FORUM GEOGRAFIC

Forum geografic. Studii și cercetări de geografie și protecția mediului
Volume **XX**, Issue **1** (**June** 2021), pp. **104-117**
http://dx.doi.org/**10.5775/fg.2021.084.i**

Heuristic Approach in Determining the Best Tourist Tours to Medieval Fruška Gora Monasteries in Serbia

Srdjevic BOJAN[1,*]

[1] Faculty of Agriculture, Department of Water Management, University of Novi Sad, Trg D. Obradovica 8, 21000 Novi Sad, Serbia

* *Corresponding author*: bojans@polj.uns.ac.rs

Received on 24-11-2020, reviewed on 08-03-2021, accepted on 14-04-2021

Abstract

This study brings the results of comprehensive analysis aimed at finding the best tourist routes between twenty important tourism sites in Serbia: sixteen medieval monasteries at Fruška Gora Mountain, two other important monasteries in the area, and cities of Novi Sad and Belgrade as starting points of tours. Several travelling salesman problems are formulated and the shortest tours are found with the application of heuristic approach and genetic algorithm developed specially for this purpose. The best tour connecting all twenty sites of great tourist interest is firstly identified in strictly geographical terms by using GPS coordinates and orthodromic distances. This result, considered as the optimal in mathematical sense is not achievable in real circumstances, but can serve as target and be compared with any other solution obtained as if the touring all sites is made by car. In this study the distances between sites are based on node-to-node ground transportation infrastructure data downloaded from the Google Maps Service at Internet. Shortest tours respect topography of the area and can be used for planning tourist and other visits to monasteries and cities of national and international importance. The results of this study are considered as input to improvement of existing government policies affecting Serbian travel and tourism marketing. An approach is applicable elsewhere with open agenda for extensions and further improvements.

Keywords: *medieval monasteries; TSP; genetic algorithm; Fruška Gora Mountain, Serbia*

Rezumat. O abordare euristică pentru determinarea celor mai bune rute turistice pentru mănăstirile medievale Fruška Gora din Serbia

Acest studiu prezintă rezultatele unei analize comprehensive ce a avut ca scop găsirea celor mai bune rute care să lege 20 de obiective turistice importante din Serbia: 16 mănăstiri medievale din Muntele Fruška Gora, alte două mănăstiri din zonă și orașele Novi Sad și Belgrad ca puncte de plecare pe traseu. Au fost formulate mai multe problemele ale voiajorilor comerciali, cele mai scurte trasee fiind găsite în urma unei abordări euristice și a unui algoritm genetic dezvoltat special pentru acest studiu. Cel mai bun traseu care leagă toate cele 20 de situri de mare interes turistic este determinat mai întâi în termeni strict geografici, folosind coordonatele GPS și distanțele ortodromice. Acest rezultat, considerat optim în sens matematic, nu este însă fezabil în realitate, dar poate fi folosit ca un reper țintă pentru a fi comparat cu toate celelalte alte soluții obținute dacă s-ar folosi mașina pentru vizitarea tuturor obiectivelor. În cadrul acestui studiu, distanțele dintre diferite obiective au fost stabilite ținând cont de principalele noduri ale infrastructurii de transport rutier, datele fiind descărcate de pe Google Maps Service. Cele mai scurte rute țin cont de topografia zonei și pot fi folosite pentru planificarea vizitelor turistice și nu numai la mănăstiri și orașe de importanță națională și internațională. Rezultatele acestui studiu pot aduce o contribuție la îmbunătățirea strategiilor guvernamentale privind călătoriile în Serbia și marketingul turistic. Această abordare este de asemenea aplicabilă și în alte domenii, care permit extinderi și îmbunătățiri ulterioare.

Cuvinte-cheie: *Mănăstiri medievale, TSP, algoritm genetic, Muntele Fruška Gora, Serbia*

Introduction

Fruška Gora is a mountain in the Autonomous Province of Vojvodina. Most of the area is administratively part of Serbia, with a small part on its western side that extends into eastern Croatia. Documents of Ottoman Empire (old Turkish Kingdom) mention 35 monasteries in territory which is now Serbian territory, out of which 16 are active at the Fruska Gora till today. Due to the large number of monasteries on one place, Fruška Gora is colloquially called Serbian Mount Athos (Bukvić, 2017). The Holy Synod of the Serbian Orthodox Church officially declared Fruška Gora the Holy Mountain in 2003 (Kovacevic, 2017). Its 16 medieval monasteries are protected by the state as monuments of culture of exceptional importance since 1990.

According to historical data, monastic communities in Fruška Gora region were historically recorded since the first decades of the 16th century. Legends, however, place their founding to the period between the 12th and 15th centuries. In the course of centuries of their existence, these monasteries sustained the spiritual and political life of the Serbian nation (Trišić, 2020; Škrbić & Milošević, 2012; Vujko et al. 2017).

This study is aimed at exploring different travel options from the cities of Belgrade (capital of Serbia) and Novi Sad (capital of Serbian Autonomous province of Vojvodina) to 16 medieval monasteries at the Fruška Gora Mountain (fore after abbreviated as FGM). Two important monasteries in the area (near Novi Sad) are also included in the analysis because of their geographically convenient position for visiting.

Different travel options are explored and best travelling routes by car are identified as shortest distances in kilometres to help tourist and other sectors to plan efficient and energy saving travels and visits to historic and religious monuments of national and international importance.

Finding the shortest routes between monasteries and cities is modelled as the Travelling Salesman Problem (TSP), well known classical combinatorial optimization problem (Holland, 1975; Lin & Kernighan, 1973). Several TSPs are formulated and solved using all relevant data on geographical positions of monasteries and cities.

The TSP formulation is simple but to solve it can be quite difficult. In short, traveller starts at given node in the network of nodes and connecting links and has to visit all nodes just once before he returns to staring node. This tour should be the shortest, that is, optimal. TSP is NP-hard problem which cannot be solved exactly in polynomial time, especially if number of points in its solution space is very large. There are exact and heuristic algorithms developed to find optimal solution to the TSP, that is, to find the tour of minimal length. As correctly stated in (Potvin, 1996), exact algorithms are computationally expensive because they are typically derived from the integer linear programming formulations of the TSP and must (implicitly) consider all solutions in integer-discrete solution space before they can identify optimum. Commonly used exact algorithm is branch and bound with different relaxation methods when generating lower bounds on the optimal value (Padberg & Rinaldi, 1991). The main problem with all such algorithms is how to handle sets of sub tour-breaking constraint which restrict the feasible solutions to solutions consisting of a single tour. The idea is to introduce constraint which prohibits sub tours, that is tours within subsets with less than total number of nodes (vertices) in the network of the TSP. Although there are good solutions in this direction, most often complete enumeration of solution space is necessary and therefore in recent times exact algorithms are not used.

Although exact algorithms for solving the TSP were not of interest in this research, mathematical descriptions and associated discussions about solving procedures is useful to consult in order to get better insight into the reasons why heuristic algorithms are almost only used by researchers and practitioners. Interested readers can find a good reviews of exact and approximate algorithms in OR literature, for instance in (He et al. 2016; Laporte, 1992; Lawler et al., 1985; Muhlenbein, Gorges-Schleuter & Kramer, 1988).

Among different classes of heuristic algorithms, genetic algorithms (GAs) (Holland, 1975; Goldberg, 1989; Brady, 1985; Cheng & Gen, 2019) are probably most popular. They are adaptive search procedures designed to simulate processes in natural systems and evolution processes following the survival of the fittest principle laid down by Charles Darwin. As such, they represent an intelligent exploitation of a random search within a defined search space and very efficiently solve extremely large TSP in reasonable computational times. GAs are widely studied since early 60s of the last century and applied in different fields of engineering, but also in biology, geography, chemistry and henceforth. With increase of computational power of modern computers, GAs successfully replaced traditional, exact, algorithms by outperforming them in many different ways, but especially by providing alternative methods to encode problems and use various heuristics to solve problems.

Recent implementations of GAs as global search techniques include installing and running the same algorithm at more than one processor/computer. Parallel processing and sharing the best-so-far-found solution among processors/computers is known as elitist reproduction that enables diversification and migration of generations of candidate solutions, escaping from local optima and extremely efficient search for global optimum (Srđević, 2019).

