Measuring performance of optimization algorithms in evolutionary computation

Ivković, Nikola; Jakobović, Domagoj; Golub, Marin

Measuring Performance of Optimization Algorithms in Evolutionary Computation

Nikola Ivkovic, Domagoj Jakobovic, and Marin Golub

Abstract—**Reporting the results of optimization algorithms in evolutionary computation is a challenging task with many potential pitfalls. The source of problems is their stochastic nature and inability to guarantee an optimal solution in polynomial time. One of the basic questions that is often not addressed concerns the method of summarizing the entire distribution of solutions into a single value. Although the mean value is used by default for that purpose, the best solution obtained is also occasionally used in addition to or instead of it. Based on our analysis of different possibilities for measuring the performance of stochastic optimization algorithms presented in this paper we propose quantiles as the standard measure of performance. Quantiles can be naturally interpreted for the designated purpose. Besides, they are defined even when the arithmetic mean is not, and are applicable in cases of multiple executions of an algorithm. Our study also showed that, on the contrary to many other fields, in the case of stochastic optimization algorithms the greater variability in measured data can be considered as an advantage.**

Index Terms—**Algorithmic performance, experimental evaluation, metaheuristics, quantile.**

I. INTRODUCTION

Experimental research in evolutionary optimization is fairly common, especially in the case of nature-inspired algorithms that often have modest theoretical foundations, or entirely lack them. Although performing experiments with a stochastic optimization algorithm and presenting the results in a way that will satisfy high scientific standards might seem easy, it can actually prove to be a tricky business. Over the past years, a number of works have been published that point out shortcomings in experimental practice or propose certain procedures, including [1]-[6]. Some problems addressed in those papers arise only when the results for different problem instances need to be summarized. In this article, we assess different measures of algorithmic performance, which is an aspect that is often omitted in similar papers and possibly even considered as trivial. As a matter of fact, the decision of choosing an appropriate measure of performance is relevant regardless of the number of problem instances that are used in experiments (i.e. one or many).

This discussion is mainly motivated by the papers of Eiben and Jelasity [3] and Birattari and Dorigo [7] that discuss the suitability of the best solution found and the arithmetic mean

as measures of the performance of stochastic optimization algorithms. Eiben & Jelasity noticed that the arithmetic mean is not well suited for measuring the performance of stochastic optimization algorithms although it is typically used for that purpose. Consequently, they suggested the best solution found as a measure of performance. On the other hand, Birattari & Dorigo cautioned that the best solution found cannot be used as a measure of algorithmic performance because the best solution is not a reproducible result. Instead, they further recommended the usage of the arithmetic mean. Although they did not offer a solution for problems pointed out by Eiben and Jelasity, they admitted that a proper research methodology should encompass a widely adopted practice of multiple executions of the algorithm.

Lately, it has become more widely recognized that parametric statistical methods are not appropriate for measuring the performance of stochastic optimization algorithms [8], [9]. Consequently, the median is sometimes used as a measure of algorithmic performance.

In this paper, we analyze different possibilities for measuring the performance of algorithms, taking into consideration the useful practice of multiple executions of an algorithm, and advocate quantiles as the most suitable measure of performance. The suitability of reporting of additional statistical information such as spread of data is also discussed.

II. MEASURING PERFORMANCE OF STOCHASTIC OPTIMIZATION ALGORITHMS

In the analysis of stochastic optimization algorithms it is important to pay attention to two important concepts. One is the quality of solutions, since stochastic optimization algorithms usually do not guarantee finding the optimal solutions, and the other is the amount of computational resources necessary for obtaining the solutions. A trade-off between the quality of solutions and the amount of computational resources is usually possible in a way that more computational resources can allow the algorithm to find better solutions. The computational resources include processing elements, memory and elapsed time, among others. In the case of experimental research, time is the resource that is usually considered, while other resources are often regarded either as available or unavailable. Time, which can be expressed directly in physical units, is often expressed indirectly by the number of iterations, number of generated solutions, and number of executed instructions, among others. In the following sections the term "computational resources" will be used in the general sense although in practice it primarily refers to time expressed directly or indirectly.

In terms of solution quality and computational resources, there are three different approaches to measuring the performance of stochastic optimization algorithms:

1) The most common approach is to restrict computational resources and to observe the solution quality.

