

Genetic Algorithm for Multi-objective p-Median Problem

Panwadee Tangpattanakul

Geo-Informatics and Space Technology Development Agency (Public Organization)
120, The Government Complex (Building B), Chaeng Wattana Road,
Laksi, Bangkok 10210, Thailand, panwadee@gistda.or.th

Abstract

This paper presents a nondominated sorting genetic algorithm II, or NSGA-II for solving a multi-objective p-Median problem which is a problem of finding p location points to install facilities from a set of m candidates. This problem considers two objectives, which are to minimize the sum of the distances from each customer to the nearest facility and to minimize the sum of the costs to install each facility in the selected location points. NSGA-II is an efficient algorithm in the area of multi-objective optimization. For the genetic algorithm operations, a binary tournament is used in the selection process; a biased gene value assignment is applied for the crossover; and a random generation is used in the mutation step. Experiments are conducted on generated instances. Hypervolume values of the approximate Pareto fronts are computed. They show that the proposed algorithm can be used for solving the multi-objective p-Median.

Keywords: p-Median problem, genetic algorithm, multi-objective optimization

1. Introduction

This paper proposed an efficient multi-objective optimization algorithm, which is called a nondominated sorting genetic algorithm II (NSGA-II), to solve the bi-objective p-Median problem. We experiment on instances, which consider different numbers of candidates and numbers of facilities.

The p-Median problem is a network-based location problem in the area of location analysis [1]-[2]. It needs to find p locals, which use to install facilities, from a set of m candidates. This problem is an optimization problem that needs to reach the objective functions and satisfy the constraints. The p-Median problem is classified as NP-hard problem [3]. Generally, the p-median problem considers only one objective, which is to minimize the sum of the distances from each customer to the nearest facility. However, two objectives are considered simultaneously in this paper. The second objective is to minimize the sum of the costs to install each facility in the selected location point. Thus, the multi-objective optimization algorithms can be applied to solve the considered problem.

The multi-objective optimization is the area of interested among researchers as it has been rapidly developed until nowadays because the formulation of multi-objective optimization assimilates to a real world question. However, the methods for solving multi-

objective optimization problems are much more complicated, and they require efficient optimization strategies. Several heuristic methods have good performance for solving this type of problems. Therefore, a heuristic method, which is nondominated sorting genetic algorithm or NSGA-II, is used to solve the considered problem in this work.

In the recent years, the single objective p-Median problem was solved by using the genetic algorithms, and it obtained good solutions [4]-[5]. For the multi-objective p-Median problem, a multi-objective genetic algorithm (MOGA) [6] and an iterated local search heuristic (ILS) [7] were proposed for solving the problem.

The article is organized as follows. Section 2 explains the multi-objective p-Median problem. The formulation of the considered problem and the difficulty of the multi-objective optimization problem are presented. NSGA-II and the strategies for the selection, crossover, and mutation process are presented in Section 3. Section 4 reports the computational results. Finally, conclusions are discussed in Section 5.

2. Multi-objective p-Median problem

Generally, the p-Median problem considers only one objective, which is to minimize the sum of the distance from each customer to the nearest facility. However, the second objective is added in our problem to minimize the sum of the costs to install each facility in the selected location point. Thus, the formulation of this problem is as follows [6]:

$$\min \sum_{i \in I} \sum_{j \in J} d_{ij} x_{ij} \quad (1)$$

$$\min \sum_{j \in J} f_j y_j \quad (2)$$

$$\text{subject to: } \sum_{j \in J} x_{ij} = 1, \quad i \in I \quad (3)$$

$$x_{ij} \leq y_j, \quad i \in I, j \in J \quad (4)$$

$$\sum_{j \in J} y_j = p \quad (5)$$

$$x_{ij}, y_j \in \{0,1\}, i \in I, j \in J. \quad (6)$$

The values of d_{ij} is the distance from customer i to facility at location point j , and f_j is a cost to install the facility at location point j . The decision variable x_{ij} equals 1, if customer i is serviced by facility at location

point j , otherwise, x_{ij} equals 0. The decision variable y_j equals 1, if the facility is installed at location point j , otherwise y_j equals 0.

