Comparison of decisions quality of heuristic methods with sequential formation of the decision in the graph shortest path problem

Eduard Vatutin
Southwest State University, Kursk, Russia
evatutin@rambler.ru

Abstract

The article deals with the problem of analyze of effectiveness of the heuristic methods with sequential formation of the decision in the test problem of getting shortest path in graph. The article brief describes the sequential heuristic methods used to solve the problem. The methodic of experimental comparison for estimation the quality of solutions based on the performing of computational experiments using the BOINC platform is considered. It also shows description of obtained experimental results which allows to identify the areas of the preferable use of selected subset of heuristic methods depending on the size of the problem and power of constraints.

1 Introduction

There is a wide class of optimizing problems with restrictions on discrete values of solution elements (arguments of the objective function) [Ros00, VTE16]. These include many problems from combinatorics, graph theory, operations research. Mathematical apparatus of derivatives and gradients is not applicable for them, but it is successfully used in solving problems of continuous optimization [Kar14]. To solve some of them, exact methods with polynomial time asymptotic (for example, methods of Prim and Kruskal [Pri57, Kru56] for getting minimal spanning tree, Kuhn–Munkres method for assignment problem solving [Kuh56, Mun57], etc.) are well known, while other problems belonging to the NP class don't have similar methods. To solve them, in practice, heuristic methods are used successfully. They provide decisions of good quality for reasonable computing time.

Known methods for solving discrete optimization problems can be divided into the following classes:

- limited-search methods (Brute Force, branch and bound method [LD60], modifications of the Brute Force with limited number of branches or limited depth of the analyzed subtree within the combinatorial search tree, SAT-based methods [CD06, ZKS16]);
- methods with sequential formation of the decision (greedy methods, random and weighted random methods, ant colony optimization method [VDMT14, Dor92, VT14]);
- methods based on the modification of the current decision or set of decisions with using some rules that are specific to the problem being solved (random walks method, simulated annealing method, evolutionary (genetic) methods, bee colony method [GK14, KGV83, Kar83, PGKORZ05]);

In: E. Ivashko, A. Rumyantsev (eds.): Proceedings of the Third International Conference BOINC:FAST 2017, Petrozavodsk, Russia, August 28 - September 01, 2017, published at http://ceur-ws.org

- methods based on the movement in arguments space with subsequent mapping of the current agent position to one of the decisions of discrete optimization problem (particle swarm optimization method, firefly method, fish school search method, gravitational search method, etc. [Kar14, GK14, KE95]).

Limited Brute Force methods cluster has exponential or factorial asymptotic time complexity in the common case, while other methods have polynomial asymptotic time complexity. By combining the elementary methods that are shown above, it is possible to construct hybrid multistep methods [Kar14].

2 Statement of the Problem

Different heuristic methods are characterized in practice both by the different costs of computer time, and by the different quality of the decisions obtained, in this view it is interesting to compare quality of decisions aimed to select the best of methods. For this purpose a test problem of getting shortest path $P(G) = \left[a_{i_1} = a_{beg}, a_{i_2}, ..., a_{i_Q} = a_{end}\right]$ in graph $G = \langle A, V \rangle$ was selected, where $A = \{a_1, a_2, ..., a_N\}$ is a set of vertices, $|A| = N$ is a number of vertices (size of a problem), $V = \{v_1, v_2, ..., v_M\} \subseteq A \times A$ is a set of arcs, $|V| = M$ is a number of arcs, $v_i = \left(a_{beg}^{(i)}, a_{end}^{(i)}\right)$, $a_{beg}^{(i)} \in A$, $a_{end}^{(i)} \in A$, and the arcs are weighted by the length value $L(v_i) > 0$, $i = \overline{1, M}$. Target function in the specified problem is a length of path $L(P) = \sum\limits_{j=1}^{R-1} L(a_{i_j}, a_{i_{j+1}}) \to \min$, density of graph $d(G) = \frac{M}{N(N-1)} \in [0{,}0; 1{,}0]$ has a role of restriction as in graphs with low density a large number of decisions are prohibited. Selected problem has exact optimal decision (*abbr.* O) that can be obtained polynomially using Dijkstra algorithm [Dij59] for a quadratic time, that allows to use it as test problem for estimating deviations of decisions quality from optimum for heuristic methods. In a number of combinatorial optimization problems it was shown [VVT15, VT15] that heuristic methods are characterized by significantly different behavior during solving problems with different size of a problem and different power of restrictions. For the purpose of careful analysis of this feature a computer experiment was organized. During this experiment it was formed a sample $\Lambda = \{G_1, G_2, ..., G_K\}$ of $K = 1000$ graphs with pseudorandom structure, given number of vertices N, density d and pseudo-random values of arcs lengths $0 < L(v_i) \leq 1$. For the obtained decisions (paths P with length $L(P)$) the following average parameters was calculated: average length of paths (*abbr.* AVG)

$$\overline{Q} = \frac{\sum\limits_{i=1}^{K} Q(G_i)\phi(G_i)}{K}, \tag{1}$$

where $Q(G_i) = L(P \in G_i)$ is the quality of the best decision (path with minimal length) for selected method, $\phi(G_i) \in \{0, 1\}$ is function that has 1 value if the decision was found with selected method and 0 otherwise; average deviation of quality of the best decision from optimum (*abbr.* DIFF)

