Balanced Allocation of Multi-criteria Geographic Areas by a Genetic Algorithm

Shahin Sharifi Noorian and Christian E. Murphy

Abstract The balanced partitioning of geographic space into regions is a common problem. This Territory Design Problem (TDP) of assigning smaller areas to larger regions with equal potential is a task mainly done manually. Therefore, the result becomes subjective and provides only a roughly approximated balanced result. This work presents an automated allocation of independent areas to regions using the Genetic Algorithm (GA), which finds an optimally balanced configuration of regions based on multiple criteria. Thereby, spatial constraints are fully respected as (1) all areas remain contiguous within a region and (2) the automated allocation facilitates a compact region shape. The developed algorithm was tested on a case study in the field of sales territory planning. The target of sales territory planning is the optimal distribution of balanced and fair sales areas based on market potentials. Results of our case study demonstrate the effectiveness of our proposed technique to find an optimal structure of sales territories in a reasonable time. The distance that salesmen need to travel is 16% lower than the existing sales territory configuration. This means that the regions are more balanced and more compact. Due to the independent nature of the GA, this method demonstrates a high flexibility to the optimization problem. It can be easily altered to any objective in territory planning as well as to familiar multi-criteria spatial allocation problems in other disciplines.

Keywords Territory design problem · Combinatorial optimization · Graph partitioning · Genetic algorithm

S. Sharifi Noorian (✉) · C.E. Murphy
Chair of Cartography, Technische Universität München, Arcisstraße 21, 80333 Munich, Germany
e-mail: shahin.sharifi@tum.de

C.E. Murphy
e-mail: christian.murphy@tum.de

M.P. Peterson (ed.), *Advances in Cartography and GIScience*, Lecture Notes in Geoinformation and Cartography, DOI 10.1007/978-3-319-57336-6_29

1 Introduction

Spatial aggregation is a procedure for grouping discrete geographical units (such as zip-code areas, municipality area, company trading areas and electoral districts) into larger geographic clusters called territories in such a way that those clusters are eligible according to the planning criteria and satisfy the conditions (Drexl and Haase 1999).

Territory Design is the aggregation of basic geographical units into larger geographic regions. Territory Design Problems (TDPs) are motivated by a broad range of applications such as administrative segmentation, design of school territories, social facilities, emergency services, sales and service territory design. Businesses with direct sales departments need sales and service territories, as well-planned decisions enable an efficient market penetration and lead to decreased costs and improved customer service. For instance, a beverage manufacturer will deploy a sales force in order to promote its products for retailers and supermarkets. This planning will be effective when some kind of balance is maintained between the workload of sales representatives, which involves the distance that representatives need to travel, and the number of customers that they have to visit.

When some kind of balancing among the territories is desired, it is very difficult to generate an acceptable solution by manual adjustment and, in some cases, it is almost impossible. Besides the balancing of territories in terms of workload or the number of customers, contiguity and compactness should be considered as other core criteria for the TDP. Therefore, an automatic algorithm is required to handle all criteria simultaneously and resolve the complexity of the TDP.

In recent years, a number of TDP heuristic and non-heuristic approaches have appeared in the literature. For some examples, Drexl and Haase (1999) proposed a non-linear mixed-integer programming in order to solve the sales force deployment problem. Kalcsics et al. (2005) utilized computational geometry techniques for solving districting problems, which are suitable for interactive use. Some others suggested heuristic approaches for TDPs such as GRASP (Ríos-Mercado and Fernández 2009), Simulated Annealing (D'Amico and Fernández 2002), and Tabu Search (Blais et al. 2003).

These approaches are mostly applied to relatively small to medium-sized problems in terms of space. In addition, these approaches usually do not take the geographical nature of the TDP into consideration and thus, totally neglect geographic obstacles such as mountain chains, river, etc. On the other hand, territory design is an iterative process which is based on planning parameters and related data. Therefore, the proposed approach must be flexible enough to deal with different kinds of objectives and constraints.

This motivated us to present a generic objective-independent approach that supports core criteria for all TDPs such as compactness, contiguity and balancedness in addition to a customized objective function and multiple problem-specific constraints like organizational limitation, locked basic areas (the basic areas that have to be excluded from planning), etc.

In this paper, we apply the genetic algorithm to the TDP. Considering the fast growing web technologies and immense utilization of software as a services (SaaS) (Gil 2016), we also used techniques such as parallelization of algorithms, simplification of search space, etc., to improve the performance of our method. This prepares a level of interactivity for planning and enables us to publish our method as a web service for a broad range of problems.

