A Versatile Adaptive Aggregation Framework for Spatially Large Discrete Location-Allocation Problems

Matej Cebecauer[a,*], Ľuboš Buzna[b,c]

[a] *Department of Transport Science, KTH Royal Institute of Technology, Teknikringen 10, SE-100 44 Stockholm, Sweden*
[b] *Department of Mathematical Methods and Operations Research, University of Žilina, Univerzitná 8215/1, SK-010 26 Žilina, Slovakia*
[c] *ERA Chair for Intelligent Transport Systems, University of Žilina, Univerzitná 8215/1, SK-010 26 Žilina, Slovakia*

Abstract

We propose a versatile concept of the adaptive aggregation framework for the facility location problems that keeps the problem size in reasonable limits. Most location-allocation problems are known to be NP-hard. Thus, if a problem reaches the critical size, the computation exceeds reasonable time limits, or all computer memory is consumed. Aggregation is a tool that allows for transforming problems into smaller sizes. Usually, it is used only in the data preparation phase, and it leads to the loss of optimality due to aggregation errors. This is particularly remarkable when solving problems with a large number of demand points. The proposed framework embeds the aggregation into the solving process and it iteratively adjusts the aggregation level to the high quality solutions. To explore its versatility, we apply it to the p-median and to the lexicographic minimax problems that lead to structurally different patterns of located facilities. To evaluate the optimality errors, we use benchmarks which can be computed exactly, and to explore the limits of our approach, we study benchmarks reaching 670 000 demand points. Numerical experiments reveal that the adaptive aggregation framework performs well across a large range of problem sizes and is able to provide solutions of higher quality than the state-of-the-art exact methods when applied to the aggregated problem.

Keywords: data aggregation, location analysis, adaptive aggregation, framework, heuristics

1. Introduction

1.1. Motivation

High quality design of public service systems or private supply chains is of significant importance due to tight public budgets and today's highly competitive globalised environment, as it has the potential to facilitate collaboration mechanisms to build innovative partnerships leading to new types of businesses and services. To ensure efficient public spendings and to reach low transportation costs and high performance of service systems, analytical tools able to provide an effective decision support are required. Three main pillars of a decisions support tool are the model that captures important aspects of a real-world system, the efficient and flexible solving method and high fidelity input data. The field data describing a system's operation and the relevant characteristics of external environment are today more available than ever, due to the recent progress in ICT technologies enabling new concepts of sensing, data transmission, data storage and data processing, and due to open data initiatives enabling data sharing. Most approaches to the design of public service systems or supply chains focus on the development of new models or solving methods. In contrast, this paper is focused on the input data, and it examines the potential of how aligning aggregation

*Corresponding author
Email addresses: `matejc@kth.se` (Matej Cebecauer), `Lubos.Buzna@fri.uniza.sk` (Ľuboš Buzna)

of the input data with high quality solutions could help increase the quality of solutions provided by a decision support tool when data aggregation is unavoidable due to the high demand of solving methods on computational resources.

1.2. Review of literature on location-allocation problems

The need to design a public service system or a private supply chain often leads to a location-allocation problem. Researchers have been formulating and studding location-allocation problems for decades, and they have recognized a variety of forms depending on the particular application, objectives and constraints (Daskin. M., 1995; Drezner, 1995; Eiselt and Marianov, 2011, 2015).

Two basic problem classes are formed by continuous location problems where facilities are located in a plain or in some other type of a continuous set, and discrete location problems, where locations of facilities are selected from a finite set of options. According to the type of the objective, it is common to differentiate between minisum problems (minimising the sum of customers' utilities), mimimax (minimising the maximum customer's utility), and covering problems (ensuring that either each or the maximum number of customers' utilities reach the pre-defined quality) (Eiselt and Marianov, 2011). Nowadays, most problems are optimized with respect to a multicriterial objective function. The examples include bi-objective and k-objective location problems that incorporate operational goals such as total setup costs, fixed costs, average time/distance travelled, number of located facilities, fuel consumption, or also more recently, environmental and social goals based on land use, congestion, noise, pollution or tourism (Doerner et al., 2007; George and ReVelle, 2003; Hamacher et al., 2002; Nickel et al., 2005). For a comprehensive overview of multicriterial location problems, see Farahani et al. (2010). Similarly, various approaches are considered as the primary attractiveness determinant when allocating a customer to the located facilities. The basic approach is to associate customers with a single, e.g. the closest (Hakimi, 1965), or with multiple facility locations (Achabal, 1982). The approach by D. L. Huff (1962), allocates a share of customer's demand to all located facilities using the gravitational force formula. Another well-known model is the multinomial logit model (Gupta Sachin et al., 1996). More complex approaches consider several determinants simultaneously, e.g. the travel time and the waiting time (Marianov et al., 2008). If customers do not have knowledge of the functionality of facilities, they may chose to follow the strategy of visiting a number of pre-assigned facilities until they acquire the service or give up trying after a given number of unsuccessful trials (Yun et al., 2015).

Over the last few decades, the location research has addressed also many modelling challenges directly induced by practical applications. Among them we mention efforts aiming at capturing uncertainties in the operation of a system and the development of complex, but tractable formulations of problems that combine strategic location decisions with tactical and operational decisions associated with the organization of services and flows. In the context of the supply chain network, the uncertainty concerns both the demand and the supply (Snyder and Daskin, 2005). The authors in An et al. (2014) consider a robust optimization model, where k facilities may fail. The problem is optimized with respect to the multicriterial objective function aiming at finding a trade-off between the operational costs of the least costly and the most costly disruptive scenarios. The research paper Fei and Mahmassani (2011) is an example of the demand uncertainty minimisation for optimal sensor locations where a multi-objective problem maximising the link information gains in the conjunction with the demand coverage applying hybrid greedy randomized adaptive search procedure to identify the Pareto frontier is presented. An early attempt to address the combined location-routing problem is the work by Perl and Daskin (1985). Ouyang et al. (2015) presented a modelling approach for the median type of facility location under elastic customer demand and traffic equilibrium in a continuous space. The study Ponboon et al. (2016) formulates a mathematical model, and it proposes a solving method based on the branch and price (column generation) algorithm for the location routing problem with time windows. Romero et al. (2016) developed a model to analyse the facility location and routing in the context of hazardous materials transportation. Detailed review on location-routing problems is provided in Prodhon and Prins (2014).

The proposed adaptive aggregation framework is broadly applicable. To evaluate it, we selected two problems. An archetypal example for a location-allocation problem is the *p-median problem* (Hakimi, 1965). The number of algorithms applicable to this problem available in the literature is large. One of the most successful exact algorithms is ZEBRA (García et al., 2011). ZEBRA is based on the radial formulation of the p-median problem and for a narrow range of located facilities allows for solving problems up to 80 000 demand points. Thus, problems of larger size are solved by classical heuristics or metaheuristics. Examples of classical heuristics include: greedy heuristics that grow the number of located facilities one by one (Whitaker, 1983), dual accent heuristics based on the dual of the relaxed integer programming formulation of the problem (Erlenkotter, 1978), interchange heuristics where facilities are moved iteratively to vacant sites if it reduces the value of the objective function (Teitz and Bart, 1968) and the alternate heuristic (ALA) (Cooper, 1964, 1972; Maranzana, 1964). In the first step of ALA heuristic customers are divided into p subsets, and single optimal location is determined for each group. In the second step, allocations of customer are re-optimized. Location and allocation steps are alternated until no further improving changes are possible. In the literature there are several papers proving the convergence of ALA heuristic (Drezner, 1992; Lawrence M. Ostresh, 1978). For comprehensive overview of heuristic and metaheuristic algorithms to the p-median problem please refer to Mladenović et al. (2007). The second selected problem is the *lexicographic minimax facility location problem* (Ogryczak, 1997). This problem is motivated by the need arising in some applications to consider the equitable access of customers to located facilities. The goal is to find the location of facilities that corresponds to the lexicographically smallest non-increasingly ordered vector of disutilities that are associated with allocations of customers to facilities. The vector can be rearranged, where the k-th term in the vector is the number of occurrences of the k-th worst possible unique outcome (disutility). The optimal solution is then found by minimising the value of the first vector element followed by the minimisation of the second element without worsening the first term and so on (Ogryczak, 1997). As an alternative, the ordered outcomes approach and the ordered values approach were proposed in Ogryczak and Śliwiński (2006). The latter approach is more efficient, however, the size of solvable instances is very small. A convenient technique for interactive analysis, where facilities are located with respect to the objective function taking into account lexicographic minimax combined with the minisum term, was proposed in Ogryczak (1999). The approach is based on the reference distribution method which can be steered by manipulating few parameters only and allows to take into account aspiration values of assigned distances defined by the user. The above mentioned approaches to the lexicographic minimax optimization result in a specific form of the mathematical model that is supposed to be solved by a general purpose solver. This limits the size of solvable problem to less than 1 000 demand points. Approximative algorithm ALEX that provides high quality solutions and enables to extend the size of solvable problems to several thousands was proposed in Buzna et al. (2014).

1.3. Review of literature on aggregation errors

Aggregation methods have been a subject of intense research in the fields of transport economics, operations research and geographic information systems. They lead to problems of smaller size where the demand points (DPs) are replaced by the aggregated demand points (ADPs). The simplest aggregation strategies are dividing geographical space using regular grid (Hillsman and Rhoda, 1978) or selecting randomly a sample of demand points that represent them all (Goodchild, 1979). As an alternative, Erkut and Bozkaya (1999) describes the iterative aggregation of pairs of DPs that are close to each other based on a defined measurement. In contrast to the previous methods, the row-column aggregation method (Francis et al., 1996; Andersson et al., 1998) is able to capture the population clusters present in the data by constructing an irregular grid, considering the spatial distribution of demand, by applying the lexicographic ordering to the lengths of the shortest paths that connect customers with the closest located facilities. An example of the method utilizing neighbourhood relations among DPs is the work Fotheringham et al. (1995) and optimal zoning for the aggregation is used in Openshaw (1977) and Openshaw and Rao (1995). Reference Salazar-Aguilar et al. (2011) introduces a mathematical model able to achieve balanced district territories with the pre-defined tolerance. The work Assunção et al. (2006) uses a graph to represent neighbourhood relations using the minimum spanning trees.

The data aggregation and the use of ADPs tends to lead to various types of location errors (Francis et al., 2009; Erkut and Bozkaya, 1999). To be able to minimise the effects of aggregation errors, the detail understanding of possible sources of errors is needed. We collected information about the sources of errors from the available literature, organized them in two groups, and we briefly explain them in Table 1.

Table 1: Overview of typical aggregation and evaluation errors that are often made by spatial analysts, their causes and how they can be minimized or eliminated.

	Source error	**Description**	**Caused by**	**Treatment**
Source errors due to loss of information about positions of DPs.	**A**	The distance between a pair of ADPs is misestimated, due to measuring it as a distance between the centroids representing ADPs.	Loss of information about DPs in the original problem.	Replace the distance by the average distance from all DPs aggregated in the ADP to its centroid.
	B	If an ADP is candidate location for a facility, and at the same time it represents a customer, the distance between the facility location and the customer is often incorrectly set to zero.		Replace the zero distance by the average distance from DPs aggregated by the ADP to the centroid of the ADP.
	C	DPs aggregated within an ADP are assigned to the same facility.		Re-aggregate ADPs and find for each DP the closest facility .
	D	Facilities are established in ADP centroids and not in DPs, thus locations of facilities are almost certainly not optimal.		Disaggregate ADPs in the close neighbourhood of located facilities.
Typical aggregation and evaluation errors made by spatial analysts according to Erkut and Bozkaya (1999)	**UD**	This error is introduced when uniform demand is assumed.	Misestimated demand.	Use high granularity data to model the demand.
	RA	Aggregation method does not take into consideration population clusters.	Aggregation method.	Use aggregation methods that consider population clusters.
	FL	Evaluation of solutions is solely based on comparing the sets of located facilities. Two very different sets of located facilities may have similar values of the objective function.	Unilateral criterion used to evaluate the quality of solutions.	Consider optimality error or objective function value, when evaluating solutions.
	EC	The effects of aggregation are often evaluated with respect to one (optimal) solution.	Neglected effects of aggregation on the relevant subset of solutions.	Evaluate the effects of aggregation on the relevant (e.g. the best quality) set of solutions.
	DF	The candidate locations for facilities, while often being a few, are aggregated equally as DPs representing customers.	Aggregation method.	Aggregate separately the candidate location and customer DPs.
	OA	Use of an unnecessarily high level of aggregation.	Choice of parameters.	Solve as large location problems as the used algorithm allows.

The aggregation errors in the first group are caused by the loss of information when DPs are replaced by the aggregated demand points. Each ADP is often modelled as a polygon that is represented by its centroid (Francis et al., 2009). Consequently, many DPs are represented by one centroid, what results in various types of errors. In Hillsman and Rhoda (1978) the authors named these errors as source errors, and introduced the source A, B and C errors. Elimination of the source A and B errors was studied in Current and Schilling (1987). Minimization of the source C error was analysed in Hodgson and Neuman (1993). Source D error and the possibilities how it can be minimised were studied by Hodgson et al. (1997). Source errors A, B, C and D are illustrated on examples in the supplementary information file.

The second group of aggregation errors involves the sources of errors that are often made by the analysts (Erkut and Bozkaya, 1999), responsible for the preparation of the input data and the execution of spatial analyses. Typical examples of such errors are the uniform demand distribution (UD), ignoring population clusters (RA), using incorrect methods to evaluate the effects of the data aggregation focusing solely on the optimal solution (FL) or location pattern (EC), the same aggregation method applied to customers and candidate locations (DF) and an exceedingly high level of aggregation (OA).

