Acta Agraria Kaposváriensis (2010) Vol 14 No 3, 1-14
Kaposvári Egyetem, Állattudományi Kar, Kaposvár
Kaposvár University, Faculty of Animal Science, Kaposvár

A Google Maps based novel approach to the optimization of multiple Traveling Salesman problem for limited distribution systems

A. Király, J. Abonyi
University of Pannonia, Department of Process Engineering, H-8200 Veszprém, P.O. Box 158.

ABSTRACT

The Vehicle Routing Problem (VRP) is a complex combinatorial optimization problem that can be described as follows: given a fleet of vehicles with uniform capacity, a common depot, and several costumer demands; find the set of routes with overall minimum route cost which service all the demands. The multiple traveling salesman problem (mTSP) is a generalization of the well-known traveling salesman problem (TSP), where more than one salesman is allowed to be used in the solution. It is well-known that mTSP based algorithms can also be utilized in several VRPs by incorporating some additional constraints. There are several exact algorithms of the mTSP with relaxation of some constraints of the problem. These approaches have serious importance because the solution what is supplied is optimal. Furthermore, due to the combinatorial complexity of mTSP, it is necessary to apply some heuristic in the solution, especially in real-sized applications. The aim of this paper is to discuss how genetic algorithms can be applied to solve these problems. It reviews the previous approaches with their disadvantages and proposes a novel, interpretable representation based algorithm. In this work, the effectiveness of the developed novel algorithm which based on the proposed representation will be demonstrated by several examples. The paper pans out about the implemented software solutions, like a new tool for calculation of distance table (on real routes) based on Google Maps API and a user-friendly tool for the drawing of the vehicle round trips. Furthermore, it proposes a complete framework and methodology to solve real problems. The new algorithm, a novel tool, and the resulted routes provide economically effective solutions for logistics which is proved by a real application.

(Keywords: mTSP, VRP, genetic algorithm, multi-chromosome, optimization)

ÖSSZEFOGLALÁS

Google Maps alapú új típusú megközelítés korlátos elosztási rendszer optimalizálására

Király A., Abonyi J.
Pannon Egyetem, Folyamatmérnöki Intézeti Tanszék, 8200 Veszprém, Pf. 158.

A fuvarszervezési probléma (VRP) egy komplex kombinatorikus optimalizálási feladat, ami a következőképpen írható le: adott a járművek egy halmaza, előre meghatározott, közös kapacitással, egy központi depó, és adottak vásárlói igények; a feladat egy minimális összköltségű útvonalhálózat keresése, mely az összes vásárlói igényt kielégíti. A többes utazóügynök probléma (mTSP) a jól ismert utazóügynök probléma (TSP) egy általánosítása, *ahol egy vagy több utazóügynök lehet a feladat megoldásában. Ismeretes, hogy a mTSP alapú algoritmusok VRP-k megoldására is használhatóak, további kikötések definiálásával. Több egzakt algoritmus is létezik az irodalomban, melyek néhány megszorítás relaxációját tartalmazzák mTSP-nek. Ezen algoritmusok elméletileg igazolt optimumot szolgáltatnak. Az mTSP probléma kombinatorikus komplexitása miatt szükséges valamilyen heurisztika alkalmazása a megoldásban, különösen valós méretű feladatok esetén. Jelen munka célja annak tárgyalása, hogy hogyan alkalmazhatóak a genetikus algoritmusok ilyen típusú problémák megoldására. Áttekinti a korábbi megközelítéseket azok hátrányaival együtt, és ajánl egy újszerű reprezentáción alapuló megoldást, és több példán keresztül szemlélteti annak hatékonyságát. A cikk kitér a megvalósított szoftver megoldásokra is, bemutat egy új Google Maps API-n alapuló távolságmátrix-generáló programot, és egy kényelmes eszközt az eredmények megjelenítésére. Továbbá ajánl egy teljes keretrendszert és módszert valós elosztási feladatok megoldására. Az újszerű algoritmus, az új eszközök és a kapott útvonalhálózatok gazdaságilag hatékonynak bizonyulnak logisztikai problémák esetén, amelyet a cikk egy valós életből vett példán szemléletet.*

(Kulcsszavak: mTSP, VRP, genetikus algoritmus, multi-kromoszóma, optimalizáció)

INTRODUCTION

The aim of logistics is to get the right materials to the right place at the right time, while optimizing a given performance measure (e.g. minimizing total operating cost) and satisfying a given set of constraints (e.g. time and capacity constraints). Supply chain management includes the planning and management of all activities involved in sourcing, procurement, conversion, and logistics management, as well as crucial components of coordination and collaboration. It deals with several problems, like Distribution Network Configuration, Trade-Offs in Logistical Activities, Inventory Management or Distribution Strategy (*Christopher*, 1999; *Burns et al*., 1985).