There is no general genetic algorithm. Rather, a specialized algorithm should be written for given problem using appropriate programming language and by following problem-oriented and evolution-inspired rules and heuristics; in addition to the main source code different shells and sub-routines can be added to perform specific tasks such as recombination, shuffling, generating random numbers, etc.

In many experiments and applications GAs demonstrated outstanding performance in optimization (Talwo et al., 2013; Bolaños, Echeverrya & Escobar, 2015), not only as a function optimizers, but also as extremely efficient problem solvers. They are commonly considered as a computational analogy of adaptive systems and intelligent replication of parallel computing (Muhlenbein, Gorges-Schleuter & Kramer, 1988). More details on genetic algorithms and their applications can be found in (Cheng & Gen, 2019; Talwo, 2015; Goldberg, 1989; Jain & Singh, 2013; Srdjevic and Srdjevic, 2016; Srdjevic, 2019).

Genetic algorithms are often used for solving the large TSPs that occur in many engineering and other fields. Although there are different classifications of TSPs in relation to heuristics, such as construction, improvement or composite procedures (Potvin, 1996), there are also sub-classifications related to strictly problem oriented heuristics that lay somewhere between these classifications. GA-FGM is loosely based on the principles of the evolution via natural selection. It employs a population of individuals (tours connecting monasteries and cities) that undergo selection in the presence of mutation and recombination (crossover) operators. A direct encoding scheme is used to model several TSPs and a fitness function to evaluate individuals is represented by sum of node-to-node distances. Reproductive success in creating individuals, randomly mating individuals as parents producing offspring and creating generations of new individuals propagated towards better solutions until global optimum (shortest tour) is found, are only few of desired outcomes of GA-FGM achieved in combination with different heuristics in multiple runs performed.

Two computer programs are developed during this study period. For solving TSPs, the GA-FGM program is written to realize comprehensive computations required by genetic algorithm. Source and executive code of this program is not available. The other program named HAVDIS is used for calculating GPS orthodromic (great-cycle) distances and is available in either source or executive form. All programming is done in FORTRAN programming language.

Monasteries and cities

The 16 medieval monasteries at the Fruška Gora Mountain (Fig. 1) are concentrated in an area approximately 50 kilometres long and 10 kilometres wide.

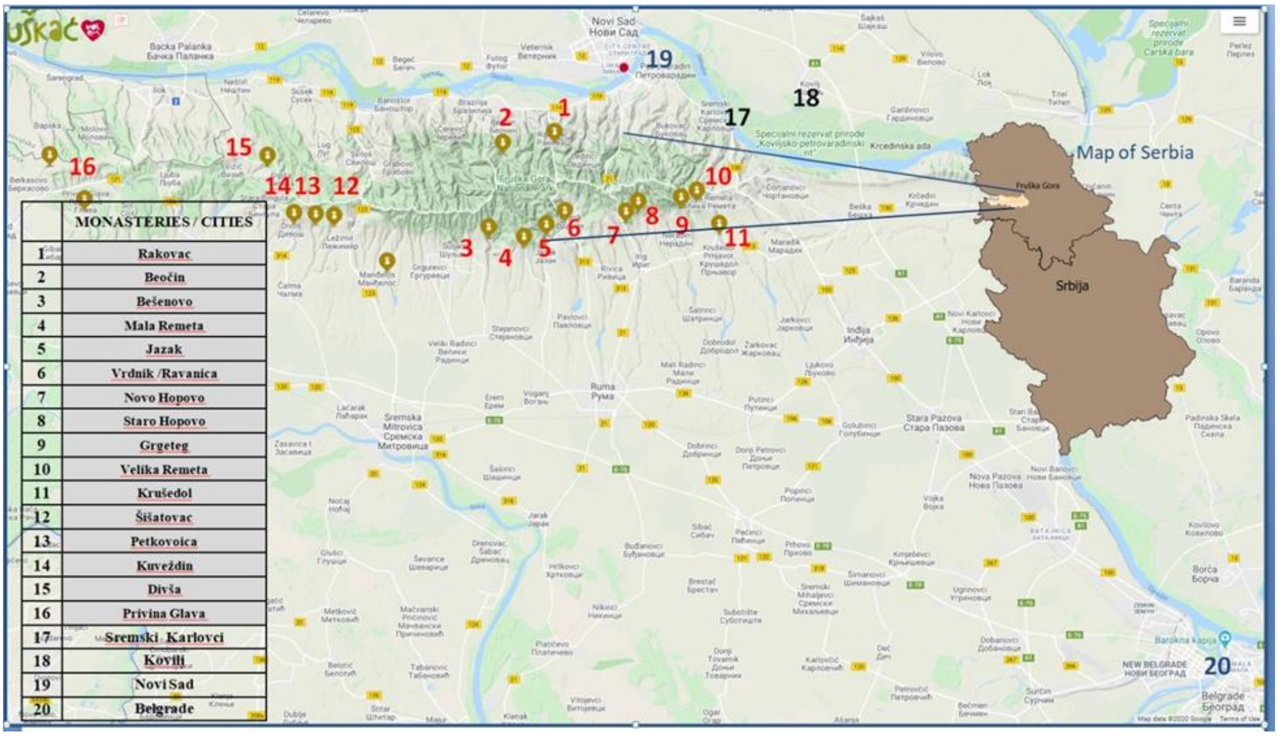

Fig. 1: Sixteen medieval monasteries at the Fruška Gora Mountain (Internet source: https://fruskac.net/rs/mapa)

Monasteries are referenced in this study in the following order:

1. **Rakovac** (according to a legend written in 1704 Rakovac was erected the monastery in 1498. The earliest historical records mentioning the monastery are dated to 1545/1546);
2. **Beočin** (the time of founding is unknown; first mentioned in Turkish records dated in 1566/1567);
3. **Bešenovo** (according to the legend, it was founded at the end of the 13th century. The earliest historical records about the monastery date from 1545);
4. **Mala Remeta** (the earliest historical records relating to the monastery are dated to the middle of the 16th century);
5. **Jazak** (founded in 1736 and reconstructed from 1926 to 1930);
6. **Vrdnik / Ravanica** (the present church in the monastery was constructed in the period from 1801 to 1811).;
7. **Novo Hopovo** (the first reliable mention of the monastery is dated to 1641);
8. **Staro Hopovo** (the first reliable mention of the monastery dates back to 1545/1546);
9. **Grgeteg** (founded in 1471. The earliest historical records about the monastery date back to 1545/1546);
10. **Velika Remeta** (traditionally, its founding is linked to the end of the 13th century. The earliest historical records about the monastery date to 1562);
11. **Krušedol** (founded between 1509 and 1516);

12. **Šišatovac** (the foundation of the monastery is ascribed to the refugee monks from the Serbian monastery of Žiča. The reliable facts illustrating the life of the monastery date back from the mid 16th century);

13. **Petkovica** (the earliest historical records mentioning the monastery are dated to 1566/1567);

14. **Kuveždin** (the first reliable records of its foundation are dated in 1566/1569);

15. **Divša** (founded in thre late 15th century. The earliest historical records about the monastery date to the second half of the 16th century);

16. **Privina Glava** (according to the legends monastery is founded by a man named Priva, in the 12th century. The earliest historical records about the monastery are dated in 1566/1567).

There are two more important religious sites in picturesque cities and important touristic destination near the city of Novi Sad: Vavadenje Presveta Bogorodica, known also as the Upper Church monastery in Sremski Karlovci, and the Svetih Arhangela (Holly Arhangels) monastery in Kovilj. In the map at Fig. 1 they are identified as monasteries number 17 and 18, respectively.

Two cities included in this study are Novi Sad and Belgrade, identified on the same map as locations 19 and 20, respectively. Belgrade is capital of Republic of Serbia, approximately 80 km southern from Novi Sad. These cities are included because tourist or other purpose tours and visits to aforementioned 16+2 Serbian religious sites would probably start from these two cities. Tour from Belgrade most often include Novi Sad, and on a way to it also visit to the monastery in Kovilj. From Novi Sad, after crossing the Danube River on a way to Belgrade and medieval monasteries in the Fruška Gora Mountain, tours often include monastery in Sremski Karlovci as an important religious and historic monument of Serbia and its cultural heritage.