2) Another approach is to specify the minimal solution quality and then observe the amount of computational resources required to reach the specified solution. However, practical issues can arise in that case since the optimal solution quality is generally not known in advance. As a result, the solution quality that surpasses the quality of an optimal solution, which is impossible to achieve, may be required by mistake.

3) In the combined approach both maximal computational resources and the minimal solution quality are prespecified simultaneously. After that, the success rate of the algorithm is observed. This approach is the most restrictive among the three described approaches since both maximal computational resources and minimal solution quality must be provided in advance, which is often not convenient. Considering that the third approach has an obvious measure of performance – the success rate – in this paper we are focused on evaluating the performance for the first two approaches.

Optimization algorithms in the evolutionary computation are stochastic and produce different solutions on different executions. Although the whole distribution represents the most comprehensive information, it often means too much information to obtain or to handle directly, which entails that a distribution is usually analyzed by looking at some numbers that describe it. In the analysis suitable representative number(s) that satisfy basic scientific requirements should be used. It is mandatory that the measure of performance can be obtained in an objective way, to allow reproducibility (i.e. that the obtained results can be reproduced by other independent scientists) and that it is informative (i.e. ensuring that the measure of algorithmic performance has a useful interpretation). The measure of performance of stochastic optimization algorithms should ideally incorporate the common and beneficial practice of multiple executions of an algorithm.

A. Arithmetic Mean

The arithmetic mean still prevails as a measure of algorithmic performance. It allows for the reproducibility of results, provided that it is possible to measure the arithmetic mean in a reliable way and that a suitably large and representative sample is used. The most recognized disadvantage of using the arithmetic mean for stochastic optimization algorithms is that it is not well suited in the case of asymmetrical distribution, which is relevant when we consider that distributions of solution quality for stochastic optimization algorithms indeed are often asymmetrical [8], [10], [11].

Another disadvantage is that the arithmetic mean is not very informative. In the case that the arithmetic mean of the solutions' quality for a certain algorithm and problem instance is known, this information can be interpreted as follows: if the algorithm is executed many times, the sum of the achieved solutions as well as their arithmetic mean is expected to attain certain values. The sum of the solutions is expected to be approximately equal to the product of the number of executions and the known arithmetic mean. In addition, the arithmetic mean of the achieved solutions is expected to be approximately equal to the known arithmetic mean. This is usually not very interesting information since the user of the algorithm is typically not concerned with the sum of solutions or their arithmetic mean. If the algorithm is executed once, the user will want to know what kind of solution he might expect. Also, if the algorithm is executed many times, the user will be interested to know what kind of solution will be the best one obtained from these executions. The arithmetic mean could still be a useful indicator of algorithmic performance since one hopes that a better solution might be acquired by an algorithm that achieves a better mean solution.

Yet another disadvantage of using the arithmetic mean is that it is not applicable to the common and very useful practice of multiple executions of an algorithm and using the best solution obtained. In this case the arithmetic mean is also only a possible indicator of algorithmic performance.

There is a specific disadvantage of using the arithmetic mean that applies only to the case when the minimal solution quality is specified and the required computational resources are the measured value. Calculating the arithmetic mean of the required computational resources might not be possible owing to some experiments that might require more resources than the researcher is able to provide, e.g. too much execution time.

Finally, a stochastic optimization algorithm may come to stagnation and reach the state from which it is not possible to obtain the required solution quality regardless of the available time and other computational resources. Such cases, although essential for the correct estimation of the arithmetic mean, might be rare and hard to observe and might not occur in a gathered sample. Consequently, the arithmetic mean of the required time could be estimated wrongly to some definite value, when in fact it is infinite or undefined

B. Best Solution

The best solution obtained in multiple executions of an algorithm is fairly often used as a measure of performance. Eiben and Jelasity [3] determined that the information wanted by the user of the stochastic optimization algorithm is the best result and therefore recommended it as a measure of algorithmic performance. Birattari and Dorigo [7] correctly observed that the best achieved result cannot be reproduced by another researcher and can thus not be used as a measure of algorithmic performance. In our opinion, it is worth noting that if the best achieved solution for some problem instance is better than the best solution that is known from the literature (or when such a solution is not known), the best achieved solution should be reported. In that case, the best achieved solution represents valuable information about a problem instance rather than information about algorithmic performance.

Using best achieved solutions (out of predefined number of attempts) inside statistical tests that can handle individual values might be acceptable. It is not relevant if in a particular experiment one algorithm yields a better "best solution" than

the other. Instead, it is important that this behavior is consistent in many experiments and confirmed as statistically significant.