In logistics, several types of problems could come up; one of the most remarkable is the set of route planning problems. One of the most studied route planning problem is the Vehicle Routing Problem (VRP) (*Toth, and Vigo*, 1987), which is a complex combinatorial optimization problem that can be described as follows: given a fleet of vehicles with uniform capacity, a common depot, and several requests by the customers, find the set of routes with overall minimum route cost which service all the demands. The complexity of the search space and the number of decision variables makes this problem notoriously difficult. There doesn’t exist open-source or free approach which can provide a complete solution procedure for these optimization problems.

A relaxation of the VRP is the multiple Traveling Salesman Problem (mTSP) (*Bektas*, 2006), where the capacity of the vehicles is infinite. This means that all the solution procedures and formulations for VRP are also valid for mTSP, by assigning accordingly large capacities to the vehicles (which is matched to salesmen). The number of solvers for mTSP is much smaller than for VRP. If there is only one salesman in the problem, the mTSP reduces to the well-studied Traveling Salesman Problem (TSP). Because of the fact that TSP belongs to the class of NP-complete problems, it is obvious that mTSP is an NP-hard problem thus it's solution requires heuristic approach.

In this paper, an mTSP with Time Windows (mTSPTW) problem will be optimized with a novel approach, where the number of salesmen is an upper bounded variable, and there exist additional constraints, like the maximum travelling distance of each salesman.

In the last two decades the traveling salesman problem received quite big attention, and various approaches have proposed to solve the problem. For example branch-and-bound (*Finke*, 1986), cutting planes (*Miliotis*, 1978), neural network (*Bhide et al*., 1993) or tabu search (*Glover*, 1990). Some of these methods are exact algorithms, while others are near-optimal or approximate algorithms. The exact algorithms usually use integer linear programming approaches with additional constraints.

The mTSP is much less studied like TSP problem. (*Bektas*, 2006) gives a comprehensive review of the known approaches. There are several exact algorithms of the mTSP with relaxation of some constraints of the problem, like (*Laporte and Nobert*, 1980), which is the first approach to solve the mTSP directly, without any transformation of the TSP. In this problem, each salesman has a fixed cost *f*, which is activated whenever a salesman is included in the solution. The solution by (*Ali and Kennington*, 1986) is based on branch-and-Bound algorithm, which is applicable for asymmetric as well as symmetric problems. Another exact solution method is given by (*Gromicho et al*., 1992), which is based on a quasi-assignment relaxation obtained by relaxing the SECs.

Lately GAs are used for the solution of mTSP too. The first result can be bound to (*Zhang et al.*, 1999). Most of the work on solving mTSPs using GAs has focused on the Vehicle Scheduling Problem (VSP) (*Malmborg*, 1996; *Park*, 2001). VSP typically includes additional constraints like the capacity of a vehicle (it also determines the number of cities each vehicle can visit), or time windows for the duration of loadings. Recent application can be found in (*Tanga et al*., 2000), where GAs were developed for hot rolling scheduling. (*Yu et al*., 2002) use GAs to solve the mTSP in path planning. There are several representations of mTSP, like *one chromosome technique* (*Zhang et al*., 1999), the *two chromosome technique* (*Malmborg*, 1996; *Park*, 2001) and the latest *two-part chromosome technique* (*Carter et al*., 2006), wich is the best known so far. Although the salesmen in mTSP are separated from each other "physically", every previous solutions of mTSP with GA has used a single chromosome to represent a solution, i.e. to represent each salesman.

Rely on these considerations, the authors introduce a new representation to solve this problem-class with genetic algorithms, which can separate the salesmen from each other, thus it is more similar to the characteristic of mTSP. Authors also give a complete methodology for the optimization of routing problems by the development of a complete framework.

MATERIAL AND METHOD

GAs are direct, random search algorithms, based on the evolutionary model (*Goldberg*, 1989), related with Darwin's evolutionary theory. The researches of GAs have begun in the sixties by (*Holland*, 1975). GAs belong to the EC methods thus their terminology is close interconnected with biology. Each solution of the problem or equivalently each point in the search space is represented by an individual, who consists of chromosomes, and chromosomes consist of genes. Individuals constitute a population, which contains all possible solutions. The method is based on the collective learning process of the population. The individuals are improved in the course of iterations by the partway forthcoming operators, like selection, crossover and mutation[1].

GA starts with an initial solution set, which contains individuals created randomly. This is called initial population. The initial step can mightily improve the efficiency of the algorithm, thus a new start strategy can be momentous. The new population is always generated from the actual population's participants by the genetic operators. The generation of new populations is continued until a predefined stop criterion is satisfied.