2 Territory Design by a Multi-objective Genetic Algorithm

TDP involves combinatorial optimization in which a set of feasible solutions are available. The goal is to find the best solution in a finite collection of solutions. Each feasible solution usually has a concise demonstration. But, the number of solutions is huge. Therefore, an exhaustive search such as checking all objects one-by-one and selecting the best one is not practical. The traveling salesman problem, the vehicle routing problem, and the knapsack problem are some other well-known problems which are classified as combinatorial optimization (Schrijver 2003).

2.1 Usability of Graphs in Order to Simulate the Problem

TDP has multiple objectives as well as several constraints which have to be taken into consideration. Moreover, the scale of the problem changes for every use case. A comprehensive model is needed for representing the problem. Due to the flexibility and scalability of graphs, we model the problem in the form of a graph. The Graph structure allows us to model all kinds of scenarios which are based on relationships. According to the definition of spatial aggregation and the geographical essence of territory design, using graphs for problem modelling is the best choice (Płuciennik and Płuciennik-Psota 2014). Generally in our graph model, each node represents an entity (person, location, product and etc.), and each relationship between nodes represents how two entities are associated.

2.2 Mapping the Territory Design Problem into a Graph Model

The goal of territory design specifies how basic units should be grouped into clusters (Ríos-Mercado and Ferández 2009). Based on the domain of a problem, a certain weight will be assigned to each basic unit that is calculated based on its attributes. As a simple example, the government wants to define a certain number of census territories in the level of zip-codes. Each territory is supposed to have an equal population. Therefore, the number of people in each basic unit indicates its weight. In the graph model, each basic unit (centroid) is represented as a node which can contain one or several attributes. Each node also has a label that specifies the cluster to which it belongs. The model structure is flexible enough to represent all types of spatial relationships between two basic units such as neighbourhood, higher administrative division, distance, etc.

As the territory design is a multi-constraint problem, our graph model is also capable to support multiple constraints (see Fig. 1). Feasible solutions of the problem are defined based on the constraints. For instance, contiguity is a common criteria for all TDPs. It means that only those territories that are not divided into separated parts are acceptable. One can easily check the contiguity constraint by considering the connectivity of the graph. In Fig. 2a, the cluster (blue arrays) is not totally connected and the contiguity constraint is violated. Figure 2b shows a connected generated cluster. This solution is valid.

It is also possible to define certain constraints based on the problem. For example, in some cases, it is necessary to reserve a group of basic areas exclusively for a certain territory. From the graph modelling point of view, the corresponding nodes will be labelled constantly in the initial graph and therefore are excluded from the process of optimization. Figure 3 illustrates how constraints are applied into the graph model.

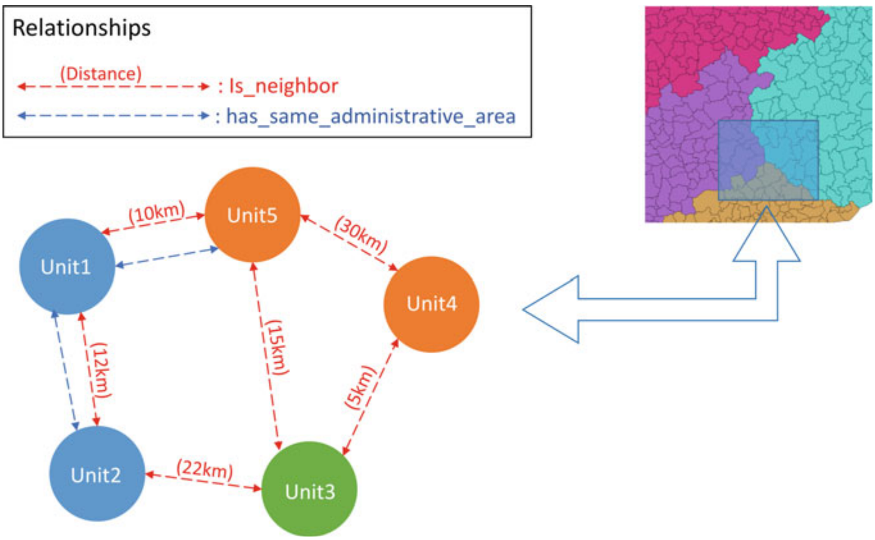

Fig. 1 Proposed scheme for mapping a real-world problem into the graph model

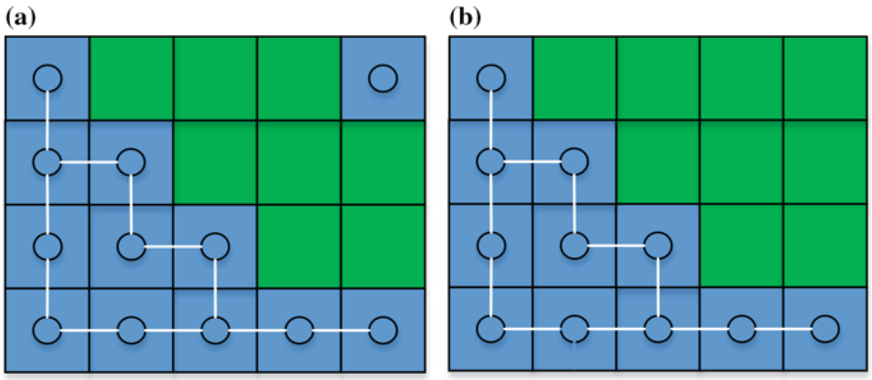

Fig. 2 (**a**) The contiguity constraint for the territory is violated; (**b**) the territory is connected