C. Quantiles

The use of the median is becoming more common owing to the increased awareness that the distribution of solution quality is often asymmetrical and that the median makes for a more robust measure of central tendency in such cases, as suggested by statistical textbooks. Here we argue that there are other important advantages of using the median or, more generally, of using quantiles in measuring the performance of stochastic optimization algorithms. The quantile Q_p is associated with probability p. For some random variable X from the population, it holds by definition that $P(X \leq Q_p) \geq p$ and $P(X \geq Q_p) \geq 1 - p$. The median is a special case, where $p = 0.5$ so there is a 0.5 probability that a random variable X from the population is greater than or equal to Q_p and at the same time there is a 0.5 probability that the random variable X is lesser than or equal to Q_p.

An important advantage of using quantiles as a measure of performance for stochastic optimization algorithms is their interpretation. If quantile Q_p of the solution quality for some algorithm is known, then the probability of achieving a solution that has equal or better quality than Q_p is greater than or equal to p. For example, if $Q_{0.7} = 150$ then it is known that there is at least a 0.7 probability of getting a solution with the quality equal to 150 or better. It is worth noting that the probability is at least p and not exactly p because in some cases the probability of obtaining a solution of a certain quality can be larger than p. This occurs when for some $p1 < p2$ it is also true that the corresponding quantiles are equal $Q_{p1} = Q_{p2}$. In that case the probability of finding a solution of the quality that is at least Q_{p1} is at least $p2$. The most extreme case is when the algorithm always finds solutions of equal quality since in that case the probability of finding any quantile Q_p is equal to 1.

In the case when a minimal solution quality is specified and quantile Q_p of the required computational time is properly measured, it is known that at least with probability p the algorithm will find a solution of the desired quality using at most Q_p time. For example, if $Q_{0.8} = 200$ s, then it is known that there is at least a 0.8 probability of getting a solution with the desired quality after at most 200 seconds.

When using quantiles one inconvenience may arise that can be easily avoided. For certain pairs of probabilities and sample sizes, e.g. median and even sample sizes, quantiles can be calculated slightly differently depending on interpolation or the rounding rule that is applied. Although the method that was used for calculating the quantiles can be explicitly stated with the results or a default method for the field of stochastic optimization algorithms might be agreed, the best way to overcome this issue is to use sample sizes in which such issues do not arise in the first place. For example, this issue does not occur when calculating quantiles $Q_{0.1}, Q_{0.2}, Q_{0.3}, \dots, Q_{0.9}$ if the sample size is set at $10k + 1$, where k is an arbitrarily selected natural number. It is also sufficient that the sample size is set at $2k + 1$ when calculating the median, or $4k + 1$ when calculating quartiles ($Q_{0.25}$ and $Q_{0.75}$), or $5k + 1$ when calculating $Q_{0.2}$. In all these cases the values of k should be

suitably large to achieve a good estimation of the true quantile value.

Another advantage of quantiles over the arithmetic mean is manifested when, although the algorithm fails to find a solution of a specified quality with probability r in reasonable time, it is still possible to measure Q_p, if $p < 1 - r$.

Besides being very informative and having other good properties quantiles are applicable to the practice of multiple executions of stochastic algorithms, which is explained in Section III of this paper.

D. Mode

The mode is usually not reported in the results of experiments with stochastic optimization algorithms. Solutions whose quality equals the mode have the highest probability of being produced by an execution of the algorithm, provided that the distribution is discrete and unimodal. Owing to this, the mode may seem as a good measure of performance in some cases. However, since the probability of gaining a solution of the quality that equals the mode is usually very small, the mode is clearly not a suitable measure of the performance of a stochastic optimization algorithm.

III. MULTIPLE EXECUTIONS OF STOCHASTIC OPTIMIZATION ALGORITHM

A big advantage of using quantiles is their applicability to multiple executions of the same algorithm for a particular problem instance. In the common approach, when computational resources are restricted and the algorithm is independently executed in a sequential or parallel manner, the best achieved solution is used as the final result. If the algorithm with restricted computational resources achieves quantile Q_p of solution quality, then for n independent executions there is probability $1 - (1 - p)^n$ of getting at least once the solution of the quality at least as good as Q_p.