$$\Delta\overline{Q} = \frac{\sum\limits_{i=1}^{K} \left(Q(G_i) - Q^*(G_i)\right)\phi(G_i)}{K}, \tag{2}$$

where $Q^*(G_i)$ is a quality of optimal decision (length of the shortest path) given by Dijkstra algorithm; probability of finding decision (*abbr.* PFD)

$$\overline{p} = \frac{\sum\limits_{i=1}^{K} \phi(G_i)}{K} \tag{3}$$

and probability of finding optimal decision (*abbr.* PFOD)

$$\overline{p}_{opt} = \frac{\sum\limits_{i=1}^{K} \theta(G_i)}{K} \tag{4}$$

where

$$\theta(G_i) = \begin{cases} 0, & Q(G_i) > Q^*(G_i), \\ 1, & Q(G_i) = Q^*(G_i). \end{cases} \tag{5}$$

To compare methods for selected values of vertices number N and density of graph d it is required from several minutes to several hours of computational time. To analyze the behavior of methods under different conditions of use there was organized a computing experiment based on distributed computing project Gerasim@Home[1] on BOINC platform [And04]. Within this experiment we studied the behavior of methods for $2 \leq N \leq 500$ and $0 < d \leq 0{,}2$ with steps $\Delta N = 1$ and $\Delta d = 0{,}001$, obtained results are presented in this article. For each set of source data for iteration methods there was organized formation of $C_{max} = 1000$ decisions with selection the best of them.

3 Brief Classification of Heuristic Methods with Sequential Formation of the Decision

We give a brief description of the heuristic methods (a detailed description is given in the works [VTE16, VDMT14, VT14]). As it has already been shown above, for optimal decisions with quality $Q^*(G_i)$ Dijkstra algorithm was used. With using a greedy method (*abbr.* G) from the current vertex a_{curr} as a direction of movement such a vertex a_i is chosen that $L(a_{curr}, a_i) \neq \infty$ and $i = \arg\min_j L(a_{curr}, a_j)$. With using random search method (*abbr.* RS) as a direction of movement random vertex a_i is chosen that $L(a_{curr}, a_i) \neq \infty$, and lengths of arcs $L(a_{curr}, a_i)$ do not effect the choice of direction. With using weighted random search method (*abbr.* WRS) direction of movement is chosen as $i = \arg\min_j \; [L(a_{curr}, a_i)(1 + 2D(r_k - 0{,}5))]$, where r_k is a pseudo-random value with a uniform distribution in the range of $r_k \in [0{,}0; 1{,}0]$, $L(a_{curr}, a_i) \neq \infty$. The meaning of the used heuristic is in combination of greedy and random methods of decisions forming by using variable parameter D of scatter about the greedy heuristic $L(a_{curr}, a_i)$ (with $D = 0$ method is equal to greedy, and with $D \gg 1$ – to random search). In the ant colony method (*abbr.* AC) direction of movement is chosen as $i = \arg\min_j \eta_j^{\alpha} \tau_j^{\beta} r_k$, where $\eta_j = \frac{1}{L(a_{curr}, a_j)}$, τ_j is amount of pheromone at arc (a_{curr}, a_j). During initializing of method all arcs have τ_0 of pheromone, after moving of Z ants of colony each of them increase pheromone value to $\Delta\tau = \frac{Q}{L(P_l)}$ value in the arcs of path P_l, that was found by l-th ant of colony, after that pheromone is evaporated from all arcs: $\tau_{ij}^{(t+1)} = \tau_{ij}^{(t)}(1 - p)$, where t is a discrete time (iteration number). Schematically choosing of direction in combinatorial tree is shown in fig. 1.

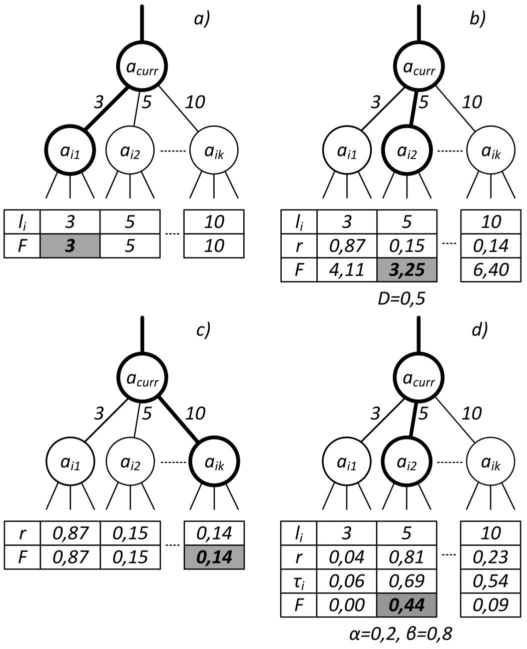

Figure 1: Schematic representation of the choice of the movement direction from the current vertex a_{curr} for greedy method (a), weighted random search method (b), random search method (c) and ant colony method (d).