[1] http://www.obitko.com/tutorials/genetic-algorithms/

Further description about general theory of genetic algorithms and about the usual genetic representations, operators, fitness assignment and genetic parameters can be found in (*Goldberg*, 1989).

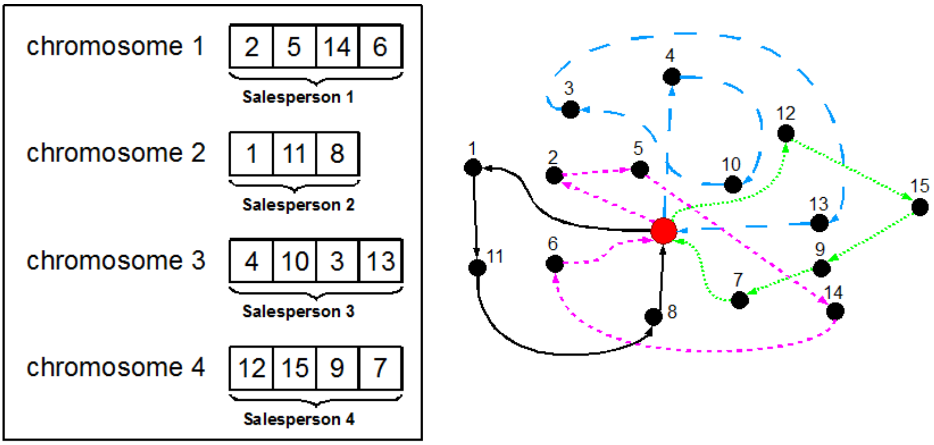
The proposed approach to solve mTSP

There are several representations of mTSP (see previous chapter), like *one chromosome technique* (*Zhang et al.*, 1999), the *two chromosome technique* (*Malmborg*, 1996; *Park*, 2001) and the latest *two-part chromosome technique* (*Carter et al.*, 2006). As mentioned in the previous chapter, every GA-based approach for solving the mTSP has used single chromosome for representation so far. The new approach presented here is a so-called multi-chromosome technique, which separates the salesmen from each other, thus may present more effective approach.

The multi-chromosome approach is used in notoriously difficult problems to decompose complex solution into simpler components. It was used in mixed integer problem (*Pierrot et al.*, 1997) or in order problems (*Yoshiji et al.*, 2001). A usage of routing problem optimization can be seen in (*Ronald and Kirkby*, 1998) and a lately solution of a symbolic regression problem in (*Cavill et al.*, 2005). This paper discusses the usage of multi-chromosomal genetic programming in the optimization of mTSP. *Figure 1* illustrates the new chromosome representation for mTSP with 15 cities (n = 15) and with 4 salesmen (m = 4). It shows a single individual of the population, which represents a single solution of the problem.

Figure 1

Example of multi-chromosome representation with 15 locations (n = 15) and 4 salesmen (m = 4)



1. ábra: Példa a multi-chromosome reprezentációra 15 város (n = 15) és 4 utazóügynök (m = 4) esetén

The first chromosome represents the first salesman itself, thus each gene denotes a city (cities was numbered previously, depot is not presented here, it is the first and the last station of each salesman). This is the so-called permutation encoding, because a sequence of numbers are encoded into the genes. It is in evidence in the example that salesperson 1 visits 4 cities: city 2, 5, 14 and 6, respectively. In the same way, chromosome 2 represents salesperson 2, which visits city 1, 11 and 8 respectively, and so on. This representation is much similar to the characteristic of the problem, because salesmen are separated from each other "physically".

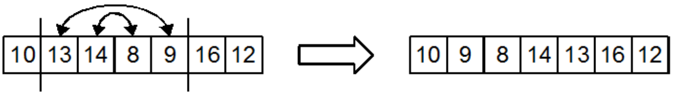
Special operators for the proposed representation

Because of our new representation, implementation of new genetic operators became necessary, like mutation operators. There are two sets of mutation operators, the so-called *In-route mutations* and the *Cross-route mutations*. Only some example of the newly created operators are given in this section. Further information with several examples about the novel operators can be found in (*Király A., Abonyi J.,* to be published in 2011).

In-route mutation operators work inside one chromosome. An example is illustrated in *Figure 2.* The operator chooses a random subsection of a chromosome and inverts the order of the genes inside it.

