EFFICIENT DECOMPOSITION ALGORITHMS FOR SOLVING LARGE-SCALE TSP

Roman Bazylevych, Marek Pałasiński, Roman Kutelmakh, Bohdan Kuz

***Abstract**: The decomposition algorithms for solving specific Traveling Salesman Problems (TSPs) are presented. The test instances are based on the national geographic data and range in size from 9,847 cities in Japan to 115,475 cities in the USA. The proposed algorithms have a few stages: partitioning of the input set of points into small subsets, finding the partial high quality solutions, merging them into the whole initial solution, and optimization the final solution. Experimental results prove the efficiency of the proposed algorithms. Developed methods provide high quality solutions for large-scale TSP within close to the linear-logarithmic computational complexity.*

***Keywords**: TSP, large-scale, decomposition, algorithm, optimization, NP-hard.*

***ACM Classification Keywords**: G.2.1 Combinatorics - Combinatorial algorithms; I.2.8 Problem Solving, Control Methods, and Search - Heuristic methods.*

Introduction

The Travelling Salesman Problem (TSP) is extensively applied in transportation systems, automated design, testing and manufacturing of integrated circuits and printed circuit boards, X-ray crystallography and many other fields. The TSP is referred to the class of *NP*-hard combinatorial problems due to its factorial computational complexity, which unables obtaining exact solutions for large-scale problems within a reasonable runtime.

The TSP research began in the 50s of the previous century. In 1954 Dantzig, Fulkerson and Johnson defined the TSP as a discrete optimization problem and proposed a branch-and-bound method, which provides finding the optimal solution [Dantzig, 1954]. They solved an instance with 49 points and proved that no other route could be shorter. Flood [Flood, 1956] was one of the first scientists who introduced heuristic method for the problem. Lin and Kernighan devised one of the most efficient heuristic methods [Lin, 1973]. In 1972 Karp substantiated the *NP-completeness* of the problem [Karp, 1972]. The problem was also studied by many other researchers [Papadimitriou, 1977, Christofides, 1979, Reinelt, 1994, Johnson, 2002].

The studies by Applegate and others focused on finding the optimal solutions [Applegate, 1995, 1999, 2003, 2006, 2009]. They developed the "Concorde" software for providing exact solution to the problem. Recently Helsgaun has improved the classic version of the Lin-Kernighan method (LKH), which is considered as the best heuristic method so far [Helsgaun, 1998, 2006].

The Travelling Salesman Problem is formulated as follows: given is a set of points P, described by the their coordinates $P=\{p_1, p_2, \dots, p_N\}$, $p_i=(x_i, y_i)$ for $i \in \{1, 2, \dots, N\}$;

and metric $dist$: $P \times P \to R$ on the set P, for instance:

R_E: $dist_E(p_i, p_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ (euclidean metrics),

or R_O: $dist_O(p_i, p_j) = |x_i - x_j| + |y_i - y_j|$ (orthogonal metrics), $i, j \in \{1, 2, \dots, N\}$.

The problem consists in finding the closed route *M* (Hamiltonian cycle), which visits all points of the set *P* and has the minimum length: $len\ M \rightarrow \min$, where

$$len\ M = \sum_{i=1}^{N-1} dist(m_i, m_{i+1}) + dist(m_N, m_1), \tag{1}$$

is the function of route length $M = <m_1, m_2, \ldots, m_N>$, $\forall i,j\ [m_i, m_j \in P, m_i \neq m_j]$, $|M| = N$.

Function $f_{quality}$ is used to measure the quality of the solution:

$$f_{quality} = \frac{len\ M - len\ M^*}{len\ M^*}, \tag{2}$$

where M^* is the optimal solution.

When M^* is minimum length route: $len\ M \geq$ len M*, and $f_{quality}(M, M^*) \geq 0$.

Given $f_{quality}(M, M^*) = 0$ the route *M* (problem solution) is considered as an optimal, otherwise it is a suboptimal. The smaller function fquality (M, M*) value is, the closer to the optimal the problem solution is.

When the optimal problem solution (route M^*) is not given, the quality function $f'_{quality}$ for problem solution (route *M*) is applied, which for the given set of point *P* instead of optimal route length uses values of Held-Karp lower bound [Held, 1970, 1971, 1974]:

$$f'_{quality}(M, HKbound) = \frac{len\ M - HK_{bound}}{HK_{bound}}, \tag{3}$$

$$f'_{quality}(M, HKbound) > 0, \tag{4}$$

where *HKbound* is value of Held-Karp lower bound for the set of points *P*. The smaller the value of function $f'_{quality}$ is, the better the solution *M* is obtained.