For convenience, the sample of probabilities for multiple executions of the algorithm is provided in Table I. For example, if $Q_{0.5} = 1000$ and the algorithm is executed 4 times independently over a particular problem instance, there is at least a 0.9375 probability that the best solution found will have a quality equal to 1000 or better.

The other possible approach is to specify the solution quality and then execute multiple instances of the algorithm in parallel until a solution of the required quality is found by at least one algorithm. In this approach, if the quantile of time Q_p required for finding a solution of the specified quality is known, then it is expected that for n parallel executions of the algorithm there is probability $1 - (1 - p)^n$ of finding a suitable solution in at most Q_p time.

For example, if the algorithm needs 1000 seconds to obtain a specified solution quality with a 0.75 probability, i.e. $Q_{0.75} = 1000$ s, then there is at least a 0.9961 probability that the required solution quality will be obtained within 1000 seconds at least in one of four parallel executions of the algorithm.

IV. ADDITIONAL STATISTICAL INFORMATION

Sometimes it is appropriate to provide additional

information along with the measure of performance. When the arithmetic mean is reported, the additional information provided is usually the standard deviation. In the case of the median, interquartile or q-quantile range may be provided. Although such practice is widely recommended in many fields, we argue that this is not the best choice in reporting the performance of a stochastic optimization algorithm. When measuring some natural phenomena, one is usually concerned with precision and accuracy. The measured value is often deterministic and dispersion is caused by imperfections in the measuring process or by some neglected or unknown phenomena. A smaller dispersion in this case is obviously preferred.

When the performance of stochastic optimization algorithms is concerned, the spread of data, i.e. distribution of values, is not caused by imperfections in measurement or by some neglected details. Obtaining different solutions for a particular problem instance on different executions is inherent to stochastic optimization algorithms.

A symmetrical measure of dispersion like the interquartile range is not very informative in this case, since the decision whether it is better to have a larger or a smaller symmetrical dispersion is generally not straightforward. With some fixed quantile value, it is preferable to have a larger dispersion on the side that is closer to the optimum and a lesser dispersion on the side that is at the opposite side of the optimum. In the case of such dispersion, and if the actual solution is not as good as the specified quantile, it is more probable that this solution will be closer to the chosen quantile than in the case in which the dispersion is reversed.

TABLE I: PROBABILITIES FOR MULTIPLE EXECUTIONS OF THE STOCHASTIC OPTIMIZATION ALGORITHM

n	$Q_{0.1}$	$Q_{0.2}$	$Q_{0.25}$	$Q_{0.5}$	$Q_{0.75}$	$Q_{0.8}$	$Q_{0.9}$
1	0.1000	0.2000	0.2500	0.5000	0.7500	0.8000	0.9000
2	0.1900	0.3600	0.4375	0.7500	0.9375	0.9600	0.9900
3	0.2710	0.4880	0.5781	0.8750	0.9844	0.9920	0.9990
4	0.3439	0.5904	0.6836	0.9375	0.9961	0.9984	0.9999
5	0.4095	0.6723	0.7627	0.9688	0.9990	0.9997	1.0000
10	0.6513	0.8926	0.9437	0.9990	1.0000	1.0000	1.0000
20	0.8784	0.9885	0.9968	1.0000	1.0000	1.0000	1.0000
30	0.9576	0.9988	0.9998	1.0000	1.0000	1.0000	1.0000
40	0.9852	0.9999	1.0000	1.0000	1.0000	1.0000	1.0000
50	0.9948	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

Similarly, if the achieved solution is better than the specified quantile, it is more probable that it will be closer to the optimum. This situation is further clarified by Fig. 1, where all the algorithms have an equal median value for solution quality, and algorithms C and D also have an equal interquartile range. Presuming that the algorithms were applied to a minimization problem and based on the presented information, it follows that algorithm D has better properties than algorithms A, B and C. This is because algorithm D has quantiles $Q_{0.25}$ and $Q_{0.75}$ closer to the optimum than the other algorithms. Also, algorithm A has $Q_{0.25}$ closer to the optimum than algorithm C, and algorithm B has $Q_{0.75}$ closer to the optimum than algorithm C. Therefore, both algorithms A and

B have better properties than algorithm C.

Deciding whether algorithm A or algorithm B is better depends on personal preference in the case of a single execution of the algorithm, but this dilemma can be eliminated in the case of multiple executions of the algorithm. Namely, if both algorithms A and B are executed many times so the probability of getting a solution that is better than the median is very high, then it might be better to choose algorithm A over algorithm B since there is a higher probability of achieving a solution that is closer to the optimum.