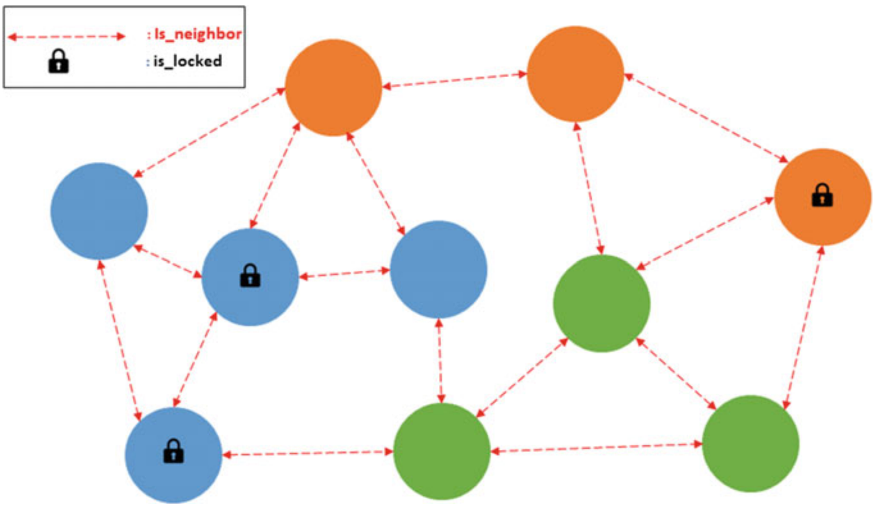

Fig. 3 Graph modelling of constraints: locked nodes are not modified by the algorithm

2.3 *The Core of the GA*

The GA is a heuristic search technique which was inspired from the principal of natural evolution to solve optimization problems. The GA is a subclass of evolutionary algorithms which attempts to mimic some of the processes that occur in natural selection based on "Survival of the fittest" (Sivanandam and Deepa 2007). The GA looks for the best solution among a set of feasible solutions. The set of feasible solutions is called Search Space. It explores the search space and evaluates the fitness of candidate solutions in order to find the best one. The GA has two distinguished elements which are *individuals* and *population*.

An individual is a single candidate solution to the optimization problem while the population is a set of individuals involved in the optimization process. GA applies variation only to the genetic structure of the problem, not the real structure. Therefore, observable properties of the individual (phenotype) should be encoded to the abstract genetic structure (genotype) (Fogel 1995). The genotype is subdivided to some small parts (genes) and mapped according to the decision variables of the problem. The mapping between phenotype and genotype is one of the challenging parts of designing a GA for a real-world application. The GA´s main operations *initialization*, *selection* and *reproduction* are based on a random procedure. In the initialization step, individuals are generated randomly or with some certain consideration for improving the performance of the algorithm and the quality of final solution (Kazimipour et al. 2014). Every individual is assigned a degree of its excellence by means of a *fitness function*. The fitness function is an evaluator that not only shows how good the solution is, but also measures how close our potential solution is to the target. The selection process involves a method for selecting two parent genotypes based on their fitness value. Those selected genotypes are given to the reproduction procedure as input in order to generate a new population. The reproduction step consists of two main operators which are *crossover* and *mutation* (Fogel 1995).

Crossover is a recombination method in which two parent genotypes are cut at a certain point and the halves are connected to each other to make a new candidate solution (offspring) which contains the characteristics of the parents. If parents have some good features, we can expect that all the good features are inherited by the generated offspring. Mutation, as the second reproduction operator, is a way of generating a new offspring which can be a little bit different from its parents (Sivanandam and Deepa 2007). Mutation does not happen frequently in nature, but it is a very important operation in the artificial GA, because the mutation operator helps the algorithm to rapidly explore the search space (Fonseca and Fleming 1995). In Fig. 4, an abstract scheme of the GA is outlined in a pseudo-code fashion.