When the optimal problem solution (route M^*) and the value of Held-Karp lower bound are not given, a compare function with the best known solution M_{known} found by the existing methods is used in order to estimate quality. If $f_{compare}(M, M_{known}) < 0$, the solution *M* is considered as the better comparatively with existing known one.

There are *N*! different alternative routes (solutions) via the given set of points *P*, and finding the optimal route can be either through the search of all possible options, that for the large-scale problems is impossible, either by branch and bound method, which also requires considerable computational cost.

Despite the fact that the general number of possible routes is finite, even advanced or future supercomputers are not able to conduct such search for many thousands or larger number of points. Therefore, many contemporary research works are focused on finding the suboptimal solutions for reasonable time, which are close to the optimal.

Decomposition and finding initial solution

The solving process involves the following main stages:

1) partitioning of the input set *P* into set $U = \{U_1, U_2, \ldots, U_K\}$ with *K* subsets: $P = U_1 \cup U_2 \cup \ldots \cup U_K$, $U_i \cap U_j = 0$, $D_{min} \leq |U_i| \leq D_{max}$, where D_{min} and D_{max} – respectively the minimum and maximum number of points in the subsets;
2) selection of the initial subset U_1 and finding its TSP solution (route M_1);
3) sequential extention of existing in *i*-step solution M_i by merging it with the partial solution ΔM_{i+1} for the adjacent subset U_{i+1} of points. New solution M_{i+1} is created;

4) continuation of the previous procedure until the inclusion of all points of set P into solution M_0 that is considered as an initial solution.

In order to extend for the (i+1)-step solutions M_i we consider the two subsets of points: Ui (all previous points) and additional U_{i+1}, that have overlapping $\Delta U_{i,i+1} = U_i \cap U_{i+1}$. The numbers of points in the set U_{i+1} and points in the set $\Delta U_{i,i+1}$ are the method's parameters which affects the quality of solution and running time. *Boundary entry* and *exit points* are defined for the set U_{i+1}. The rest part of the route M_i, which is not included in the set U_{i+1} is replaced by the fixed edges of the zero length.

With the help of the chosen method the TSP solution ΔM_{i+1} for the points within the set U_{i+1} is found. A new route M_{i+1} is formed by the route ΔM_{i+1} and segments of the route M_i, as a result of merging of the solutions in the subsets U_i and U_{i+1} (Fig. 1).

The procedure of the solutions' merging in the subsets continues till all subsets of the set U are united. The resulting route, covering all points of the set P, is viewed as the *initial solution* M_0 of the problem.

