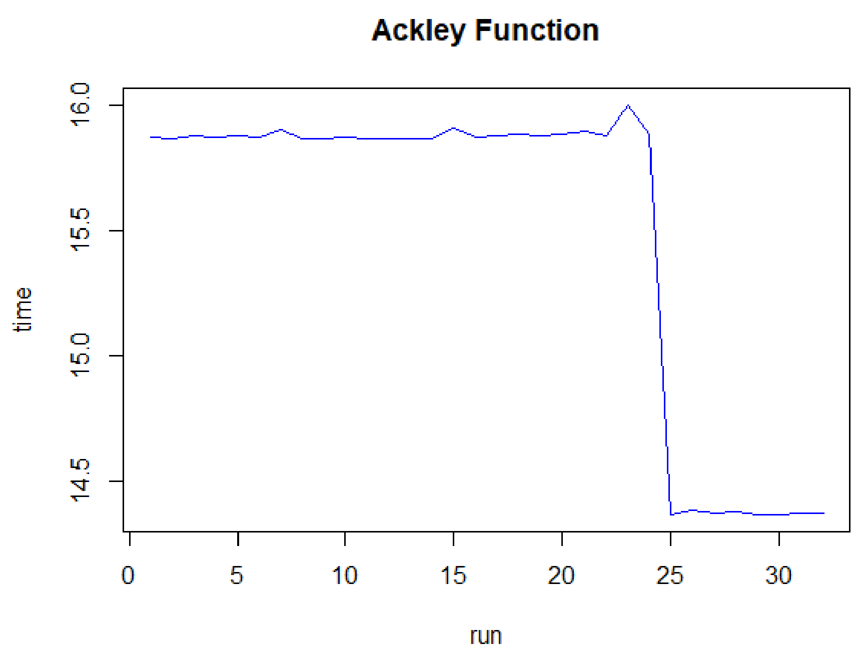
Plot 3. 5 dimensions



Plot 4. 5 dimensions



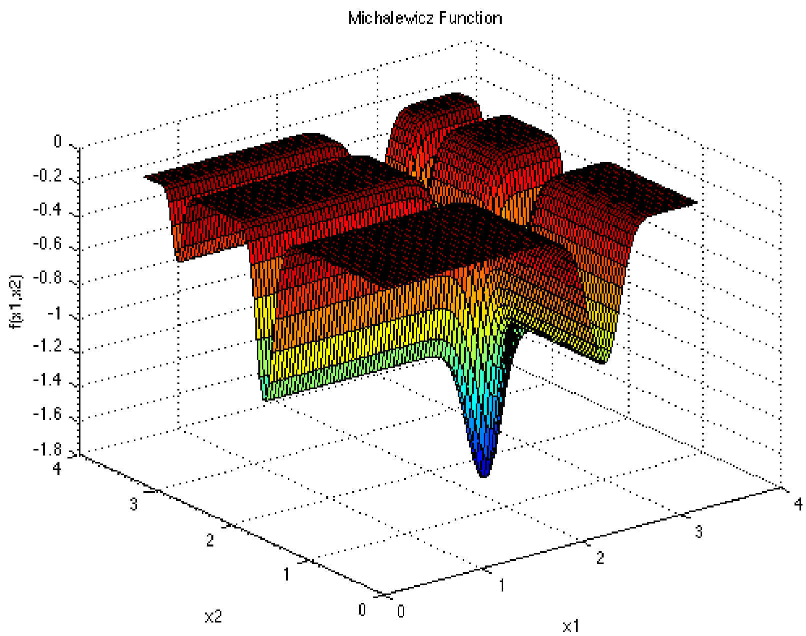
Plot 5. 20 dimensions



Plot 6. 20 dimensions

Michalewics Function

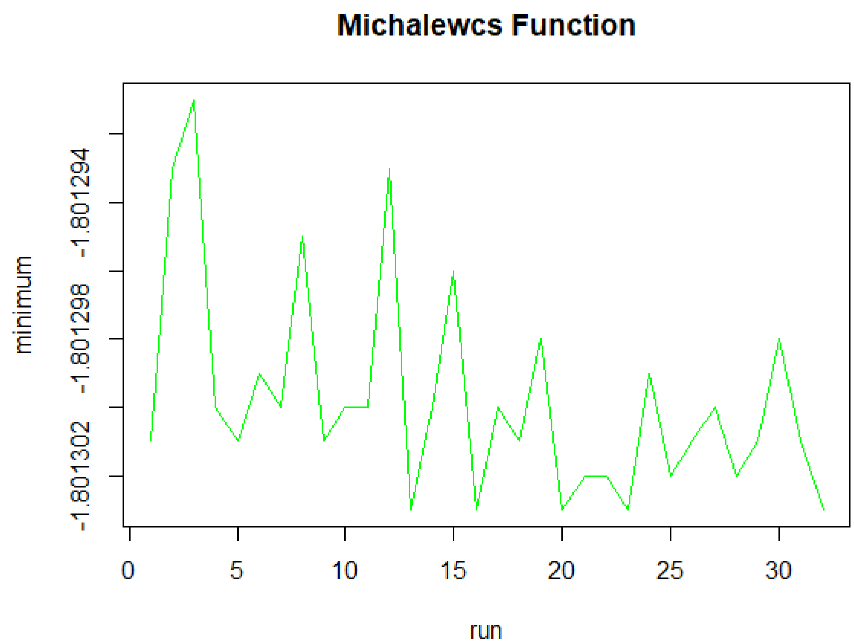
- Number of variables: n variables
- Definition: $f(\mathbf{x}) = f(x_1, ..., x_n) = -\sum_{i=1}^{n} \sin(x_i) \sin^{20}(\frac{ix_i^2}{\pi})$
- Search domain: $0 \leqslant x_i \leqslant \pi,\ i = 1, 2, ..., n$
- Global minimum: $n = 2 : f(x^*) = -1.801300$, $n = 2 : f(x^*) = -1.801300$, $n = 20 : f(x^*) = -19.637013$
- Function graph: for $n = 2$