1.4. Scientific contributions and structure of the paper

The majority of approaches to location-allocation problems focus either on the formulation of more realistic mathematical models or on the design of improved solving methods. The role of the data aggregation is often overlooked. Aggregation is often unavoidable, as, for example, the state-of-the-art exact algorithms (García et al., 2011) when applied to the p-median problem are limited to 20 000 DPs and to 90 000 DPs either when the number of the located facilities falls into a narrow interval, or when using the heuristic algorithms (Avella et al., 2012). Although the sources of aggregation errors are well understood, traditionally, the data describing the problem being solved are spatially aggregated only once, before executing the numerical analysis. Thus, the minimisation of aggregation errors in the input data does not reflect specifically on the spatial patterns that are associated with high quality solutions, and thus aggregation errors are not curtailed. The development of computational approaches that could better utilize opportunities that big data age presents in solving location problems has been also recently recognized to have big potential to solve emerging issues concerning sustainability and environmental challenges (Tong and Murray, 2017). By integrating together existing aggregation methods, elimination of aggregation errors and solving algorithms to the adaptive aggregation framework, this paper makes two main scientific contributions. Our first contribution is to propose the concept of the adaptive aggregation framework that minimises the aggregation errors by iteratively aggregating the input data for the solved problem by taking into account the best found solutions from individual iterations. Hence, the input data are at the final stage aggregated in a way that minimises aggregation errors for the resulting solution. The proposed framework is applicable to a large spectrum of location-allocation problems and can be viewed in two ways: as a solving algorithm or as an aggregation method while minimising aggregation errors. Our second contribution is to carry out extensive computational experiments and analysis to gain the understanding of possible benefits of this approach and the role of its parameters.

The paper is organized as follow: section 2 introduces the generalized formulation of the location-allocation problem. In section 3, we describe the data requirements, and we introduce the adaptive aggregation framework. The results of numerical experiments are reported in section 4. To conclude, we summarise our main findings in section 5.

2. Problem formulation

The applicability of the adaptive aggregation framework is broad and thus it is difficult to describe formally all applicable problems. To outline a class of applicable problems, while keeping the description concise, we formulate a generalized location-allocation problem. It should be noted that it is not only the mathematical formulation that determines the applicability of the proposed framework. Another important aspect is that the problem is solved within large spatial area for a large number of customers, and to adjust the size of the problem to the available solving method the input data have to be to a large degree aggregated.

We assume a graph $\mathcal{G}(\mathcal{V},\mathcal{E})$ representing the transportation infrastructure, $\mathcal{V}$ is a set of nodes and $\mathcal{E}$ is a set of edges (i,j), where $i,j \in \mathcal{V}$. The set $\mathcal{J} \subseteq \mathcal{V}$ of demand points describes the locations of customers and the set $\mathcal{I} \subseteq \mathcal{V}$ represents the possible locations of facilities. Often, to each customer location $j \in \mathcal{J}$ we associate the weight w_j expressing its relative importance, for example, in terms of the number of inhabitants living in the area represented by the DP. By the symbol $\mathbf{x} \in \mathbf{B}^{|\mathcal{I}|,|\mathcal{J}|}$, where $\mathbf{B} = \{0,1\}$ is boolean domain, we denote the vector of binary allocation variables x_{ij}, for $i \in \mathcal{I}$ and $j \in \mathcal{J}$, taking value 1, when the customer j is allocated to the location i and value 0, otherwise. For the vector of location variables we use the symbol $\mathbf{y}$, where for $i \in \mathcal{I}$, $y_i = 1$ if the facility is located at DP i and $y_i = 0$, otherwise. We define the vector of variables $\mathbf{z} \in \mathcal{Q}$ to account for additional decisions that can be associated with the location-allocation problem. The domain $\mathcal{Q}$ is a Cartesian product of domains of individual variables $\mathbf{z}$ that are typically real or binary, but other domains are also possible. Making use of this notation, we define the generalized location-allocation problem as:

$$
\begin{aligned}
\text{Minimize } & f(\mathbf{x},\mathbf{y},\mathbf{z}) && && (1)\\
\text{subject to } & \sum_{i\in\mathcal{I}} x_{ij} = 1 && \text{for } j \in \mathcal{J} && (2)\\
& x_{ij} \leq y_i && \text{for } i \in I, j \in \mathcal{J} && (3)\\
& \sum_{i\in\mathcal{I}} y_i = \hat{p} && && (4)\\
& f_l(\mathbf{x},\mathbf{y},\mathbf{z}) = 0 && \text{for } l \in \mathcal{L} && (5)\\
& \mathbf{x} \in \mathbf{B}^{|\mathcal{I}|,|\mathcal{J}|} && && (6)\\
& \mathbf{y} \in \mathbf{B}^{|\mathcal{I}|} && && (7)\\
& \mathbf{z} \in \mathcal{Q}. && && (8)
\end{aligned}
$$

The core of the problem (1)-(8) is comprised of the location-allocation constraints (2)-(4). The constraints (2) ensure that each DP is allocated exactly to one facility. The constraints (3) allow for allocating customers only to the located facilities, and the constraints (4) make sure that exactly $\hat{p}$ facilities are located. An alternative to the constraints (4) is to minimise the sum of the fixed costs that are associated with the location of facilities (Erlenkotter, 1978). The objective function $f : \mathbf{B}^{|\mathcal{I}|,|\mathcal{J}|} \times \mathbf{B}^{|\mathcal{I}|} \times \mathcal{Q} \to \mathbf{R}$ typically represents costs or another kind of disutilities that are associated with the quality of the solution as it is perceived by individual customers. The set of the constraints (5) generalizes restrictions that are implied by additional decisions to the location-allocation decisions by combining them with some other types of problems. Please note that this formulation captures inequalities that can be turned to equalities by adding slack variables.

Typically, it is assumed that $\mathcal{I} = \mathcal{J}$, i.e., facilities can be located in all demand points. To simplify the notation, from here on we use this assumption and we recover the p-median (Hakimi, 1965; García et al., 2011) for $\mathcal{L} = \emptyset$, $\mathcal{Q} = \emptyset$ and $f(\mathbf{x},\mathbf{y}) = \sum_{i\in\mathcal{J}}\sum_{j\in\mathcal{J}} c_{ij}x_{ij}$, where c_{ij} is the cost associated with the allocation of the customer $j \in \mathcal{J}$ to the facility location $i \in \mathcal{J}$. One possibility is to set $c_{ij} = w_j d_{ij}$, where the symbol d_{ij} denotes the shortest path length from the customer $j \in \mathcal{J}$ to the potential facility location $i \in \mathcal{J}$ and weight w_j represents the number of inhabitants associated with the customer location $j \in \mathcal{J}$.

While again considering $\mathcal{L} = \emptyset$, $\mathcal{Q} = \emptyset$, we can define the lexicographic location problem (Ogryczak, 1997). We identify all unique shortest path lengths D_k, $k = 1,\ldots,K_{max}$ in the set of all feasible values d_{ij}, where K_{max} is the number of unique d_{ij} values. We order the lengths D_k into the descending sequence. Thus, D_1 will denote the maximal possible distance between a customer and a facility. Then, for any feasible solution, we create a system of subsets $\mathcal{W}_k$, for $k = 1,\ldots,K_{max}$, where the pair $(i,j) \in \mathcal{W}_k$ if $x_{ij} = 1$ and $d_{ij} = D_k$. Further, for each subset $\mathcal{W}_k$ we define the following function:

$$
h_k(\mathbf{x}) = \sum_{(i,j)\in\mathcal{W}_k} w_j x_{ij} \qquad \text{for } k = 1,\ldots,K_{max}. \tag{9}
$$

Using this definition, we formulate the lexicographic location problem, where the objective is to find the solution that corresponds to the lexicographically smallest vector:

$$(h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_{K_{max}}(\mathbf{x})). \tag{10}$$

Apart from these two generic problems, we can find in the literature plethora of optimization problems that add various types of constraints to meet the requirements of specific applications but still include the location-allocation constraints (2)-(4) to determine strategic decisions about the spatial organization of a system. In Frade et al. (2011) an approach was proposed to locate the charging stations for electric vehicles where the additional constraints (5) make sure that each located facility is equipped with a sufficient number of charging points to meet the estimated number of daytime and nighttime refuelling operations. The problem to determine sales representative locations and allocate to them contiguous sales territories, while maximising the sum of profit contributions, was described in Haase and Müller (2014). The multiobjective police districting problem that is balancing the demand and considering the compactness of the districts, where the blocks of a city are allocated only to the located district centres, was introduced in Bucarey et al. (2015). The location-allocation problems were used in the context of optimizing vessels for maritime and rescue operations, while balancing the workload (Pelot et al., 2015). Furthermore, location-allocation constraints (2)-(4) naturally come out, when optimally locating the vehicle identification sensors (Gentili and Mirchandani, 2015). In order to measure the travel time information and maximise the volume of the monitored traffic, two (upstream and downstream) readers are located on the network edges and allocated to monitored paths. In all these, but also in many other cases, the adaptive aggregation framework could be either directly or after some small rearrangements applied to enhance the spatial accuracy of the location-allocation decisions.

3. Adaptive aggregation framework

The main objective is to handle large instances of location-allocation problems and to provide solutions of higher quality than the conventional methods. We will use the term original location problem (OLP) to refer to the unaggregated location-allocation problem that we aim to solve. The OLP is defined by the geometric graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ interconnecting the demand points $j \in \mathcal{J} \subseteq \mathcal{V}$ and by values w_j defining the associated demand (see Figure 2a). Original location problem is too large to be solved directly by conventional solvers, thus, the aggregation is used to reduce its size. The aggregated location problem (ALP) is defined by the set of aggregated demand points $\mathcal{K}$. Each aggregated demand points is represented by the centroid of the corresponding area. Here, we use as the centroid the most central demand point (see Figure 2b). Thus, the set of aggregated demand points is a subset of original demand points ($\mathcal{K} \subseteq \mathcal{J}$). To characterize the size of the ALP, we define the relative reduction coefficient:

$$\alpha = (1 - \frac{number\ of\ ADPs}{number\ of\ original\ DPs})100\%. \tag{11}$$

Thus, the unaggregated problem has the value of $\alpha = 0\%$. The conventional approach is to aggregate the original location problem only once before the solving process starts. Such aggregation is inevitably associated with the aggregation errors. The adaptive aggregation framework integrates the aggregation method, elimination of source errors and the solving algorithm and it is based on the iterative process solving the ALP and adjusting it in order to achieve more accurate solution in the next iteration. To minimise the aggregation errors, the crucial task is to identify ADPs that if disaggregated affect the positions of located facilities and ADPs that include DPs that are allocated to incorrect facilities and to lower the level of the aggregation in their surrounding. Thus, the adaptive aggregation framework leads to the irregular level of the aggregation across the space, while the data are aggregated less in areas that may lead to aggregation errors. To the best of our knowledge, this is the first attempt to present integrated framework for location-allocation problems that iteratively adapts granularity of the aggregated location problem.

3.1. Algorithm

The main idea behind the adaptive aggregation framework is to adjust the *local level* of data aggregation to the results of optimization. To do so, we use the iteration mechanism, while embedding into it the conventional optimization algorithm, the re-aggregation and (optionally) also the elimination of source errors. Thus, the adaptive aggregation framework can be considered as novel type of solving approach or as an aggregation method.

Figure 1 describes the phases of the adaptive aggregation framework and visualizes the associated data and work flows. Phases 0,2,4,5 and 6 form the core of the adaptive aggregation framework. Phases 1 and 3 extend the adaptive aggregation framework by the elimination of sources errors A, B, C and D. Iterative elimination of source errors C and D resembles the ALA heuristic (Cooper, 1964, 1972), however, phase 3 is performed only once in each iteration. For the description of all source errors and methods to eliminate them, see Table 1. Examples illustrating source errors are available in the supplementary information file.

The notation used to describe the concept of the adaptive aggregation framework is summarised in Table 2. The adaptive aggregation framework is applied to the original location problem and it starts with phase 0, where OLP is aggregated to the problem ALP_1 of computable size $\hat{\alpha}_1$. To prepare the initial ALP, we use the row-column aggregation method (Andersson et al., 1998), which is able to capture the population clusters, and directly it eliminates UD and RA sources of errors (see Table 1). To improve the performance of the row-column aggregation method, we apply it separately to individual municipalities to consider better the population clusters present in the geographical area. After accomplishing phase 0, the adaptive aggregation framework continues executing the main iteration loop. The inputs to each iteration round i are OLP and ALP_i. The proposed framework iterates over the phases identifying the source errors, re-aggregating the data and running the optimization solver. The information needed to target the adaptive aggregation of the problem ALP_i is collected in phases 2-5. To solve the basic instances of the location-allocation problem that are typically significantly smaller than OLP, we use either a specialized solver or a general purpose optimizer. Phase 6 re-aggregates ALP_i to prepare it for the next iteration $i + 1$. The algorithm of the adaptive aggregation framework is as follows:

Phase 0: Initialization
Set $i = 1$, aggregate the OLP to the aggregated problem ALP_1 of size corresponding to the reduction coefficient α_1.
Phase 1: Elimination of source A and B errors
Update the distance matrix corresponding to ALP_i accounting for source A and B errors (see Table 1).
Phase 2: Location of facilities
Solve the location problem ALP_i using the algorithm $\hat{A}\_p$. We derive solution of the OLP by considering $\hat{p}$ located facilities from the ALP_i solution and DPs are assigned to the facilities the same way as their ADP centroids in the ALP_i solution.
Phase 3: Elimination of source C and D errors
Eliminate the source C error by reallocating DPs to the most suitable facilities (see Table 1) and update the OLP solution. In addition, minimise the source D error by decomposing the problem into $\hat{p}$ location subproblems, each consisting of one located facility and of all associated DPs according to the OLP solution. For each decomposed problem locate a single facility using the algorithm $\hat{A}\_1$. As a result, $\hat{p}$ newly located facilities are obtained.
Phase 4: Identification of ADPs located in the central area of service zones
Set $\mathcal{A} = \emptyset$ and $\mathcal{B} = \emptyset$. Considering ADPs in ALP_i, process all ADPs and if distance from ADP to the closest facility is less or equal than $\hat{\epsilon}$, then insert ADP into the set $\mathcal{A}$, otherwise insert ADP into the set $\mathcal{B}$.
Phase 5: Identification of ADPs that if disaggregated may affect the positions of located facilities
Move from the set $\mathcal{B}$ into the set $\mathcal{A}$ all ADPs that could influence the positions of located facilities. This step needs to be adjusted to the specific location-allocation problem. To see some examples please refer to sections 4.3 and 4.4.
Phase 6: Re-aggregation
Update the best found OLP solution. If every facility is established in an ADP that cannot be further

Table 2: Notation used to describe the adaptive aggregation framework. We add symbol ˆ to distinguish the quantities representing the input parameters from the quantities characterizing the execution of the adaptive aggregation framework.