[1] http://gerasim.boinc.ru

During solving problems with restrictions using heuristic methods with sequential formation of the decision the appearance of "dead ends" is possible where from the current vertex of the combinatorial tree it is impossible to go to any of the vertices of the underlying tier (in the problem of getting the shortest path in graph it is impossible to move to one of the vertexes which has not been visited yet). During increasing power of restrictions (in the problem of getting the shortest path in graph – during decreasing density of graph d) the probability of this situation in practice increases. For a greedy method this leads to the fact that if you get into the "dead end" the decision can't be found. For random and weighted random methods this leads to the appearance of iterations of work, where there is no correct decisions and, as a result, improvement of the record. With restriction on maximal number of iterations C_{max} for these methods it is possible that a decision exists, but the algorithm does not find it because all attempts have been completed in "dead ends". For ant colony method presence of "dead ends" leads to situation when hitting the "dead ends" the ant does not mark the path by the pheromone and the algorithm actually stops functioning, because there is no finding new decisions and updating the record.

In the article [VMT14] as one of the ways out of this situation, it was suggested to make a combinatorial return to one tier of a combinatorial search tree upwards and then to visit the next (in lexicographical order) vertex behind the "dead end" (fig. 2).

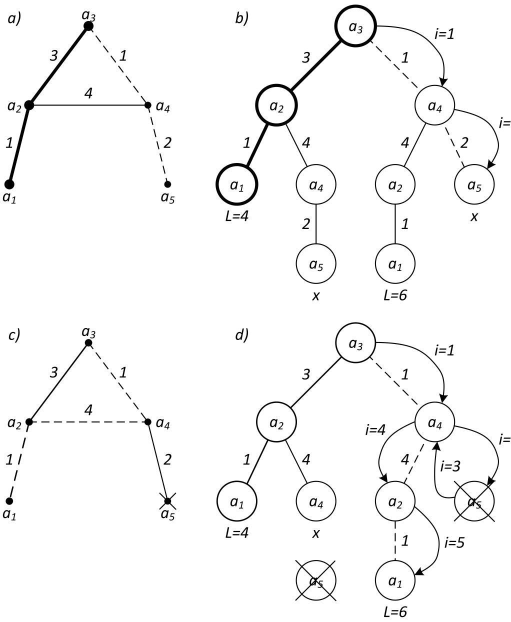

Figure 2: Illustration of the principle of combinatorial returns: a – source graph (the shortest path between vertexes a_3 and a_1 is bolded, dashed line corresponds to the greedy decision ended in the "dead end"); b – corresponding combinatorial tree; c – the greedy decision with return from disallowed "dead end" vertex a_5; d – order of movement in combinatorial tree with return from – "dead end" vertex.

Number R of combinatorial returns must be limited by a reasonable value (in organized experiments value $R = 1000$ was used), otherwise when $R \to \infty$ methods with sequential formation of the decision turns into the Brute Force when it is used in conditions of strong restrictions, that has negative influence on the time complexity. For selected value of R computing time costs was increased by no more than 1,5x times [VT116]. For all heuristic methods discussed above (G, RS, WRS and AC) the program implementations were developed both in basic version and with support of combinatorial returns principal (*abbr.* GR, RSR, WRSR and ACR correspondently).

Before using weighted random search and ant colony methods in practice it is required to select values of its tuning parameters (solve the problem of meta-optimization). For this purpose a number of computing experiments was organized where in correspondence with articles [VDMT14, VT14] the following values are selected: spread value D = 1,0, greed by path length α = 0,28, greed by pheromone β = 0,82, amount of pheromone at one act of colony Q = 1, initial pheromone amount $\tau^{(0)}$ = 100, pheromone evaporation speed p = 0,1, colony size Z = 100. In all experiments selected tuning parameters are constant.

4 Computing Experiments and Analysis of the Experimental Data

The computing experiment took 4 months in the grid with real performance about 2 TFLOP/s. Figure 3 shows the experimental results for optimal decisions provided by Dijkstra algorithm.

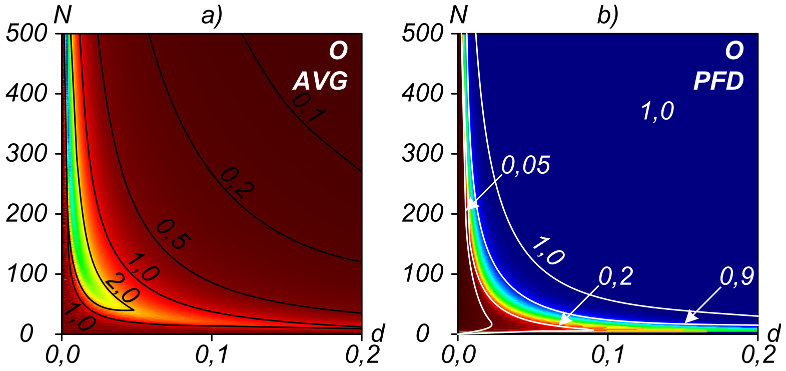

Figure 3: Average path length $\overline{Q}$ (a) and probability of finding decision $\overline{p}$ (b) depending on the number of vertexes N and graph density d (optimal decisions).