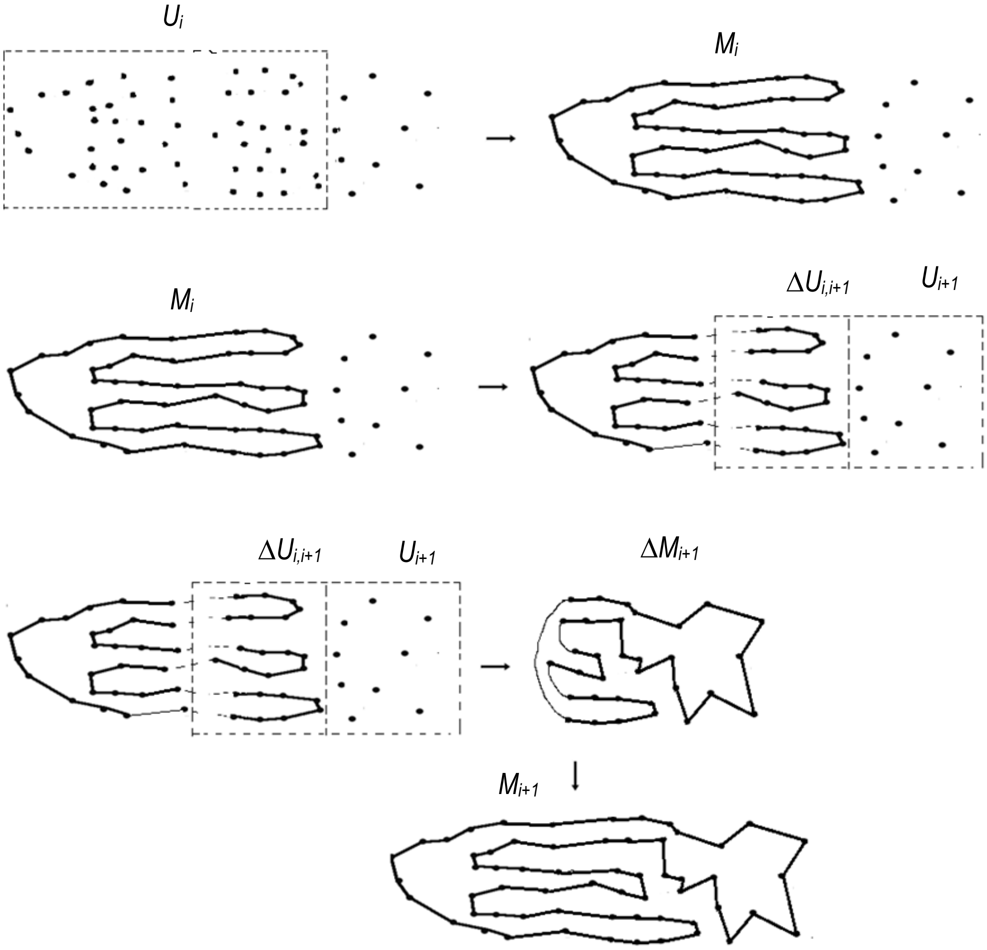

Fig. 1. Solution extension process

There are a number of algorithms of selecting the initial subset and subsequent subsets for solution extending. For example, from the left to the right merging of subsets, or alternatively, zigzag, spirally from some corner or center etc [Bazylevych, 2012].

Solution optimization

The results of experiments show that applying the extension method allows finding the initial route M_0 which on average 0,2-2% exceeds the length of the optimal one. It is required to use optimization methods to improve its quality. The reduction of the length of the route M_0 is provided through its iterative reduction in the certain Local Optimization Areas (LOA).

The method of optimization [Bazylevych, 2008, 2009] have the following features:

• size of the *optimization area* – the number of its points;

• size of the *overlapping area* – the number of points of the intersection area of two or more adjacent *optimization areas;*

• *strategy* (sequence, or direction) of optimization;

• *basic method* - the known method used for the TSP in a given LOA.

For the solution optimization the certain LOA is selected. In case of route length reduction, this area is replaced with a new one. The result is the route M_1, where $len\ M_1 < len\ M_0$. The process is repeated for all LOAs until all points of the route M_0 are reviewed.

The sequence of routes M_0, M_1, M_2, …, M_k, is obtained, where $len\ M_{i+1} < len\ M_i$ for $i \in \{0, 1, …, k\}$, and k is the number of area replacements on the route M_0 for shorter ones. Complete optimization process can be repeated several times until the length stops changing or the changes are insignificant.

The results of experiments prove that with the LOA size increase the quality of the solution improves, but computation time also increases. The quality also depends on the selected *basic method*. We recommend applying efficient *Lin-Kernighan* or *Lin-Kernighan-Helsgaun* methods.

Delaunay triangulation based optimization

This method is aimed to decrease the length of the route M_0 by sequential "scanning" the different LOAs along the initial route including also not only the points belonging to this route segment, but also points of other segments, which may be far away from this route segment, but close geometrically.

The initial route M_0 is divided into set of segments (LOAs) $S = \{S_1, S_2, …, S_r\}$, each of which has given number D of points (its size), and every two adjacent LOAs have the *overlapping* area which given number C of points (its size). The third parameter of the method is the *Depth* of the LOA (Fig. 2).

The set of points P is triangulated by the Delaunay algorithm [Guibas, 1985], obtaining the set of triangles $T=\{t_1, t_2, …, t_w\}$, $|T|=w$, $w \approx 3N$, every of which is described by their points $t_i=(p_{i1}, p_{i2}, p_{i3})$; $p_{i1}, p_{i2}, p_{i3} \in P$ for $i \in \{1, 2, …, w\}$. At the first step we choose an arbitrary point on the existed M_i road and spread around it the waves in triangles until the resulting region (LOA) includes the desired D number of points (dot line in the Fig. 3a). At the second step (Fig. 3b) we eliminate all pieces of existing road M_i and replace it external pieces (dashed line outside of LOA in the Fig. 3a) by fictitious pieces of zero length (continuous line beyond the LOA). At the third step (Fig. 3c) we solve the TSP in selected LOA. Finally, at the last step (Fig. 3d), the external fictitious pieces are replaced by the real ones (dashed line).