Symbol	Description
i	Iteration counter.
i_{last}	Index of the last iteration.
$\hat{i}_{max}$	Input parameter determining the upper bound for the number of iterations.
α_i	Value of the relative reduction coefficient corresponding to the solved aggregated problem in the iteration i.
$\hat{\alpha}_{max}$	Input parameter (the value of the relative reduction coefficient) determining the maximum size of the aggregated problem.
$\hat{\alpha}_1$	Input parameter (the value of the relative reduction coefficient) determining the aggregation level for the initially aggregated problem.
$\hat{\epsilon}$	Input parameter defining the radius of the circular neighbourhood of an ADP. This parameter separates the set $\mathcal{K}$ of all ADPs into two disjoint subsets $\mathcal{A}, \mathcal{B} \subseteq \mathcal{K}$, thus $\mathcal{A} \cap \mathcal{B} = \emptyset$ and $\mathcal{A} \cup \mathcal{B} = \mathcal{K}$. Subset $\mathcal{A}$ includes all ADPs that are located at a distance less or equal than $\hat{\epsilon}$ from the closest facility. Subset $\mathcal{B}$ is then determined as $\mathcal{B} = \mathcal{K} - \mathcal{A}$. Thus, if $\hat{\epsilon} = 0$, then $\mathcal{A}$ includes only ADPs with located facilities.
$\hat{\lambda}$	Input parameter defining the maximum number of newly created ADPs when de-aggregating an ADP.
$\hat{p}$	Number of the facilities to be located.
$\hat{A}\_p$	Input parameter that determines the exact or heuristic algorithm for locating $\hat{p}$ facilities.
$\hat{A}\_1$	Input parameter specifying the exact or heuristic algorithm for locating a single facility.

de-aggregated or if $i > i_{max}$ then terminate. Output the best found solution as the final result. Otherwise, increment i by 1 and de-aggregate each ADP in the set $\mathcal{A}$ to at maximum λ new ADPs using an aggregation method. If $\alpha(i) > \alpha_{max}$, then terminate otherwise go to phase 1.

The iteration loop starts with phase 1 by eliminating the source errors A and B. Phase 1 leads to changes in the distance matrix associated with ALP_i. Further, in the phase 2, $\hat{p}$ facilities are located by solving ALP_i using the algorithm $\hat{A}\_p$.

When the facility locations are known, in phase 3 source C errors are eliminated by re-allocating affected DPs to the most suitable facilities. In addition, this phase minimises the source D error by decomposing the problem into $\hat{p}$ location subproblems, each consisting of one located facility and of all associated DPs. The algorithm $\hat{A}\_1$ is applied to each subproblem, to find more efficient location of a facility considering DPs instead of ADPs. As a result, we obtain $\hat{p}$ newly located facilities. The elimination of the source C errors is applied again taking into account positions of newly located facilities.

Finally, in phases 4 and 5, ADPs that may have an impact on the accuracy of the location of facilities are identified. We aim to dis-aggregate ADPs situated in the close vicinity of the located facilities to ensure more accurate positioning of facilities. Therefore, the phase 4 extracts the subset $\mathcal{A}$ of ADPs$\in \mathcal{K}$, that are located in the central area of service zones, having the distance to the closest facility smaller or equal

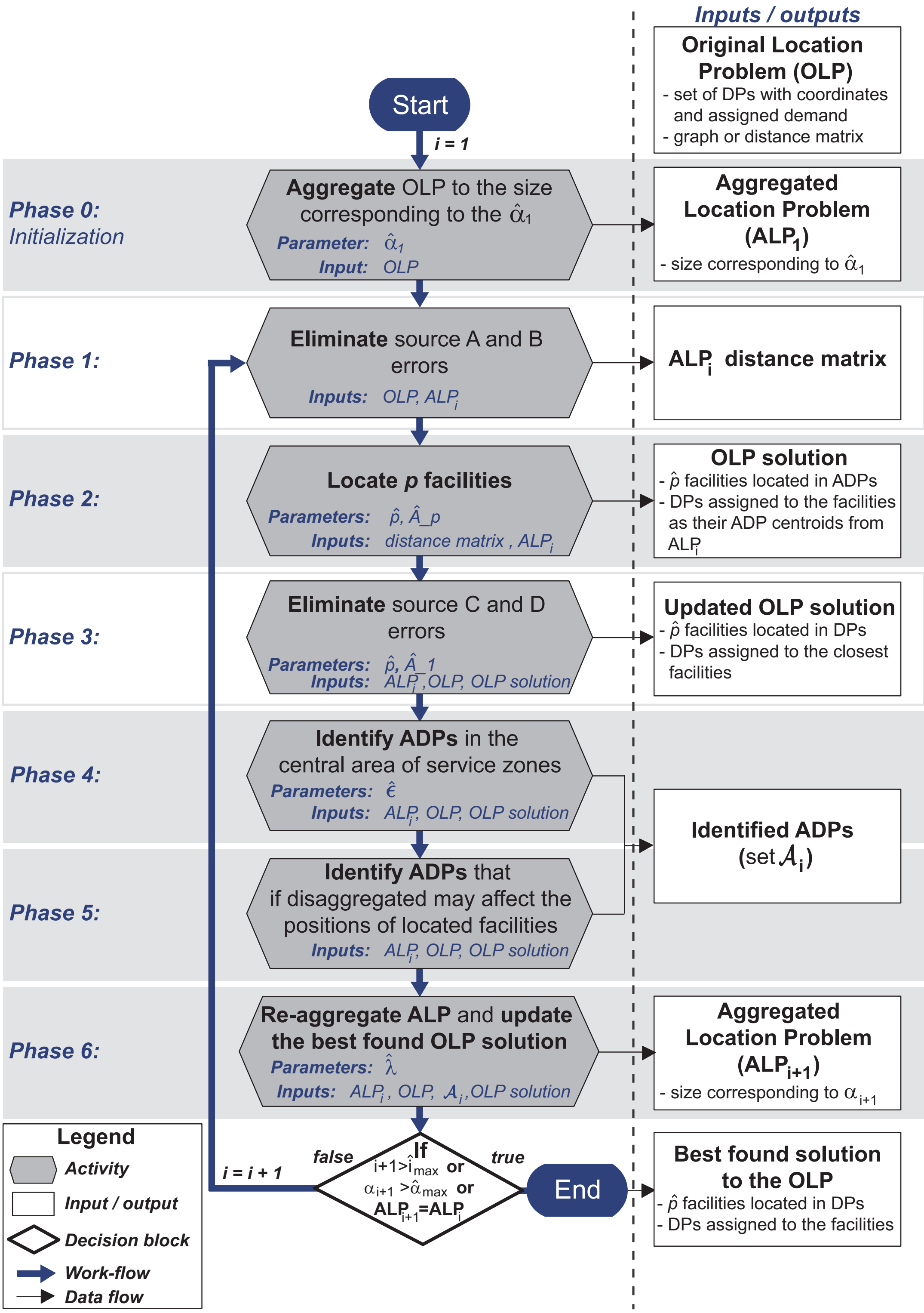

Figure 1: Flowchart of the adaptive aggregation framework. Phases produce outputs (displayed on the right from the vertical dashed line) that are later used as inputs by the follow up phases. The parameters and inputs are listed within the hexagonal boxes representing phases. Phases 0, 2, 4, 5 and 6 are highlighted as they form the core parts of the adaptive aggregation framework.

than value $\hat{\epsilon}$. Furthermore, we define the set $\mathcal{B} = \mathcal{K} - \mathcal{A}$. Phase 5 aims to identify ADPs in the subset $\mathcal{B}$ that could influence the positions of the located facilities, and they are moved to the set $\mathcal{A}$. The factors influencing positions of such ADPs are dependent on the particular type of the location-allocation problem, hence, phase 5 has to be specified separately for each problem.

In the phase 6, the best found solution for the OLP is updated. Then each ADP in the set $\mathcal{A}$ is re-aggregated to maximum $\hat{\lambda}$ new ADPs, using an aggregation method (we applied the row-column aggregation method (Andersson et al., 1998) in this paper). The parameter $\hat{\lambda}$ ensures that the size of the aggregated problem does not grow too fast. The framework runs until it reaches the predefined number of iterations $\hat{i}_{max}$, the aggregation level of the problem α_{i+1} exceeds the predefined threshold $\hat{\alpha}_{max}$, or the aggregated problem has not changed since the last iteration. In Figure 2, we use small example to illustrate the application of the adaptive aggregation framework to the p-median problem with $\hat{p} = 2$. The original location problem consists of 199 demand points (see Figure 2a) and is reduced to 11 aggregated demand points, hence, $\hat{\alpha}_1 = 94.5\%$ (see panel (b) of Figure 2). Panels (c)-(f) of Figure 2 visualize the output of the adaptive aggregation framework after executing the first four iterations, i.e. until no further change in the aggregated location problem is detected. Please note that when the algorithm terminates, the density of ADPs is higher in the close vicinity of located facilities and in border areas that are served from different facilities. This illustrates that the adaptive aggregation framework is able to locally suit the data aggregation to the resulting solution.

4. Results

In this section we describe computational experiments and discuss the results and our findings. Subsection 4.1 details the design of benchmarks and basic organization of experiments. In order to study the relevance of the particular phases, we proposed several versions of the adaptive aggregation framework as described in the subsection 4.2. Subsections 4.3 and 4.4 are devoted to the application of the adaptive aggregation framework to the p-median problem and to the lexicographic minimax problem, respectively.

4.1. Benchmarks

The benchmarks are prepared using a large amount of GIS data covering the geographical area of the Slovakia. What is a suitable approach to determine the set of customer locations depends on the application. Here, we utilize the dataset that we used previously to study the design and efficiency of public service systems such as networks of hospitals, schools or medical emergency centres (Cebecauer et al., 2016). All details regarding the data and the processing procedure are given in Cebecauer and Buzna (2017). To construct the dataset, freely available geographical data originating from the OpenStreetMap are used. Furthermore, data layers describing the positions of buildings, roads, residential, commercial and industrial areas are considered to estimate the spatial distribution of customers that are modelled by the set of DPs. The dataset is composed of the graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ representing the Slovak road network and of the set $\mathcal{J}$ of DPs characterized by the weights w_j derived from the residential population data (Batista e Silva et al., 2013). The graph includes 2 080 694 edges and 1 956 067 nodes, out of which 663 203 are considered as DPs. The average area representing one DP in dense urban areas is about 0.01 km^2 and the average distance between the closest DPs is approximately 0.1 km. Figure 3 shows the choropleth map of Slovakia and the border lines of geographical areas that constitute our benchmarks.

To evaluate the proposed approach, three classes of benchmarks that differ in the problem size are used. The first class is used to compare the objective value and time consumption with the exact or high quality methods. The main purpose of these benchmarks is to evaluate how far the adaptive aggregation framework can deviate from the optimal solution. Thus, problems are small enough to be computable to optimality and the distance matrix can be easily stored in the computer memory. For this purpose we created the benchmarks Partizánske and Košice (see Table 3), where all distances are calculated in advance to the solving process. Please note that the size of these problems is in the location analysis already considered as large. In addition, two standard benchmarks d2103 and pcb3038 with 2 103 and 3 038 DPs, used in García et al. (2011), are considered to demonstrate the universality and independence of results on datesets generated in Cebecauer and Buzna (2017). Please note that for these benchmarks the coordinate system is unknown,