First of all it should be noted that the level lines in fig. 3a are very similar to hyperbolas in coordinates $(d; N)$. Minimal length of paths corresponds to the graphs with high density and big number of vertexes (top right corner in fig. 3). The greatest length of the paths is observed for graphs with low density with big number of vertexes (top left corner in fig. 3). Dijkstra's algorithm guarantees a decision in case of its existence (if the graph in which the path is found is connected), therefore the dependence in fig. 3b can be treated as probability of getting connected graph for selected values of N and d. During decreasing of graph d density for each N such value d' can be chosen that probability of finding decision is $\overline{p} \approx 1$, but $\overline{p} < 1$, and also value d'', that $\overline{p} \approx 0$, but $\overline{p} > 0$. For example, for $N = 100$ these values are $d' \approx 0{,}063$ and $d'' \approx 0{,}002$. If in the selected range of densities $d' \leq d \leq d''$ one of methods X provides the probability of obtaining a decision lower than the Dijkstra algorithm ($\overline{p}_X < \overline{p}_O$), then this indicates that for some test cases decisions exist but have not been found.

Figures 4 and 5 show dependencies for greedy method (G) and for its modification with combinatorial returns (GR). Their analysis allows us to draw a number of conclusions. First of all, the analysis of deviations of decisions quality from the optimum (fig. 4a and 5a) shows that greedy decisions are several times worse than optimal ones, that is a known fact for most problems. Analysis of dependencies of decision finding probabilities for the considered pair of methods shows that greedy method is a subject to falling into the ”dead ends” and its border d' is located more to the right than the border of Dijkstras algorithm. Adding to it the possibility of making combinatorial returns shifts this boundary to the left (fig. 5b, the boundary shift is indicated by an arrow), providing a significantly higher probability of finding decisions, although the quality of the decisions themselves still leaves much to be desired. Probability of finding the optimal decision for both the methods is very small: it is equal to $\overline{p} = 0{,}2$ for graphs with small amount of vertexes and high density, and significantly decreased with the growth of the problem size.

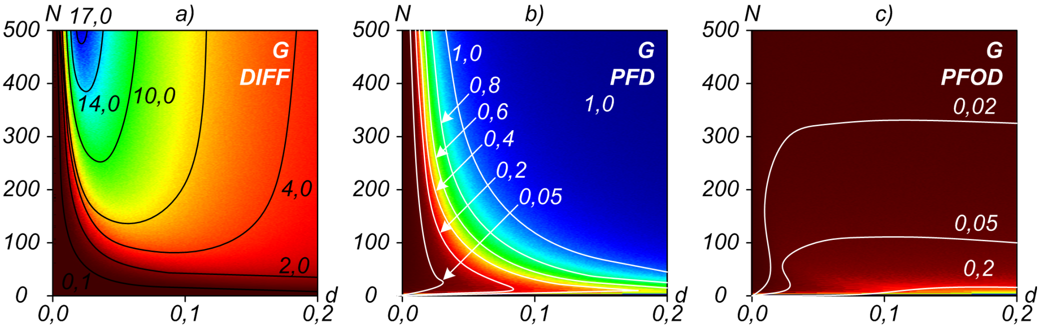

Figure 4: Dependencies of average deviation of quality of the best decision from optimum $\Delta\overline{Q}$ (a), probability of finding decision $\overline{p}$ (b) and probability of finding optimal decision $\overline{p}_{opt}$ (c) for greedy method.

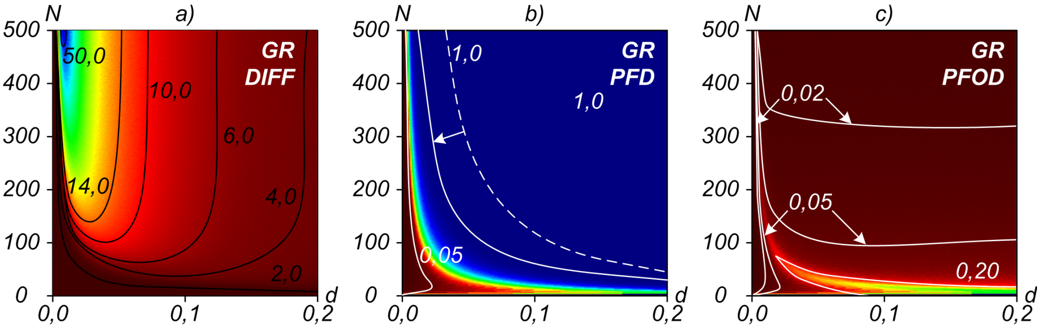

Figure 5: Dependencies of average deviation of quality of the best decision from optimum $\Delta\overline{Q}$ (a), probability of finding decision $\overline{p}$ (b) and probability of finding optimal decision $\overline{p}_{opt}$ (c) for greedy method with combinatorial returns.

In order to identify the advantages of the probability to obtain the optimal decision a comparison of values $\overline{p}_G$ and $(\overline{p}_{GR}$ for different values of N and d was made. As a result, regions were obtained where one of two methods has the advantage (fig. 6).