The replacement of the segments S_i ($i= 1,…, r$) continue until the optimization of all areas of the route M_0. As a result, the route M_1 is obtained, which is considered as *optimized*. The computational complexity of the optimization method is $O(N \log N + KD^{2,2})$, where K is the number of LOAs of optimization. Since the value D

is constant, K is the linear function of N and $K \ll N$, the computational complexity is $O(N \log N + K) \approx O(N \log N)$.

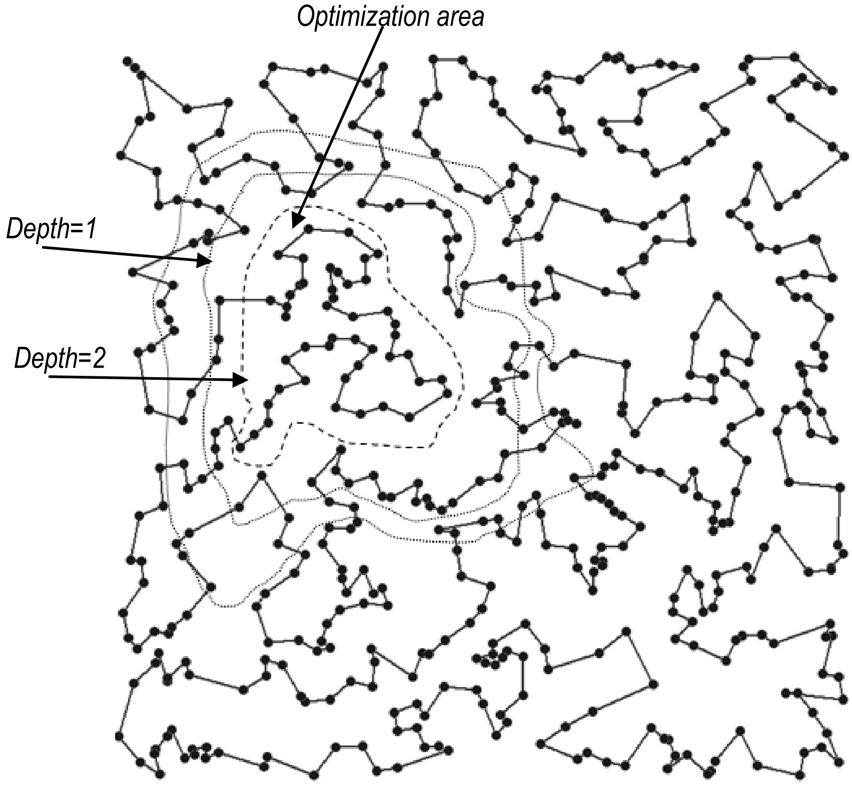

Fig. 2. Delaunay triangulation based optimization method and its features: size and depth

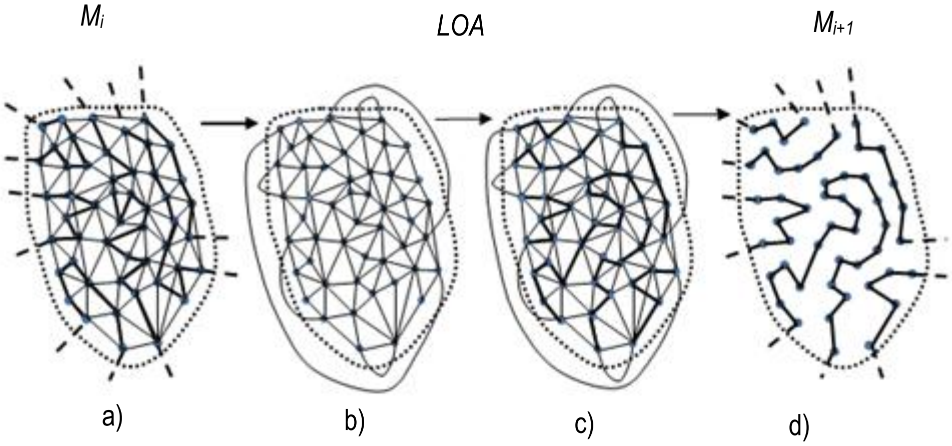

Fig. 3. Steps of replacement of route areas for shorter ones using optimization by route scanning method with Delaunay triangulation

Experimental results

Investigated how the parameters of optimization affect on quality of the solution optimization. The problem ch71009 was chosen for testing [National TSPs]. Initial route, 0,53% longer than the current best solution, was obtained by solution extending method [Bazylevych, 2012]. The following parameters were investigated: *size* of the LOA and *overlapping*. The size of the LOA varied from 100 to 2000 points, overlapping value varied from 10% to 80%. Test were performed on 3,5 GHz CPU. Table 1 provides the quality and running time of solution optimization.