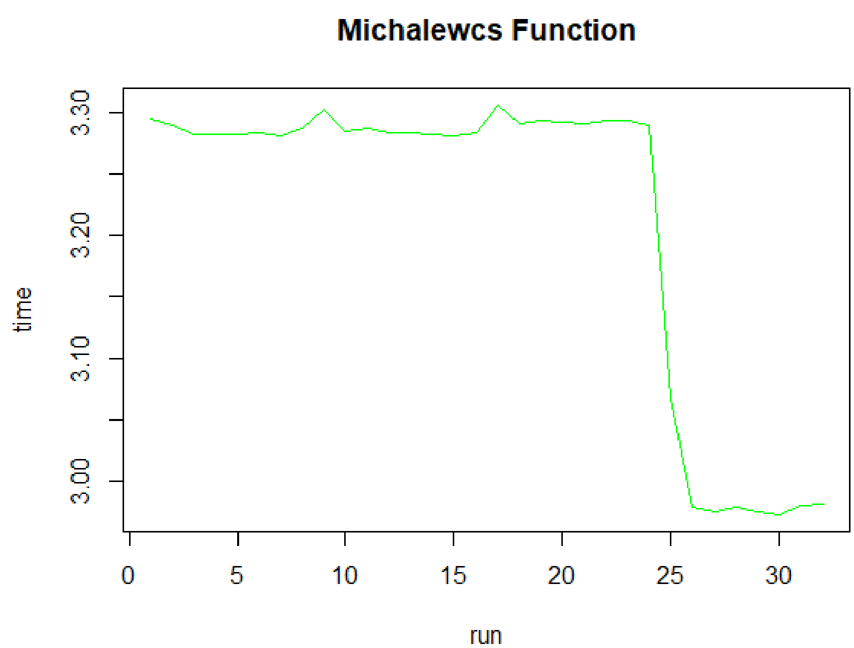
2 dimensions	**Lowest**	**Highest**	**Mean**	**Median**	**Standard deviation**
Minimum	-1.801303	-1.801291	-1.801299	-1.801300	3.090046
Time	2.973214	3.305908	3.213462	3.283472	0.132731

5 dimensions	**Lowest**	**Highest**	**Mean**	**Median**	**Standard deviation**
Minimum	-4.611290	-4.293530	-4.427474	-4.408140	0.075044
Time	7.658718	8.495882	8.265861	8.455539	0.354581

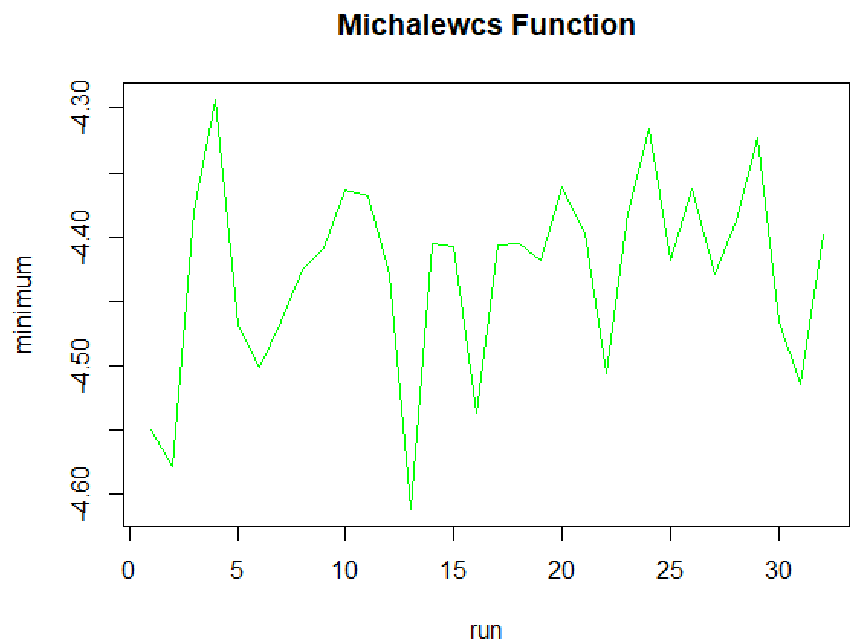
20 dimensions	**Lowest**	**Highest**	**Mean**	**Median**	**Standard deviation**
Minimum	-10.309255	-8.923238	-9.499013	-9.478783	0.298714
Time	30.782483	35.637232	33.277868	33.950150	1.489665