When it is suitable to provide additional information we would recommend providing more than one quantile, e.g. $Q_{0.5}$ and $Q_{0.2}$, instead of the quantile and symmetrical measures of dispersion.

Naturally, this does not mean that statistical errors should not be controlled. The quantile estimated from the gathered sample might not perfectly match the actual quantile. It is therefore advisable to control and provide some information about estimation errors like the standard error of a quantile estimate, e.g. by using the bootstrap technique [12].

V. DISCUSSION

Currently, we are not aware of any good reason why quantiles should not become the standard measure of performance of stochastic optimization algorithms. Drawing on the numerous arguments in favor of quantiles that were presented in this paper we would highly recommend using quantiles for that purpose. Currently, the arithmetic mean is a well-established and widely known measure and thus some researchers might continue to use it rather than quantiles. In the opinion of the authors of this paper, such reasoning does not seem scientifically acceptable.
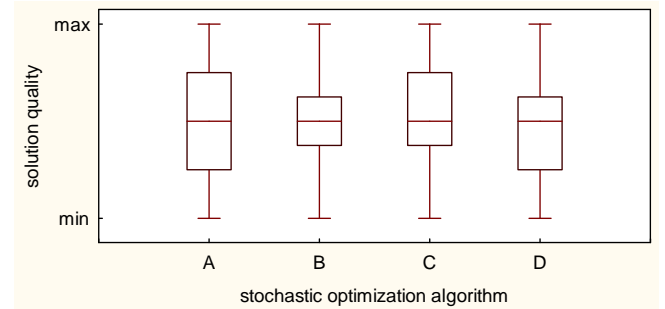

Fig. 1. Example of a boxplot of stochastic optimization algorithms that have an equal median value but different quartiles.

Some of the quantiles that we propose as standard choices for presenting results are $Q_{0.50}$, $Q_{0.20}$, $Q_{0.10}$, and $Q_{0.90}$, although other good choices are also possible. It is easy to choose sample sizes for these quantiles to avoid the inconvenience during the calculation of the sample's quantile. Increasing the sample size to control the estimation error in these cases can be gradually performed. As a counterexample, if $Q_{0.51}$ is chosen instead of $Q_{0.50}$, then the smallest sample size for mitigating the inconvenience in the calculation is 101. Even if this size is not large enough to bound the estimation error to an acceptable magnitude, then the next convenient sample size is 201, which can be considered as a large increase in the

sample size.

In the case of a symmetrical distribution, which however should not be regularly expected in stochastic optimization, $Q_{0.50}$ should match the arithmetic mean. Quantile $Q_{0.50}$ has a special role since it is also a well-known measure of central tendency. Choosing $Q_{0.50}$ can ensure the high probability of finding a solution of the specified quality after 4 or 5 repetitions of the algorithm.

Finding a solution with the quality $Q_{0.90}$ and a high probability can be obtained by only one execution of the algorithm, while multiple executions can ensure that this kind of solution is found with an almost absolute certainty. When it is feasible to use ten or a few tens of executions, then $Q_{0.10}$ and $Q_{0.20}$ might be adequate. Small quantiles like $Q_{0.10}$ or even smaller can capture the peak performance of the algorithm, for which the best found solution is not adequate because it is irreproducible, but require larger sample sizes. A good estimation of the quantile depends on the chosen probability and the sample size in a way that furthering from $p = 0.5$ towards $p = 0$ or towards $p = 1$ requires an increase in the sample size.

The same quantiles can be used when the solution quality is specified and computational resources are measured, and similar arguments apply as in the case when resources are constrained and the solution quality is observed.

In the case when only the solutions of a specified quality are regarded as useful and computational resources are strictly tied, the success rate is well suited. In most other cases this might not be an appropriate measure of performance because it does not preserve the information on how close the obtained solutions are to the specified quality

VI. Conclusion

In this paper, we have shown that quantiles demonstrate very good properties and are a natural measure of performance of stochastic optimization algorithms. At the same time, quantiles make it possible to specify the solution quality and the probability of achieving that kind of solution or, alternatively, to state the required time and the probability of achieving a solution of some preselected quality. Quantiles are useful in the case of multiple executions of an algorithm, which is a frequent and beneficial practice. They enable a practitioner to make an informed decision when choosing a trade-off between computational resources and the probability of achieving desired solutions. Quantiles are also defined in cases when the arithmetic mean is not, or when the arithmetic mean is hard to estimate. Taking into consideration numerous advantages of quantiles over the arithmetic mean, we propose quantiles as the standard measure of performance of stochastic optimization algorithms. In addition, in our study it was established that a greater variability of results in stochastic optimization is not necessarily a bad characteristic of algorithms, as it would be in many other domains, but may actually prove to be an advantage in certain cases.