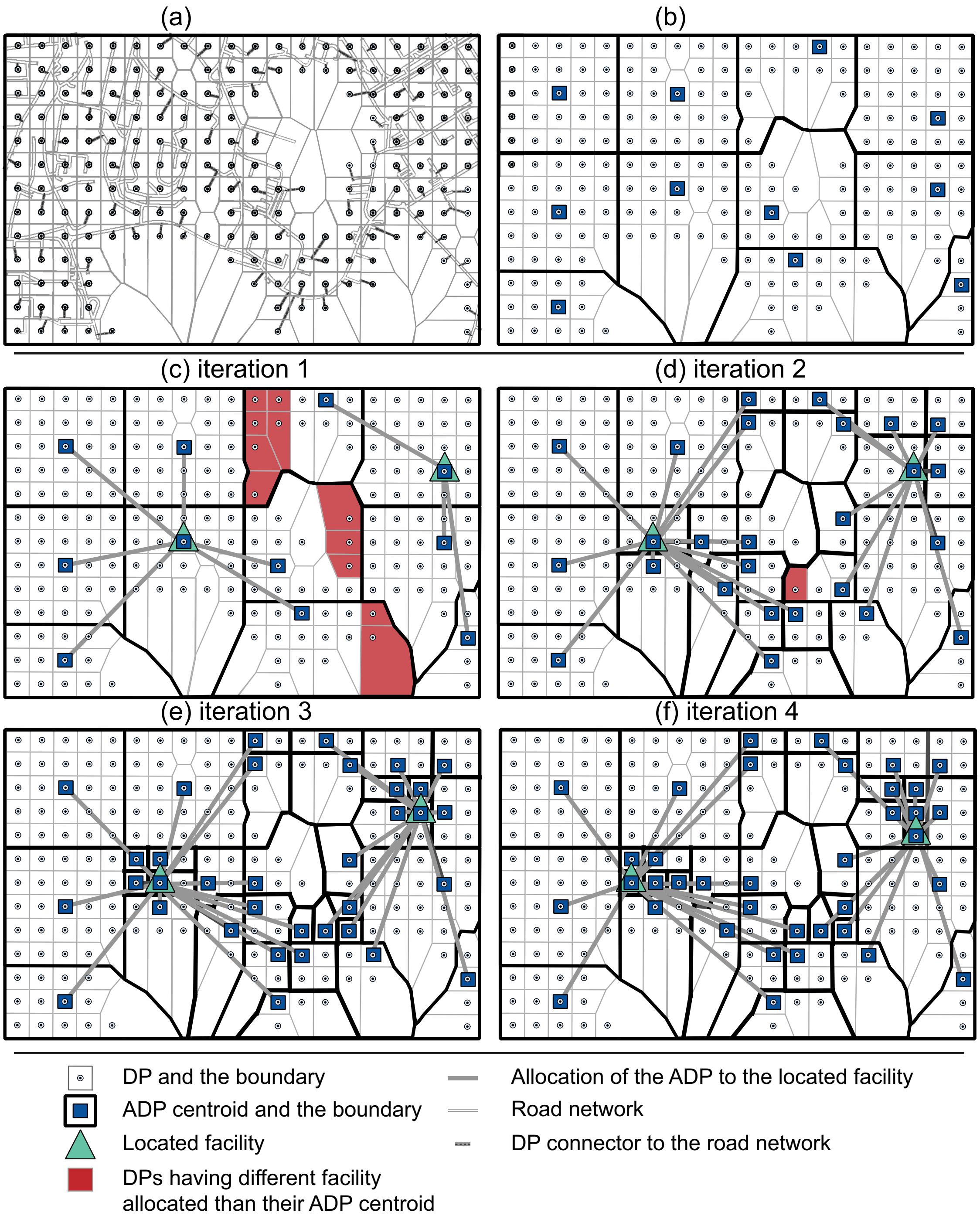

Figure 2: Re-organizations of the aggregated location problem within individual iterations of the adaptive aggregation framework when it is applied to the p-median problem for $\hat{p} = 2$. (a) Demand points that constitute the original location problem and the underlying road network forming the geometric graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ (for clarity the network is not shown in other panels). (b) Aggregated location problem produced by the row-column aggregation method in the phase 0 of the adaptive aggregation framework. (c)-(f) Aggregated location problem after each of the first four iterations of the adaptive aggregation framework. Source errors D are minimised as the close neighbourhood of the located facilities becomes disaggregated to larger degree than other areas. Areas of demand points filled with the colour visualise all demand points that are incorrectly assigned to facilities due to the aggregation (source error C). These areas are eliminated within the first two iterations of the adaptive aggregation framework.

Table 3: Basic information about the geographical areas that constitute our benchmarks. The number of DPs is equal to the size of sets $\mathcal{I}$ and $\mathcal{J}$ and thus determines the size of solved problems.

Benchmark	**Number of DPs**	**Area** $[km^2]$	**Population**
Partizánske	4 873	301	47 801
Košice	9 562	240	235 251
Žilina	79 612	6 809	690 420
Slovakia	663 203	49 035	5 418 561

and therefore, it is hard associate values of the parameter ϵ with some meaning. Furthermore, the Euclidian distances between demand points have to used instead of the road network distances.

The second class concerns the benchmark Žilina (see Table 3) that covers more than 300 municipalities in the region of Žilina. The number of DPs considered in the benchmark Žilina is similar to the maximal number that we found in the literature to be solved in the context of the p-median problem without the use of any aggregation method (García et al., 2011; Avella et al., 2012). Problems of this size are usually computable only for very small or large enough value of the parameter $\hat{p}$. Another difficulty is that the matrix of the shortest path distances is exceedingly large, and thus it is problematic to store it in the computer memory. Therefore, some authors use Euclidean distances instead (García et al., 2011). They are easy to calculate from geographical coordinates whenever they are needed. Here, another approach is applied. Although it is challenging, the size of the benchmark Žilina makes it still possible to repeatedly calculate the shortest path lengths from the graph of the road network. Therefore, we do not store distances in the memory but calculate them whenever they are needed to be known. This approach allows for quantifying the contributions of the individual phases of the adaptive aggregation framework on the quality of the final solution, while the level of aggregation is changing.

The third class is represented by the exceedingly large benchmark Slovakia covering the entire generated dataset (see Table 3) reaching 663 204 DPs. This benchmark significantly exceeds the capabilities of current exact and also many heuristic methods. Our aim is to quantify the effect of the adaptive aggregation framework when it is applied to extremely large problems where the level of the used aggregation is typically high.

Repeated computations of the shortest path lengths consume non-negligible time, and because we applied a different approach to the calculation of distances, dependent on the size of problems, it is difficult to compare computational times across the different problem sizes. However, please note that location problems are typically conceived as strategic decision problems, and thus computational time, while being within the reasonable limits, is not the most sensitive issue. All the benchmarks are available for download from the web site `http://frdsa.uniza.sk/~buzna/page5/page5.html` and to facilitate the reproducibility of results, the description of data has been turned to a short data paper Cebecauer and Buzna (2017).

4.2. Versions of the adaptive aggregation framework

The purpose of computational experiments is to evaluate the adaptive aggregation framework and to investigate the importance of the elimination of source errors (phases 1 and 3) and the importance of customizing the selection of ADPs that are further re-aggregated (phase 5) to the particular type of the problem. Therefore, we have created three versions V1, V2 and V3 that are characterized by the absence of some phases. For overview please refer to the Table 4.

Versions V1 and V2 both omit phases 1 and 3 that eliminate A, B, C and D source errors. Thus, these two versions represent the core of the adaptive aggregation framework. By comparing V1 and V2, we may quantify the effect of the phase 5 that is not present in the version V1. When phase 5 is omitted, more DPs remain incorrectly assigned to located facilities, however, some computational time is saved, and the value of the reduction coefficient α_i remains higher for longer time.

Version V3 is a reference version that extends the adaptive aggregation framework by elimination of source errors A, B, C and D. Elimination of source errors is computationally very expensive for large

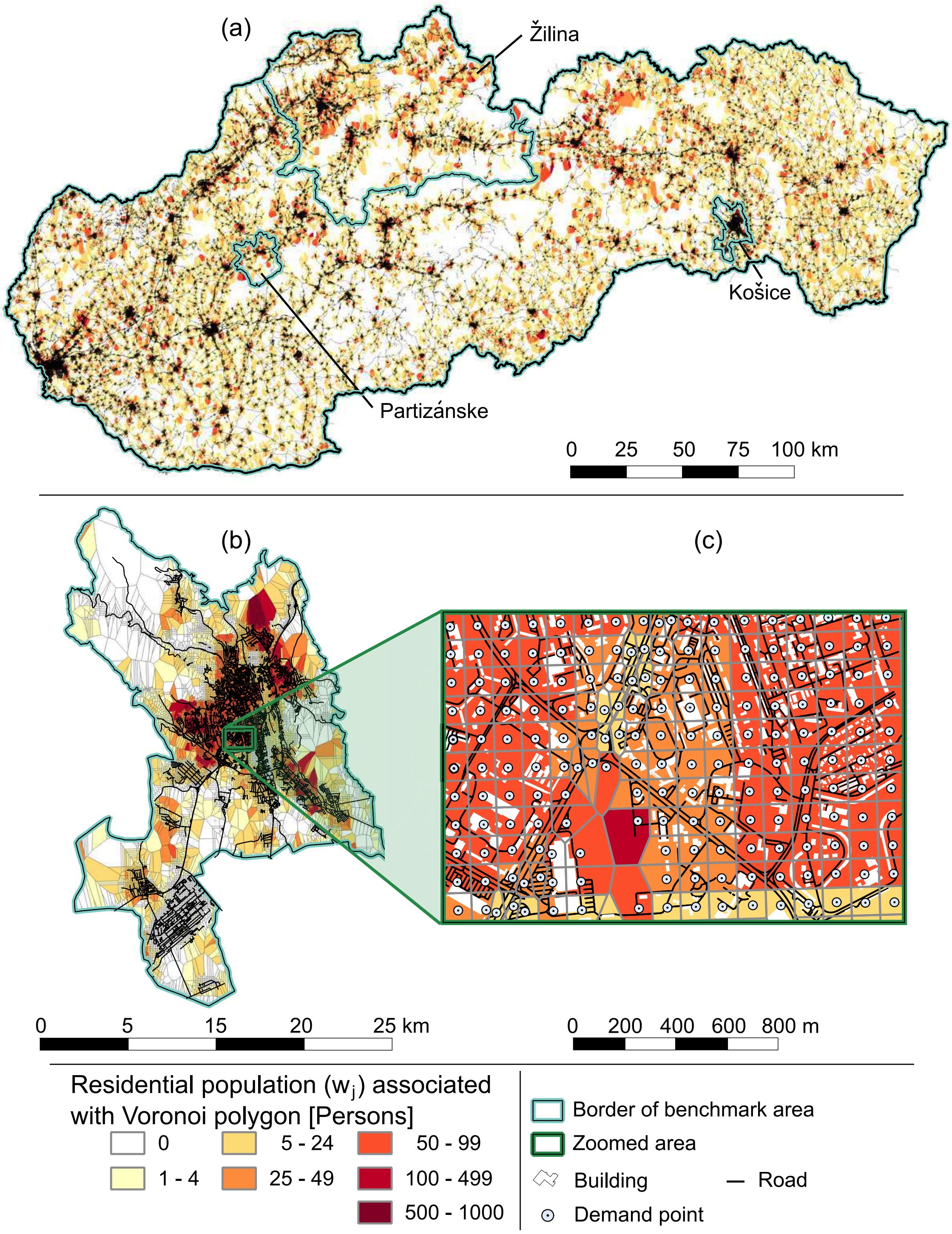

Figure 3: Choropleth maps of the selected areas that belong to our benchmarks. The colours indicate the residential population associated with the Voronoi polygons derived from the positions of DPs. (a) Map of Slovakia with highlighted borders of the areas that constitute our benchmarks. (b) Benchmark Košice. (c) Zoomed area of benchmark Košice illustrating the level of details that we considered across the entire area of Slovakia.

Table 4: Versions of the adaptive aggregation heuristic. We use the symbol "+" to denote the presence of phases. The versions of the framework were designed to evaluate the importance of the elimination of source errors (phases 1 and 3) and the importance of customizing the selection of ADPs to the particular type of the problem (phase 5).

Version	Phases 1 and 3	Phase 5
V1		
V2		+
V3	+	+

location problems such as Žilina and Slovakia. Phases 0, 2, 4 and 6 are the integral parts of the framework and therefore we do not present results when experimenting with their removal. Preliminary experiments have shown that the missing combination of the removed phases in table 4 does not lead to significant effects.

4.3. Application to the p-median problem

Before applying the proposed adaptive aggregation framework, to achieve higher performance, the phase 5 has to be adjusted to the p-median problem. Here, ADPs that may erroneously (due to source errors C and D) affect the optimal location of facilities are located in the boundary area of service zones. Such ADPs are likely to be composed of DPs that should be allocated to other facilities than a given ADP, and thus their demand is incorrectly distributed among the facilities. This way, ADPs located in the boundary areas of facilities affect the set of the locations found in the phase 2. Therefore, we adjusted the phase 5 of the framework to identify ADPs in the boundary areas of the service zones as the candidate ADPs for the re-aggregation:

Phase 5: Identify ADPs that if disaggregated may affect the positions of located facilities. Process the set $\mathcal{B}$ and move an ADP to the set $\mathcal{A}$ if it includes at least one DP that has shorter distance to another facility than its ADP centroid.

To solve the p-median problem in the phase 2, we use as the parameter $\hat{A}\_p$ the state-of-the-art algorithm ZEBRA (García et al., 2011). In the phase 3, to minimise the source D errors, we use as the parameter $\hat{A}\_1$ the enumeration algorithm to solve the 1-median problem. The results of preliminary experiments with the early version of the adaptive approach applied to the p-median problem were presented in the conference contribution Cebecauer and Buzna (2015).

4.3.1. Performance evaluation

To quantify the performance, we adopt the indicators described in Erkut and Neuman (1992) and we formalize the relative error Φ as:

$$\Phi(\alpha_I, \alpha_{II}) = \frac{f(\mathbf{x}^{\alpha_{II}}, \mathbf{y}^{\alpha_{II}}) - f(\mathbf{x}^{\alpha_I}, \mathbf{y}^{\alpha_I})}{f(\mathbf{x}^{\alpha_I}, \mathbf{y}^{\alpha_I})} 100\%, \tag{12}$$

where $(\mathbf{x}^{\alpha}, \mathbf{y}^{\alpha})$ is the optimal solution to the problem that has been aggregated to the level of α and $f(\mathbf{x}^{\alpha}, \mathbf{y}^{\alpha})$ is the corresponding objective function value evaluated with respected to the unaggregated problem (OLP). If $\alpha_I = 0\%$ in the equation (12), then the optimality error (Erkut and Neuman, 1992), often called the relative gap, is recovered.

Using the same notation, we define the relative time efficiency τ as:

$$\tau(\alpha_I, \alpha_{II}) = \frac{t(\mathbf{x}^{\alpha_{II}}, \mathbf{y}^{\alpha_{II}}) - t(\mathbf{x}^{\alpha_I}, \mathbf{y}^{\alpha_I})}{t(\mathbf{x}^{\alpha_I}, \mathbf{y}^{\alpha_I})} 100\%, \tag{13}$$

where $t(\mathbf{x}, \mathbf{y})$ is the time spent by computing the solution $(\mathbf{x}, \mathbf{y})$.

Table 5: Results of the numerical experiments for the p-median problem and the benchmark Košice. The symbol $\alpha_{i_{last}}$ denotes the value of the reduction coefficient after the last iteration i_{last} of the framework. Values $\Phi(0, \alpha_{i_{last}})$ and $\tau(0, \alpha_{i_{last}})$ are calculated by contrasting solutions with the optimal solution found by the algorithm ZEBRA.