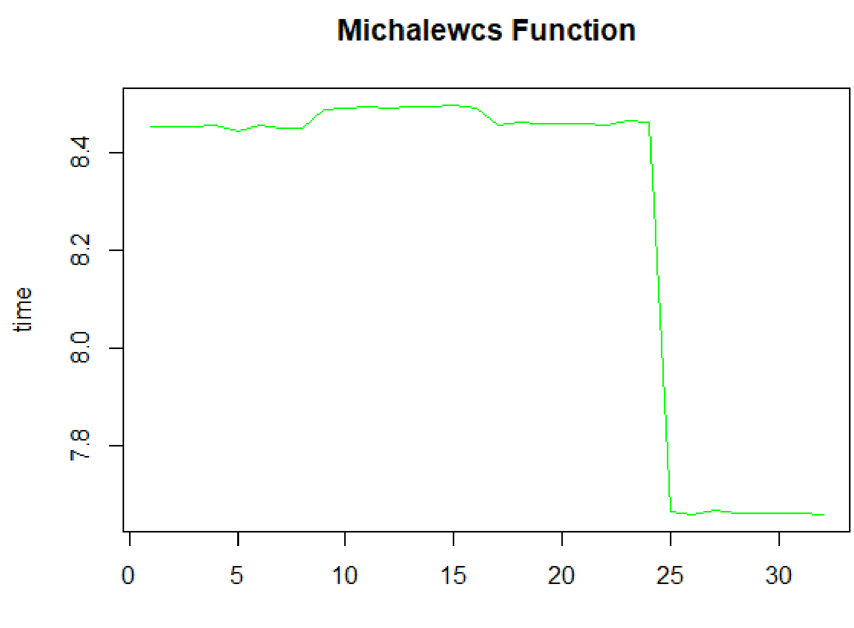
Plot 7. 2 dimensions



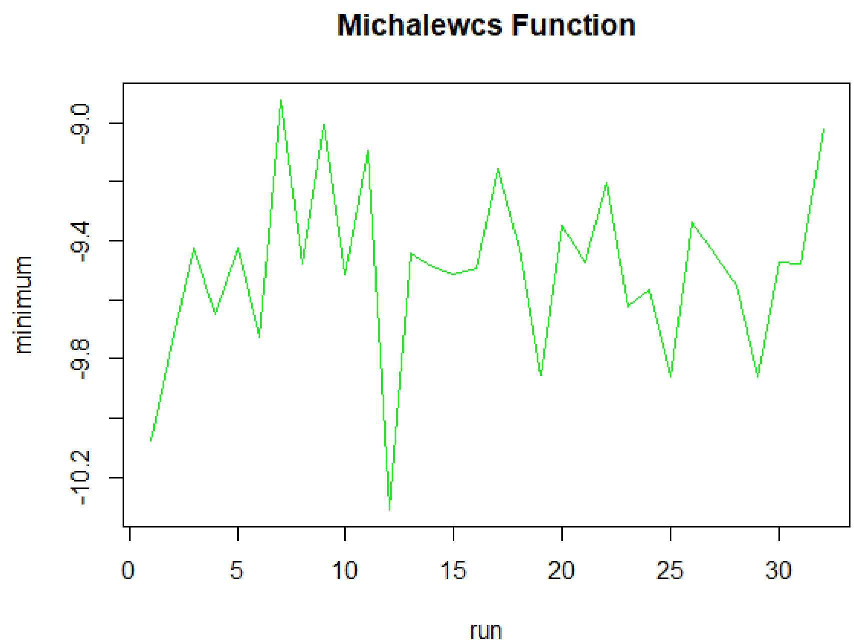
Plot 8. 2 dimensions



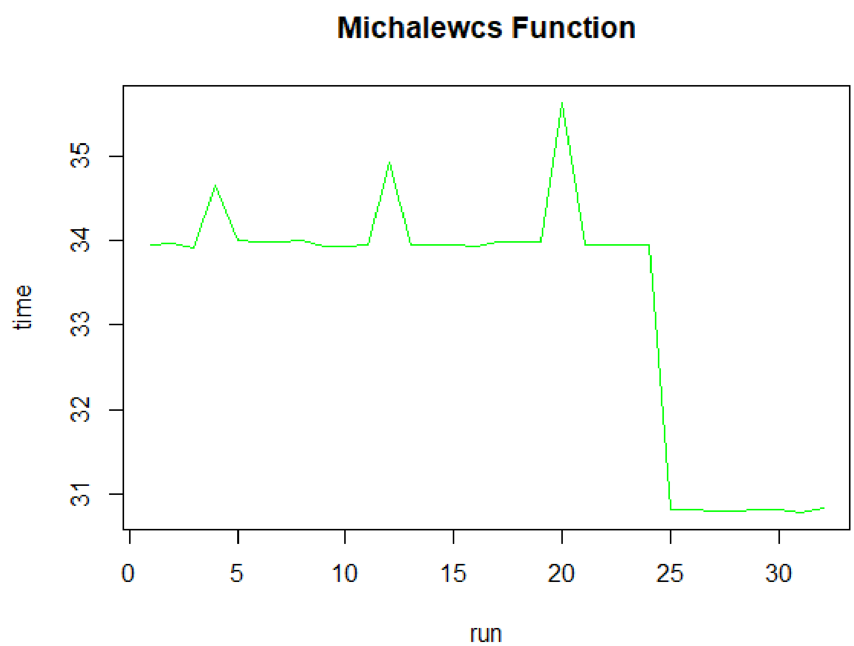
Plot 9. 5 dimensions



Plot 10. 5 dimensions



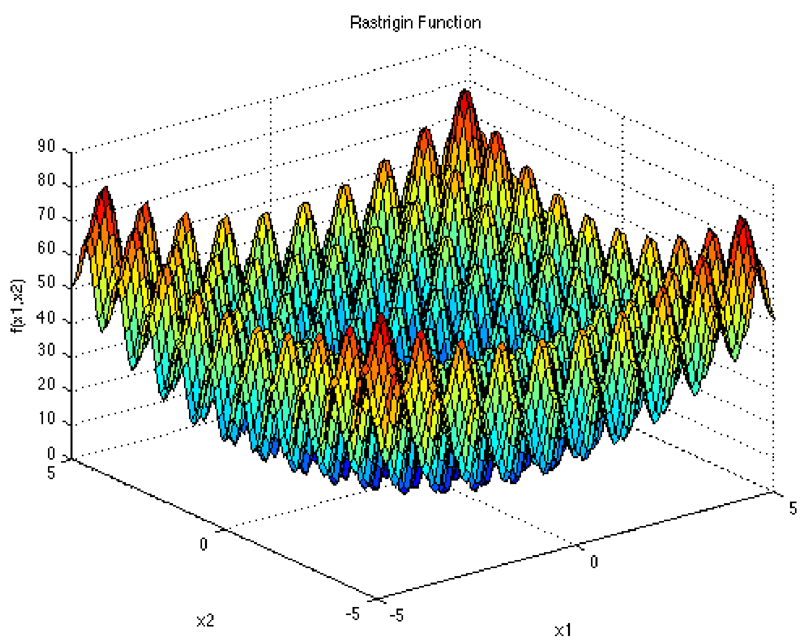
Plot 11. 20 dimensions



Plot 12. 20 dimensions

Rastrigin Function

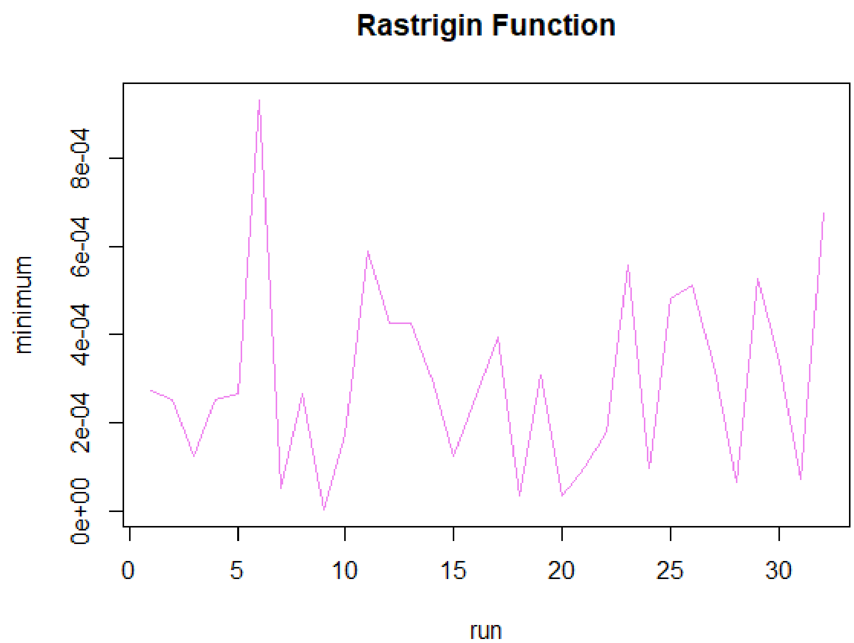
- Number of variables: n variables
- Definition: $f(\mathbf{x}) = f(x_1, ..., x_n) = 10n + \sum_{i=1}^{n}(x_i^2 - 10cos(2\pi x_i))$