Problem statement

There are thousands of billions of possible car routes connecting FGM in a 'travelling salesman' manner. Recall that Travelling Salesman Problem (TSP) is one of most popular linear programming optimization problems in combinatorial mathematics. The problem here belongs to this class of problems and is stated as to find the shortest tour (route) between n points in a connected network of monasteries and cities, assuming that traveller may start in any given point, visit all the others and return to his origin. Each point is to be visited only once, except starting/stopping point. If all points are connected by links and travelling is possible in both directions between any two points, assuming that distances are the same in either direction, then TSP is complete and symmetrical.

Solution space for *n* points consists of n! closed paths. In combinatorial mathematics *n!* is the number of permutations of *n* elements without repetition. If there are 16 nodes in the TSP network, labeled by integers, then there are $16! = 20.92279 \times 10^{12}$ possible closed loops connecting all nodes. The number of paths is slightly more than 20 trillion and to find the shortest one (or multiple paths with the same length) is not that easy task. In this study several TSPs has been solved with 20 modes and $20! = 2.43291298 \times 10^{18}$ large solution spaces. In principle, any TSP can be solved by enumerating procedure known as brute-force search through complete solution space. This process would be obviously tedious, and instead of it most often solutions are searched for by intelligent stochastic algorithms combined with certain heuristics. In later case candidate algorithms, besides genetic, are from the families known as ant colony systems (Dorigo & Stützle, 2019; Bonabeau, Dorigo & Theraulaz, 1999; Dorigo & Gambardella, 1997; Dorigo, Maniezzo & Colorni, 1996; Srdjevic, 2004), taboo search (Glover, 1989 and 1990; Glover & Laguna, 1997), simulated annealing (Blagojevic, Srdjevic, Srdjevic & Zoranovic, 2016; Laarhoven & Aarts, 1987; Aarts, Korst & Laarhoven, 1988), particle swarm optimizarion (Jahandideh-Tehrani, Bozorg- Haddad & Loáiciga, 2020), among many others (e.g. Yang et al., 2016).

In our study we defined a set of travelling salesman problems to enable analysis of travelling by car options between 16+2 monasteries and cities of Belgrade and Novi Sad. Problems are structured in a way to first find ideal shortest paths measured in kilometres and consider them as targets to be met along the best possible roads if one decides to drive the car and visit all the sites. GPS coordinates and Google Maps distances for all performed computations are downloaded from reliable Internet sources.

Worth to mention is that a set of additional TSPs related to travelling by car and on foot and required times for doing that has been explored in initial phase of this study, too. A decision has been made however to postpone solving these problems for later times as an addition to analyses provided here that is related to only car travelling along national, regional and local transport infrastructure.

Solution methodology

Applied methodology in this research consisted of the following eight steps:

1. review historical, cultural, religious and geographical documents and other sources of information about medieval monasteries at the Fruška Gora Mountain;

2. identify position of monasteries at the Google Maps internet platform and label them with integers 1 through 16 in a direction from centre to East and to West. Also, identify position of two other important monasteries in the area and position of cities Belgrade and Novi Sad;

3. download from the internet the GPS (Earth) spherical coordinates (latitudes and longitudes) for all 20 nodes in corresponding network with 18 monasteries and two cities;

4. develop computer program and based on GPS coordinates calculate orthodromic (great-circle) distances in kilometres between all nodes in a network. Use Haversine formula to perform this task;

5. download from Google Maps platform distances in kilometres between all 20 points for by-car travelling;

6. develop genetic algorithm for solving general symmetric TSP and find optimal (shortest) tours - (a) between all 18 monasteries and two cities (with start and return to Belgrade) and (b) between city of Novi Sad and monasteries at Fruška Gora Mountain with start and return to Novi Sad;

7. apply genetic algorithm and compute shortest routes between identified 20 sites for two types of distances - (a) orthodromic (spherical) and (b) on-Earth as suggested by Google Maps;

8. document the results of the study by describing the results textually and graphically, provide discussion and recommend future research.

Genetic algorithm GA-FGM for solving TSPs

For solving several stated travelling salesman problems (TSPs), the genetic algorithm GA-FGM (acronym of Genetic Algorithm for Fruška Gora Monasteries) is developed with pseudo-code shown in Fig. 2.

Starting point in development of genetic algorithm was to adopt encoding scheme for the problem in hand, that is to define representation of network of nodes (vertices) and connecting links (edges) in the TSPs. Routes (tours with lengths expressed as distances in kilometres) through a given network of mutually connected monasteries and cities are encoded directly as permutations of integer numbers. In this research there are in total 17 or 20 nodes depending on particular TSP formulation. Therefore, GA- FGM is dimensioned to search any solution space with up to $20! = 2.43291298 \times 10^{18}$ feasible tours between 18 monasteries and two cities. Tours are considered as chains of travelling segments between nodes (monasteries and cities) and the following search spaces are explored thoroughly by the algorithm:

(A) Orthodromic distances in kilometres

A1: Network with 20 nodes consisting of 16 medieval monasteries at Fruška Gora Mountain + two monasteries in cities near Novi Sad + cities of Belgrade and Novi Sad.

(B) Google Map distances in kilometres /travel by car

B1: The same network as A1, but with Google Maps distances if travel between all nodes is made by car.

B2: The 17 nodes network consisting of 16 medieval monasteries + city of Novi Sad. Travel is to be made by car from Novi Sad to visit only monasteries at the mountain.

Listed solution spaces could be even extended by introducing more nodes, for instance with durations of travels instead of distances in kilometres. As analogous spaces to B1 and B2, data related to travelling by foot in minutes are also available the Google Maps platform. These spaces and several other spaces (such as travel by car to monasteries only from Belgrade) were explored but not described here because of limits in presenting the results. There are possibilities to create other TSPs, for instance to explore distances of tours or their durations if travel is made from Belgrade to monasteries on foot; simply, Belgrade is too far from all monasteries and finding shortest tours has no sense with meaningful results.

Recall that points in solution space explored by GA-FGM are in TSP terminology vertices to be visited exactly once. Because the shortest travelling route in TSP terminology is the shortest tour through a set of *n* vertices, the problem in each presented TSP was to find that shortest tour. Adopted direct encoding scheme in algorithm meant that vertices (monasteries and cities) are genes and tours are individuals or chromosomes. Each element of each tour is monastery or city, that is 'gene', and chain of genes is a chromosome – tour. In other words, in GA-FGM gene is one monastery or city and individual (chromosome) is a chain of genes in given order.

Analogous to biological inheritance, algorithm realizes two evolution operations: (a) crossover with given probability represented by or gene recombination, that is, a single-point exchange of strings of genes from two randomly selected chromosomes considered as parents to be sexually mated; and (b) mutation with very low probability, represented by randomly selected two genes within a single chromosome and exchange of their position. These operations are performed within given generation of chromosomes and uniform recombination pattern is used.

Generating initial and all other generations of chromosomes requires defining number of individuals in each generation. This number is usually low, for instance 4 or 5 as we used it as parameter for running the algorithm. In the beginning, computer generates multiple random numbers uniformly distributed within interval (0,1), based on which (by sampling procedure), the initial set of chromosomes is created

Forum geografic. Studii și cercetări de geografie și protecția mediului
Volume **XX**, Issue **1** (**June 2021**), pp. **104-117**
http://dx.doi.org/**10.5775/fg.2021.084.i**

in generation #1. This generation of individuals is a matting pool from which process of creation of new generations will evolve.

Based on programmed strategy, computer randomly selects two chromosomes, consider them as parents and perform crossover and/or eventually perform mutation operation. Crossover always performs between two parents picked up from generation by tournament selection procedure. By single point random cut at parents' genes, their segments of genes are swapped and two offspring are created. If some gen(s) are repeated within any offspring, circular inspection from left to right replaces duplicated genes with missing ones. Mutation occurs only with very low probability at randomly selected individual; two genes in this individual simply change their positions in a permutation. Mutation operation does not apply at individual selected earlier via elitist reproduction. If randomly computed probabilities for performing crossover or mutation operations are outside given boundaries, one (randomly selected) parent will be copied into the next generation; heuristic applied is that at least one recombination of genes must be performed in one generation.