Figure 2

In-route mutation operator - gene sequence inversion

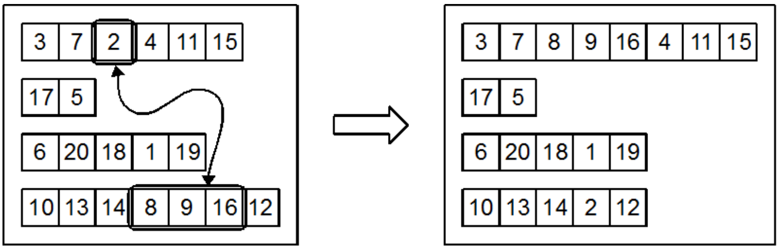


2. ábra: Úton belüli mutációs operátor – gén szekvencia megfordítása

Cross-route mutation operates on multiple chromosomes. If we think about the distinct chromosomes as individuals, this method could be similar to the regular crossover operator. *Figure 3* illustrates the method when randomly chosen subparts of two chromosomes are transposed. If the length of one of the chosen subsections is equal to zero, the operator could transform into an interpolation.

Figure 3

Cross-route mutation operator - gene sequence transposition



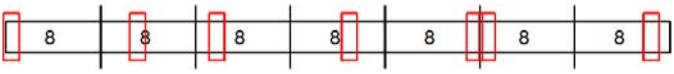
3. ábra: Utak közti mutációs operátor – gén szekvenciák felcserélése

The genetic loop

An overview of the realization of the novel representation will be discussed in the next section, a review of the evolution strategy will be discussed here.

Figure 4

The partition of the population and the selection of the best individual inside a part



4. ábra: A populáció felosztása kisebb részekre, és az egyes részeken belül a legjobb egyed kiválasztása

Figure 3 illustrates the division of the population. Firstly, a random permutation of the population is executed. The algorithm divides the population into *k* pieces, each piece contains *l* individuals. Here *l* is the number of applicable genetic operators, currently it is *8*. Thus, the population size must be divisible by *l*. The strategy is the following. The tournament selection selects the best member according to its fitness function (see next section) inside the first block of individuals. This member is transmitted into the next population, and this fellow is used for the new individual creation. The genetic operators (in-route and cross-route, see previous sections in this chapter) are applied with a certain feasibility on the selected individual *l* times. The best *l-1* new fellow and the winner of the tournament are transmitted into the new population. Next, the fitness assignment is executed for each individual considering the restrictions; it will be discussed in the next chapter. It is noticeable, that the algorithm doesn’t use the classical crossover operators. However, the cross-route operators realize a crossover between the chromosomes, which can be caught as a kind of crossover. Note that the classical Evolution Strategy (ES) doesn't contain crossover.

RESULTS AND DISCUSSION

Implementation of the novel representation based genetic algorithm

To analyze the new representation, a novel genetic algorithm using this approach was developed in MATLAB. This novel approach was compared with the most effective one so far (the two-part chromosome) which is available on MATLAB Central . The novel algorithm can optimize the traditional mTSP problems, furthermore, it is capable to handle the additional constraints and time windows. In this case, time windows are realized as an extra time at each location, e.g. it is the duration of loadings.

It requires two input sets, like the coordinates of the cities and the distance table which contains the travelling distances between any pair of cities. Naturally, the determination of the constraints, time windows and the parameters of the genetic algorithms are also necessary.

The fitness function simply summarizes the overall route lengths for each salesman inside an individual. The selection is tournament selection, where tournament size i.e. the number of individuals who compete for survival is 8. Therefore, population size must be divisible by 8. The winner of the tournament is the member with the smallest fitness, this individual is selected for new individual creation, and this member will get into the new population without any modification.

The penalty of the too long routes (over the defined constraint) instead of a proportionally large fitness value assignment is implemented by a split operator, which separates the route into smaller routes, which do not exceed the constraints (but the number of salesmen is incremented). Because there exists a constraint for the number of the salesmen, the algorithm involves the minimization of this amount, hence this penalty has a remarkable effect in the optimization process. Further information about the implemented algorithm can be found in (*Király and Abonyi*, to be published in 2011).

Complexity analysis of the proposed representation

Using the multi-chromosome technique for the mTSP reduces the size of the overall search space of the problem. Let the length of the first chromosome be k_1, let the length of the second be k_2 and so on. Of course $\sum_{i=1}^{m} k_i = n$. Determining the genes of the first chromosome is equal to the problem of obtaining an ordered subset of k1 element from a set of n elements. There are $\frac{n!}{(n-k_1)!}$ distinct assignment. This number is $\frac{(n-k_1)!}{(n-k_1-k_2)!}$ for the second chromosome, and so on. Thus, the total search space of the problem can be formulated as equation (1).

$$\frac{n!}{(n-k_1)} * \frac{(n-k_1)!}{(n-k_1-k_2)!} * \ldots * \frac{(n-k_1-\ldots-k_{m-1})!}{(n-k_1-\ldots-k_m)!} = \frac{n!}{(n-n)!} = n! \tag{1}$$

It is necessary to determine the length of each chromosome too. It can be represented as a positive vector of the lengths (k_1, k_2,…, k_m) that must sum to n. There are $\binom{n-1}{m-1}$ distinct positive integer-valued vectors that satisfy this requirement (*Ross, S. M., 1984*). Thus, the solution space of the new representation is $n!\binom{n-1}{m-1}$. It is equal with the solution space in (*Carter et al.*, 2006), but this approach is more similar to the characteristic of the mTSP, so it can be more problem-specific therefore more effective.