- Search domain: $-5.12 \leqslant x_i \leqslant 5.12,\ i = 1, 2, ..., n$
- Global minimum: $x^* = (0, 0, ..., 0),\ f(x^*) = 0$
- Function graph: for $n = 2$



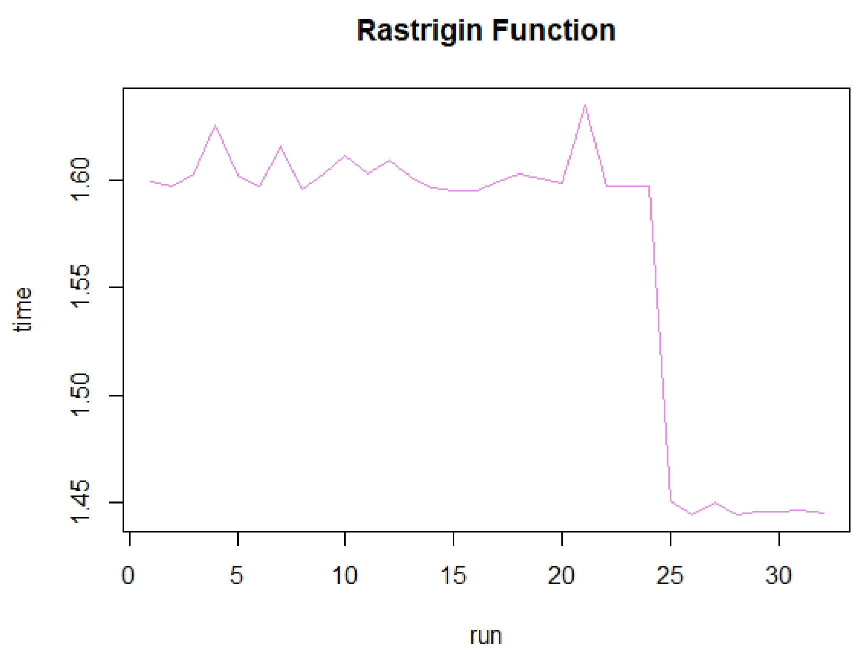
2 dimensions	**Lowest**	**Highest**	**Mean**	**Median**	**Standard deviation**
Minimum	0.000003	0.000931	0.000293	0.000265	0.000215
Time	1.444618	1.634849	1.564097	1.597118	0.069344

5 dimensions	**Lowest**	**Highest**	**Mean**	**Median**	**Standard deviation**
Minimum	1.524483	5.166165	3.665185	3.677805	0.912492
Time	3.666486	4.391095	3.978580	4.051079	0.190211

