

The formation of initial point initialization methods for optimization algorithms

V V Tynchenko^{1,2}, A A Pavlenko¹, V V Bukhtoyarov^{1,2}, D V Tikhonenko¹, S V Tynchenko^{1,2} and A V Tsvettsykh¹

¹ Reshetnev Siberian State University of Science and Technology, 31, Krasnoyarsky Rabochy Av., Krasnoyarsk, 660037, Russian Federation

² Siberian Federal University, 79, Svobodny pr., Krasnoyarsk, 660041, Russian Federation

E-mail: ms_f1_pavlenko@mail.ru

Abstract. The initial point initialization method is one of the main parameters for global optimization algorithms. Many scientists are engaged in its construction. The importance of this parameter for the entire algorithm is still not proven at all. Today, initialization methods based on stochastic algorithms are used. Six algorithms for constructing multidimensional points for global optimization algorithms — boolean strings — have been designed. The available algorithms have been analyzed. We used the starting points scattering algorithms, which are: LP τ sequence, UDC sequence, uniform random scatter. A large number of algorithms relaunches were used. The best way to initialize the starting points for the non-parametric, genetic algorithm, the MIVER scheme algorithm and the collective optimization method based on the Co-Operation of Biology Related Algorithms (COBRA) for these test functions has been determined.

1. Introduction

When solving optimization problems, one of the main points is the way to initialize the starting points. In optimization algorithms [1 - 3], the points are represented as Boolean strings.

UDC sequence, LP τ -sequence [4], uniform random scatter - effective starting point scattering algorithms that were used during the experiments. In these experiments, only some of the initialization methods were used, the initialization methods were not used without randomness. The applied variations used a weak algorithm for the Boolean strings formation and were tested on a small number of practical problems [5]. Uniform random scatter is a stochastic point scatter algorithm using the normal distribution law. UDC sequences and LP τ -sequences are nonrandom starting point scattering algorithms. This may lead to more confident final results of optimization algorithms when using them.

2. Explanatory part

Experiments were carried out on the Rastrigin function, Shekel function, Acley function, Grievanka function [6]. LP τ -sequences are a non-random algorithm of forming points based on a matrix of irreducible Marshal polynomials. UDC sequences are a non-random algorithm of absolutely uniform points formation along all coordinates in a multidimensional space [7], regardless of the number of scatter points [8]. Boolean strings are filled in different ways:

- Directly using random sequences with a uniform distribution law.
- Filled with random sequences (with a uniform distribution law) in the space of real numbers, and then convert real numbers to boolean ones.

- Filled with non-random sequences.

Recent experiments in this field were carried out in the works of the scientist [8]. In his works binary lines were filled without taking into account the check for repeatability.

3. Experimental part

The first way is that using random sequences (with a uniform distribution law) get a real number from 1 to 100. If the resulting number is less than 50, then the corresponding bit of the Boolean string takes the value 1, otherwise 0. So, boolean strings are obtained (Figure 1).

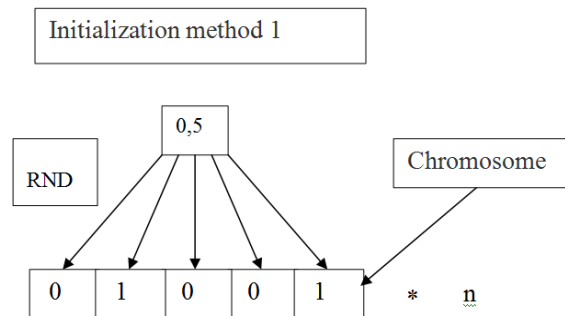


Figure 1. First initialization method.

The second way is that using random sequences (with a uniform distribution law) get a real number from 1 to K, where K is the maximum real number, obtained if each bit of the Boolean string was equal to 1. Then, this real number is converted to a boolean string by the conversion rules. So, boolean strings are obtained (Figure 2).

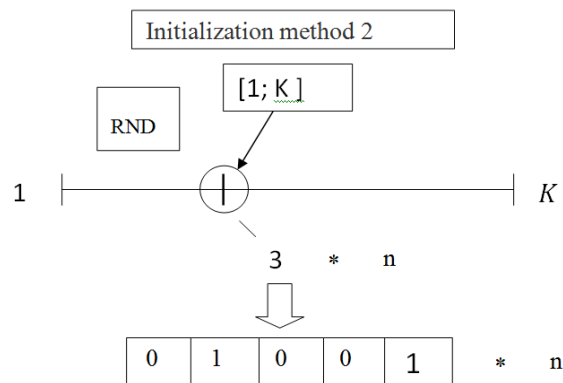


Figure 2. Second initialization method.