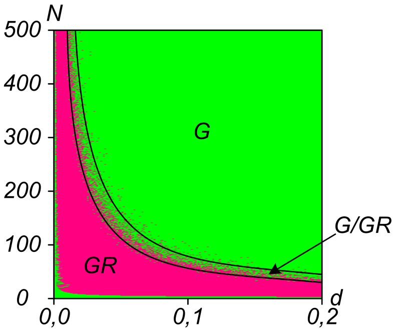

Figure 6: Comparison of methods G and GR decisions quality by the criterion of the probability of obtaining a better decision $\overline{p}_{opt}$.

Analysis of given areas shows that for graphs of high density $d > d'$ combinatorial returns are not required, but during density decreasing (or increasing power of restriction) they start to have a positive impact. Border between the areas of preferred usage of the considered pair of methods can be empirically approximated by a hyperbola $N = \frac{A}{d}$, where A is some constant, from where it is possible to obtain the following expression:

$$N = A\frac{N(N-1)}{M} \Rightarrow 1 = A\frac{N-1}{M} \Rightarrow M = A(N-1). \tag{6}$$

For the given pair of methods $A \approx 8$. Using this expression, it can be concluded that in case of $M > A(N-1)$ a classical greedy method is efficient, but in case of $M < A(N-1)$ its efficiency can be statistically significantly increased by implementing support for combinatorial returns.

Similar dependencies were obtained for methods RS, RSR, WRS, WRSR, AC and ACR (fig. 7 – 9). These methods demonstrate a substantially smaller deterioration in the quality of the decisions found with respect to the optimum, which particularly applies to weighted random search and ant colony methods as they have higher convergence rate [VT216]. Dependencies of the probability of finding decisions for basic implementations and implementations with support of combinatorial returns are qualitatively similar to the dependencies considered above for the pair of greedy methods: support of the combinatorial returns increases probability of finding decisions in the area $d' \leq d \leq d''$. The area of finding optimal decisions $\overline{p}_{opt} \to 1$ is located mainly in the lower right corner of the diagrams, that corresponds to graphs with small number of vertexes. Adding support of the combinatorial returns expands the indicated area both in the direction of graphs with a large number of vertices (to the top) and toward graphs with a lower density (to the left).

Results of pair wise comparison of methods with and without combinatorial returns by criterion of the maximum probability to find optimal decision are shown in fig. 10.

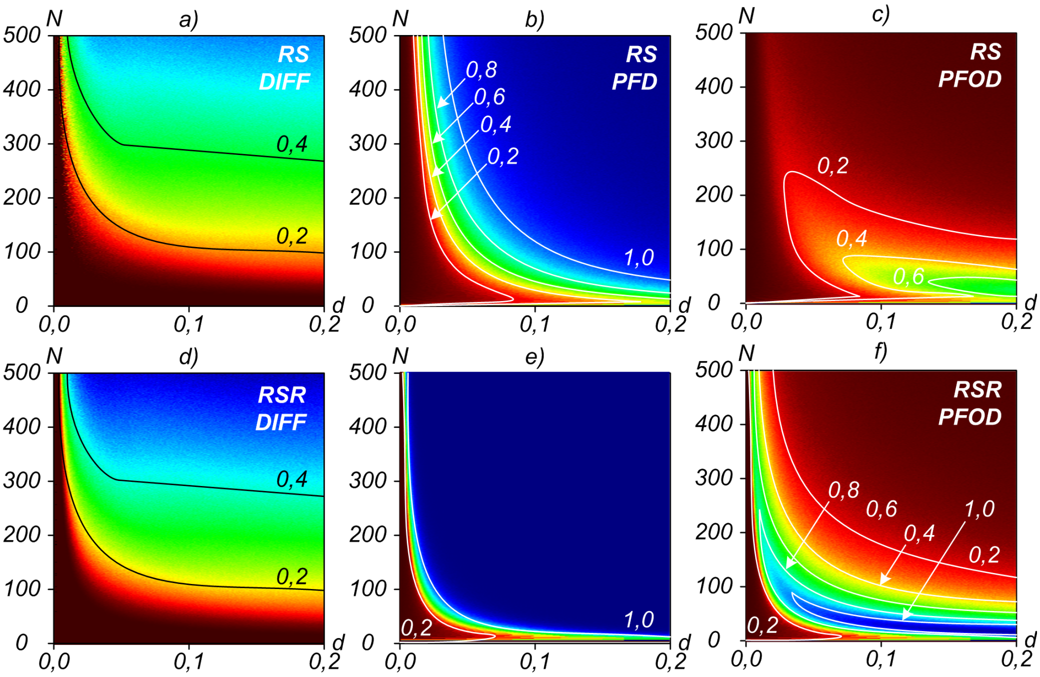

Figure 7: Dependencies of average deviation of the best decision quality from optimum $\Delta\overline{Q}$ (a, d), probability of finding decision $\overline{p}$ (b, e) and probability of finding optimal decision $\overline{p}_{opt}$ (c, f) for random search method and its modification with combinatorial returns support.