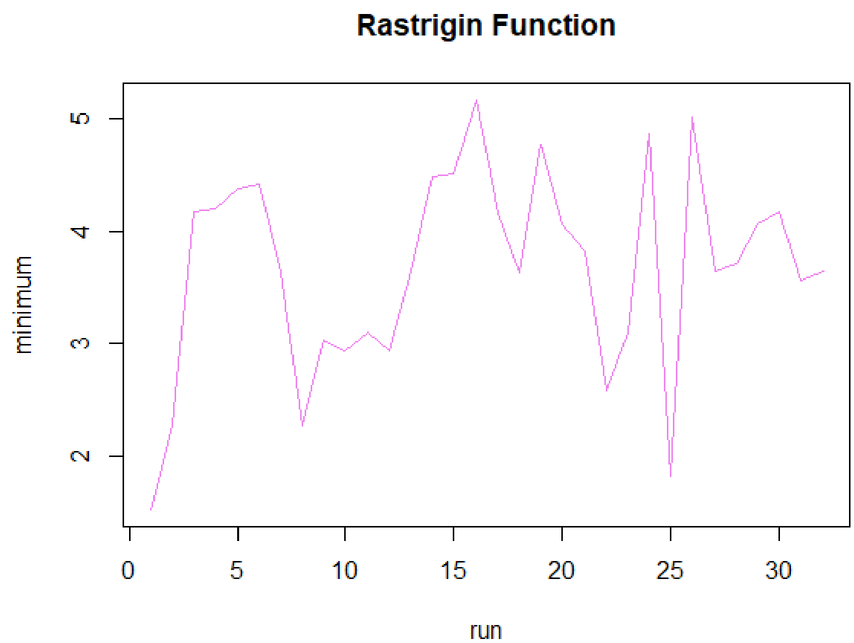
20 dimensions	**Lowest**	**Highest**	**Mean**	**Median**	**Standard deviation**
Minimum	125.985879	152.372851	143.630441	145.242357	6.356427
Time	14.104089	18.084485	15.915156	15.622196	0.936238