The set of test instances, based on the national geographic data were chosen [National TSPs]. They vary in size from 9,847 cities in Japan to 115,475 cities in the USA (Fig. 4). The solving algorithm had a few stages: partitioning of the input set of points into small subsets, finding the initial solution, and optimization phase. Table 2 provides the results of finding solutions for test instances using the proposed decomposition and optimization algorithms.

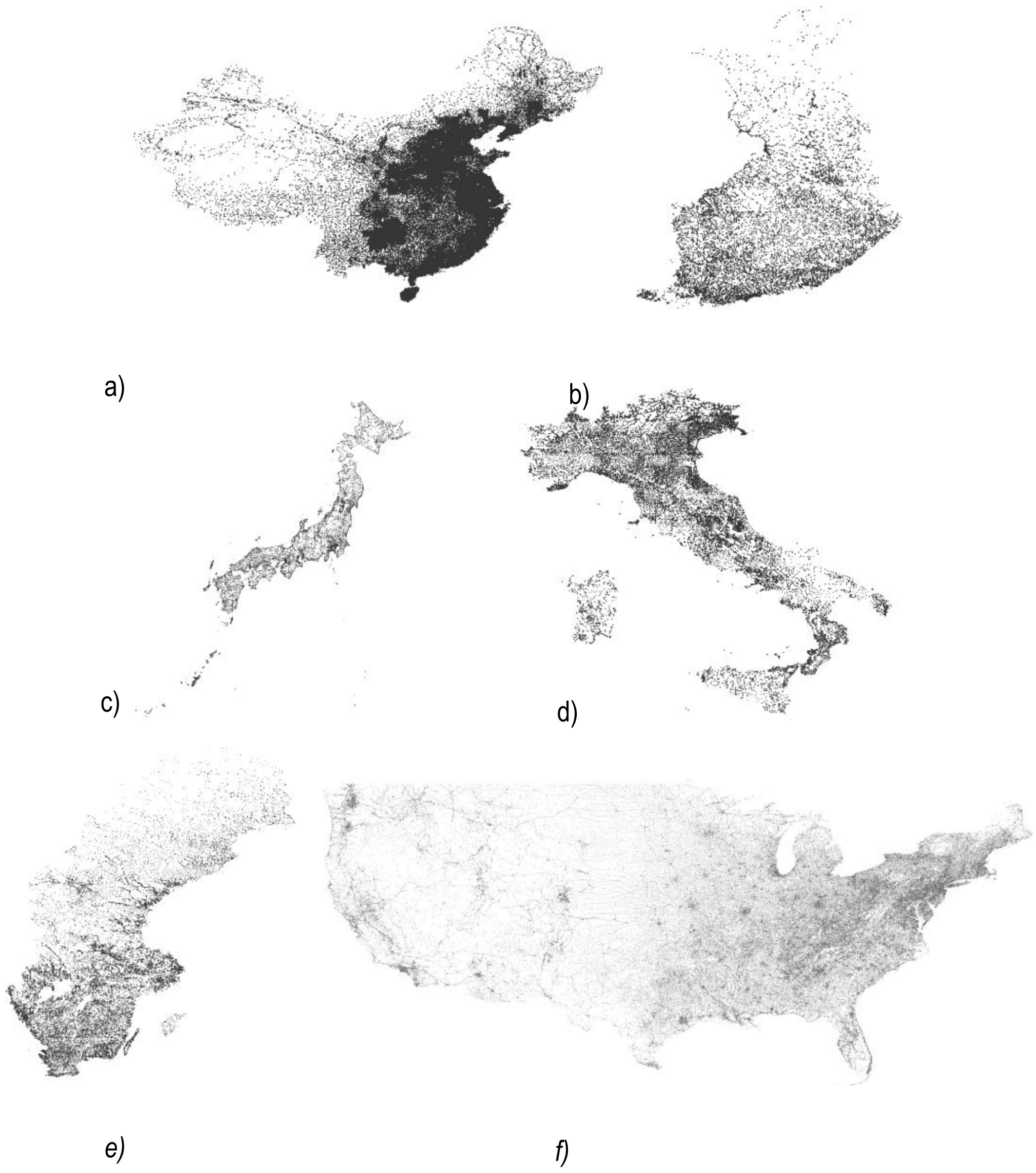

Fig. 4. Test instances: a) China (71009 points), b) Finland (10639), c) Japan (9847),
d) Italy (16862), e) Sweden (24978), f) the USA (115475)

The following pictures show the comparison in some areas of the route between the initial route and the optimized one. In some cases, the optimized route has no “inefficient” areas with long edges (Fig. 5). Also there are some changes in “global” route (Fig. 6). Figure 7 also shows some changes of the route segments.