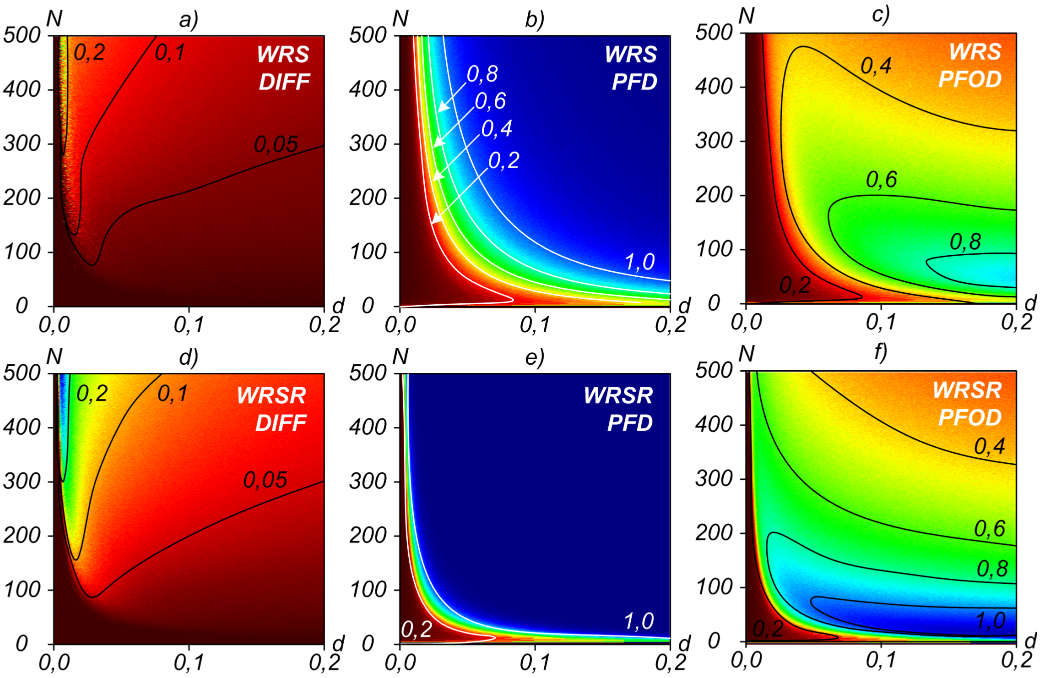

Figure 8: Dependencies of average deviation of the best decision quality from optimum $\Delta\overline{Q}$ (a, d), probability of finding decision $\overline{p}$ (b, e) and probability of finding optimal decision $\overline{p}_{opt}$ (c, f) for random search method and its modification with combinatorial returns support.

They are qualitatively similar to the results obtained above for a pair of algorithms G and GR and differ

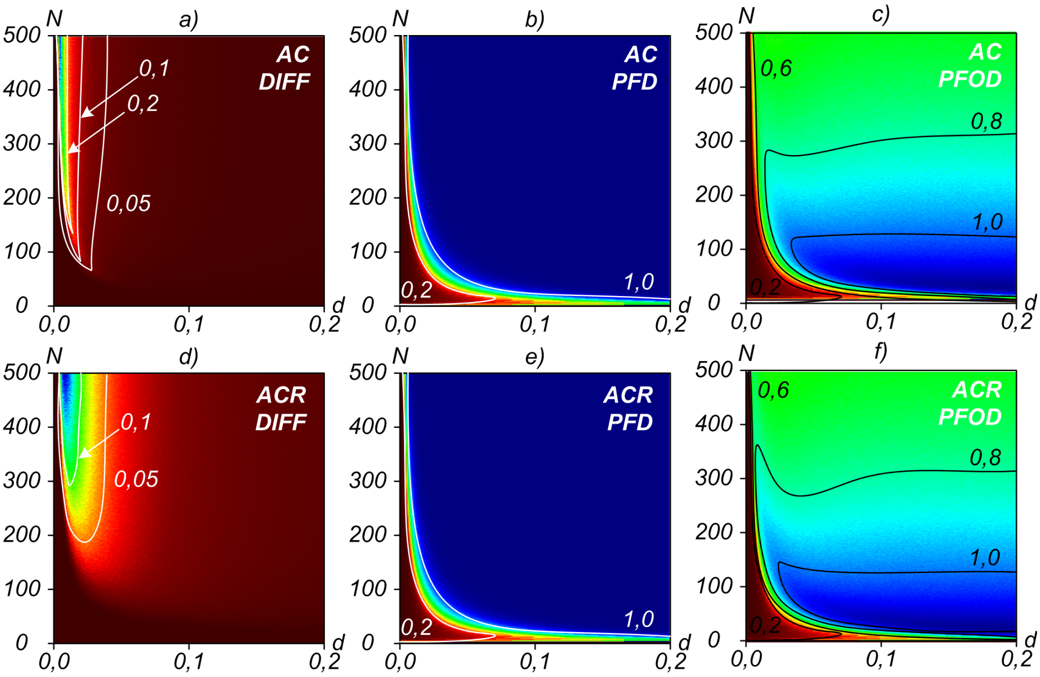

Figure 9: Dependencies of average deviation of the best decision quality from optimum $\Delta\overline{Q}$ (a, d), probability of finding decision $\overline{p}$ (b, e) and probability of finding optimal decision $\overline{p}_{opt}$ (c, f) for ant colony method and its modification with combinatorial returns support.