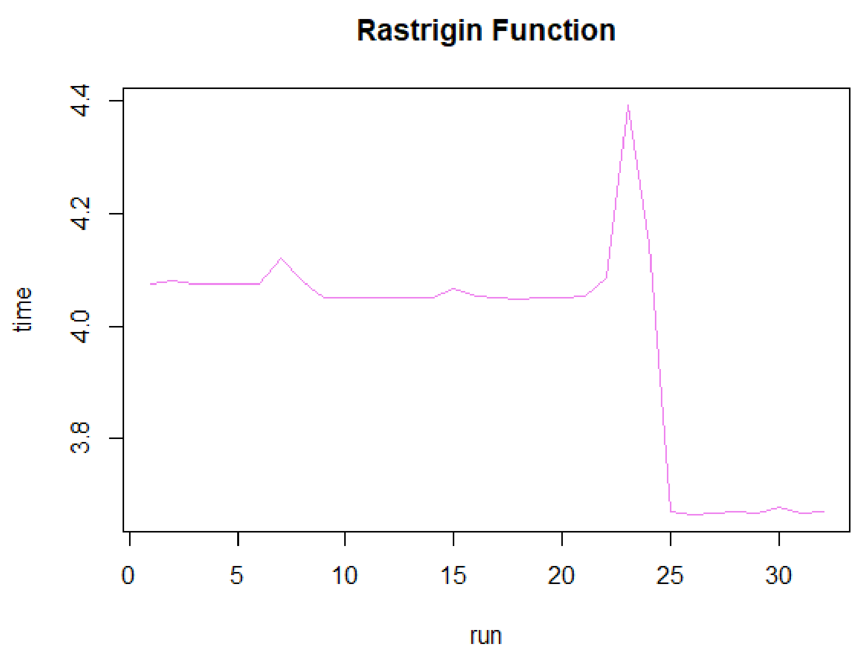
Plot 13. 2 dimensions



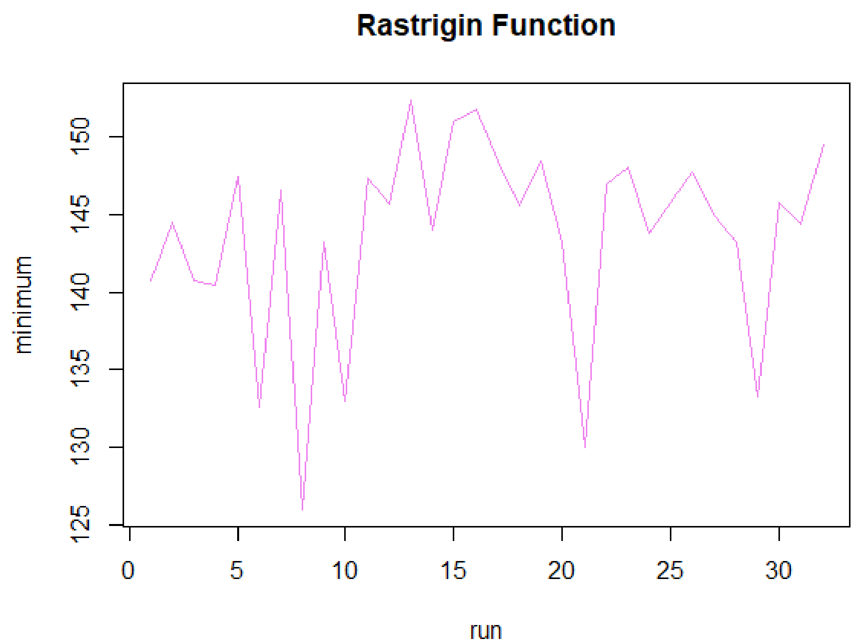
Plot 14. 2 dimensions



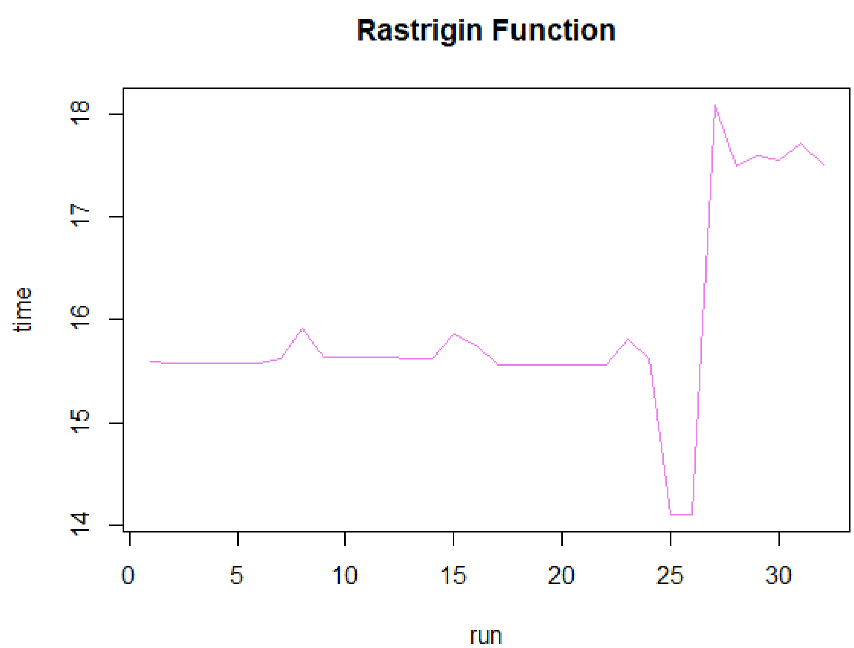
Plot 15. 5 dimensions



Plot 16. 5 dimensions



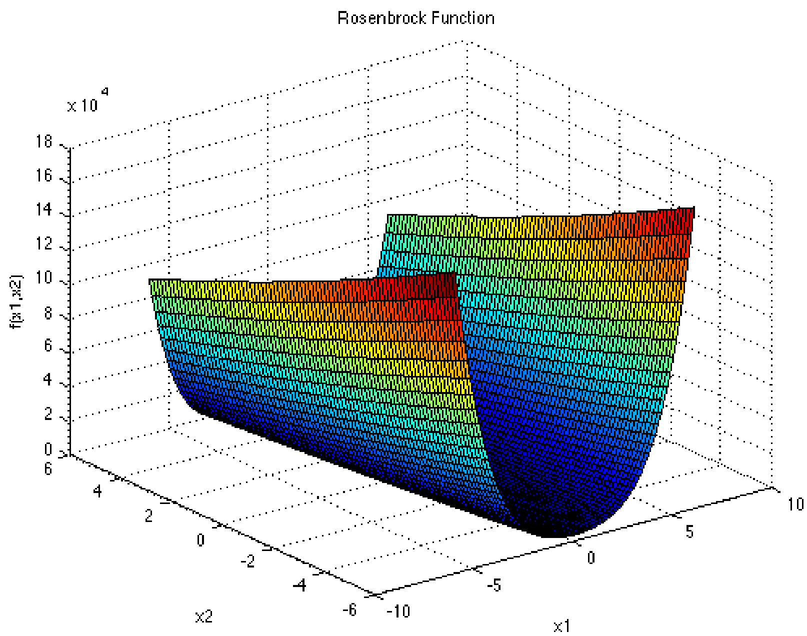
Plot 17. 20 dimensions



Plot 18. 20 dimensions

Rosenbrock Function

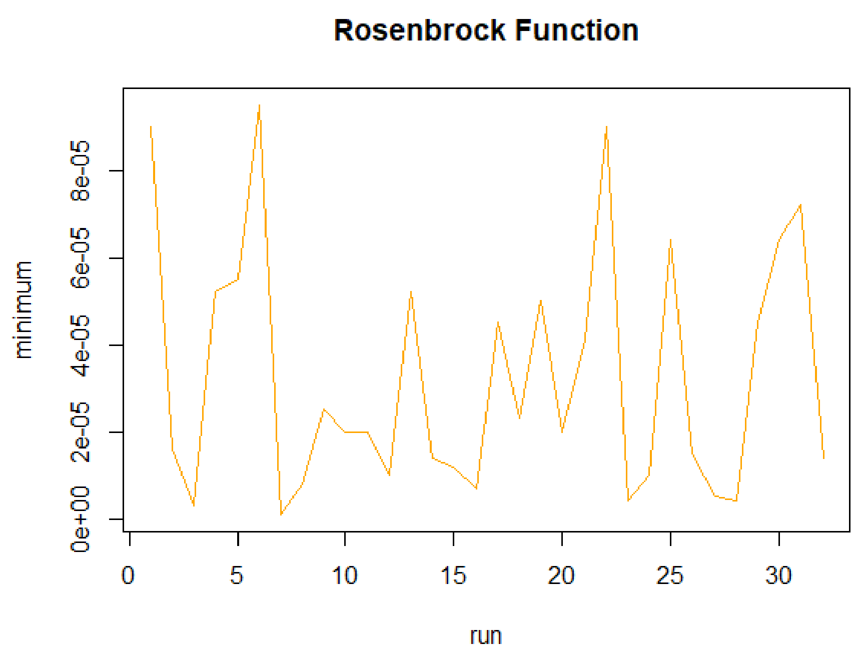
- Number of variables: n variables
- Definition: $f(\mathbf{x}) = f(x_1, ..., x_n) = \sum_{i=1}^{n-1}(100(x_i^2 - x_{i+1})^2 + (1 - x_i)^2)$
- Search domain: $-5 \leqslant x_i \leqslant 10,\ i = 1, 2, ..., n$
- Global minimum: $x^* = (1, 1, ..., 1),\ f(x^*) = 0$
- Function graph: for $n = 2$