Algorithm 1 Genetic Algorithm

```
initialize a population with random individuals;
while (termination condition is satisfied) do
    evaluate each individual;
    select fitter individuals and copy into a temporary set;
    while (size of new generation = size of population) do
        select a pair of individuals from temporary pool;
        recombine parent individuals to make an offspring;
        mutate the generated offspring;
        copy the offspring into the next generation;
    end while
end while
return the fittest individual(solution)
```

Fig. 4 Abstract scheme of the GA

Self-adaptive GA:

Local optima is one of the common issues in all GAs. It means that the GA can easily converge to an incorrect solution. Thus, the GA should be capable to escape from a local optimum and to converge to the global optimum. In order to tackle this issue, we propose a self-adaptive GA in which the probability of crossover and mutation, Pc and Pm, are varied depending on the fitness values of the solutions. Fitness assignment of SEA realizes the twin goals of maintaining diversity in the population and guiding the population to the global optimum (Cao et al. 2007). In other words, the mutation probability should be increased when the diversity of candidate solutions is decreasing and should be decreased when the diversity is increasing.

Parallelization of the GA:

GAs operate on a population in which each individual represents a potential solution to the problem. This feature of the GA makes it extremely amenable to parallelization which can significantly improve the efficiency vof the algorithm (Luque and Alba 2011). The GA which is designed in this research project, is capable to be executed concurrently so that the population is divided into smaller populations and each subpopulation is assigned to one processor of the parallel processing system.

3 Case Study: Sales Territory Planning

Sales Territory Planning is considered the problem of grouping smaller area units, so-called sales coverage units (SCUs), into larger geographic clusters called territories in such a way that those clusters are eligible according to the planning criteria and satisfy the conditions. Each territory must have a responsible sales representative. To be responsible means that the sales representative has to provide service for all customers located in the corresponding sales territory (Zoltners and Sinha 1983). For example, a sales representative who is working for a company that provides materials concerning dental surgery will be responsible for all dentists practicing in her sales territory. She is only allowed to sell products in SCUs that belong to her sales territory (Haase and Müller 2014). Reasons for aligning existing or designing new territories can be changes in the number of sales representatives or balancing their workload.

Nowadays, marketers and decision-makers face three different scenarios in sales territory planning. These scenarios are the following (Zoltners et al. 2009):

1. We have neither the location of sales representatives nor the existing sales territory structure. In this case, the number of needed territories should be determined by decision-makers. Then, the algorithm consists of two parts, respectively:

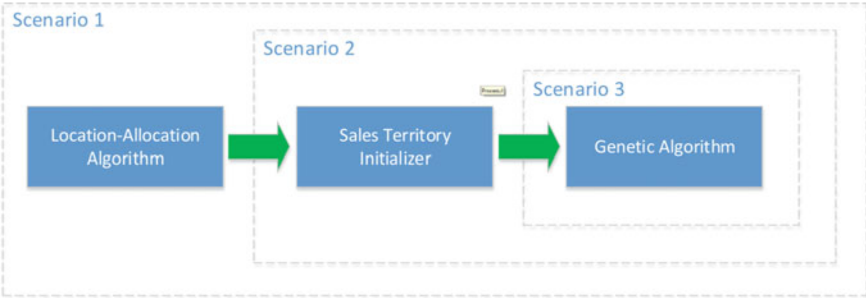

Fig. 5 A generic workflow for sales territory planning

- Finding the best location for each sales representative. The number of sales representatives are equal to the number of territories.
- Performing a heuristic search algorithm for finding an acceptable sales territory structure.

This scenario is commonly called Green Field Planning.

2. We have a number of sales representatives with fixed locations but we do not have any existing territory structure.
3. We have either fixed sales representatives or an existing sales territory structure. But, the territories should be optimized due to the new strategy of the sales organization.

For each scenario, a respective strategy is chosen. But, we are going to design a generic approach for all scenarios. So, we should have a workflow which supports all scenarios. Figure 5 presents how our model handles all scenarios:

For the case study, we considered a real situation in which an agricultural machinery manufacturer is selling a certain product (tractor) in Germany. The sales team consists of 27 representatives who must cover the entire country. Each salesman must be responsible for one territory only. The territory planning is carried out on the level of municipalities (administrative districts), which consist of 11,212 basic areas as SCUs. For this case study, we considered the number of farms inside each coverage unit.

4 Results

To assess the correctness and effectiveness of our method, we performed some experiments for both evaluating the balance of workload among sales representatives and evaluating the travel time improvement.

5 Tuning of the GA Parameters

The Genetic Algorithm is very sensitive to parameters such as population size (N), elitism percentage (E), cross-over probability (Pc) and mutation probability (Pm). The elitism rate identifies the percentage of individuals that should be transferred to the next generation without any changes (Sivanandam and Deepa 2007). After several preliminary experiments, we have tuned the parameters as follows:

- N: 30
- E: 26%
- Pc: [0.6 − 0.95]
- Pm: [0.005 − 0.05]

Since the algorithm is self-adaptive, a range of probabilities are defined for Pc and Pm in which the appropriate probability is chosen depending on the average fitness of the population.