[km]	[%]	Indicator	$\hat{p} = 10$			$\hat{p} = 20$			$\hat{p} = 40$		
			V1	**V2**	**V3**	**V1**	**V2**	**V3**	**V1**	**V2**	**V3**
$\hat{\epsilon} = 0$	$\hat{\alpha}_1 = 99$	i_{last}	7	13	19	11	9	14	14	21	16
		$\alpha_{i_{last}}[\%]$	97.8	91.8	91.6	96.3	89.0	87.2	92.7	82.3	84.0
		$\Phi(0, \alpha_{i_{last}})[\%]$	4.025	2.155	0.099	2.539	2.02	0.177	3.252	1.177	0.018
		$\tau(0, \alpha_{i_{last}})[\%]$	-99.9	-98.4	-91.1	-99.8	-98.3	-91.5	-97.5	-66.5	-63.2
	$\hat{\alpha}_1 = 90$	i_{last}	12	12	12	6	9	8	6	12	16
		$\alpha_{i_{last}}[\%]$	87.6	87.5	86.8	89.0	84.6	85.5	88.0	81.8	80.4
		$\Phi(0, \alpha_{i_{last}})[\%]$	1.167	1.167	0.005	1.592	0.801	0	2.364	2.151	0.035
		$\tau(0, \alpha_{i_{last}})[\%]$	-96.8	-96.7	-89.2	-99.1	-96.3	-93.7	-97.2	-78.5	-44.5
	$\hat{\alpha}_1 = 75$	i_{last}	4	6	7	5	6	7	4	10	4
		$\alpha_{i_{last}}[\%]$	74.9	73.9	73.9	74.6	72.5	72.4	74.1	71.0	70.7
		$\Phi(0, \alpha_{i_{last}})[\%]$	0.093	0.033	0	1.004	0.74	0	1.298	1.198	0.191
		$\tau(0, \alpha_{i_{last}})[\%]$	-96.5	-94.7	-81	-93.2	-92.2	-82.1	-88.8	-63.1	-12.2
$\hat{\epsilon} = 1$	$\hat{\alpha}_1 = 99$	i_{last}	16	16	14	11	16	13	11	17	9
		$\alpha_{i_{last}}[\%]$	71.2	71.1	70.9	63.2	57.4	57.6	50.4	44.5	44.6
		$\Phi(0, \alpha_{i_{last}})[\%]$	0.008	0.008	0	0.359	0.186	0.036	0.089	0.023	0.014
		$\tau(0, \alpha_{i_{last}})[\%]$	-85.7	-83.3	-64.6	-84.7	-56.7	-45.5	-27.7	106.7	196.8
	$\hat{\alpha}_1 = 90$	i_{last}	12	12	12	9	12	12	8	18	8
		$\alpha_{i_{last}}[\%]$	68.8	68.9	69.3	59.0	55.4	57.8	48.0	43.0	43.7
		$\Phi(0, \alpha_{i_{last}})[\%]$	0.004	0.004	0	0.229	0.19	0.15	0.079	0.02	0
		$\tau(0, \alpha_{i_{last}})[\%]$	-84.7	-78.3	-51.2	-72.7	-51	-29	-32	239.3	342.9
	$\hat{\alpha}_1 = 75$	i_{last}	6	8	8	7	13	13	7	14	7
		$\alpha_{i_{last}}[\%]$	63.8	62.0	61.9	53.5	50.9	52.1	44.2	41.5	42.2
		$\Phi(0, \alpha_{i_{last}})[\%]$	0.009	0.009	0	0.003	0.003	0	0.024	0.02	0.014
		$\tau(0, \alpha_{i_{last}})[\%]$	-86.1	-82.6	-57.9	-65.6	-39.9	-11.5	-25.7	102.3	127.4
$\hat{\epsilon} = 2$	$\hat{\alpha}_1 = 99$	i_{last}	19	16	11	19	18	11	18	18	18
		$\alpha_{i_{last}}[\%]$	47.9	44.7	44.8	33.9	31.1	31.4	24.2	20.2	22.1
		$\Phi(0, \alpha_{i_{last}})[\%]$	0.008	0	0	0.039	0.039	0	0.032	0	0
		$\tau(0, \alpha_{i_{last}})[\%]$	-26.1	-24.1	23.7	-25.3	79	110	146.3	473.4	563.6
	$\hat{\alpha}_1 = 90$	i_{last}	10	12	12	10	18	16	10	15	15
		$\alpha_{i_{last}}[\%]$	46.8	44.0	44.1	33.0	29.3	29.7	23.1	21.0	21.6
		$\Phi(0, \alpha_{i_{last}})[\%]$	0	0	0	0.036	0	0	0.011	0	0
		$\tau(0, \alpha_{i_{last}})[\%]$	-54.3	-24.4	18.3	-13.9	109.4	99.6	187.7	392.7	459.5
	$\hat{\alpha}_1 = 75$	i_{last}	7	17	17	8	17	18	10	17	17
		$\alpha_{i_{last}}[\%]$	42.2	40.5	41.1	29.5	27.9	27.8	21.0	19.2	20.0
		$\Phi(0, \alpha_{i_{last}})[\%]$	0	0	0	0.004	0.004	0	0.004	0	0
		$\tau(0, \alpha_{i_{last}})[\%]$	-54.4	45.1	151	7.6	98	171.8	285.7	555.7	699.6

4.3.2. Benchmark Košice.

The aim of this section is to evaluate the ability of the adaptive aggregation framework to find high-quality solutions by comparing it to the exact algorithm ZEBRA and to evaluate the effects of selected phases. In addition, we study how the initial level of the problem aggregation ($\hat{\alpha}_1$), the number of located facilities ($\hat{p}$) and the radius determining the disaggregated area ($\hat{\epsilon}$) influence the optimality error and the time effectiveness. Benchmark Košice ,as well as, the benchmarks d2103 and pcb3038 can be solved to optimality and therefore there is no need to limit the number of iterations and we set $\hat{\alpha}_{max} = 0\%$ and $\hat{i}_{max} = \infty$. Thus, the framework terminates when the solved problem stops changing.

We present values of the performance indicators $\Phi(0, \alpha_{i_{last}})$ and $\tau(0, \alpha_{i_{last}})$ in Table 5. Except the cases when we limit the size of the re-aggregated area in phase 4 by setting $\hat{\epsilon} = 0$, the optimality error $\Phi(0, \alpha_{i_{last}})$ is below 1%. Setting $\hat{\epsilon} = 2$ significantly increases the quality of solutions and increases the chances to find the optimal solution. For instance, version V2 provides optimal solutions in 77% of cases when $\hat{\epsilon} = 2$ and no optimal solution for $\hat{\epsilon} = 1$ and $\hat{\epsilon} = 0$. This indicates that the quality of the resulting solution is improved, when we re-aggregated not only ADPs where the facilities are located ($\hat{\epsilon} = 0$), but also ADPs located in their close neighbourhood.

As expected full version of the framework (V3) provides the best quality of solutions. It finds the optimal solution in all cases when parameter $\hat{\epsilon} = 2$, in 55% of cases for $\hat{\epsilon} = 1$ and in 33% of cases if $\hat{\epsilon} = 0$. By removing the phases 1, 3 and 5 the quality of the solutions quickly deteriorates. The version V1 finds the optimal solution only in 22% of cases for $\hat{\epsilon} = 2$ and it fails to find any optimal solution when setting $\hat{\epsilon} = 1$ and $\hat{\epsilon} = 0$. The comparison between V1 and V2 reveals that phase 5 improves the quality of solutions, and it thus partially compensates for phase 4, especially when $\hat{\epsilon} = 0$.

The version V1 is the most time efficient, and it leads to the lowest number of iterations and to the largest reduction coefficient $\alpha_{i_{last}}$. The most time consuming is the version V3. The time efficiency of V2 closely resembles V3, and it is smaller than V1, which implies that the elimination of source errors in phases 1 and 3 is more time consuming than phase 5. The elimination of source errors and phase 5 take less computational time when the number of located facilities $\hat{p}$ is small. Consequently, the adaptive aggregation framework has higher time efficiency $\tau(0, \alpha_{i_{last}})$, when $\hat{p}$ is small. This is particularly beneficial as the algorithm ZEBRA systematically consumes more computational time and computer memory for smaller values of the parameter $\hat{p}$.

To summarise, the experiments with the benchmark Košice showed that the adaptive aggregation framework is able to find a high quality solutions. For small values of $\hat{p}$, it can even find the optimal solution while saving some computational time when compared to the exact algorithm ZEBRA (see the cells in Table 5 where $\Phi(0, \alpha_{i_{last}}) = 0$ and $\tau(0, \alpha_{i_{last}})$ is negative). We conducted similar numerical experiments with the benchmarks Partizánske, d2103 and pcb3038 that further support our conclusions (see Tables S1, S2 and S3 in the supplementary information file).

4.3.3. Benchmark Žilina

The benchmark Žilina is too large to be solved to optimality by the algorithm ZEBRA. To ensure that the aggregated problem can be solved in phase 2, $\hat{\alpha}_{max}$ is set to 70%. The computational time is limited to 12 days to execute several iterations of the framework. Due to the size of the benchmark Žilina, it is impossible to evaluate the optimality error $\Phi(0, \alpha_{i_{last}})$. In the location analyses DPs are often considered as geographically large areas, e.g. the individual municipalities (Yun et al., 2015; Romero et al., 2016; Janáček et al., 2012; Buzna et al., 2014; McLay and Moore, 2012). Here, we use the problem aggregated to the level of 314 municipalities as a basic reference case to evaluate the quality of solutions. This problem is also used to initialize the adaptive aggregation framework, i.e. $\hat{\alpha_1} = 99.6\%$. As a second reference case we use the problem that was aggregated by the row-column aggregation method (Andersson et al., 1998) to 23 833 ADPs, corresponding to $\alpha = 70\%$. Projecting the numbers of emergency services centres operated in Slovakia (113 fire brigades, 273 ambulances, 405 police stations and 1 500 post offices) proportionally to the population of the region of Žilina, we obtained the following values of the parameter $\hat{p}$: 10, 35, 50 and 190. To solve such computationally demanding problems efficiently, we used the computer cluster. Table 6 summarises our results.

In comparison to the problem aggregated to the level of municipalities, the adaptive aggregation framework enables to lower the objective function by 8% to 45%. The reduction grows with the parameter $\hat{p}$. The framework provides similar quality of solutions than the exact algorithm ZEBRA when applied to the aggregated problem with $\alpha = 70\%$. Here we should point out that the aggregation level is in the majority of cases significantly higher than 70% (see values of $\alpha_{i_{last}}$).

The version V1 terminates with less aggregated problem, however, it provides solutions of very similar quality to the version V3 while consuming significantly less computational time. As the number of located facilities grows, the relative improvement in the solution quality grows as well. In other words, the relative benefit of using less aggregated problem is larger when the service zones are small. This is beneficial when

Table 6: Results of the numerical experiments for the p-median problem and the benchmark Žilina. The symbol $\alpha_{i_{last}}$ denotes the value of the reduction coefficient after the last iteration i_{last} of the adaptive aggregation framework that was successfully completed before reaching the time limit of 12 days. The relative error Φ was evaluated with respect to the reference problems aggregated to the level of $\alpha = 99.6\%$ and $\alpha = 70\%$, respectively.

[km]	Indicator	$\hat{p} = 10$			$\hat{p} = 35$			$\hat{p} = 50$			$\hat{p} = 190$		
		V1	**V2**	**V3**	**V1**	**V2**	**V3**	**V1**	**V2**	**V3**	**V1**	**V2**	**V3**
$\hat{\epsilon} = 0$	i_{last}	6	8	3	8	10	4	8	10	5	10	13	6
	$\alpha_{i_{last}}[\%]$	98.9	84.6	95.2	98.5	82.3	90.8	98.3	81.3	86.1	96.1	76.9	78.5
	$\Phi(99.6, \alpha_{i_{last}})[\%]$	-8.19	-8.32	-8.69	-15.8	-15.9	-16.3	-18.6	-19.0	-19.6	-43.9	-44.5	-45.4
	$\Phi(70.0, \alpha_{i_{last}})[\%]$	0.47	0.32	-0.07	0.68	0.51	0.01	1.17	0.69	-0.06	2.14	0.9	-0.62
	$t[h]$	2.6	288	288	3.6	242.3	288	4.5	209.6	288	11.1	212.5	288
$\hat{\epsilon} = 1$	i_{last}	9	7	3	13	8	5	11	9	5	9	5	5
	$\alpha_{i_{last}}[\%]$	97.4	82.8	94.9	93.3	76.8	82.9	92.1	75.2	81.4	75.6	70.0	70.0
	$\Phi(99.6, \alpha_{i_{last}})[\%]$	-8.56	-8.66	-8.69	-16.1	-16.2	-16.4	-19.4	-19.5	-19.7	-45.2	-45.1	-45.4
	$\Phi(70.0, \alpha_{i_{last}})[\%]$	0.07	-0.05	-0.07	0.23	0.24	-0.13	0.19	0.13	-0.16	-0.28	-0.28	-0.74
	$t[h]$	6.5	288	288	23.5	288	288	22.4	288	288	57.4	59.2	244.9
$\hat{\epsilon} = 2$	i_{last}	10	6	3	11	6	4	13	5	4	5	4	5
	$\alpha_{i_{last}}[\%]$	93.0	80.2	94.1	82.9	70.0	84.4	79.5	70.0	82.7	70.0	70.0	70.0
	$\Phi(99.6, \alpha_{i_{last}})[\%]$	-8.77	-8.76	-8.49	-16.4	-16.4	-16.4	-19.7	-19.6	-19.7	-45.0	-44.9	-45.4
	$\Phi(70.0, \alpha_{i_{last}})[\%]$	-0.16	-0.15	-0.07	-0.10	-0.13	-0.13	-0.14	-0.02	-0.20	-0.01	0.18	-0.70
	$t[h]$	35.8	288	118.1	76.0	240.7	204.2	131.5	114.6	193.1	33.0	39.0	269.2

dealing with extremely large problems. The version V2 provides solutions of quality and time effectiveness between V1 and V3.

4.3.4. Benchmark Slovakia

The set of preliminary tests showed that the size of the benchmark Slovakia does not allow for the elimination of source errors (phases 1 and 3) due to the demanding computation of the shortest path distances d_{ij}. Therefore, we studied only versions V1 and V2. Please note that experiments with the benchmark Žilina revealed that the versions V1 and V2 when applied to large aggregated problems provide solutions of comparable quality to the version V3, so it is very likely that such enormous computational effort would not pay off anyway.