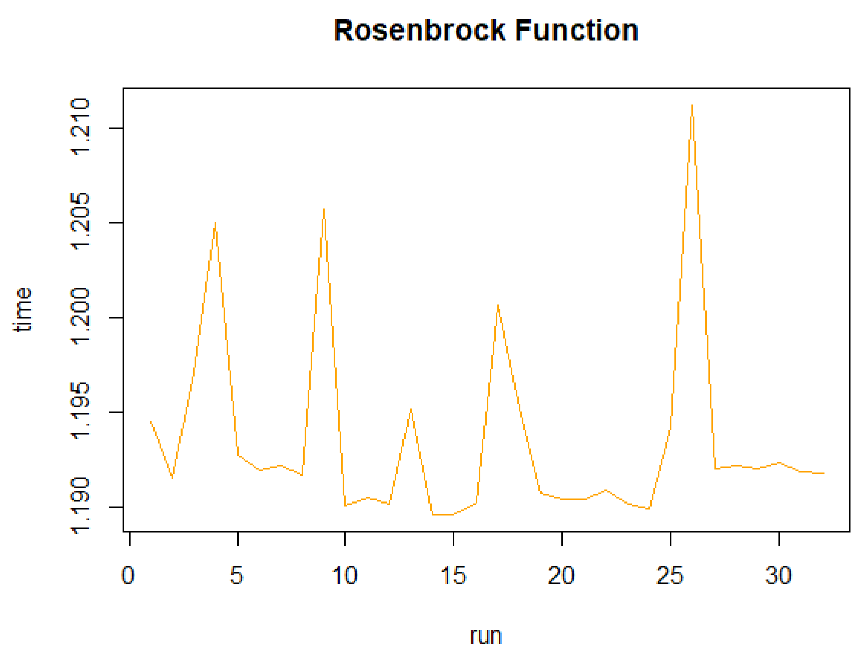
2 dimensions	**Lowest**	**Highest**	**Mean**	**Median**	**Standard deviation**
Minimum	0.000001	0.000095	0.000032	0.000020	0.000028
Time	1.189639	1.211185	1.193579	1.1919325	0.005135

5 dimensions	**Lowest**	**Highest**	**Mean**	**Median**	**Standard deviation**
Minimum	3.106602	14.524255	7.44607025	7.057480	3.014583
Time	3.254926	3.281803	3.264170	3.260069	0.009825

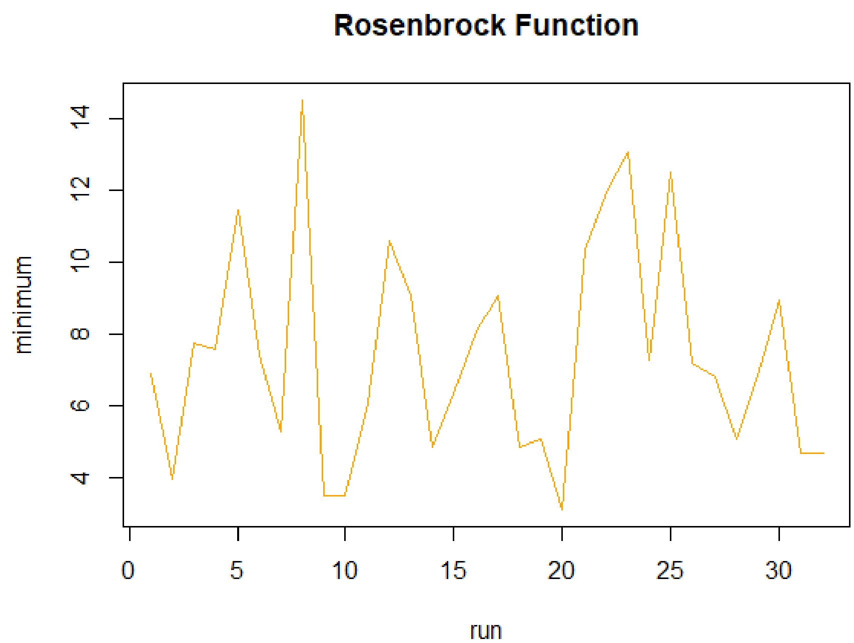
20 dimensions	**Lowest**	**Highest**	**Mean**	**Median**	**Standard deviation**
Minimum	16056.336493	49680.038313	33542.298794	32733.812702	7900.207404
Time	13.333528	13.360450	13.349122	13.348457	0.006334