A proposed Google Maps API based framework

For a real application, from the definition of the initial map to the final routes, the development of a whole methodology and a complete framework is needed. Out of the implemented algorithm, the Google Maps based determination of the distance matrix and the lightly comprehensible visualization of the results is necessary. There were several aspects to which the proposed solution had to match:

- It has to use open-source or free technologies
- Users need to easily define an initial map
- The information retrieval of coordinates from the defined map has to be automatic
- The distance table computation has to be automatic
- The solution need to handle additional constraints, like maximum number of salesmen, maximum travelling time / distance per salesmen, and time windows
- The solution need to be feasible and it has to be optimal as far as possible
- Easily interpretable visualization is needed
- Minimal user interaction

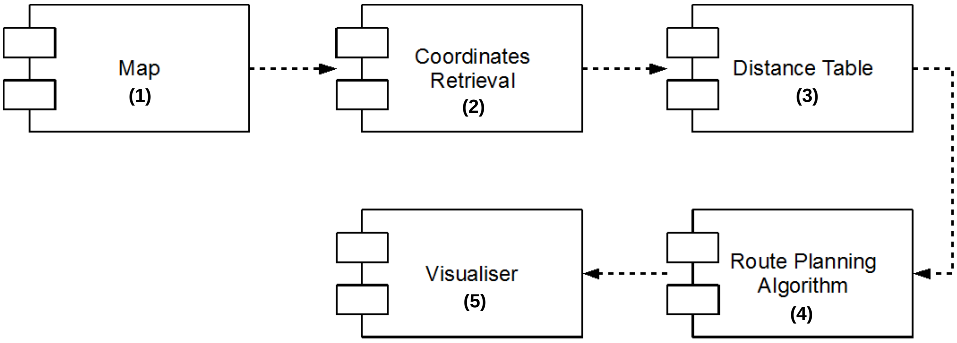
These expectations determine the usable technologies. In this paper we publish this novel framework at the first time, where the Google's web map service is used, because it is one of the most widely used mapping service on earth, and it provides a complete open-source programming API. According to the Google Maps API Terms, the service can be available only via Javascript programming language; hence the chosen client-side programming language is Javascript. To import initial data and to export the results, a dynamic server-side programming language is required too. Authors' choice is the PHP programming language, because it is one of the most popular server-side scripting languages running today, which is used to create dynamic web pages.

The requirements above necessitate the solution of problems with a different type, hence an approach consisting of discrete softwares can be appropriate. This realization effects a modular solution, where the components are not strongly dependent from each other. Namely, the delivery of data between the components can be resolved by temporary files. This approach makes each component usable itself, thus the framework is easily expandable and the components are highly reusable.

In *Figure 5* the component diagram of the proposed solution is illustrated. First of all, a definition of input data (*Map*) is needed. This first object on the figure represents the determination of locations on a Google Maps map. The second component (*Coordinates Retrieval*) provides an automatic tool for the retrieval of longitude and latitude values of the locations on the *Map*. *Distance Table* component involves the computation of distances and duration between every pair of locations and uses the data determined by the previous component. The next step is the determination of optimal routes (*Route Planning Algorithm*), which is the proposed GA discussed earlier. This component requires the distance table provided by the previous component. The last component (*Visualiser*) is a visualiser, which can show the resulted routes in an easily interpretable form on a Google Maps map.