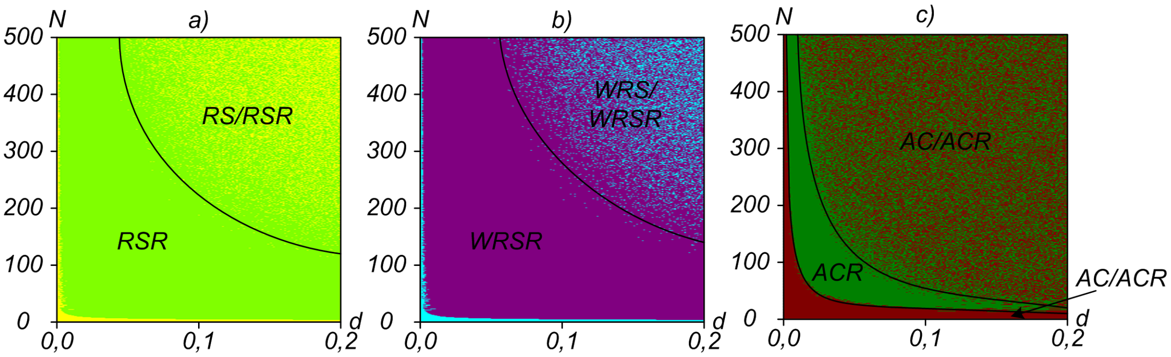

Figure 10: Pair wise comparison of methods RS and RSR (a), WRS and WRSR (b), and AC and ACR (c); $A_{RS} \approx 22, A_{WRS} \approx 26, A_{AC} \approx 4{,}5$.

from them by the location of the boundary between the areas of preferential use of methods (by values of constants A_X, $X \in \{RS, WRS, AC\}$). It should be noted that among the three algorithms without the support of combinatorial returns the ant colony method feels relatively comfortable in a larger range of graph densities (for them $A_{AC} \ll A_{RS}$ and $A_{AC} \ll A_{WRS}$). However, there is also an area of low-density graphs where support of combinatorial returns provides an advantage.

Fig. 11 shows comparison results of all above-mentioned heuristic methods by the criterion of the maximum probability of finding optimal decision $\overline{p}_{opt}$.

Their analysis allows us to conclude that for dense graphs the ant colony method is the best among considered ones. During density decreasing from value $d < d' \approx \frac{4{,}5}{N}$ the modified method of an ant colony with support of combinatorial returns turns out to be more effective. In a narrow area of very small graph density the algorithms AC, ACR and RSR demonstrate approximately equal efficiency.

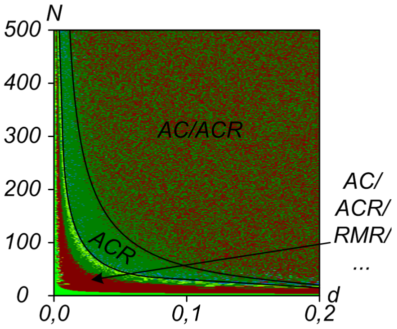

Figure 11: Comparison of decision quality of all above-mentioned heuristic methods by the criterion of the maximum probability of finding optimal decision $\overline{p}_{opt}$.

5 Prospects for the Future Investigations

The current series of experiments performed within Gerasim@Home project has several objectives and aimed to subset of heuristic methods with sequential formation of the decision only. In addition, it would be interesting to study the quality of solutions obtained using different iterative methods such as modifications of limited depth first search, group intelligence approaches [Kar83, PGKORZ05], modifications of simulated annealing [KGV83], random walks, particle swarm optimization [KE95] and so on. Also all of heuristic methods requires testing in different combinatorial optimization problems (logic multicontrollers design [Ros00], different graph theory problems, etc.). Fine tuning of variable parameters of methods during meta-optimization or adaptation is another one important problem that needs to be investigated.

6 Acknowledgements

The author would like to thank all volunteers who took part in the calculation within the distributed computing project Gerasim@Home. The author also wish to thank Anna Vayzbina for assistance in preparing the English version of the article. The author also wish to thank Sergey Valyaev for developing and supporting technical part of Gerasim@Home project. The article was partially supported by RFBR (grant 17-07-00317-a) and by council on grants of the President of the Russian Federation (grant MK-9445.2016.8).

References

[Ros00] K. H. Rosen et al. Handbook of discrete and combinatorial mathematics. CRC Press, New York, 2000.

[VTE16] E. I. Vatutin,V. S. Titov, S. G. Emelyanov. Basics of discrete combinatorial optimization (in Russian). Argamac-Media, Moscow, 2016.

[Kar14] A. P. Karpenko. Modern algorithms of search optimization. Algorithms inspired by nature (in Russian). Bauman Moscow State Technical University, Moscow, 2014.