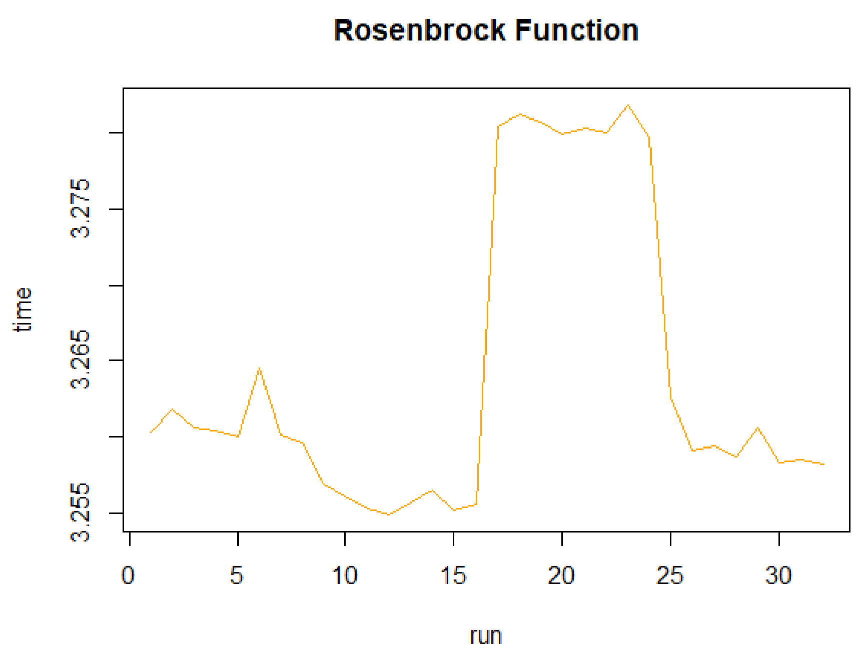
Plot 19. 2 dimensions



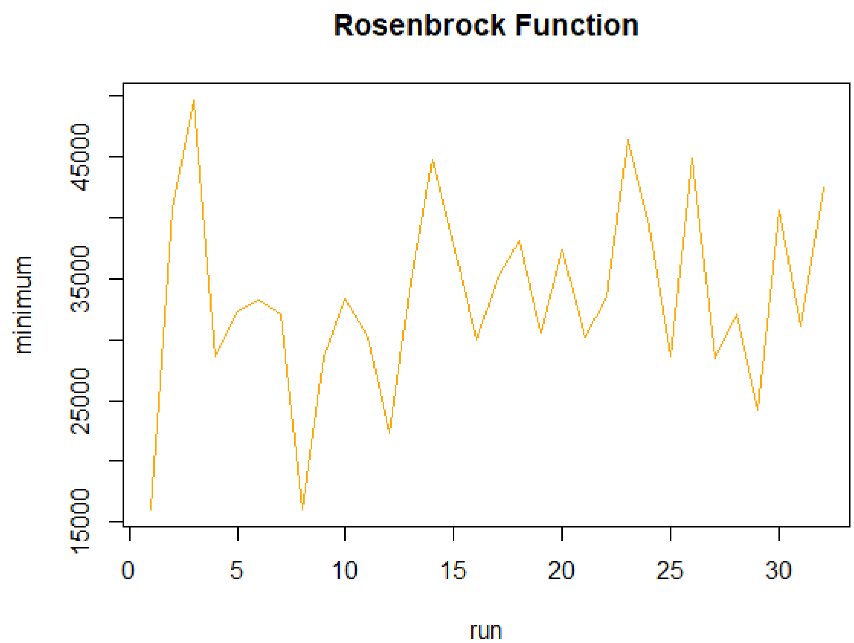
Plot 20. 2 dimensions



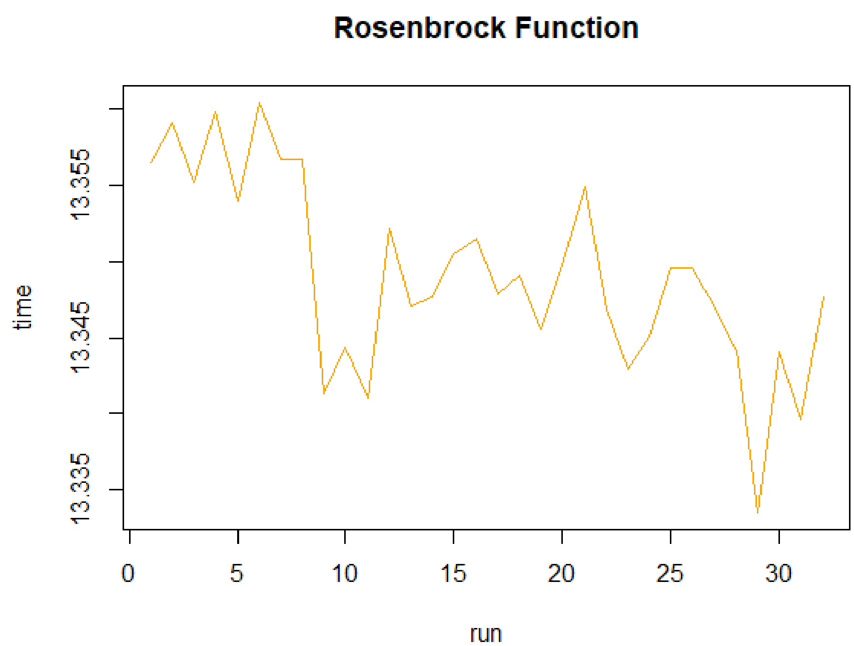
Plot 21. 5 dimensions



Plot 22. 5 dimensions



Plot 23. 20 dimensions



Plot 24. 20 dimensions

4 Conclusions

Pure random search is much faster than *exhaustive search*, but the deterministic algorithm gives a better result than the heuristic algorithm. In the case of pure random search, the more the number of dimensions increases, the higher the calculated minimum is, due to fluctuations in generating the function argument. Therefore, because the deterministic algorithm cannot determine the global minimum of a function with 20 dimensions in a reasonable time, it is recommended to use the heuristic algorithm, which can be improved.

References

- Global optimization
- Random search
- Exhaustive search
- Benchmark functions
- Ackley function
- Michalewics function
- Rastrigin function
- Rosenbrock function