The 3rd method is the same 1st method, but also with checking the boolean rows for repeatability (Figure 3).

The fourth method is the same second method, but also with the verification of real numbers for repeatability (Figure 4).

The fifth method is that using a $LP\tau$ -sequence, a real number from 1 to 100 is obtained. If the number obtained is less than 50, then the corresponding bit of the Boolean string takes the value 1, otherwise 0. So, boolean strings are obtained (Figure 5).

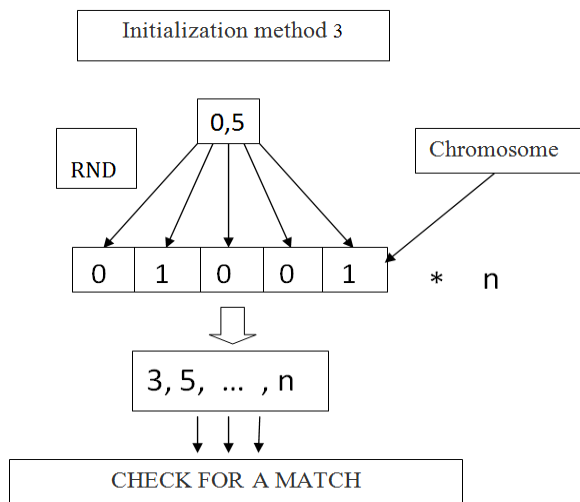


Figure 3. Third initialization method.

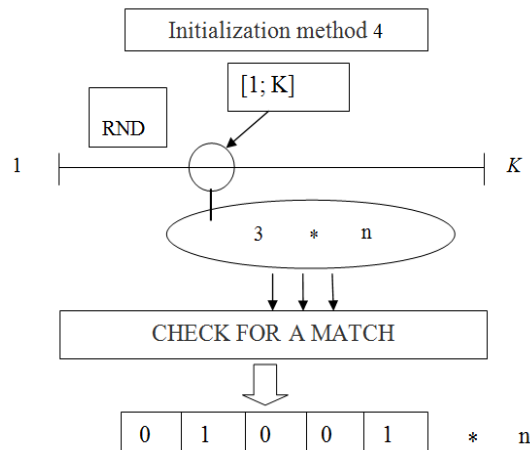


Figure 4. Fourth initialization method.

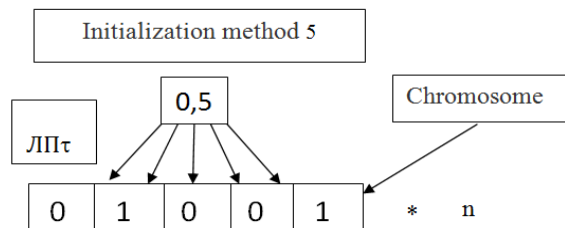


Figure 5. Fifth initialization method.

The 6th way is that with the help of LP_{τ} -sequences a real number is obtained from 1 to K , where K is the maximum real number, which would result if each bit of a Boolean string was equal to 1. Then, this real number is converted into a boolean string by conversion rules.

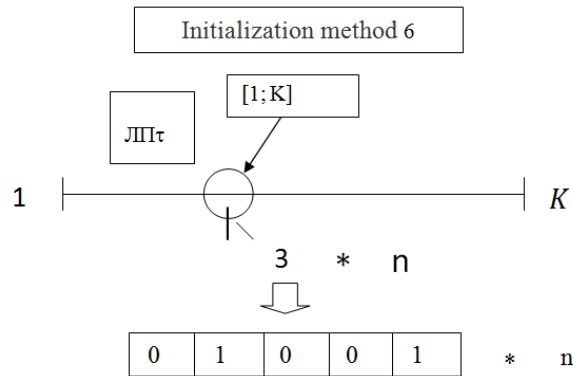


Figure 6. Sixth initialization method.

Initialization methods were tested on different algorithms.

4. Results

Figures 1-6 shows the schemes for constructing methods of initializing boolean strings.