The benchmark Slovakia is extremely large and cannot be solved to optimality and we again evaluate the quality of the final solutions by comparing them with the optimal solution to two aggregated problems. The first problem, where the benchmark is aggregated to 2 924 municipalities ($\alpha = 99.6\%$), and the second problem where we used the row-column aggregation method to create as large as possible benchmark comprising of 39 792 DPs ($\alpha = 94\%$). The instances of this problem could still be solved within 12 days, using 64-bit version of algorithm ZEBRA and 35 GB RAM memory. We set the value of parameter $\hat{\alpha}_{max} = 96.98\%$ (20 000 ADPs) and the number of the located facilities $\hat{p} = 113$, 273, 405 and 1 500. We limit the computational time to 18 days and parameter $\hat{\alpha}_1 = 99.6\%$. Table 7 summarises the results of computational experiments.

Not surprisingly, the computational time grows with the parameter $\hat{p}$, resulting in smaller values of the reduction coefficient α_i. The adaptive aggregation framework improves the quality of the solution from 12 to 44% with respect to the optimal solution to the problem that is aggregated to the level of municipalities ($\alpha = 99.6\%$). The gained improvements are also significant, ranging from 11% to 18%, when compared to the problem that was aggregated to the maximum computable size ($\alpha = 94.0\%$). For example, for $\hat{p} = 273$ and $\hat{\epsilon} = 0$ the version V1 of the adaptive aggregation framework reached $\Phi(94.0, 98.97) = -13.59\%$, while using less than one fifth of ADPs. In absolute numbers, this improvement corresponds to 3 354 988 km, when the solution is evaluated considering the unaggregated location problem. These results underline the significant benefits of the adaptive aggregation framework.

Like in the case of the benchmark Žilina, the results confirmed that the parameter $\hat{\epsilon} > 0$ does not effect

Table 7: Results of the computational experiments obtained when applying the adaptive aggregation framework to the benchmark Slovakia and the p-median problem. Symbol $\alpha_{i_{last}}$ denotes the value of the reduction coefficient after the last iteration, i_{last}, of the algorithm. We evaluate the relative error Φ with respect to the optimal solution to problems aggregated to the levels of $\alpha = 99.6\%$ and $\alpha = 94.0\%$, respectively.

[km]	Indicator	$\hat{p} = 113$		$\hat{p} = 273$		$\hat{p} = 405$		$\hat{p} = 1500$	
		V1	**V2**	**V1**	**V2**	**V1**	**V2**	**V1**	**V2**
$\hat{\epsilon} = 0$	i_{last}	9	3	12	3	10	3	6	3
	$\alpha_{i_{last}}[\%]$	99.32	96.98	98.97	96.98	98.65	96.98	96.98	96.98
	$\Phi(99.6, \alpha_{i_{last}})[\%]$	-13.15	-12.36	-19.94	-18.87	-23.22	-21.95	-43.93	-40.99
	$\Phi(94.0, \alpha_{i_{last}})[\%]$	-12.18	-11.39	-13.59	-12.44	-14.82	-13.40	-18.79	-14.52
	$t[h]$	130.7	181.2	243.6	139.8	244.7	134.8	272.2	146.0
$\hat{\epsilon} = 1$	i_{last}	9	3	6	3	6	3	5	3
	$\alpha_{i_{last}}[\%]$	97.74	96.98	96.98	96.98	96.98	96.98	96.98	96.98
	$\Phi(99.6, \alpha_{i_{last}})[\%]$	-13.45	-12.40	-19.94	-18.87	-22.96	-21.97	-42.49	-41.03
	$\Phi(94.0, \alpha_{i_{last}})[\%]$	-12.48	-11.43	-13.59	-12.44	-14.53	-13.43	-16.71	-14.59
	$t[h]$	432	180.7	335.4	139.8	351.2	136.9	160.1	144.7
$\hat{\epsilon} = 2$	i_{last}	6	3	6	3	4	3	3	3
	$\alpha_{i_{last}}[\%]$	96.98	96.98	97.29	96.98	97.29	96.98	96.98	96.98
	$\Phi(99.6, \alpha_{i_{last}})[\%]$	-13.16	-12.46	-20.04	-18.91	-22.59	-22.06	-41.10	-41.09
	$\Phi(94.0, \alpha_{i_{last}})[\%]$	-12.19	-11.49	-13.70	-12.48	-14.11	-13.53	-14.70	-14.68
	$t[h]$	253.2	233.7	196.5	136.6	127.6	136.4	115.9	133.9

the quality of the solution so strongly. The values $\alpha_{i_{last}}$ are similar independently on $\hat{\epsilon}$, thus, we conclude that re-aggregation of ADPs, where facilities are located, is sufficient for very large benchmarks. The version V1 provides in all cases larger improvements than the version V2, thus, it confirms that omitting phase 5 is beneficial for very large p-median problems. Here, including the phase 5 can make α_i exceeding the α_{max} in just few iterations. It is typical for iterative heuristics that gains in the quality of the solution quickly diminish with the number of iterations. However, it is important to find a suitable combination of input parameters $\hat{\epsilon}$, $\hat{\alpha}_1$ and $\hat{\lambda}$ to ensure that the number of executed iterations is sufficient to fully exploit the potential of the re-aggregation. To save some space, we report in the supplementary information file (see Table S4) the results confirming that the number of the executed iterations seems to be a reasonable trade off between the quality of solutions and the computational time.

To highlight the benefits of the adaptive aggregation framework when solving extensively large location problems, we visualise in Figure 4 values $\Phi(99.6, \alpha_{i_{last}})$ (triangles) from Table 7 obtained for the largest benchmark Slovakia. We compare these results with values $\Phi(99.6, \alpha_2)$ corresponding to the exact solutions obtained by the algorithm ZEBRA considering data aggregated to levels $\alpha_2 = \{94, 95, 96, 97, 98, 99\}$ prepared by the row-column aggregation method (circles). Values $\Phi(99.6, \alpha_2)$ clearly show that benefits of increasing the size of the aggregated location problem are considerable and grow together with the number of located facilities $\hat{p}$. However, almost in all cases, the benefits of the adaptive aggregation framework are significantly larger and these improvements were achieved on problems aggregated to higher level of $\alpha_{i_{last}}$.

4.4. Application to the lexicographic minimax problem

The lexicographic minimax approach interatively minimizes the maximum distance from DPs to the closest facilities. Therefore, it is likely that DPs, that have larger distance to the closest facility than is the distance from their ADP to the closest facility may cause aggregation errors. Therefore, to apply the adaptive aggregation framework to the lexicographic minimax problem, we customized phase 5 in the following way:

Phase 5: Identify ADPs that if disaggregated may affect the positions of located facilities. For each located facility f we define the distance d_f^{max} as the maximum distance between an ADP and the assigned facility f. Then all ADPs that include at least one DP that has distance to the closest facility f larger than d_f^{max} are moved from the subset $\mathcal{B}$ into the subset $\mathcal{A}$.

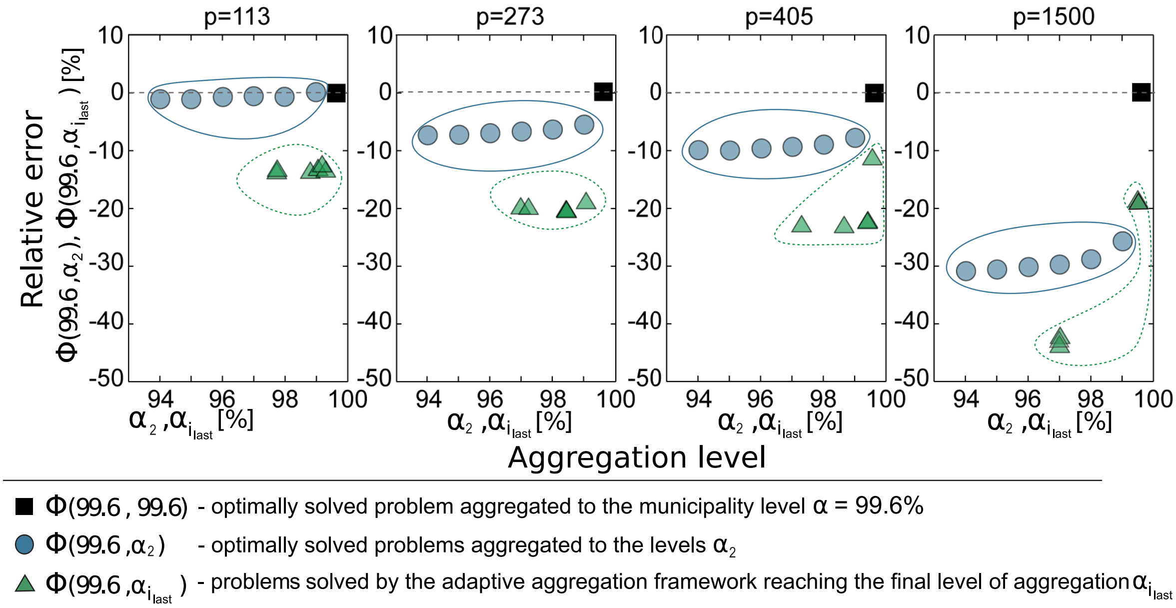

Figure 4: Comparison of the adaptive aggregation framework with the row-column aggregation method when applying them to the p-median problem and large benchmark Slovakia. We visualise values of the relative change in the value of the objective function $\Phi(99.6, \alpha_{i_{last}})$ obtained by the adaptive aggregation framework (triangles), where $\alpha_{i_{last}}$ is the final level of aggregation and values $\Phi(99.6, \alpha_2)$ obtained by the exact algorithm (circles), while aggregating the problem to the levels $\alpha_2 = \{94, 95, 96, 97, 98, 99\}$ using the row-column aggregation method. We use transparent symbols to facilitate identification of overlaps.

To solve the lexicographic minimax problem in the phase 2, we use as the parameter $\hat{A}\_p$ the approximative algorithm A-LEX (Buzna et al., 2014). In the phase 3, to minimise the source D errors, we use as the parameter $\hat{A}\_1$ the enumeration algorithm to solve the 1-centre problem. The lexicographic minimax problem is computationally much more demanding than the p-median problem, therefore, we did not apply the adaptive aggregation framework to the largest benchmark Slovakia.

4.4.1. Performance evaluation

To evaluate the lexicographic minimax problem is more complex than the p-median problem. We need to quantify the degree of equity among all distances between DPs and the closest facilities and at the same the proximity of located facilities to DPs should be assessed. To evaluate the degree of equity, we follow the performance indicators used in Ogryczak (1997). The important aspect in the lexicographic minimax approach is the minimisation of the maximal distance between a DP and its closest facility. Therefore, the relative difference Φ_{MAX} in maximal distances of two solutions is evaluated as follows:

$$\Phi_{MAX}(\alpha_I, \alpha_{II}) = \frac{b_{MAX}(\mathbf{x}^{\alpha_{II}}, \mathbf{y}^{\alpha_{II}}) - b_{MAX}(\mathbf{x}^{\alpha_I}, \mathbf{y}^{\alpha_I})}{b_{MAX}(\mathbf{x}^{\alpha_I}, \mathbf{y}^{\alpha_I})} 100\%, \tag{14}$$

where $b_{MAX}(\mathbf{x}^{\alpha}, \mathbf{y}^{\alpha})$ is the value of the maximum distance from an unaggregated DP to the closest facility in the solution $(\mathbf{x}^{\alpha}, \mathbf{y}^{\alpha})$ to the problem aggregated to the level of α. Further, as an equity measure, we define the relative difference Φ_{GINI} between two solutions as:

$$\Phi_{GINI}(\alpha_I, \alpha_{II}) = \frac{g(\mathbf{x}^{\alpha_{II}}, \mathbf{y}^{\alpha_{II}}) - g(\mathbf{x}^{\alpha_I}, \mathbf{y}^{\alpha_I})}{g(\mathbf{x}^{\alpha_I}, \mathbf{y}^{\alpha_I})} 100\%, \tag{15}$$

where $g(\mathbf{x}^\alpha, \mathbf{y}^\alpha)$ is the value of the gini coefficient computed by taking all distances from unaggregated DPs to their closest facilities. The gini coefficient is a measure of inequality (Ullah and Giles, 1998). Let b_j denote the distance from the customer j to the closest facility in the solution $(\mathbf{x}^\alpha, \mathbf{y}^\alpha)$ and we denote $\langle b \rangle$ as its mean. For the set $\{b_j, j = 1, 2, \ldots, n\}$, the gini coefficient $g(\mathbf{x}^\alpha, \mathbf{y}^\alpha)$, can be estimated by a sample mean:

$$g(\mathbf{x}^\alpha, \mathbf{y}^\alpha) = \frac{\sum_{k=1}^{n} \sum_{l=1}^{n} |b_k - b_l|}{2n^2 \langle b \rangle}. \tag{16}$$

4.4.2. Benchmark Košice

This section, in the same spirit as in the case of p-median problem, evaluates how the presence of phases affects the time efficiency and the solution quality trade-off and the role of parameters $\hat{\alpha}_1$ and $\hat{\epsilon}$. We compare three values of $\hat{\alpha}_1$: 99%, 90% and 75% and three values of $\hat{\epsilon}$: 0, 1 and 2 km. The benchmark Košice is reasonably small, and thus we set $\hat{\alpha}_{max} = 0\%$ and $\hat{i}_{max} = \infty$ not to limit the number of iterations. The values of the performance indicators are calculated by contrasting the values obtained with the adaptive aggregation framework with the A-LEX algorithm.