References

[1] R. Barr, B. Golden, J. Kelly, M. Resende, and W. Stewart, "Designing and reporting on computational experiments with heuristic methods," *Journal of Heuristics*, vol. 1, no. 1, pp. 9–32, September 1995.

[2] T. Bartz-Beielstein, M. Chiarandini, L. Paquete, and M. Preuss, *Experimental Methods for the Analysis of Optimization Algorithms*. Berlin Heidelberg: Springer-Verlag, 2010.

[3] A. E. Eiben and M. Jelasity, "A critical note on experimental research methodology in ec," in *Proc. 2002 Congress on Evolutionary Computation*, 2002, vol. 1. IEEE Press, Piscataway, NJ, pp. 582–587.

[4] P. J. Fleming and J. J. Wallace, "How not to lie with statistics: the correct way to summarize benchmark results," *Communications of the ACM*, vol. 29, no. 3, pp. 218–221, March 1986.

[5] J. N. Hooker, "Testing heuristics: We have it all wrong," *Journal of Heuristics*, vol. 1, no. 1, pp. 33–42, September 1995.

[6] R. L. Rardin and R. Uzsoy, "Experimental evaluation of heuristic optimization algorithms: A tutorial," *Journal of Heuristics*, vol. 7, no. 3, pp. 261–304, May 2001.

[7] M. Birattari and M. Dorigo, "How to assess and report the performance of a stochastic algorithm on a benchmark problem: Mean or best result on a number of runs?" *Optimization Letters*, vol. 1, pp. 309–311, 2007.

[8] J. Derrac, S. Garcia, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm and Evolutionary Computation*, vol. 1, pp. 3–18, March 2011.

[9] D. Shilane, J. Martikainen, S. Dudoit, and S. J. Ovaska, "A general framework for statistical performance comparison of evolutionary computation algorithms," *Information Sciences*, vol. 178, no. 14, pp. 2870–2879, July 2008.

[10] M. Chiarandini, D. Basso, and T. Stützle, "Statistical methods for the comparison of stochastic optimizers," *in Proc. the 6th Metaheuristics International Conference*, Vienna, Austria, 2005, pp. 189–196.

[11] N. Ivkovic, "Modeling, analysis and improvement of ant colony optimization algorithms (in croatian)," Ph.D. dissertation, Faculty of Electrical Engineering and Computing, University of Zagreb, Zagreb, Croatia, 2014.

[12] R. R. Wilcox, *Introduction to Robust Estimation and Hypothesis Testing*, 2nd ed. Academic Press, 2004.

Nikola Ivkovic received the MS degree in computer engineering and the Ph.D. degree in computer science all from the Faculty of Electrical Engineering and Computing, University of Zagreb, in Zagreb, Croatia. His PhD thesis was in the area of swarm intelligence and evolutionary computation.

From 2007, he is a member of the research and teaching staff at the Department of Information Technologies and Computing of Faculty of Organization and Informatics, University of Zagreb. His research interests include computational intelligence and optimization, especially ant colony optimization, and also parallel programming, formal methods, and computer networks.

Domagoj Jakobovic received BS degree in December 1996. and MS degree in December 2001 in electrical engineering. Since April 1997. He is a member of the research and teaching staff at the Department of Electronics, Microelectronics, Computer and Intelligent Systems of Faculty of Electrical Engineering and Computing, University of Zagreb. He received the PhD degree in December 2005 on the subject of generating scheduling heuristics with genetic programming.

Marin Golub received the BS degree in 1992 in electrical engineering, MS degree in 1996 and Ph.D. degree in 2001 in computer science, all at the Faculty of Electrical Engineering and Computing, University of Zagreb. Currently he is working as an associate professor at the Department of Electronics, Microelectronics, Computer and Intelligent Systems, Faculty of Electrical Engineering and Computing, University of Zagreb. His interests include parallel algorithms, operating systems, evolutionary algorithms and computer system security.