Figure 5

Component diagram of the proposed framework



5. ábra: A tervezett keretrendszer komponens diagramja

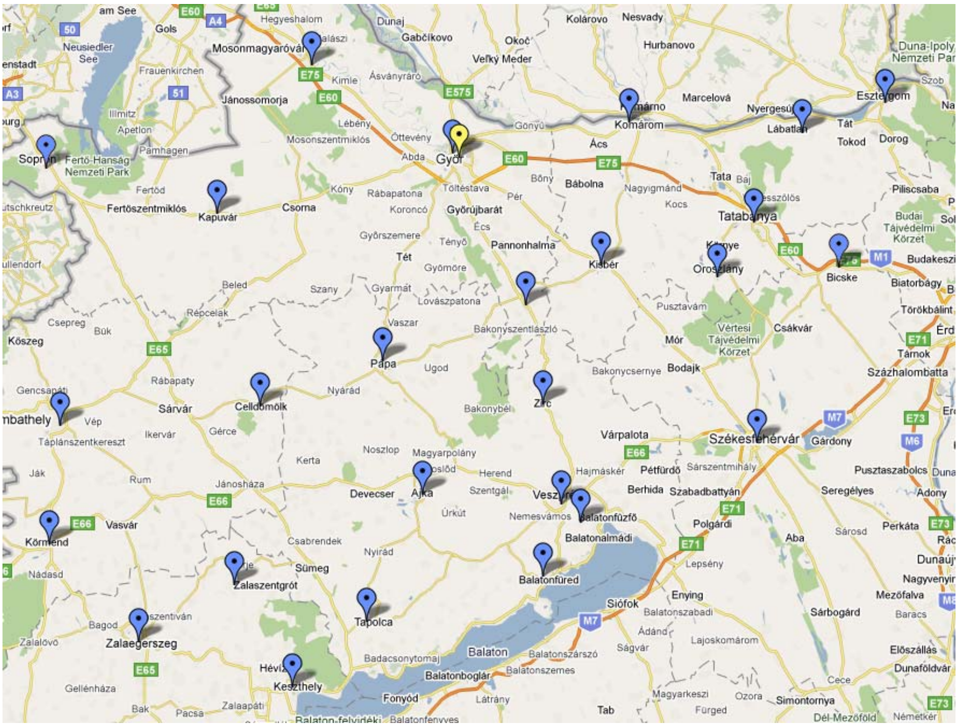
Térkép(1), Koordináták kinyerése(2), Távolságmátrix(3), Útvonaltervező algoritmus(4), Megjelenítő(5)

Application study

In this section every steps of the methodology will be presented during a solution of a real problem. The input data is given by a map as it is illustrated in *Figure 6*. It contains 25 locations (with the depot).

Figure 6

The input map of the studied application



6. ábra: A tárgyalt valós alkalmazás bemeneti térképe

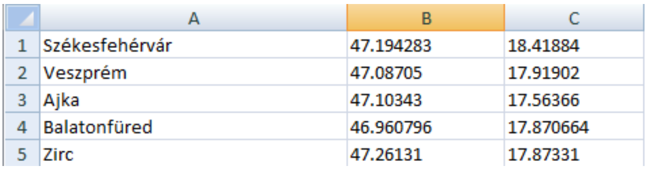
The task is to determine the optimal routes for these locations with the following constraints:

- maximum number of salesmen is 5
- maximum travelling distance of each salesman is 400 km

The first step is the information retrieval from the map. It can be done by the help of *Coordinates Retrieval* component. A piece of the output is illustrated in *Figure 7*.

Figure 7

A piece of the coordinates of the placemarks from the map (output of Coordinates Retrieval component)



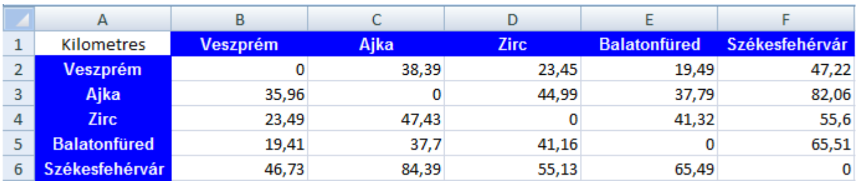
	A	B	C
1	Székesfehérvár	47.194283	18.41884
2	Veszprém	47.08705	17.91902
3	Ajka	47.10343	17.56366
4	Balatonfüred	46.960796	17.870664
5	Zirc	47.26131	17.87331

7. ábra: Részlet a térképen definiált jelölők koordinátáiból (a Coordinates Retrieval komponens kimenete)

Thereafter this spreadsheet is applicable for the input of distance matrix determination. Some of the resulted data is shown in *Figure 8*.