Table 1. Quality (in %) and runtime (in seconds) of the optimized solutions (ch71009)

Size of LOA / Overlapping	100	500	1000	1500
10%	0,36% 3,3s	0,27% 4,7s	0,24% 6,7s	0,17% 8,4s
30%	0,29% 4,5s	0,30% 6,2s	0,19% 7,6s	0,13% 9,9s
40%	0,27% 4,9s	0,26% 6,9s	0,14% 10,3s	0,13% 11,8s
80%	0,26% 13,8s	0,19% 19,8s	0,08% 26,4s	0,10% 31,5s

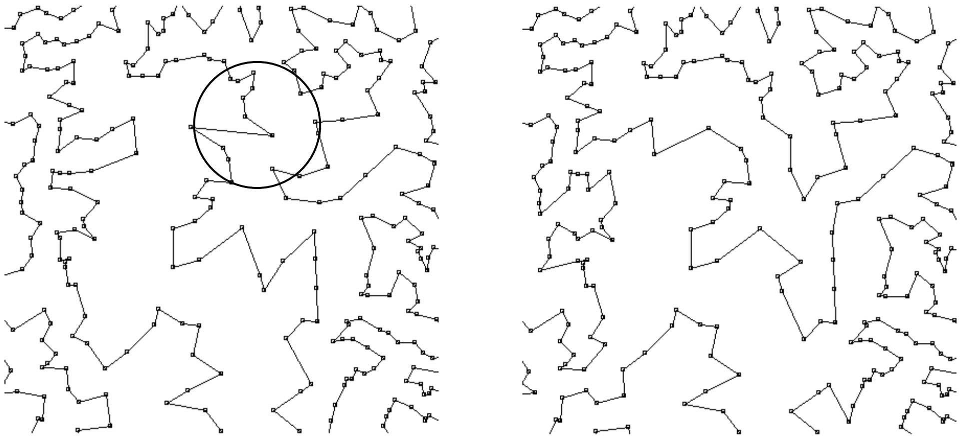

Fig. 5. Comparison of some segments of initial route and the optimized route (long edge)

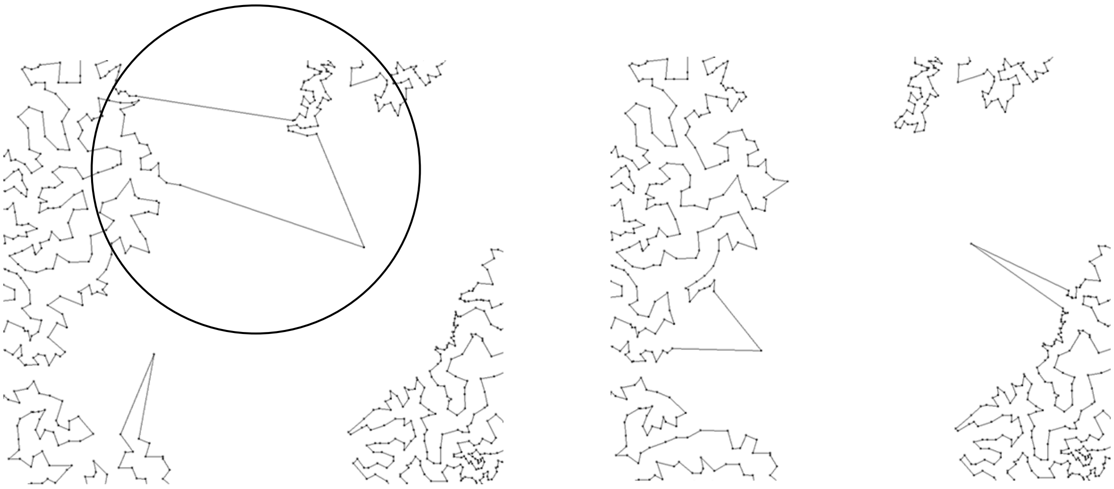

Fig. 6. Comparison of some segments of initial route and the optimized route (optimized more "globally")