6 Location-Allocation and Initializing the Territories

As we explained in Sect. 3, we have three scenarios. Since the first scenario (Green Field Planning) includes other scenarios, we evaluate here only the Green Field Planning case. We defined three steps for solving the Green Field Planning problem. Firstly, we found the best positions for locating salesmen. In order to create the initial form of territories, we clustered the administrative districts based on the nearest salesman. Then, the location of salesmen and the initial form of territories are used as input for the GA. Since the residence of salesmen is an essential input for sales territory planning, finding the best location is crucial.

In order to avoid unnecessary computational effort, we have selected a subset of municipalities that have more convenience for locating the sales team such as accessibility, and amenities, etc. For choosing the best candidate municipalities, we considered the capital municipalities of higher-level administrative districts (German *Landkreis*). Figure 6 illustrates the candidate municipalities and the connections between them.

After simplification of the dataset, the size of basic units decreases from 11,212 to 402 administrative districts. Similar to the graph model in Sect. 2, each candidate district is shown as a node. In addition, we have considered the highway network for showing the connections between nodes in our graph model. Thus, the highway which connects two districts is shown as an edge which is labelled by distance. We assigned a certain weight to each node by summing up the total travel time from the node to all other nodes. Then, the mapping of the location-allocation graph model

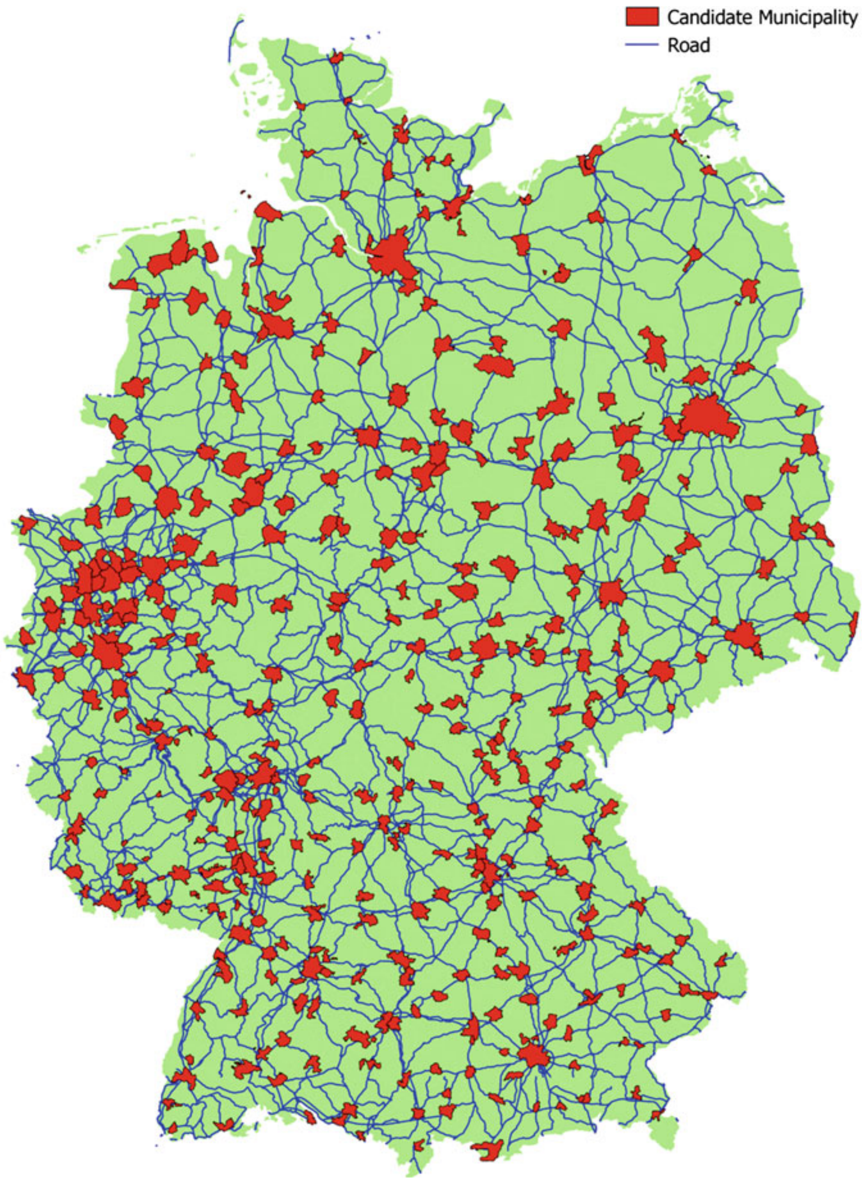

Fig. 6 Location possibilities of the sales teams in the candidate municipalities of Germany

into the genotype is done (see Sect. 2). The objective is to minimize the total travel time (DT) by which those municipalities are chosen that are more accessible. The fitness value for each solution in the GA is calculated as:

$$minimize\, D_T = \sum_{i=1}^{L} \sum_{j=1}^{M} d_{ij} \quad , i \neq j \tag{1}$$