Each chromosome in generation has its own fitness, here represented as the total length of a tour (one feasible point in discrete space of all possible tours). Based on a survival principle, better chromosomes (shorter tours) will survive and will be propagated as members of the next generation. The average fitness of generation is a measure used to compare any two adjacent generations, but also for occasional checks if algorithm converged into the local optimum. To escape from possible traps (local optima), only the-best-so-far chromosome will survive (heuristic known as elitist reproduction) and other chromosomes will be eliminated. The new generation of possible solutions will be created with elite chromosome in it, and this strategy obviously corresponds to random jumps within solution space in search for better generations of better chromosomes (tours).

Recall that any tour is a closed loop meaning that its starting point (gene) is also its end point. In GA-FGM fitness of given tour is its length along which traveller will visit monastery by monastery or city. In other words, fitness of a tour is a sum of distances (orthodromic or as downloaded from Google Maps) between genes in a permutation of monasteries and cities. Sum includes distance between the last and the first gene because the tour is a closed loop. Shorter tours by definition have better fitness and therefore are better candidates to be picked up for mating.

Algorithm imitates natural process of selection by applying so called tournament selection; note that in this study, the algorithm did not use imbedded options for applying selection strategies known as 'roulette wheel' and 'rank based selection'. The tournament selection strategy is performed in the following way (pseudo code):

1. choose d (the tournament size) individuals from the population at random;

2. choose individual with the best fitness from the tournament with probability p;

3. choose the second best individual with probability p(1-p);

4. choose the third best individual with probability p(1-p)(1-p), and so on until mating poll is filled up.

In GA-FGM applications tournament selection is performed with tournament size $d = 1$ and probability $p = 0.7$). Selection is in fact equivalent to random walk strategy and, in a way stochastic noise is avoided (which can be a feature of intelligent stochastic algorithms).

```
Parameter setting:

  -  Search space for loops = indefinite
  -  Problem size  = 20 individuals (max)
  -  Encoding = real-valued scheme, floating point distances between genes (locations); number of genes in each
     individual is 16, 17, or 19, depending on number of monasteries, and cities includes); real values of genes in one
     individual sum to length of the loop.
  -  Population size = 4 or 5
  -  Number of offspring per pair of parents = 1
  -  Crossover =  used (probability 0,7)
  -  Mutation probability = used (0.004)
  -  Elitist reproduction = yes
  -  Worst individual elimination = yes (with probability less than 0.1)
  -  Selection = tournament
  -  Perturbation = shuffle
  -  Number of generations = arbitrary (max to be defined)
Pseudo Code:

  BEGIN       [ GENETIC ALGORITHM]

                                       [ Input and parameter setting]

       c : = 0;  [ Initialization. Start with an initial value of counter and

       Init   population P (c);        [ Initialization  initialize at random population of individuals. ]

       Eval population P (c);          [ Evaluation. Evaluate fitness of all individuals of initial population by
                                          summing distances between genes ]

       WHILE NOT finished DO           [ Test for termination criterion (Max. no. of generations) ]

         BEGIN

            c := c + 1;                [ Population counter. Increase generation counter]

            P' := select parents P(c); [ 'Mating'. Select stochastically two parents for two offspring
                                          production. ]

            mutate P'(c);              [ Mutation. For randomly selected individual in generation, two
                                          different genes are selected at random. These genes change
                                          places to generate mutated individual. ]

            perturb P'(c);             [ Perturbation. Shuffle stochastically individuals in the population. ]

            evaluate P'(c);            [ Evaluation. Compute fitness of all individuals in the generation
                                          (including offspring) ]

            P : = survive P, P'(c);    [ New generation. Select the survivors by using actual fitness, and
                                          do Elitism, i.e. copy the best individual into the next generation;
                                          Delete worst individual . ]

         END

  END

Output:

The best individual, i.e. the shortest distance between all genes.
```

Fig. 2: Pseudo code for GA-FGM genetic algorithm

Elitist reproduction strategy is implemented in a way that best chromosome in given generation is automatically copied into the next one, while the worst chromosome is eliminated with given probability q (in our applications $q = 0.1$) before mating starts; if it happen, eliminated chromosome is replaced with randomly generated new one. Elitism means that the best-so-far solution cannot be lost. Transition principle is preserved until the moment when better solution appears. Then, this solution becomes the elite solution and is propagated forward to the new generation. And so on, until solution process is finished. Algorithm ends after given number of iterations or if solution process converges after pre-specified number of consecutive generations with the same average fitness.

Pseudo code of GA-FGM algorithm is shown on Fig. 2. Its source code in FORTRAN programming language has 1,467 lines. On standard PC platform its runs are completed in up to 5 seconds.

Data, results and discussion

The following two categories of data are downloaded from the Internet for medieval monasteries at the Fruška Gora Mountain, two monasteries near Novi Sad, and for cities Belgrade and Novi Sad:

1. GPS (on Earth spherical) coordinates given as decimal degrees of latitudes and longitudes;
2. Google Maps distances in kilometres for travelling by car across national, regional and local motorways (as suggested by Google Maps).

The distances based on GPS data are computed by the Haversine formula.

Distances between monasteries and cities

Orthodromic distances

There are different formulas for finding the shortest distance between points on a sphere. This distance, measured along the surface of the sphere (as opposed to a straight line through the sphere's interior), is called the *orthodromic distance* or the *great-circle distance*. Recall that the distance between two points in Euclidean space is the length of a straight line between them. Differently, on the sphere there are no straight lines and in such space there are only curvature straight lines which are replaced by geodesics. Geodesics on the sphere are circles whose centres coincide with the centre of the sphere, and are called *great circles*.

Through any two points on a sphere that are not directly opposite each other, such as P and G in Fig. 3, there is a unique great circle which is separated by two points into two arcs. The length of the shorter arc is the great-circle distance between the points; in Fig. 3 this is represented by red segment of a circle u-v.

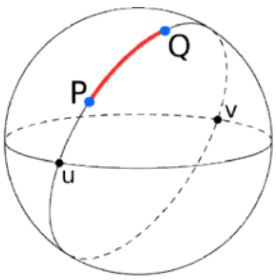

Fig. 3: Sphere and orthodromic (great-circle) distancebetween two points

If the sphere is Earth and two points are given by their sphere coordinates latitudes and longitudes in decimal degrees, to compute distance between points commonly used is the Haversine formula given as an Eq. (1).

$$D = 3963 \cdot \text{arc cos} \, [(\sin(lat1) \cdot \sin(lat2) + \cos(lat1) \cdot \cos(lat1) \cdot \cos(long2 - long1)] \; (km) \qquad (1)$$

The first step in applying Eq. /1) is to convert the latitude and longitude values from decimal degrees to radians. To do this the values of longitude and latitude of points P (latitude1, longitude1) and Q (latitude2, longitude2) should be divided by 180/which is approximately 57.29577951. The result of division are coordinates of points P and Q in radians, that is P (lat1, long1) and Q (lat2, long2). The distance in kilometers between points is obtained if coordinates (in radians) are inserted in Eq. 1.

Example:

For Belgrade (capital of Serbia) and Novi Sad (capital of Vojvodina Province in the northern part of Serbia), co-ordinates in decimal degrees are:

Belgrade: 44.6866 ^{0}N (latitude1)
20.4489 ^{0}E (longitude1)
Novi Sad: 45.2671 ^{0}N (latitude2)
19.8335 ^{0}E (longitude2)

After conversion into radians these values are:

Belgrade: 0.77993 (lat1) 0.35690 (long1)
Novi Sad: 0.79006 (lat2) 0.34616 (long2)

and the distance computed by Eq. 1 is:

$D_{BG\text{-}NS}$ = 80.68 km.