Table 8 shows numerical results obtained for the benchmark Košice. If the values of the maximum distance between the customer DP location and the closest located facility are not optimal, the quality of the solution is strongly deteriorated. The version V1 has never reached the value $\Phi_{MAX}(0, \alpha_{i_{last}}) = 0$ and versions V2 and V3 perform similarly, thus we conclude that phase 5 has stronger effect on the quality of the solutions than in the case of the p-median problem, and it is more relevant than the elimination of source errors. The values of the parameter $\hat{\epsilon} > 0$ contribute to the higher quality of solutions, although the effect is not so pronounced. The time efficiency lowers as the parameter values $\hat{\epsilon}$ and $\hat{\alpha}_1$ are growing. Although, when they are selected properly, the adaptive aggregation framework can find solutions having $\Phi_{MAX}(0, \alpha_{i_{last}}) = 0\%$ in shorter time than the algorithm A-LEX. We conducted similar numerical experiments with the benchmark Partizánske that further support our conclusions (see Table S7 in the supplementary information file). Results obtained on benchmarks d2103 and pcb3038, where the Euclidian distances are used (see Tables S5 and S6 in the supplementary information file), show that consistently only the version V2 provides very good results. Version V1 and occasionally also version V3 produce high values of $\Phi_{MAX}(0, \alpha_{i_{last}})$, indicating low quality solutions. Thus, the results confirm that phase 5 plays an important role and elimination of source errors may sometimes lead to even counterproductive effects. Independently on the way how the distances are computed the results of numerical experiments on small problems show that the V2 is the most reliable version of the algorithm when solving the lexicographic minimax location problem.

4.4.3. Benchmark Žilina

The experiments with smaller benchmark Košice already illustrated that the version V1 is not competitive with V2 and V3 and therefore here we do not report the results for V1. The algorithm A-LEX is not able to compute problems of this size, therefore, we evaluate the performance of the adaptive aggregation framework by comparing it to the algorithm A-LEX on aggregated problems. Again, we use the problem aggregated to the level of the individual municipalities $\alpha = 99.6\%$ and the problem aggregated to $\alpha = 80.0\%$ which can be considered as very large but still computable with the algorithm A-LEX. We initiate the framework by using the problem aggregated to the individual municipalities, and we set the maximum computational time to 10 days and $\hat{\alpha}_{max}$ to 80%. We set the numbers of the located facilities to the same values as for the p-median problem, hence, $\hat{p} = 10$, 35, 50 and 190. The results of the computational experiments are summarised in Table 9.

In all the cases the adaptive aggregation framework provides significant improvements in the quality of the solution as well as in the equity. The reduction of the maximum distance increases by growing the number of the located facilities $\hat{p}$, and it ranges from 3% to 50%. It is noteworthy that the reduction with respect to the problem reduced to the level of municipalities leads to the values of Φ_{MAX} ranging from 38% to 80%. In absolute values, the adaptive aggregation framework for $\hat{p} = 190$ reduces the maximum distance about 22 km compared to the solution obtained by the algorithm A-LEX on the problem reduced

Table 8: Results of the numerical experiments after applying the adaptive aggregation framework to the lexicographic minimax problem and benchmark Košice. The symbol $\alpha_{i_{last}}$ denotes the value of the reduction coefficient after the last iteration i_{last} of the framework. We evaluate the relative difference in maximal distances Φ_{MAX}, the relative difference in gini coefficients Φ_{GINI} and the relative time efficiency τ.

[km]	[%]	Indicator	$\hat{p}=10$			$\hat{p}=20$			$\hat{p}=40$		
			V1	**V2**	**V3**	**V1**	**V2**	**V3**	**V1**	**V2**	**V3**
$\hat{\epsilon}=0$		i_{last}	7	22	14	10	14	14	13	18	21
		$\alpha_{i_{last}}[\%]$	97.8	92.3	91.7	96.0	86.7	85.8	92.6	78.1	78.0
	$\hat{\alpha_1}=99$	$\Phi_{MAX}(0,\alpha_{i_{last}})[\%]$	137.1	11.29	11.29	227.5	5	0	137.93	0	0
		$\Phi_{GINI}(0,\alpha_{i_{last}})[\%]$	21.58	1.87	4	5.27	-3.9	-3.82	7.59	0.52	0.08
		$\tau(0,\alpha_{i_{last}})[\%]$	-99.88	-96.11	-94	-98.09	-79.03	-66.09	-98.22	-55.58	-47.82
		i_{last}	5	24	23	6	18	19	5	18	16
		$\alpha_{i_{last}}[\%]$	89.3	87.1	87.3	88.7	84.0	83.6	86.9	76.7	75.9
	$\hat{\alpha_1}=90$	$\Phi_{MAX}(0,\alpha_{i_{last}})[\%]$	16.13	1.61	0	25	2.5	0	37.93	0	0
		$\Phi_{GINI}(0,\alpha_{i_{last}})[\%]$	4.12	0.08	-0.27	-4.94	-1.31	1.23	3.3	-0.12	-0.6
		$\tau(0,\alpha_{i_{last}})[\%]$	-98.4	-91.4	-89.27	-95.3	-65.05	-56.99	-95.22	-57.97	-61
		i_{last}	4	11	14	5	12	12	8	14	14
		$\alpha_{i_{last}}[\%]$	74.5	72.7	72.4	73.7	70.6	70.1	70.1	66.3	66.0
	$\hat{\alpha_1}=75$	$\Phi_{MAX}(0,\alpha_{i_{last}})[\%]$	3.23	0	0	5	2.5	0	20.69	0	0
		$\Phi_{GINI}(0,\alpha_{i_{last}})[\%]$	0.88	-0.04	-0.23	-0.72	-2.43	-3.15	0.91	0.24	-0.99
		$\tau(0,\alpha_{i_{last}})[\%]$	-90.58	-46.83	-42.91	-65.75	29.59	16.24	-70.71	4.23	-6.46
$\hat{\epsilon}=1$		i_{last}	11	18	15	12	20	18	10	21	23
		$\alpha_{i_{last}}[\%]$	87.3	76.6	78.6	74.8	55.6	61.9	50.3	30.6	32.0
	$\hat{\alpha_1}=99$	$\Phi_{MAX}(0,\alpha_{i_{last}})[\%]$	138.71	0	0	100	0	0	162.07	0	0
		$\Phi_{GINI(0,\alpha_{i_{last}})}[\%]$	21.65	-0.04	-4.88	4.94	-2.35	-1.79	4.06	0.68	-1.03
		$\tau(0,\alpha_{i_{last}})[\%]$	-99.08	-79.26	-85.5	-77.59	197.08	47.1	-49.89	396.2	468.39
		i_{last}	8	25	22	9	17	16	10	17	17
		$\alpha_{i_{last}}[\%]$	78.0	72.5	73.9	70.5	61.4	62.3	40.8	32.1	32.5
	$\hat{\alpha_1}=90$	$\Phi_{MAX}(0,\alpha_{i_{last}})[\%]$	17.74	0	0	2.5	0	0	41.38	0	0
		$\Phi_{GINI}(0,\alpha_{i_{last}})[\%]$	6.52	-0.04	-1.18	-4.46	-2.51	-2.51	2.9	1.75	-1.11
		$\tau(0,\alpha_{i_{last}})[\%]$	-85.77	-64.31	-73.35	-69	8.82	5.13	24.83	213.61	256.57
		i_{last}	8	12	12	7	18	25	12	10	18
		$\alpha_{i_{last}}[\%]$	63.4	61.2	62.1	55.1	46.8	48.8	28.1	27.7	26.9
	$\hat{\alpha_1}=75$	$\Phi_{MAX}(0,\alpha_{i_{last}})[\%]$	8.06	0	0	5	0	0	20.69	0	0
		$\Phi_{GINI}(0,\alpha_{i_{last}})[\%]$	0.04	-0.04	-0.34	-3.19	0	-0.52	1.15	0.68	-0.99
		$\tau(0,\alpha_{i_{last}})[\%]$	-62.4	5.5	-1.32	23.55	365.18	592.25	258.67	208.19	471.98
$\hat{\epsilon}=2$		i_{last}	11	27	22	12	12	14	10	17	11
		$\alpha_{i_{last}}[\%]$	52.8	46.6	48.9	19.6	13.5	16.1	2.6	1.3	2.7
	$\hat{\alpha_1}=99$	$\Phi_{MAX}(0,\alpha_{i_{last}})[\%]$	74.19	0	0	15	0	0	10.34	0	0
		$\Phi_{GINI}(0,\alpha_{i_{last}})[\%]$	15.97	0	-0.27	-7.57	-0.12	-0.68	-0.36	0.28	-1.03
		$\tau(0,\alpha_{i_{last}})[\%]$	-73.45	247.74	130.59	400.44	514.26	763.7	429.08	782.4	418.05
		i_{last}	9	13	16	10	8	11	16	16	9
		$\alpha_{i_{last}}[\%]$	48.5	41.8	42.9	22.7	16.8	17.6	2.1	1.2	2.3
	$\hat{\alpha_1}=90$	$\Phi_{MAX}(0,\alpha_{i_{last}})[\%]$	9.68	0	0	7.5	0	0	10.34	0	0
		$\Phi_{GINI}(0,\alpha_{i_{last}})[\%]$	3.32	0	-1.18	-4.98	0	-0.72	0.76	1.27	0.83
		$\tau(0,\alpha_{i_{last}})[\%]$	-44.16	177.59	265.71	302.45	546.8	806	851.81	785.86	868.9
		i_{last}	8	14	10	8	14	14	6	7	7
		$\alpha_{i_{last}}[\%]$	43.7	41.7	43.1	16.5	15.5	17.4	3.4	5.5	3.8
	$\hat{\alpha_1}=75$	$\Phi_{MAX}(0,\alpha_{i_{last}})[\%]$	8.06	0	0	7.5	0	0	10.34	0	0
		$\Phi_{GINI}(0,\alpha_{i_{last}})[\%]$	-0.23	-0.19	-1.18	-4.46	0	-0.68	1.23	0.48	-0.99
		$\tau(0,\alpha_{i_{last}})[\%]$	-21.94	202	88.36	412.75	697.95	851.46	2587.43	158.19	653.93

Table 9: Results of the numerical experiments after applying the adaptive aggregation framework to the benchmark Žilina. By the symbol $\alpha_{i_{last}}$ we denote the aggregation level after accomplishing the last iteration i_{last} of the framework. We evaluate the relative difference in maximal distances Φ_{MAX}, the relative difference in gini coefficients Φ_{GINI} and the computational time t.

[km]	Indicator	$\hat{p}=10$		$\hat{p}=35$		$\hat{p}=50$		$\hat{p}=190$	
		V2	**V3**	**V2**	**V3**	**V2**	**V3**	**V2**	**V3**
$\hat{\epsilon}=0$	i_{last}	15	9	14	8	10	8	8	7
	$\alpha_{i_{last}}[\%]$	98.35	98.29	95.81	95.18	94.93	95.18	87.23	87.11
	$\Phi_{MAX}(99.6\%, \alpha_{i_{last}})[\%]$	-38.68	-39.12	-62.24	-62.24	-68.37	-68.62	-78.72	-79.79
	$\Phi_{GINI}(99.6\%, \alpha_{i_{last}})[\%]$	-22.90	-21.46	-21.16	-20.90	-26.52	-25.77	-27.27	-27.43
	$\Phi_{MAX}(80.0\%, \alpha_{i_{last}})[\%]$	-3.46	-4.15	-26.00	-26.50	-12.68	-13.38	-48.28	-50.86
	$\Phi_{GINI}(80.0\%, \alpha_{i_{last}})[\%]$	-1.21	0.62	0.69	1.02	1.60	2.65	-0.35	-0.57
	$t[h]$	13.8	240.0	32.1	240.0	22.49	240.0	48.0	240.0
$\hat{\epsilon}=1$	i_{last}	23	9	16	6	12	6	5	5
	$\alpha_{i_{last}}[\%]$	96.77	97.16	90.20	95.64	87.81	94.54	80.00	80.00
	$\Phi_{MAX}(99.6\%, \alpha_{i_{last}})[\%]$	-39.12	-39.12	-62.24	-60.97	-68.62	-68.37	-79.08	-80.14
	$\Phi_{GINI}(99.6\%, \alpha_{i_{last}})[\%]$	-21.41	-21.49	-21.87	-20.71	-25.88	-26.74	-27.71	-27.61
	$\Phi_{MAX}(80.0\%, \alpha_{i_{last}})[\%]$	-4.15	-4.15	-26.00	-23.00	-13.38	-12.68	-49.14	-51.76
	$\Phi_{GINI}(80.0\%, \alpha_{i_{last}})[\%]$	0.70	0.59	-0.22	1.27	2.50	1.30	-0.96	-0.82
	$t[h]$	29.76	240.0	83.17	240	68.75	240	36.4	176.5

to $\alpha = 99.6\%$. With respect to the problem reduced to $\alpha = 80.0\%$ the maximum distance is reduced about 6 km. Strongly affected are not only the maximum distances, but all other distances.

The values of indicators Φ_{MAX} and Φ_{GINI} for the lexicographic minimax problem are often higher than indicator Φ for the p-median problem suggesting that the lexicographic minimax problem is more sensitive to the quality of input data than the p-median problem and the adaptive aggregation framework brings here larger benefits. The versions V2 and V3 provide solutions of comparable quality, with slight improvements in Φ_{GINI} that are in favour of V3. However, V2 requires less computational time. Thus, the experiments on the benchmark Žilina confirm that the effect of the elimination of source errors is very small. The lexicographic minimax problem is computationally significantly more demanding than the p-median problem and the benchmark Slovakia aggregated to the level of individual municipalities is already close to the limit of what the algorithm A-LEX can compute (Buzna et al., 2014). Therefore, we did not apply the adaptive aggregation framework to this benchmark.