The variables in Eq. (1) are as follows:

- *M: number of candidate municipalities*
- *L: number of needed locations*
- *d_ij: the travel time from municipality i to municipality j*

The best found locations in this step are just used as a starting point. The ultimate locations for salesmen will be determined at the final stage based on the optimal form of territories. Figure 10a, b show the allocated locations for salesmen before and after balancing the workload.

6.1 Evaluating the Balance of Workload

Since balancing the workload among sales representatives (territories) is one of the main objectives in the sales territory planning, we want to prove the efficiency of our algorithm for this objective. Figure 7 shows the unbalanced distribution of workload after generating the initial structure of territories. Since each territory only belongs to one salesman, the ID of salesmen in Fig. 8 represents the belonging territory.

To balance the workload, we consider the standard deviation (σ) as the fitness value of each solution which is to be minimized. The smaller the standard deviation, the more equality will be gained. We formulated the objective function as:

Fig. 7 Distribution of workload deviations from the mean before optimization

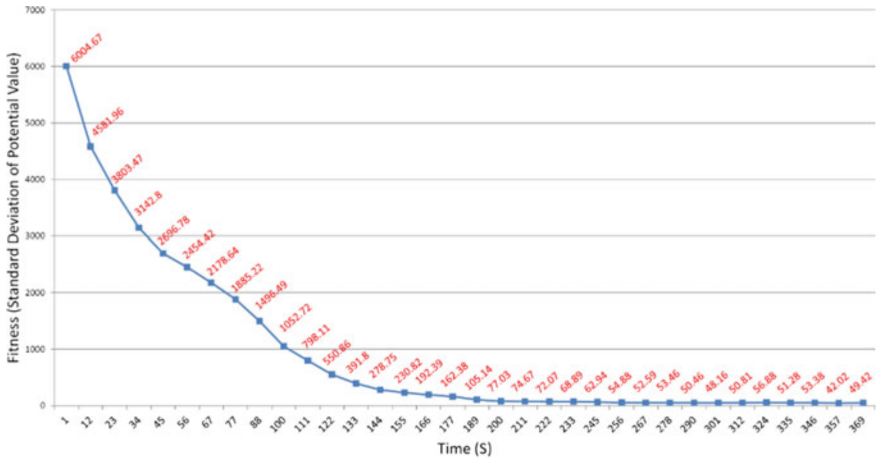

Fig. 8 Fitness progress of the GA

$$minimize\, \sigma = \sum_{i=0}^{T} |R_i - \bar{r}| \tag{2}$$

where,

$$\bar{r} = \frac{1}{T}\sum_{j=1}^{n} r_j, R_i = \sum_{j \in T_i} r_j \quad and\ |R_i - \bar{r}| \le \delta\bar{r}$$

The variables in Eq. (2) are as follows:

- *n: number of basic areas (SCUs)*
- *T: number of territories*
- *T_i: the ith territory*
- *r_j: the workload contained in 〖SCU〗 _j*
- *R_i: the total workload in territory i*
- *δ: the maximum allowable percentage (0 ≤ δ ≤ 1)*

The GA can be terminated in different ways such as elapsed time, generation count, stagnation, target fitness and intentional abort. In this experiment, we have chosen the stagnation strategy which halts the algorithm if no improvement in the average fitness of population is observed within a predetermined number of generations or during an interval of time. The processing of our GA has finished after 369 s, which is a promising computational time for the scale of our problem. Figure 8 represents the progress of our algorithm in which the standard deviation progressively decreases over time until improvement becomes negligible.

As shown in Fig. 8, the standard deviation has decreased from ~6000 to ~50. Figure 9 also illustrates that the workload is almost completely balanced between salesmen.

After balancing the potential between territories, the location of sales representatives are updated according to the new structure of territories. Thus, we run our location-allocation algorithm to find the best location for the sales persons based on the current structure of territories. Figure 10 illustrates the structure of territories as well as the locations of salesmen before and after optimization.