[Pri57] R. C. Prim. Shortest connection networks and some generalizations. Bell System Technical Journal. Vol. 36 (6). pp. 1389–1401 (1957). DOI: 10.1002/j.1538-7305.1957.tb01515.x

[Kru56] J. B. Kruskal. On the shortest spanning subtree of a graph and the traveling salesman problem. Proceedings of the American Mathematical Society. Vol. 7. pp. 48–50 (1956). DOI: 10.1090/S0002-9939-1956-0078686-7

[Kuh56] H. W. Kuhn. Variants of the Hungarian method for assignment problems. Naval Research Logistics Quarterly. pp. 253–258 (1956). DOI: 10.1002/nav.3800030404

[Mun57] J. Munkres. Algorithms for the Assignment and Transportation Problems. Journal of the Society for Industrial and Applied Mathematics. No. 5, Vol. 1. pp. 3–38 (1957). DOI: 10.1137/0105003

[LD60] A. H. Land, A. G. Doig. An Automatic Method of Solving Discrete Programming Problems. Econometrica. Vol. 28. pp. 497–520 (1960). DOI: 10.2307/1910129

[CD06] C. J. Colbourn, J. H. Dinitz. Handbook of Combinatorial Designs, 2nd ed. (Discrete Mathematics and Its Applications). Chapman & Hall/CRC, New York, 2006.

[ZKS16] O. S. Zaikin, S. E. Kochemazov, A. A. Semenov. SAT-based search for systems of diagonal Latin squares in volunteer computing project SAT@home. 39th international convention on information and communication technology, electronics and microelectronics (MIPRO 2016). pp. 277–281 (2016). DOI: 10.1109/MIPRO.2016.7522152

[VDMT14] E. I. Vatutin, E. N. Dremov, I. A. Martynov, V. S. Titov. Weighted random search method for discrete combinatorial problems solving (in Russian). Proceedings of Volgograd State Technical University. Series: Electronics, Measuring Equipment, Radiotechnics and Communication. No. 10 (137). Iss. 9. pp. 59–64 (2014)

[Dor92] M. Dorigo. Optimization, Learning and Natural Algorithms. PhD thesis. Politecnico di Milano, Milan (1992)

[VT14] E. I. Vatutin, V. S. Titov. Analysis of results of ant colony optimization method in the problem of getting shortest path in graph with constraints (in Russian). Proceedings of South State University. Technical sciences. No. 12 (161). pp. 111–120 (2014)

[GK14] F. Glover, G. A. Kochenberger. Handbook of Metaheuristics. Kluwer Academic Publishers, New York, 2003.

[KGV83] S. Kirkpatrick, C. D. Gelatt, M. P. Vecchi. Optimization by Simulated Annealing. Science. Vol. 220. No. 4598. pp. 671–680 (1983). DOI: 10.1126/science.220.4598.671

[Kar83] D. D. Karaboga. An Idea Based On Honey Bee Swarm for Numerical Optimization. Technical Report-TR06, Erciyes University, Engineering Faculty, Computer Engineering Department (2005)

[PGKORZ05] D. T. Pham, A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim, M. Zaidi. The Bees Algorithm. Technical Note, Manufacturing Engineering Centre, Cardiff University (2005)

[KE95] J. Kennedy, R. Eberhart. Particle Swarm Optimization. Proceedings of IEEE International Conference on Neural Networks. pp. 1942–1948 (1995). DOI: 10.1109/ICNN.1995.488968

[Dij59] E. W. Dijkstra. A note on two problems in connexion with graphs. Numerische Mathematik. Vol. 1. pp. 269–271 (1959)

[VVT15] E. I. Vatutin, S. Yu. Valyaev, V. S. Titov. Comparison of Sequential Methods for Getting Separations of Parallel Logic Control Algorithms Using Volunteer Computing. CEUR Workshop Proceedings. Proceedings of the Second International Conference BOINC-based High Performance Computing: Fundamental Research and Development (BOINC:FAST 2015). Vol. 1502. pp. 37–51 (2015). urn:nbn:de:0074-1502-3

[VT15] E. I. Vatutin, V. S. Titov. Analysis of the areas of qualitative superiority of the sequential heuristic methods for getting separation during logic multicontrollers design (in Russian). Proceedings of the Higher Educational Institutions. Instrument Making. Vol. 58. No. 2. pp. 115–122 (2015). DOI: 10.17586/0021-3454-2015-58-2-115-122

[VMT14] E. I. Vatutin, I. A. Martynov, V. S. Titov. Method of workaround deadlocks for solving discrete combinatorial optimization problems with constraints (in Russian). Perspective information technologies. pp. 313–317. Samara scientific center of RAS, Samara (2014)

[VT116] E. I. Vatutin, V. S. Titov. Time costs analysis during getting decisions of heuristic methods at the shortest path problem (in Russian). Information-measuring and diagnosing control systems (Diagnostics – 2016). pp. 26–33. Southwest State University, Kursk (2016)

[VT216] E. I. Vatutin, V. S. Titov. Convergence rate analysis of quality of decisions of heuristic methods at the shortest path problem (in Russian). Information-measuring and diagnosing control systems (Diagnostics – 2016). pp. 19–25. Southwest State University, Kursk (2016)

[And04] D. P. Anderson BOINC: a system for public-resource computing and storage. Fifth IEEE/ACM International Workshop on Grid Computing, 2004. Proceedings (2004). DOI: 10.1109/GRID.2004.14