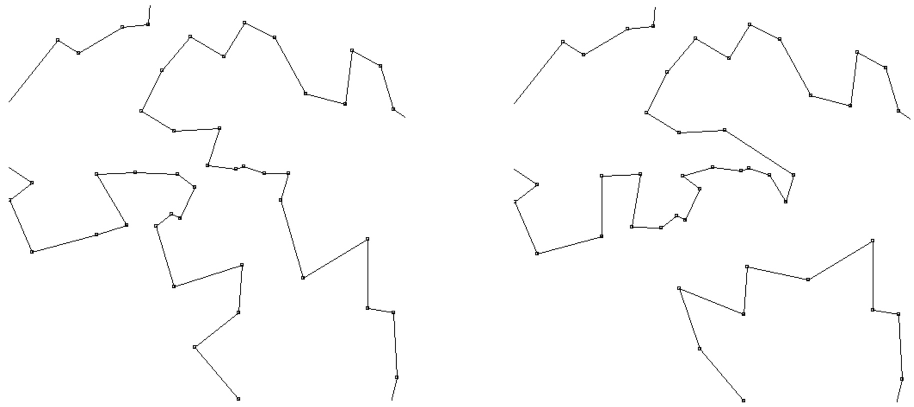

Fig. 7. Comparison of some segments of initial route and the optimized route

Table 2. Experimental results

Test problem	Number of points	Time, minutes	Quality
ch71009, China	71009	50,7	0,07%
fi10639, Finland	10639	10,7	0,05%
it16862, Italy	16862	15,9	0,02%
ja9847, Japan	9847	9,5	0,02%
sw24978, Sweden	24978	22,5	0,06%
usa115475, USA	115475	85,8	0,08%

Conclusion

New efficient decomposition and optimization methods, based on Delaunay triangulation, have been investigated for solving the large-scale travelling salesman problem. The computational complexity is close to linear-logarithmic. The problem is solved in several stages: partitioning the input set of points into subsets of limited sizes (≈ 800-2000 points); receiving the initial solution by merging partial solutions and its improvement by the developed optimization method. Methods provide at most 0.08% deviation from the best known solutions of the investigated problems of national TSPs and require much less time in comparison with the best existing heuristic or exact methods.

The work was performed as part of the project that has been partially funded by the State Agency for Science, Innovation and Informatics Implementation of Ukraine.

Bibliography

[Applegate, 1995] D. Applegate, R. Bixby, V. Chvátal V, W. Cook. Finding Cuts in the TSP // DIMACS Technical Report 95-05. – Rutgers University. – 1995.

[Applegate, 1999] D. Applegate, R. Bixby, V. Chvátal V, W. Cook. Finding Tours in the TSP // Technical report 99885, Research Institute for Discrete Mathematics. – Universität Bonn. – 1999.

[Applegate, 2003] D. Applegate, R. Bixby, V. Chvátal V, W. Cook. Implementing the Dantzig-Fulkerson-Johnson Algorithm for Large Traveling Salesman Problems // Mathematical Programming (2003). – (Series B) 97. – pp. 91-153.

[Applegate, 2006] D. Applegate, R. Bixby, V. Chvátal V, W. Cook. The Traveling Salesman Problem – A Computational Study // Princeton Series in Applied Mathematics. – Princeton University Press. – 2006.

[Applegate, 2009] D. Applegate, R. Bixby, V. Chvátal V, W. Cook, D. Espinoza, M. Goycoolea, K. Helsgaun. Certification of an Optimal TSP Tour Through 85,900 Cities // Operations Research Letters. – 37 (2009). – pp. 11—15.

[Bazylevych, 2008] R. Bazylevych, R. Kutelmakh, B. Prasad, L. Bazylevych. Decomposition and Scanning Optimization Algorithms for TSP // Proceedings of the International Conference on Theoretical and Mathematical Foundations of Computer Science. – Orlando, USA. – 2008. – pp. 110-116.

[Bazylevych, 2009] R. A. Bazylevych, B. Prasad, R. Kutelmakh, R. Dupas, L. Bazylevych. A Decomposition Algorithm for Uniform Traveling Salesman Problem / Bazylevych R., // Proceedings of the 4th Indian International Conference on Artificial Intelligence. – Tumkur, India. – 2009. – pp. 47-59.

[Bazylevych, 2009] R. Bazylevych, R. Kutelmakh. Optimization of TSP solutions by sequential scanning method // Visnyk of Lviv Polytechnic National University. – 2009. – № 638: “Computer sciences and information technologies”. – pp. 254-260 (In Ukrainian).