Figure 8

A piece of distance matrix of the studied application



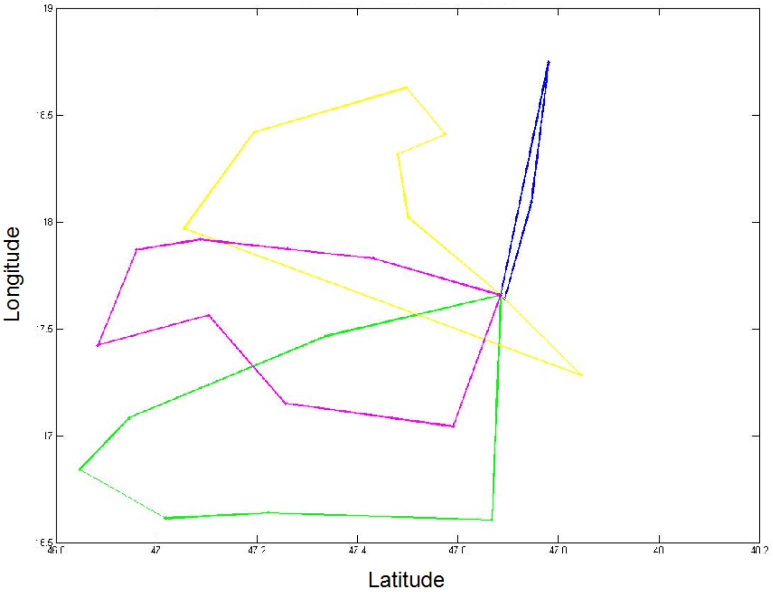
	A	B	C	D	E	F
1	Kilometres	Veszprém	Ajka	Zirc	Balatonfüred	Székesfehérvár
2	Veszprém	0	38,39	23,45	19,49	47,22
3	Ajka	35,96	0	44,99	37,79	82,06
4	Zirc	23,49	47,43	0	41,32	55,6
5	Balatonfüred	19,41	37,7	41,16	0	65,51
6	Székesfehérvár	46,73	84,39	55,13	65,49	0

8. ábra: A tárgyalt valós alkalmazás távolságmátrixának részlete

After distance matrix determination the algorithm computes the solution with the new representation. The GA ran with population size 320 and it did 200 iterations. The result of the optimization provided by MATLAB is illustrated on *Figure 9*. It found this optimum through 107 iterations, which required about 20 seconds. It resulted that 3 salesman is enough to satisfy the constraints. With the Visualiser component we can visualize the results, as it is shown on *Figure 10*. The length of the routes are 349 km, 369 km and 333 km respectively, i.e. they satisfy the constraints, thus the algorithm provided a feasible solution of the problem. The overall route length is 1051 kilometers.

Figure 9

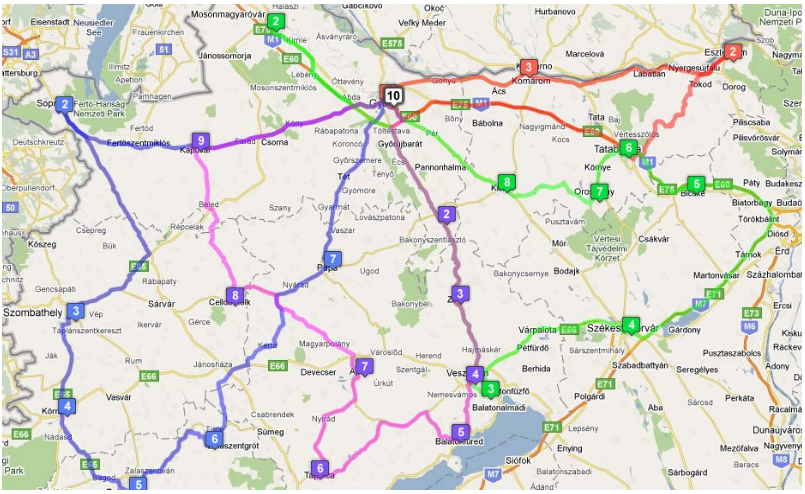
Results of the optimization by MATLAB with at most 5 salesmen and at most 400 km tour length per salesperson



9. ábra: A MATLAB optimalizáció eredménye legfeljebb 5 utazóügynök és utazóügynökönként maximálisan 400 km megtehető útvonalhossz figyelembe vételével

Figure 10

Visualization of the results by the Visualiser component



10. ábra: Az eredmények megjelenítése a Visualiser komponens segítségével

CONCLUSIONS

In the paper, a modified mTSP with additional constraints was introduced and solved by the novel approach. The literature review has showed unambiguously that the existed representations for solving mTSP by the help of GA uses only one chromosome to represent the whole problem, although the nature of the problem could necessitate the separation of the salesmen on chromosome design level. This observation motivated the author to introduce a novel representation in the individual design, where a separate chromosome is assigned to all salesmen. The aim of the paper was to propose a methodology for the optimal route planning with multiple salesmen and with side constraints from an initial Google Maps map to the final optimal or near-optimal routes visualized on a Google Maps map. The approach presented here is innovative in the representation of individuals, in the handling of the constraints, and it gives a whole methodology and a novel complete framework to solve an NP-hard problem, the mTSPTW.

ACKNOWLEDGEMENTS

The financial support from the TAMOP-4.2.2-08/1/2008-0018 (Livable environment and healthier people – Bioinnovation and Green Technology research at the University of Pannonia) project is gratefully acknowledged. This work has been supported in part by the TAMOP-4.2.2/B-10/1-2010-0025 project.

REFERENCES

Ali, A.I., Kennington, J.L. (1986). The asymmetric m-traveling salesmen problem: a duality based branch-and-bound algorithm. Discrete Applied Mathematics 13, 259-276. p.