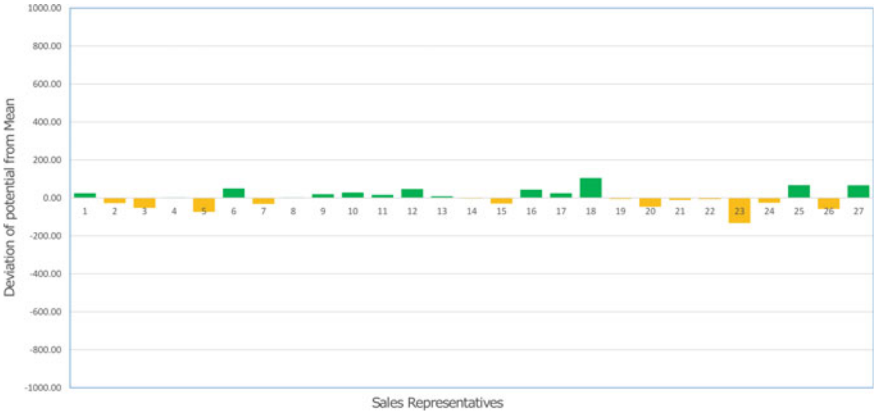

Fig. 9 Distribution of workload deviations from the mean after optimization

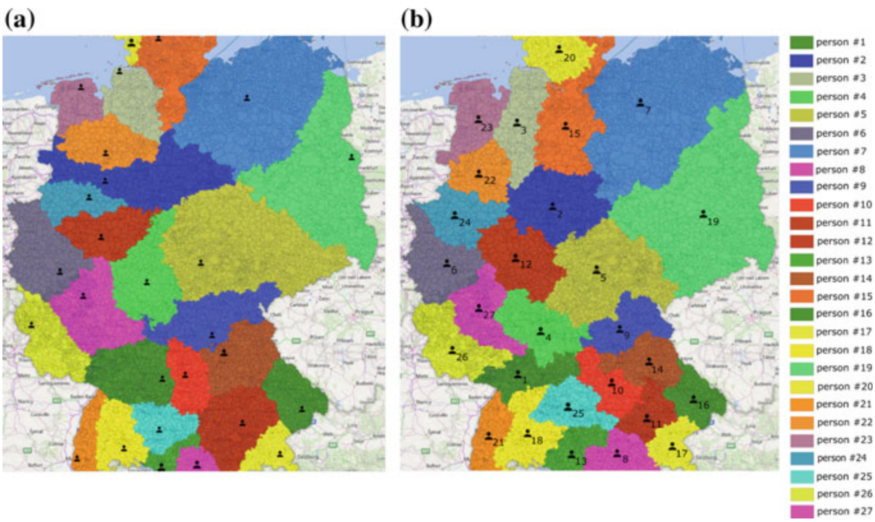

Fig. 10 The shape of sales areas before (**a**) and after (**b**) optimization

6.2 Evaluating the Travel Time Improvement

As the travel cost is one of the important parameters in sales territory planning, we have defined the minimizing of travel time and distance as an objective in our algorithm—besides the balancing of the potential. In our case study, a low travel cost is not the first priority. So we defined a lower weight for minimizing the travel cost than balancing the potential among salesmen.

The calculation of travel cost depends on several parameters and can vary among different sales organization. In our algorithm, we assumed that the salesmen only visit the potential customers of one district in each journey. In addition, we did not

Table 1 Travel time optimization by salesman and in total

Salesman	Travel time before optimization (Hour)	Travel time after optimization (Hour)
Person #1	45.8	112.6
Person #2	2177.9	2767.00
Person #3	2423.4	1775.7
Person #4	359.5	126.8
Person #5	479.6	169.5
Person #6	732.0	525.6
Person #7	212.6	204.2
Person #8	748.8	533.1
Person #9	440.7	256.6
Person #10	1093.2	1086.8
Person #11	137.7	217.8
Person #12	639.6	466.8
Person #13	157.7	140.9
Person #14	1402.2	590.7
Person #15	738.2	631.6
Person #16	124.7	398.6
Person #17	1513.6	464.5
Person #18	467.2	257.2
Person #19	203.07	144.3
Person #20	97.5	148.2
Person #21	24.4	195.2
Person #22	415.04	711.2
Person #23	78.9	83.6
Person #24	201.6	314.6
Person #25	108.4	137.6
Person #26	542.3	481.0
Person #27	164.2	248.2
Total travel time	15730.975	13191.324

take empty districts into account. The total travel cost for each salesman is the sum of all travel costs from the living location to the area (excluding empty ones) belonging to the salesman. Table 1 shows, the total travel time after the optimization process has decreased by around 2,539 h. This is a travel time saving of 16%.

6.3 Evaluating the Contiguity and Compactness

In addition to the balanced workload and the minimum travel cost, we also have two other objectives, compactness and contiguity. This is guaranteed by using a clustering method for initializing the first population of candidate solutions. In order to keep the size of the search space and the performance of the algorithm on a feasible level, we first create a temporary structure of territories which are compact and totally connected, and then we start balancing the potential among salesmen from the borders of territories. In other words, we randomly select some districts on the border of territories and assign them to the neighbour territories. If this operation decreases the standard deviation value, we transfer the new structure of territories as a good solution into the next generation. Moreover, we define operations to detect and reject the invalid solutions to keep the compactness and contiguity of the temporary structure.