[Bazylevych, 2012] R.P. Bazylevych, M. Palasinski et al. “Decomposition methods for large-scale TSP”. In book: G. Setlak, M. Alexandrow, K. Markow. “Artificial intelligence methods and techniques for business and engineering application”. ITHEA, Rzeszow–Sofia, 2012, pp. 148 – 157.

[Christofides, 1979] N. Christofides. The Traveling Salesman Problem // Combinatorial Optimization. N. Christophides, A. Mingozzi, P. Toth and C. Sandi. Eds. – John Wiley and Sons, New York. – 1979.

[Dantzig, 1954] G. Dantzig, R. Fulkerson, S. Johnson. Solution of a Large-Scale Traveling-Salesman Problem // Operations Research. – 1954. – Vol. 2. – pp. 393-410.

[Flood, 1956] M. Flood. The traveling-salesman problem // Oper. Res. – 1956. – N. 4. – pp. 61-75.

[Guibas, 1985] L. Guibas, J. Stolfi. Primitives for the manipulation of general subdivisions and the computation of Voronoi // ACM Transactions on Graphics (TOG) . – 1985. – Volume 4, Issue 2. – pp. 74-123.

[Held, 1970] M. Held, R.M. Karp. The Traveling Salesman Problem and Minimum Spanning Trees // Operations Research. – 1970. – Vol. 18. – P. 1138–1162.

[Held, 1971] M. Held, R.M. Karp. The Traveling Salesman Problem and Minimum Spanning Trees: part II // Mathematical Programming. – 1971. – Vol. 1. – P. 6-25.

[Held, 1974] M. Held, P. Wolfe, H.P. Crowder. Validation of subgradient optimization // Mathematical Programming. – 1974. – Vol. 6. – P. 62-88.

[Helsgaun, 1998] K. Helsgaun. An Effective Implementation of the Lin–Kernighan Traveling Salesman Heuristic // Datalogiske Skrifter (Writings on Computer Science). – 1998. – No. 81. – Roskilde University.

[Helsgaun, 2006] K. Helsgaun. An Effective Implementation of k-Opt Moves for the Lin–Kernighan TSP Heuristic // Datalogiske Skrifter (Writings on Computer Science). – 2006. – No. 109. – Roskilde University.

[Karp, 1972] R. Karp. Reducibility among combinatorial problems // In Raymond E.Miller and James W.Thatcher, editors, Complexity of Computer Computations. – 1972. – Plenum Press, New York. – pp. 85-103.

[Lin, 1973] S. Lin, B.W. Kernighan. An Effective Heuristic Algorithm for the Traveling-Salesman Problem // Operations Research. – 1973. – Vol. 21, No. 2. – pp. 498-516.

[National TSPs] http://www.math.uwaterloo.ca/tsp/world/countries.html

[Papadimitriou, 1977] C.H. Papadimitriou. The Euclidean traveling salesman problem is NP-complete // Theoret. Comput. Sci. – 1977. – No. 4. – pp. 237-244.

[Reinelt, 1994] G. Reinelt. The Traveling Salesman Problem: Computational Solutions for TSP Applications // Lecture Notes in Computer Science. – 840, Springer-Verlag. – Berlin. – 1994.

Authors' Information

Roman Bazylevych – Full Professor, Ph.D., D.Sc, Mathematics and Computer Science Foundations, University of Information Technology and Management in Rzeszow, Poland and Software Engineering Department, Lviv Polytechnic National University, Ukraine; e-mail: *rbaz@polynet.lviv.ua*

Major Fields of Scientific Research: Computer Science, Design Automation, Algorithms, Combinatorial Optimization

Marek Pałasiński – Prof. nadzw. dr.hab., Mathematics and Computer Science Foundations, University of Information Technology and Management in Rzeszow, Poland e-mail: *mpalasinski@wsiz.rzeszow.pl*

Major Fields of Scientific Research: Theoretical computer science, Theory of algorithms, Graph theory, Data mining and Algebraic logic

Roman Kutelmakh – Assistant Professor, Ph.D., Software Engineering Department, Lviv Polytechnic National University, Ukraine; e-mail: *rkutelmakh@polynet.lviv.ua*

Major Fields of Scientific Research: Software technologies, Combinatorial Optimization, Algorithm design, Vehicle Routing Problems

Bohdan Kuz – Assistant Professor, Software Engineering Department, Lviv Polytechnic National University, Ukraine; e-mail: *bohdankuz@gmail.com*

Major Fields of Scientific Research: Software technologies, Combinatorial Optimization