Google Maps distances

Google Maps service at the Internet offers many opportunities to find a route between selected points on the Earth, mapped in standard geographical way. Up to ten different points of interest can be defined and suggested route by car, train, or by walk can be suggested as the best with distance in kilometers and time required to travel. Google Maps does not offer optimization of ordering of points on the route. For our purpose it was enough to define two points on the map, for instance cities Belgrade and Novi Sad, or two monasteries, or city and monastery, and Google Maps offered the best routes by distance and time for various means of transport; for our study only distances for travelling by car were of interest. To be more precise, several alternative routes are commonly offered by Google maps and people usually select most desired or shortest when travelling around.

This straight-arrow spherical distance of 80.58 km between Belgrade and Novi Sad can be compared with the ground optimal route distance of 93.70 km if travelling is made by car via highway E-75 (Belgrade – Budapest) and local roads. According to Google

Forum geografic. Studii și cercetări de geografie și protecția mediului
Volume **XX**, Issue **1** (**June 2021**), pp. **104-117**
http://dx.doi.org/**10.5775/fg.2021.084.i**

Maps (see Fig. 4), increase in distance is approximately 16%.

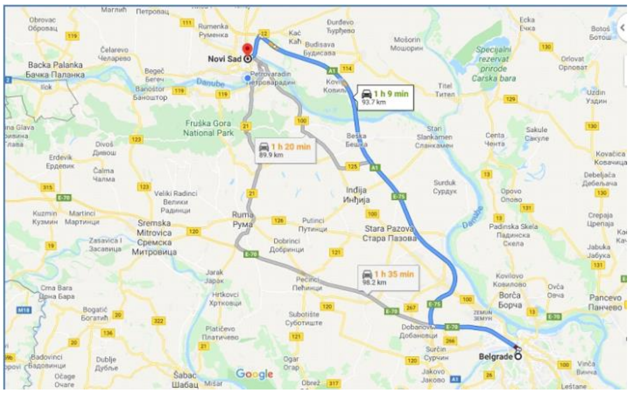

Fig. 4: Distance between Belgrade and Novi Sad *(source: Google Maps)*

The TSP is formulated to consist of 20 nodes: 16 monasteries at Fruška Gora Mountain + Sremski Karlovci monastery + Kovilj monastery + city of Novi Sad + city of Belgrade. By using spherical distances between points as computed by Eq. (1) described in sub-section 5.2, genetic algorithm identified the shortest tour of 251.31 km connecting nodes in the following order (Fig. 5):

20 – 18 – 17 – 19 – 1 – 2 – 12 – 13 – 14 – 16 – 15 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 20

Starting from Belgrade, tour continues straight to Kovilj monastery (node 18), makes left turn to Sremski Karlovci monastery (19) and reach the city of Novi Sad (19). From Novi Sad, closest are two neighbour monasteries (1 and 2) to the south, across the Danube River. What follow is a visit to the group of five medieval monasteries on the west (12, 13, 14, 16 and 15). The final part of tour is visit to the group of monasteries on the east (3 through 11) and return to Belgrade.

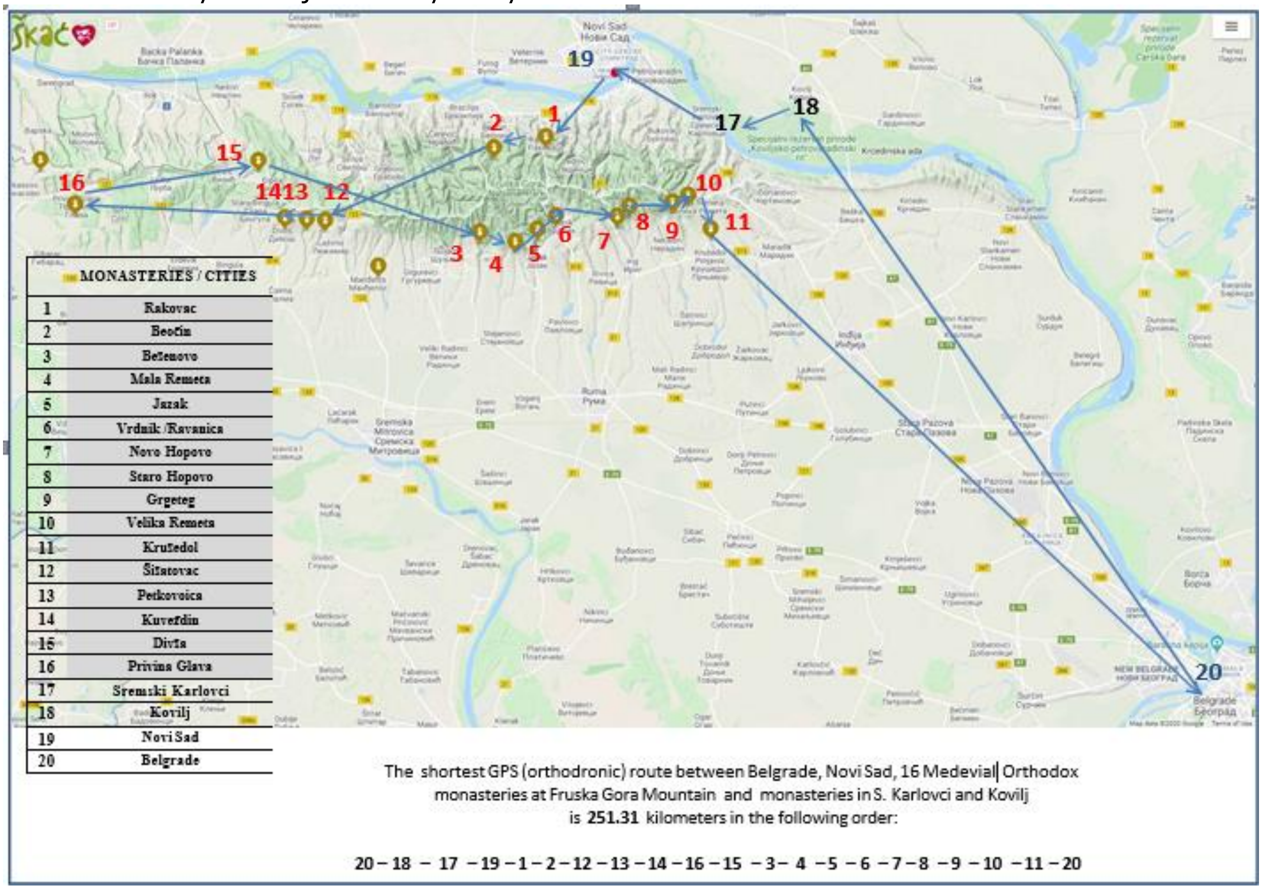

Fig. 5: The shortest GPS (orthodromic) tour between 20 points

Solution space in this case consisted of 20! = 2.4329020×10^{18} possible permutations of 20 integer numbers. Note that this number is with eighteen zeros (billion of billions). The best solution is found after 1,061 iterations in 4 seconds; to illustrate improvement of solution during GA-FGM run, several iterations are presented in Table 1. Close to global optimum is the solution obtained after 480 generations. Length of this tour is 252.12 km, which differs from the shortest one (251.31 km) for only 810 m. Heavy exploration of narrowed local solution space was necessary to handle the part of a tour between monasteries on the west with very small distances.