5. Conclusions

We proposed the concept of the adaptive aggregation framework,which integrates the solving methods, aggregation, elimination and minimization of aggregation errors and iteratively re-aggregates the solved problem. To validate its benefits, we applied it to two location-allocation problems that use different basic types of optimization criteria. Both these problems are NP hard (García et al., 2011; Ogryczak, 1997). Solving methods, that we used in the phase 2, use as a component the exact algorithm ZEBRA. Thus, when counting the number of operations as a function of problem parameters, the complexity of the adaptive aggregation framework is NP. However, even algorithms with NP complexity can perform fast, if the problem size is kept limited. This simple fact in combination with improvements in the quality of solutions that are achieved when iteratively adapting the aggregated problem to high quality solutions are the primary causes behind the benefits that the adaptive aggregation framework brings. We constituted our benchmarks in a such way that we first compare the framework to the basic versions of solving algorithms to study the performance of the framework using the problems that approach the upper limit of sensible computational time. By the numerical experiments, we aim to reveal the importance of the individual phases as well as the suitable parameter values. We derived the following main conclusions:

- The adaptive aggregation framework outperforms the row-column aggregation method, algorithm ZEBRA and algorithm A-LEX, even when applying it to smaller (more aggregated) problems. Benefits of the framework grow with the level of the used aggregation.
- Versions where the elimination of source errors is skipped and the value of the parameter $\hat{\epsilon}$ is adjusted to the size of the problem (V1 for the p-median and V2 for the lexicographic minimax problem) are the most suitable for large location-allocation problems. Thus, the re-aggregation of ADPs that are identified in phases 4 and 5 is more beneficial than the elimination of the source errors (phases 1 and 3).
- Lexicographic minimax problem is more sensitive to the level of aggregation than the p-median problem. Thus, as expected, the magnitude of possible improvements that results from re-aggregation is affected by the complexity of the problem. The combination of the detailed data model and the adaptive aggregation framework enables to enhance the capabilities of conventional methods while significantly improving the values of the performance indicators by 12-45% for the p-median and by 38-80% for the lexicographic minimax problem.

On one hand, it is important to note that such large gains in the quality of solutions could be reached because we allow the facilities to be located at any DP of the original non-aggregated problem. This corresponds to the standard definition of the selected location problems. Thus, when disaggregating the problem, we do not only reduce the source errors but we also get more options where to locate the facilities. This is also one of the reasons why we obtain solutions with more favourable values of the performance indicators. In cases, when the set of the candidates for facility locations is more restricted, we can expect that the gains will be smaller.

On the other hand, the p-median and lexicographic minimax problems represent archetypal location problems, which integrate location and allocation decisions with the system efficiency and fairness objectives giving us the main reason why we have selected them. In the case of more complex problems it is necessary to enhance the mathematical formulation of the problem, which increases the computational complexity of the solving algorithms and creates more incentives for data aggregation. Currently, the available tools usually do not offer methods that can utilize the potential for improvements that is in the use of more accurate data by re-aggregating them within the optimization process, and we do not observe a breakthrough in the available optimization methods for NP-hard integer programming problems. Therefore, we believe that the concept of the adaptive aggregation framework can find practical applications in various areas of engineering, where large and detail datasets are more available than ever before.

Supporting information

File S1 Supplementary information file which includes complete tables with the numerical values of the defined performance indicators.

Acknowledgement

This work was supported by the research grants VEGA 1/0463/16 "Economically efficient charging infrastructure deployment for electric vehicles in smart cities and communities", APVV-15-0179 "Reliability of emergency systems on infrastructure with uncertain functionality of critical elements", TRENoP Strategic Research Area and it was facilitated by the FP 7 project ERAdiate [621386]"Enhancing Research and innovation dimensions of the University of Zilina in Intelligent Transport Systems". We thank Dirk Helbing from ETH Zurich for granting the access to the Brutus high-performance cluster and Karl Ernst Ambrosch, Erik Jenelius and Ľudmila Jánošíková for valuable comments and suggestions.

References

Achabal, D.D., 1982. MULTILOC : a multiple store location decision model. Journal of Retailing 58, 5–25.

An, Y., Zeng, B., Zhang, Y., Zhao, L., 2014. Reliable p-median facility location problem: two-stage robust models and algorithms. Transportation Research Part B: Methodological 64, 54–72.

Andersson, G., Francis, R.L., Normark, T., Rayco, M., 1998. Aggregation method experimentation for large-scale network location problems. Location Science 6, 25–39.

Assunção, R.M., Neves, M.C., Câmara, G., da Costa Freitas, C., 2006. Efficient regionalization techniques for socio-economic geographical units using minimum spanning trees. International Journal of Geographical Information Science 20, 797–811.

Avella, P., Boccia, M., Salerno, S., Vasilyev, I., 2012. An aggregation heuristic for large scale p-median problem. Computers & Operations Research 39, 1625–1632.

Batista e Silva, F., Gallego, J., Lavalle, C., 2013. A high-resolution population grid map for Europe. Journal of Maps 9, 16–28.

Bucarey, V., Ordóñez, F., Bassaletti, E., 2015. Applications of Location Analysis. Springer International Publishing, Cham. chapter Shape and Balance in Police Districting. pp. 329–347.

Buzna, L., Koháni, M., Janáček, J., 2014. An approximation algorithm for the facility location problem with lexicographic minimax objective. Journal of Applied Mathematics 2014.

Cebecauer, M., Buzna, L., 2015. Re-aggregation Heuristic for Large P-median Problems, in: Operations Research and Enterprise Systems. Springer, pp. 54–70.

Cebecauer, M., Buzna, L., 2017. Large-Scale Test Data Set for Location Problems. Data in Brief , submitted.

Cebecauer, M., Rosina, K., Buzna, L., 2016. Effects of demand estimates on the evaluation and optimality of service centre locations. International Journal of Geographical Information Science 30, 765–784.

Cooper, L., 1964. Heuristic Methods for Location-Allocation Problems. SIAM Review 6, 37–53.

Cooper, L., 1972. The Transportation-Location Problem. Operations Research 20, 94–108.

Current, J.R., Schilling, D.A., 1987. Elimination of Source A and B Errors in p-Median Location Problems. Geographical Analysis 19, 95–110.

D. L. Huff, 1962. Determination of intra-urban retail trade areas. Los Angeles: Univ. of California.

Daskin. M., S., 1995. Network and discrete location: Models, Algoritmhs and Applications. John Wiley & Sons.

Doerner, K., Focke, A., Gutjahr, W.J., 2007. Multicriteria tour planning for mobile healthcare facilities in a developing country. European Journal of Operational Research 179, 1078–1096.

Drezner, Z., 1992. A note on the Weber location problem. Annals of Operations Research 40, 153–161.

Drezner, Z., 1995. Facility location: A survey of Applications and Methods. Springer Verlag.

Eiselt, H.A., Marianov, V., 2011. Foundations of Location Analysis. International Series in Operations Research and Management Science, Springer, Science + Business.

Eiselt, H.A., Marianov, V., 2015. Applications of Location Analysis. International Series in Operations Research and Management Science, Springer, Science + Business.

Erkut, E., Bozkaya, B., 1999. Analysis of aggregation errors for the p-median problem. Computers & Operations Research 26, 1075–1096.

Erkut, E., Neuman, S., 1992. A multiobjective model for locating undesirable facilities. Annals of Operations Research 40, 209–227.

Erlenkotter, D., 1978. A Dual-Based Procedure for Uncapacitated Facility Location. Operations Research 26, 992–1009.

Farahani, R.Z., SteadieSeifi, M., Asgari, N., 2010. Multiple criteria facility location problems: A survey. Applied Mathematical Modelling 34, 1689–1709.

Fei, X., Mahmassani, H.S., 2011. Structural analysis of near-optimal sensor locations for a stochastic large-scale network. Transportation Research Part C: Emerging Technologies 19, 440–453.

Fotheringham, A.S., Densham, P.J., Curtis, A., 1995. The Zone Definition Problem in Location-Allocation Modeling. Geographical Analysis 27, 60–77.

Frade, I., Ribeiro, A., Gonçalves, G., Antunes, A., 2011. Optimal location of charging stations for electric vehicles in a neighborhood in Lisbon, Portugal. Transportation research record: journal of the transportation research board , 91–98.

Francis, R., Lowe, T., Rayco, M., 1996. Row-column aggregation for rectilinear distance p-median problems. Transportation Science 30, 160–174.

Francis, R., Lowe, T., Rayco, M., Tamir, A., 2009. Aggregation error for location models: survey and analysis. Annals of Operations Research 167, 171–208.

García, S., Labbé, M., Marín, A., 2011. Solving Large p-Median Problems with a Radius Formulation. INFORMS Journal on Computing 23, 546–556.

Gentili, M., Mirchandani, B.P., 2015. Applications of Location Analysis. Springer International Publishing. chapter Locating Vehicle Identification Sensors for Travel Time Information. pp. 307–327.

George, J., ReVelle, C., 2003. Bi-Objective Median Subtree Location Problems. Annals of Operations Research 122, 219–232.

Goodchild, M.F., 1979. The Aggregation Problem in Location-Allocation. Geographical Analysis 11, 240–255.

Gupta Sachin, Chintagunta Pradeep, Kaul Anil, Wittink Dick R., 1996. Do Household Scanner Data Provide Representative Inferences from Brand Choices: A Comparison with Store Data. Journal of Marketing Research 33, 383–398.

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.

Hakimi, S.L., 1965. Optimum Distribution of Switching Centers in a Communication Network and Some Related Graph Theoretic Problems. Operations Research 13, pp. 462–475.

Hamacher, H.W., Labbé, M., Nickel, S., Skriver, A.J., 2002. Multicriteria Semi-Obnoxious Network Location Problems (MSNLP) with Sum and Center Objectives. Annals of Operations Research 110, 33–53.

Hillsman, E., Rhoda, R., 1978. Errors in measuring distances from populations to service centers. The Annals of Regional Science 12, 74–88.

Hodgson, M.J., Neuman, S., 1993. A GIS approach to eliminating source C aggregation error in p-median models. Computers & Operations Research .

Hodgson, M.J., Shmulevitz, F., Körkel, M., 1997. Aggregation error effects on the discrete-space p-median model: The case of Edmonton, Canada. Canadian Geographer / Le Géographe canadien 41, 415–428.

Janáček, J., Jánošíková, L., Buzna, L., 2012. Optimized Design of Large-Scale Social Welfare Supporting Systems on Complex Networks, in: Thai, M.T., Pardalos, P.M. (Eds.), Handbook of Optimization in Complex Networks. Springer US. volume 57 of *Springer Optimization and Its Applications*, pp. 337–361.

Lawrence M. Ostresh, J., 1978. On the Convergence of a Class of Iterative Methods for Solving the Weber Location Problem. Operations Research 26, 597–609.

Maranzana, F.E., 1964. On the Location of Supply Points to Minimize Transport Costs. Journal of the Operational Research Society 15, 261–270.

Marianov, V., Ríos, M., Icaza, M.J., 2008. Facility location for market capture when users rank facilities by shorter travel and waiting times. European Journal of Operational Research 191, 32–44.

McLay, L., Moore, H., 2012. Hanover County Improves Its Response to Emergency Medical 911 Patients. Interfaces 42, 380–394.

Mladenović, N., Brimberg, J., Hansen, P., Moreno-Pérez, J.A., 2007. The p-median problem: A survey of metaheuristic approaches. European Journal of Operational Research 179, 927–939.

Nickel, S., Puerto, J., Rodríguez-Chía, A.M., Weissler, A., 2005. Multicriteria Planar Ordered Median Problems. Journal of Optimization Theory and Applications 126, 657–683.

Ogryczak, W., 1997. On the lexicographic minmax approach to location problems. European Journal of Operational Research 100, 566–585.

Ogryczak, W., 1999. On the distribution approach to location problems. Computers & Industrial Engineering 37, 595–612.

Ogryczak, W., Śliwiński, T., 2006. On Direct Methods for Lexicographic Min-Max Optimization. Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 802–811.

Openshaw, S., 1977. A geographical solution to scale and aggregation problems in region-building, partitioning and spatial modelling. Transactions of the institute of british geographers , 459–472.

Openshaw, S., Rao, L., 1995. Algorithms for reengineering 1991 Census geography. Environment and planning A 27, 425–446.

Ouyang, Y., Wang, Z., Yang, H., 2015. Facility location design under continuous traffic equilibrium. Transportation Research Part B: Methodological 81, Part 1, 18–33.

Pelot, R., Akbari, A., Li, L., 2015. Applications of Location Analysis. Springer International Publishing. chapter Vessel Location Modeling for Maritime Search and Rescue. pp. 369–402.

Perl, J., Daskin, M.S., 1985. A warehouse location-routing problem. Transportation Research Part B: Methodological 19, 381–396.

Ponboon, S., Qureshi, A.G., Taniguchi, E., 2016. Branch-and-price algorithm for the location-routing problem with time windows. Transportation Research Part E: Logistics and Transportation Review 86, 1–19.

Prodhon, C., Prins, C., 2014. A survey of recent research on location-routing problems. European Journal of Operational Research 238, 1–17.

Romero, N., Nozick, L.K., Xu, N., 2016. Hazmat facility location and routing analysis with explicit consideration of equity using the Gini coefficient. Transportation Research Part E: Logistics and Transportation Review 89, 165–181.

Salazar-Aguilar, M.A., Ríos-Mercado, R.Z., González-Velarde, J.L., 2011. A bi-objective programming model for designing compact and balanced territories in commercial districting. Transportation Research Part C: Emerging Technologies 19, 885–895.

Snyder, L.V., Daskin, M.S., 2005. Reliability Models for Facility Location: The Expected Failure Cost Case. Transportation Science 39, 400–416.

Teitz, M.B., Bart, P., 1968. Heuristic Methods for Estimating the Generalized Vertex Median of a Weighted Graph. Oper. Res. 16, 955–961.

Tong, D., Murray, A.T., 2017. Location Analysis: Developments on the Horizon. Springer International Publishing, Cham. pp. 193–208.

Ullah, A., Giles, D.E.A., 1998. Handbook of Applied Economic Statistics. CRC Press, New York.

Whitaker, R., 1983. A Fast Algorithm For The Greedy Interchange For Large-Scale Clustering And Median Location Problems. INFOR: Information Systems and Operational Research 21, 95–108.

Yun, L., Qin, Y., Fan, H., Ji, C., Li, X., Jia, L., 2015. A reliability model for facility location design under imperfect information. Transportation Research Part B: Methodological 81, Part 2, 596–615.