According to Table 1, we can say that the best method in terms of convergence and efficiency, and the criterion relating convergence and efficiency is the initialization method based on the $LP\tau$ -sequence. In terms of convergence, the initialization method based on the UDC spread is worse than the initialization method based on a uniform random spread only when the number of points is less than 40. In all other cases, in terms of convergence and in all cases, in terms of efficiency, the initialization method based on the UDC spread exceeds the initialization method based on a uniform random spread, but doesn't exceed the initialization method based on the $LP\tau$ spread.

Table 1. Efficiency of the initialization methods.

Initialization method	Convergence, steps	Efficiency, %	Criterion
$LP\tau$	12	80	7,3
RND	17	54,6	3,5
UDC	20	75,5	4,6

There Criterion is the ratio of finding extremum percent to the algorithm steps.

The percentage of finding an extremum, on average, in all optimization functions is better for the initialization method 5 (Table 2), it follows that the initialization method 5 has the greatest reliability (probability) of finding the extremum.

Table 2. Efficiency of the initialization method 5 on different functions.

Test function	Initialization method number						Number of the best initialization method
	1	2	3	4	5	6	
1	13.75	12.00	15.50	17.50	20.75	1.50	5
2	21.25	25.75	23.50	22.50	68.50	6.25	5
3	0	0	0	0	0	0	-
4	5.50	3.25	4.50	3.00	5.25	1.00	1
Total							5

The percentage of finding an extremum, on average, in all optimization functions is better for the initialization method 5, it follows that the initialization method 5 has the greatest reliability (probability) of finding the extremum.

In terms of speed, we can say that with 50 points, the algorithm calculates 200 steps (generations), on average, in 1/400 seconds. Accuracy is the inverse characteristic of an error. For all optimization functions, the accuracy is better with the initialization method 5. The highest accuracy of finding the extremum (the quality of the solution) according to the expectation of the initialization method 5.

5. Conclusion

Six algorithms for constructing multidimensional points for global optimization algorithms — boolean strings — were designed and analyzed. The available algorithms were analyzed. The studies were carried out on the function of Akli, functions of Rastrigin, functions of Shekel, functions of Grivanka. Three initial point scattering algorithms were used: LP τ -sequence, UDC sequence, uniform random scatter. A large number of optimization re-launches algorithms were used. The experiments were carried out with high precision calculations. The best way to initialize the starting points for the non-parametric, genetic algorithm, MIVER scheme algorithm and the collective optimization method based on common bionic algorithms for these test functions was revealed - Co-Operation of Biology Related Algorithms (COBRA).

References

- [1] Zaloga A N, Yakimov I S and Dubinin P S 2018 Multipopulation genetic algorithm for determining crystal structures using powder diffraction data *Journal of surface investigation: X-ray, synchrotron and neutron techniques* **1(12)** 128-34
- [2] Tynchenko V S, Petrovsky E A and Tynchenko V V 2016 The parallel genetic algorithm for construction of technological objects neural network models *2nd International Conference on Industrial Engineering, Applications and Manufacturing, ICIEAM 2016* 7911573
- [3] Tynchenko V S, Tynchenko V V, Bukhtoyarov V V, Tynchenko S V and Petrovskiy E A 2016 The multi-objective optimization of complex objects neural network models *Indian Journal of Science and Technology* **9(29)** 99467
- [4] Akhmedova S, Stanovov V and Semenkin E 2018 Soft island model for population-based optimization algorithms *International Conference on Swarm Intelligence* 68-77
- [5] Zaloga A 2015 Genetic algorithm for automated X-Ray diffraction full-profile analysis of electrolyte composition on aluminium smelters *Informatics in control, automation and robotics 12th International Conference, ICINCO* 79-93
- [6] Stanovov V, Akhmedova S and Semenkin E 2018 Selective pressure strategy in differential evolution: exploitation improvement in solving global optimization problems *Swarm and Evolutionary Computation* 183-9
- [7] Yakimov I 2018 Application of evolutionary rietveld method based XRD phase analysis and a self-configuring genetic algorithm to the inspection of electrolyte composition in aluminum electrolysis baths *Crystals* **11(8)** 402
- [8] Du X 2019 Genetic algorithm optimized non-destructive prediction on property of mechanically injured peaches during postharvest storage by portable visible/shortwave near-infrared spectroscopy *Scientia Horticulturae* **249** 240-9