Table 1: Partial output results for 20 points TSP problem (GPS) Identification of locations to be visited;

Monasteris (1-16) + S. Karlovci (17)+Kovilj (18) + Novi Sad (19) + Belgrade (20)

*** Generation # 1	*** Generation # 41
Best tour is: 20 10 1 8 17 15 3 5 2 13 16 12 7 6 9 18 19 14 4 11	Best tour is: 20 8 18 19 14 15 16 12 13 3 4 6 7 2 5 1 17 10 9 11
Tour length: 409.28	Tour length: 310.36
*** Generation # 260	*** Generation # 326
Best tour is: 20 18 17 19 1 13 14 16 15 12 3 4 5 6 2 7 8 9 10 11	Best tour is: 20 18 17 19 2 1 13 14 16 15 12 3 4 5 6 7 8 9 10 11
Tour length: 265.69	Tour length: 259.31
*** Generation # 480	*** Generation # 1061
Best tour is: 20 18 17 19 1 2 13 14 16 15 12 3 4 5 6 7 8 9 10 11	Best tour is: 20 18 17 19 1 2 12 13 14 16 15 3 4 5 6 7 8 9 10 11
Tour length: 252.12	Tour length: 251.31

Shortest Google Maps tour between 20 points

With the distances across roads in the region, as downloaded from the Google Maps, genetic algorithm identified the shortest tour of 348.2 km connecting nodes in the following order:

20 – 18 – 17 – 10 – 11 – 9 – 8 – 7 – 3 – 12 – 13 – 15 – 16 – 14 – 19 – 2 – 1 – 6 – 5 – 4 – 20

Starting from Belgrade (node 20), this tour is heading straight to node 18 (Kovilj monastery), makes turn to the left to the node 17 (S. Karlovci monastery) and continues towards six of nine eastern medieval monasteries in order: Velika Remeta (10), Kriušedol (11), Grgeteg (10), Staro Hopovo (9), Novo Hopovo (8) and Bešenovo (3). Tour then continues to the 'western group' of monasteries in order: Šišatovac (12), Petkovica (13), Divša (15), Privina Glava (16) and Kuveždin (14). Travel should continue to the city of Novi Sad (19), monasteries Beočin (2) and Rakovac (1) across the Danube river with the final visit to three remaining eastern monasteries Vrdink/Ravanica (6), Jazak (5) and Mala Remeta (4). The final segment of the tour is travel to Belgrade (20).

Solution space in this case has again 2.4329020e+18 (more than two billion of billions) possible tours. The best solution is found after 1,612 iterations. Initially generated solution was the path long 892.5 km. In next 315 generations much better solution is found (370.90 km) and in approximately 400 next generations it is improved to 359.4 km.

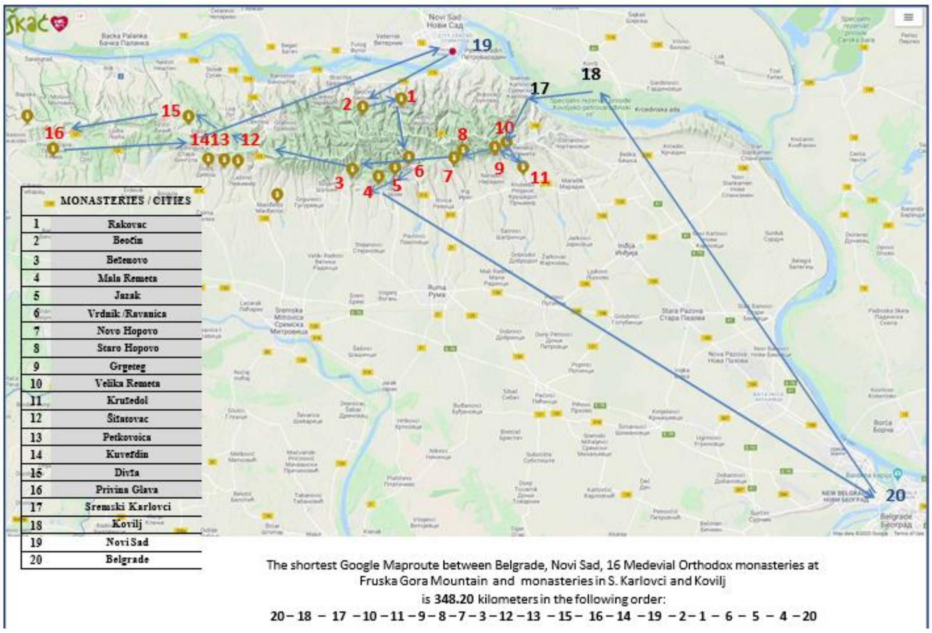

Fig. 6: The shortest Google Maps tour between 20 points

The final search identified the shortest tour with length of 348.2 km. Several runs of the algorithm from different initial solutions, that is, from different starting points in extremely large solution space, did not identify any better solution. In accordance with premise that TSP solution found by approximate

Forum geografic. Studii și cercetări de geografie și protecția mediului
Volume **XX**, Issue **1** (**June 2021**), pp. **104-117**
http://dx.doi.org/**10.5775/fg.2021.084.i**

algorithms can be suboptimal, in this particular case and with possibly very low uncertainty, the distance 348.2 km is adopted as global optimum, that is, the shortest tour from Belgrade to 16 monasteries at the Fruška Gora Mountain, monasteries in S. Karlovci and Kovilj, including visit to the city of Novi Sad.

Shortest Google Maps tour from Novi Sad to 16 medieval monasteries

Genetic algorithm identified the shortest tour of 273.8 km which connects nodes in the following order:

19 – 1 – 2 – 3 – 15 – 16 – 14 – 12 – 13 – 4 – 5 – 6 – 7 – 8 – 9 – 11 – 10 – 19

If travel by car from city of Novi Sad to 16 monasteries (without visit to Belgrade) should follow distances suggested by Google Maps shortest distance of 273.80 km is less for about hundred kilometres than the analogous travel starting in Belgrade (379.4 km), analogously without visiting Novi Sad. The result is expected because monasteries are closer to Novi Sad.

The shortest tour presented in Fig. 7 is global minimum of stated TSP. Interesting to note is that between several western monasteries at first glance seems that better than found best ordering (14 – 12 – 13) could be visually more logical ordering (14 – 13 – 12). Inspection of distances for these two sections shows that identified ordering (14 – 12 – 13) is shorter for 1.7 km. According to Google Maps suggestion for travelling and geographical position of monasteries Šišatovac (12), Petkovica (13) and Kuveždin (14), the order of visits found by GA-FGM is correct, as can be easily clarified by Fig. 8.

To get better visual perspective of travelling by car from city of Novi Sad to visit all 16 monasteries at the Fruška Gora Mountain and get back to the city along the shortest tour found by GA-FGM algorithm (Cf. Fig. 7), two segments of this tour identified by the Google Maps, are presented at Fig 9. The total travelling distance along two segments is 272.5 km and total duration of travelling (without stops!) should be 5 hours and 56 minutes. Note that small difference between distance 273.8 km, computed as shortest tour by algorithm (Fig. 7), and distance of 272.5 along the same route computed by Google Maps (Fig. 9) is due to different reference points for periphery and centre of the city of Novi Sad.