Bektas, T. (2006). The multiple traveling salesman problem: an overview of formulations and solution procedures. Omega 34. 209-219 p.

Bhide, S., John, N., Kabuka, M.R. (1993). A boolean neural network approach for the traveling salesman problem. IEEE Transactions on Computers 42. 10. 1271. p.

Burns, L.D., Hall, R.W., Blumenfeld, D.E., and Daganzo, C.F. (1985). Distribution strategies that minimize transportation and inventory costs. Operations Research 33. 3. 469-490. p.

Carter, A.E., Ragsdale, C.T. (2006). A new approach to solving the multiple traveling salesperson problem using genetic algorithms. European Journal of Operational Research 175. 246-257. p.

Cavill, R., Smith, S., Tyrrell, (2005). A. Multi-chromosomal genetic programming. In Proceedings of the 2005 conference on Genetic and evolutionary computation, ACM New York, NY, USA, 1753-1759. p.

Christopher, M. (1999). Logistics and supply chain management: strategies for reducing cost and improving service. International Journal of Logistics Research and Applications 2. 1. 103-104. p.

Finke, G. (1986). Network flow based branch and bound method for asymmetric traveling salesman problems. In XI Symposium on Operations Research, Darmstadt, 117-119. p.

Glover, F. (1990). Artificial intelligence, heuristic frameworks and tabu search. Managerial and Decision Economics 11. 5. p.

Goldberg, D.E. (1989). Genetic algorithms in search, optimization and machine learning. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA

Gromicho, J., Paixao, J., Bronco, I. (1992). Exact solution of multiple traveling salesman problems. Combinatorial optimization: new frontiers in theory and practice, 291-292. p.

Holland, J.H. (1975). Adaptation in natural and artificial systems. The University of Michigan Press

Király A., Abonyi J. (to be published in 2011). Optimization of multiple traveling salesmen problem by a novel representation based genetic algorithm. In M. Koeppen, G. Schaefer, A. Abraham, and L. Nolle, editors, Intelligent Computational Optimization in Engineering: Techniques & Applications, Advances in Intelligent and Soft Computing. Springer

Laporte, G., Nobert, Y. (1980). A cutting planes algorithm for the msalesmen problem. Journal of the Operational Research Society 31. 1017-1023. p.

Malmborg, C. J. (1996). A genetic algorithm for service level based vehicle scheduling. European Journal of Operational Research 93. 1. 121-134. p.

Miliotis, P. (1978). Using cutting planes to solve the symmetric travelling salesman problem. Mathematical Programming 15. 1. 177-188. p.

Park, Y.-B. (2001). A hybrid genetic algorithm for the vehicle scheduling problem with due times and time deadlines. International Journal of Productions Economics 73. 2. 175-188. p.

Pierrot, H.J., Hinterding, R. (1997). Multi-chromosomal genetic programming, vol. 1342/1997 of Lecture Notes in Computer Science. Springer, Berlin / Heidelberg, ch. Using multi-chromosomes to solve a simple mixed integer problem, 137-146. p.

Ronald, S., Kirkby, S. (1998). Compound optimization. solving transport and routing problems with a multi-chromosome genetic algorithm. In The 1998 IEEE International Conference on Evolutionary Computation, ICEC'98, 365-370. p.

Ross, S.M. (1984). Introduction to Probability Models. Macmillian, New York,

Tanga, L., Liu, J., Rongc, A., Yanga, Z. (2000). A multiple traveling salesman problem model for hot rolling scheduling in shangai baoshan iron & steel complex. European Journal of Operational Research 124, 267-282

Toth, P., Vigo, D. (1987) The vehicle routing problem. Society for Industrial Mathematics

Yoshiji, F., Yuki, A., Tsuyoshi, Y. (2001). Applying the genetic algorithm with multi-chromosomes to order problems. Proceedings of the Annual Conference of JSAI 13. 468-471. p.

Yu Z, Jinhai L, Guochang G, Rubo Z, and Haiyan Y. (2002). An implementation of evolutionary computation for path planning of cooperative mobile robots. Proceedings of the fourth world congress on intelligent control and automation 3. 1798-1802. p.

Zhang, T., Gruver, W., Smith, M. (1999). Team scheduling by genetic search. Proceedings of the second international conference on intelligent processing and manufacturing of materials 2. 839-844. p.

Corresponding author (*Levelezési cím*):

András Király
University of Pannonia, Department of Process Engineering
H-8200 Veszprém P.O. Box 158.
Pannon Egyetem, Folyamatmérnöki Intézeti Tanszék
H-8200 Veszprém, Pf. 158.
Tel.: +36-88-624-770, Fax: +36-88-624-171
e-mail: kiralya@fmt.uni-pannon.hu