7 Conclusions

This work presents a decision-support system for the Territory Design Problem (TDP), the balanced partitioning of geographic areas into regions. This can substitute the subjective, time consuming and mostly less balancing manual allocation. The decision support system has been developed by applying a Genetic Algorithm (GA) in combination with a graph model, and can automatically allocate large numbers of areas to regions, through simultaneous, multi-criteria potential balancing.

Sales territory planning was selected as the case study, as this is one of the most important issues for a commercial organization. An intelligent decision support system for sales territory planning is highly needed. One major benefit of a decision support system is the ability to plan the outcomes of various scenarios before making a real investment. The results of the case study reveal highly balanced regions with contiguous and very compact region shapes. According to our results, locating the sales team based on the distribution of potential, extremely reduces the total calculation time of Green Field Planning. The decision support system proves to be an effective approach for optimizing territories on a large scale.

Due to the independent nature of the GA and the graph model, our method is easily configured to any other area allocating objective. By changing the attributes of nodes and edges in the graph model, applying constraints in the graph model and changing the objective function based on the problem, our method can be used to solve other TDPs.

Acknowledgements This research was supported by the WIGeoGIS GmbH. We wish to thank WIGeoGIS for kindly providing the case study data. We specially thank Michael Steigemann for insightful discussions in the conception of this work.

References

Blais, M., Lapierre, S. D., & Laporte, G. (2003). Solving a home-care districting problem in an urban setting. *The Journal of the Operational Research Society, 54,* 1141–1147.

Cao, R., Li, G., & Wu, Y. (2007). A self-adaptive evolutionary algorithm for multi-objective optimization. In D.-S. Huang, L. Heutte, & M. Loog (Eds.), *Advanced intelligent computing theories and applications with aspects of artificial intelligence* (pp. 553–564). Berlin, Heidelberg: Springer.

D'amico, S. J., Wang, S.–J., & Batta, R., et al. (2002). A simulated annealing approach to police district design. *Computers & Operations Research*, *29*, 667–684.

Drexl, A., & Haase, K. (1999). Fast approximation methods for sales force deployment. *Management Science, 45,* 1307–1323.

Fogel, D. B. (1995). Phenotypes, genotypes, and operators in evolutionary computation. In *Proceedings of 1995 IEEE International Conference on Evolutionary Computation* (Vol. 1).

Fonseca, C. M., & Fleming, P. J. (1995). An overview of evolutionary algorithms in multiobjective optimization. *Evolutionary Computation, 3,* 1–16.

Gil, P. "What is 'SaaS' (Software as a Service)?". Retrieved June 13, 2016 from http://netforbeginners.about.com/od/s/f/what_is_SaaS_software_as_a_service.htm.

Haase, K., & Müller, S. (2014). Upper and lower bounds for the sales force deployment problem with explicit contiguity constraints. *European Journal of Operational Research, 237,* 677–689.

Kalcsics, J., Nickel, S., & Schröder, M. (2005). Towards a unified territorial design approach—Applications, algorithms and GIS integration. *Reports of the Fraunhofer Institute for Industrial Mathematics, 71,* 1–56.

Kazimipour, B., Li, X., & Qin, A. K. (2014). A review of population initialization techniques for evolutionary algorithms. In *2014 IEEE Congress on Evolutionary Computation (CEC)* (pp. 2585–2592).

Luque, G., & Alba, E. (2011). *Parallel genetic algorithms: Theory and real world applications* (Vol. 367). Springer.

Płuciennik, T., & Płuciennik-Psota, E. (2014). Using graph database in spatial data generation. In D. A. Gruca, T. Czachórski, & S. Kozielski (Eds.), *Man-machine interactions* 3 (pp. 643–650). Springer International Publishing.

Ríos-Mercado, R. Z., & Fernández, E. (2009). A reactive GRASP for a commercial territory design problem with multiple balancing requirements. *Computers & Operations Research, 36,* 755–776.

Schrijver, A. (2003). *Combinatorial optimization: Polyhedra and efficiency*. Berlin, Heidelberg: Springer.

Sivanandam, S. N., & Deepa, S. N. (2007). *Introduction to Genetic Algorithms*. Berlin Heidelberg: Springer.

Zoltners, A. A., Sinha, P., & Lorimer, S. E. (2009). *Building a winning sales force: Powerful strategies for driving high performance*. New York: AMACOM A division of American Management Association.

Zoltners, A. A., & Sinha, P. (1983). Sales territory alignment: A review and model. *Management Science, 29,* 1237–1256.