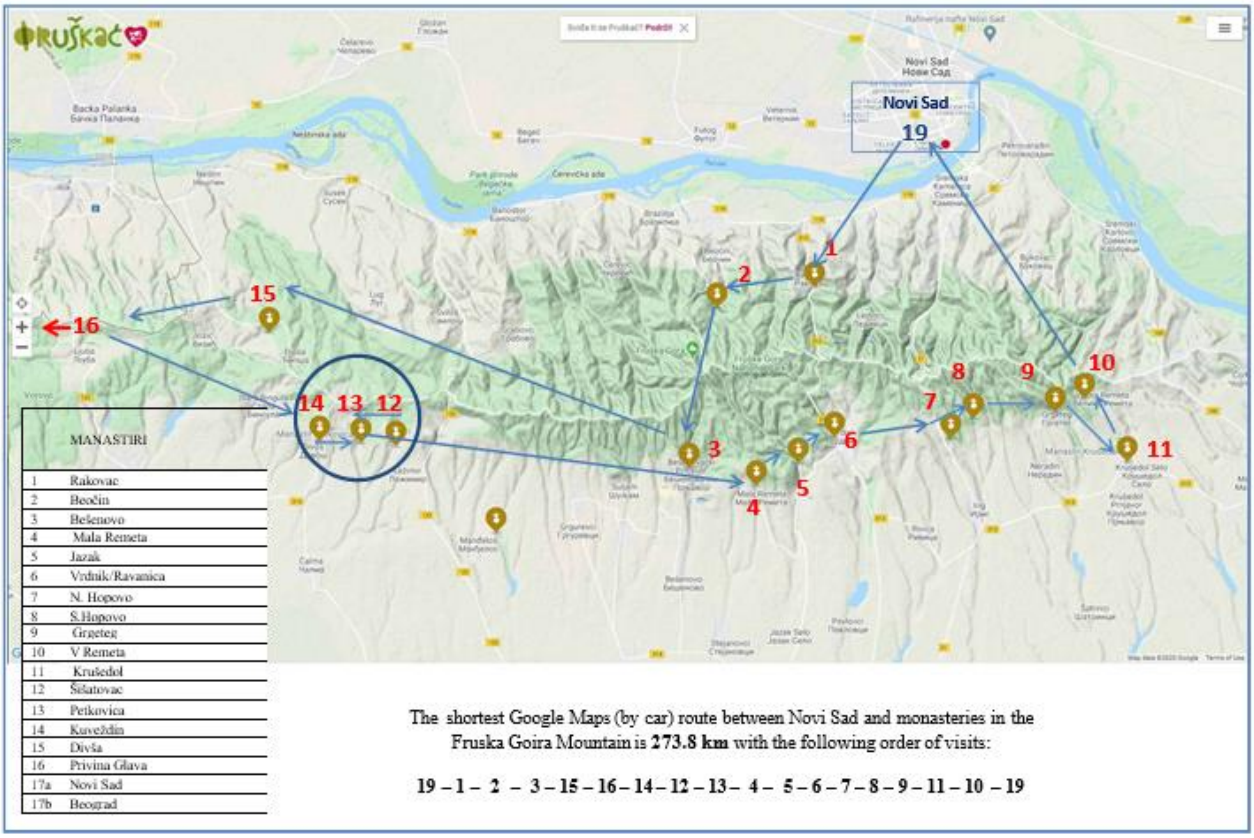

Fig. 7: The shortest (by car) Google Maps tour between Novi Sad and monasteries at Fruška Goira Mountain (source: Google Maps)

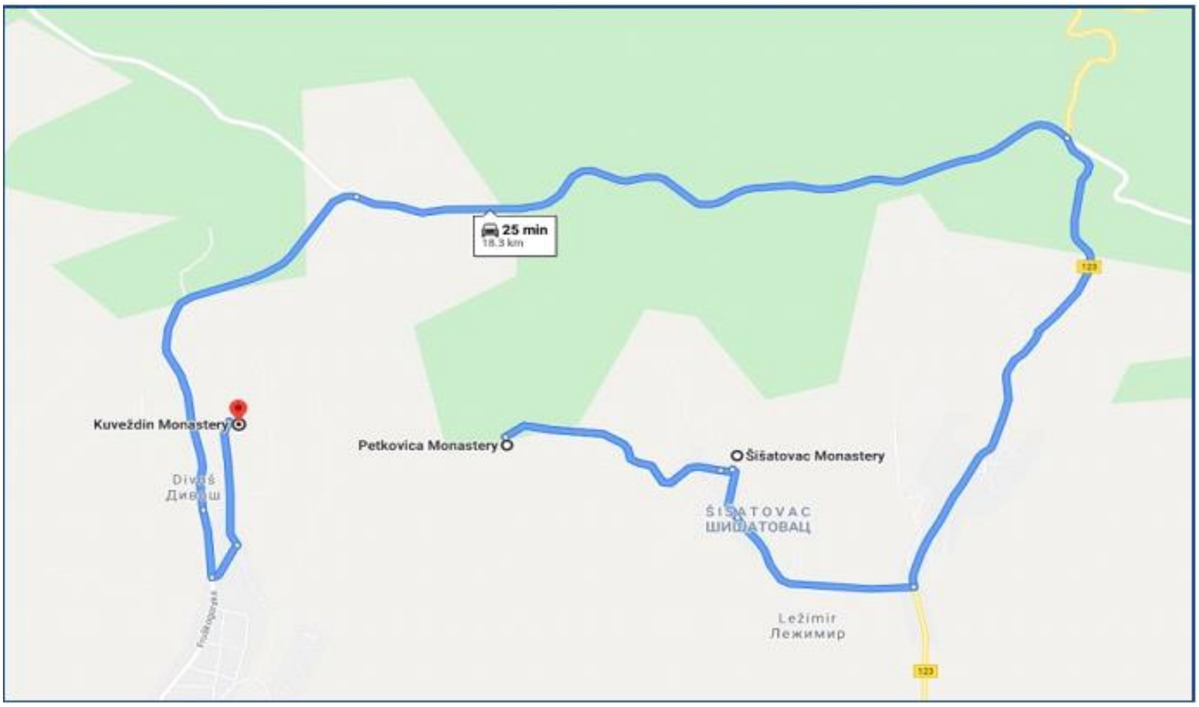

Fig. 8: Geographical position and distances of monasteries Šišatovac (12), Petkovica (13) and Kuveždin (14) (source: Google Maps)

Fig. 9: Shortest route between Novi Sad and 16 monasteries at Fruška Gora Mountain found by GA-FGM algorithm, as suggested by the Google Maps

CONCLUSIONS

The Serbian monasteries known as 16 medieval Orthodox monasteries at Fruška Gora Mountain are important religious and cultural monuments witnessing national heritage for many centuries. These monasteries create unique group of sacral objects of Serbian Orthodox Church worth to be visited starting from Belgrade (capital of Serbia) or Novi Sad (capital of Vojvodina Province), two cities from where tourists and other interested people usually start their tours to interesting and/or historically important sites in the northern part of

Serbia. To this group of monasteries, worth to add were two more monasteries in corridor connecting Belgrade and Novi Sad: (1) Vavadenja Presvete Bogorodice Monastery in the city of Sremski Karlovci, and (2) the Holy Archangels Monastery in the city of Kovilj; these two cities and their monasteries are near Novi Sad and easily reachable from Belgrade.

In this study, several travel options are explored to visit 20 sites by car (16 medieval monasteries at the mountain, two monasteries near Novi Sad, and cities of Novi Sad and Belgrade) following the shortest routes measured in kilometres. Each travel option is treated as a travelling salesman problem (TSP) with given set of sites to be visited. One option was with all 20 sites included, and the shortest possible length has been found for the tour connecting all sites. The GPS data are used and spherical coordinates of sites enabled finding the shortest tour, narrowed across the globe but not achievable in reality. Solution of this TSP may serve as ideal point or target in other related evaluations of real travelling options. Such option was to use distances suggested by Google Maps service at the Internet and solve related TSP with the same number of sites. Two TSPs with all 20 nodes are solved, and the shortest GPS tour found was 251.31 km versus Google Maps tour of length 348.20 km. The difference of approximately hundred kilometres (38.6%) means that spherical solution, considered as ideal target, although not achievable, is good to know for any future actions in (re)constructing transport infrastructure according to existing topography.

The third TSP included only monasteries at Fruška Gora Mountain and visit starting from Novi Sad (without visiting Belgrade). Corresponding network of TSP had 17 nodes (16 monasteries and the city), and the shortest tour is found by using the distance data suggested by the Google Maps Service for best available national, regional and local routes.

Solved travelling salesman problems with 17 and 20 nodal points in three presented cases are considered with great probability as global optima, although corresponding search spaces of possible tours are of sizes up to 1018 order. Best solutions are obtained after multiple runs of computer program which realizes genetic algorithm and applies many different heuristics. For instance, different crossover probabilities are used to enable escapes from local traps if solutions tend to be local optima. Elitist reproduction strategy is also consequently realized and reasonable mating pools of four parents are used for creating offspring in each reproduction phase. An important heuristic was also to assure random jumps to different regions of solution space followed by intensive exploration of local solutions in fully parallel processing environment.

Collected and computed GPS and Google Maps data and presented heuristic approach to exploration of travelling option offer many opportunities in planning tourist and/or religious tours, or recreational tours such as family excursions, marathon events and other outdoor activities. ‘The results of this study could be useful for different stakeholders. Shortest routes identified in this research offer choices such as defining the shorter routes that will include only selected number of monasteries to visit in one day from Novi Sad or Belgrade. This is obviously realistic option for any individual or group of tourists. By tracing the route along the shortest one identified in our study, and by inserting it into the Google Maps (which permits only ten locations to include), it is easy to determine how much time is required for travelling. By adding stopping times for visiting monasteries and other sightseeing along the route one can plan the whole travelling event. To visit all 20 sights (cities and monasteries) in one turn is obviously too much for any tourist, no matter how keen he/she (or group) is. One of possible options is to combine several pieces of complete route and just follow the indications of shortest global (complete) route. This is in accordance with one of the basic principles when conceiving any tourist route: not to visit too many attractions in one day or in several consecutive days. Tourists usually make a selection of the flagship attractions and our study can be considered as a service on how to choose and make relaxing, interesting and reasonable time consuming visit to medieval Serbian monasteries in Fruška Gora mountain. Keeping in mind that tourists may have different interests we are not proposing shorter tours and allocating information on times required. We also decided not to include information about the time it would take to make a complete shortest tour because there are differences in handling spatial and temporal distances. Obviously it is an interesting topic for advanced studies and setting up an agenda for future research. Not used in this study, there are also data related to distances and durations for on-foot and by-car travelling means that can be used for geographical, historical and other studies in different research agendas.

Acknowledgements

This research was supported by the Ministry of Education, Science and Technological Development of Republic of Serbia (grant 451-03-68/2020-14/ 200117).

References

Aarts, EH. L., Korst, J. H. M., & van Laarhoven, P.J.M. (1988). A quantitative analysis of the simulated annealing algorithm: A case study for the traveling salesman problem, *J. Statist. Phys*. 50, 189-206. https://doi.org/10.1007/BF01022991

Blagojevic, B., Srdjevic, B., Srdjevic, Z., & Zoranovic, T. (2016) Heuristic aggregation of individual judgments in AHP group decision making using simulated annealing algorithm. *Information Sciences* 330: 260-273, https://doi.org/10.1016/j.ins.2015.10.033

Bolaños, R. I., Echeverrya, M. G., & Escobar, J. W. (2015). A multiobjective non-dominated sorting genetic algorithm (NSGA-II) for the multiple traveling salesman problem. *Decision Science Letters* 4, 559–568. https://doi.org/10.5267/j.dsl.2015.5.003

Bonabeau, E., Dorigo, M., & Theraulaz, G. (1999), Swarm Intelligence: From Natural to Artificial Systems, *Oxford Univ. Press*, New York

Brady, R. M. (1985). Optimization strategies gleaned from biological evolution, *Nature* 317, 804 - 806. https://doi.org/10.1038/317804a0

Bukvić, D. (2017). Pod krilima orla krstaša, *Politika* ,24 July 2017, pp. 8 (in Serbian).

Cheng, J. R., & Gen, M. (2019) Accelerating genetic algorithms with GPU computing: A selective overview, *Computers & Industrial Engineering* 128, February 2019, 514-525

Dorigo, M., & Stützle, T. (2019) Ant Colony Optimization: Overview and Recent Advances. In: Gendreau M., Potvin JY. (eds) Handbook of Metaheuristics. *International Series in Operations Research & Management Science*, vol 272. Springer, Cham. https://doi.org/10.1007/978-3-319-91086-4_10

Dorigo, M., & Gambardella, L. M. (1997), Ant Colonies for the Traveling Salesman Problem, *BioSystems* 43, 73-81. https://doi.org/10.1016/s0303-2647(97)01708-5

Dorigo, M., Maniezzo, V., & Colorni, A. (1996), The Ant System: Optimization by a Colony of Cooperating Agents, *IEEE Trans. Syst. Man Cybern*. B 6, 29-41. https://doi.org/10.1109/3477.484436

Glover, F. (1989). Tabu Search — Part I. *ORSA Journal on Computing* 1 (3), 190-206. https://doi.org/10.1287/ijoc.1.3.190

Glover, F. (1990). Tabu Search — Part II. *ORSA Journal on Computing* 2 (1), 14-32. https://doi.org/10.1287/ijoc.2.1.4

Glover, F. & Laguna, M. (1997), Tabu Search. *Kluwer Academic Publishers*, Boston Norwell, MA. USA, pp. 263-282. https://doi.org/10.1007/978-1-4613-0303-9_33

Goldberg, D.E. (1989). Genetic algorithm in search, optimization and machine learning. *Addison-Wesley*

He, Y-C., Wang, X-Z., He, Y-L., Zhao, S-L., & Li, W-B. (2016) Exact and approximate algorithms for discounted {0-1} knapsack problem , *Information Sciences* 369, 634-647 http://dx.doi.org/10.1016/j.ins.2016.07.037 0020-0255

Holland, J. (1975). Adaptation in natural and artificial systems. *Ann Arbor*: University of Michigan Press

Jahandideh-Tehrani, M., Bozorg-Haddad, O. & Loáiciga, H.A. (2020) Application of particle swarm optimization to water management: an introduction and overview. *Environ Monit Assess* (2020) 192:281. https://doi.org/10.1007/s10661-020-8228-z

Jain, L., & Singh, G.D. (2013). A review: Meta - heuristic approaches for solving rectangle packing problem. *International Journal of Computer Engineering & Technology (IJCET)* 4 (2), 410-424

Kovačević, S. (2017). "Fruška sveta gora nije u turističkoj ponudi Srbije", *Politika*, p. 8 (in Serbian),

Laporte, G. (1992). The traveling salesman problem; An overview of exact and approximate algorithms. *European Journal of Operations Research* 59, 231-247. https://doi.org/10.1016/0377-2217(92)90138-Y

Laarhoven, P. van & Aarts, E.H.L. (1987). Simulated Annealing: Theory and Applications. *Kluwer Academic* https://doi.org/10.1007/978-94-015-7744-1

Lawler, E. L., Lenstra, J. K., Rinnooy Kan, A.H.G., & Shmoys, D. B. (1985). The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization. *Wiley*

Lin, S. & Kernighan B.W. (1973). An affective heuristic algorithm for the traveling-salesman problem. *Operations Research* 21(2), 498-516

Muhlenbein, H., Gorges-Schleuter, M., & Kramer, O. (1988). Evolution algorithms in combinatorial optimization, *Parallel computing* 7, 65-85 https://doi.org/10.1016/0167-8191(88)90098-1

Padberg, M. & Rinaldi, G. (1991). A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems, *Society for Industrial and Applied Mathematics*, SIAM Review 33 (1), 60-100. https://doi.org/10.1137/1033004

Potvin, J-Y. (1996). Genetic algorithms for the traveling salesman problem, *Annals of Operations research* 63, 339-370 https://doi.org/10.1007/BF02125403

Srđević, B. (2019). Heuristic approach in determining shortest routes to ramsar protected natural reserve areas in Serbia. *Annals of Agronomy* 43(2): 126-136. (in Serbian)

Srđević, B., & Srđević, Z. (2016). Water resources systems analysis with applications in water management. Book, *Faculty of Agriculture*, University of Novi Sad, Serbia (In Serbian)

Srdjevic, B. (2003). Ant colonies and combinatorial optimization. *Annals of Scientific Work* 27 (1), 128-137 (in Serbian)

Taiwo, O.S., Mayowa, O.O., & Ruka, K.B. (2013). Application of genetic algorithm to solve traveling salesman problem, *International Journal of Advance Research* 1(4), 27-46

Trišić, I. (2020). The goals of the sustainable tourism development and management in the Fruška Gora national park, *Turističko poslovanje* 22, 5-17. (in Serbian). https://doi.org/10.5937/TurPos1822005T

Vujko, A., Plavša, J., Petrović, M.D., Radovanović, M. & Gajić, T. (2017). Modelling of carrying capacity in National Park - Fruška Gora (Serbia) case study. *Open Geosci*. 2017; 9:61–72 https://doi.org/10.1515/geo-2017-0005

Škrbić, I., & Milošević, S. (2012), Attractiveness of the complex tourist offer of monasteries and villages of Fruška Gora from the aspect of potential tourists. *TIMS Acta* 6, 73-80 (in Serbian) https://doi.org/10.5937/timsact1202073S

Yang, X.-S., Chien, S. F., & Ting, T. O. (2015). Bio-Inspired Computation and Optimization. *Bio-Inspired Computation in Telecommunications*, 1–21. doi:10.1016/b978-0-12-801538-4.00001-x

Internet sources: Google maps. Wikipedia, etc.