The algorithms for solving a multi-objective optimization problem are studied. The fitness of each individual must be taken into account for all objective functions; therefore, we will not obtain only one optimal solution, but a set of solutions. The obtained tradeoff solutions are called Pareto optimal solutions. Because of this, the algorithms that use to solve the multi-objective optimization problem are more complicated than the algorithms for solving the single objective optimization problem. In this work, the nondominated sorting genetic algorithm II is proposed to solve the considered problem.

3. Nondominated sorting genetic algorithm II

Nondominated sorting genetic algorithm II or NSGA-II was proposed in [8]. The genetic algorithm is a population-based metaheuristic method which is developed from the natural survival concept. It operates by using several individuals in the population. Each individual consists of a chromosome, which represents one solution of the search space. For NSGA-II, the individuals in the population are classified into several layers of classification. NSGA-II uses the dominance relation in the Pareto sense for ranking the individuals. This is called the fast nondominated sorting. All nondominated individuals are classified in the first rank. The process continues until all individuals in the population are classified. Then, NSGA-II uses the crowding distance assignment for ensuring the diversity in the considered front. The crowding distance assignment sorts the individuals depending on the distances between the two adjacent points in the objective space. The individuals are ordered in the sorted set according to the distance values from high to low. For the boundary points, they always stay in the first positions of the sorted set, so that they are always selected.

Each chromosome represents one solution. In this work, the size of each chromosome is set equal to the value of p (number of facilities that need to be installed). Each gene represents the location point, which is selected to install the facility. Thus, the gene value is an integer in the interval $[1, m]$, where m is the number of candidate location points.

In the NSGA-II process, the initial chromosomes are generated for the first iteration. Two methods, which are a random generation and a distributed generation, are used. The random generation assigns randomly the gene values of each chromosome. The size of population (p_0) is fixed and it is set equal to 200. For the distributed generation, it is borrowed from [4]. The size of population is computed by using the idea that the population size should be proportional to the number of solutions. All candidates must be assigned in the genes of at less two chromosomes in the population. In Section

5, the results, which are obtained from these two initial population generation, are also compared.

To generate offspring of NSGA-II, the selection, crossover and mutation operators are used. For the selection step, the p_c offspring are generated. To generate each offspring, two parents are randomly chosen from the population, and the tournament selection is applied. The winner from the tournament selection becomes one offspring in the selection set. For the crossover process, the number of crossover offspring is set equal to p_c . A biased gene values assignment, which is borrowed from [9], is applied. Each offspring is built by using the crossover operator between a randomly selected chromosome from the nondominated set and another randomly selected chromosome from the whole current population. Each element in the offspring is obtained from the element of nondominated parent with the probability p_c . Otherwise, the element of offspring is copied from the other parent. For the mutation step, the offspring is generated by using the random generation. The number of mutation offspring is equal to p_m . The offspring set and the current population are combined. Then, the fast nondominated sorting and the crowding distance assignment are used for selecting p_0 best fitness individuals to become the new population as in Fig.1. The generations are repeated until a stopping criterion is satisfied.

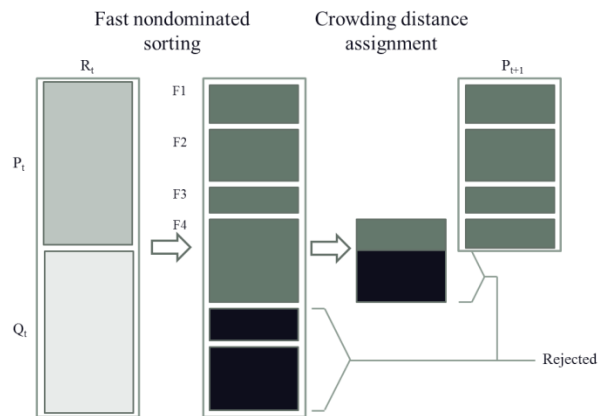


Fig. 1 NSGA-II procedure [8]

4. Computational results

The proposed algorithm is tested on the randomly generated instances. The format of instance names is set to "Instance_m_p" where m is the number of candidate location points, and p is the number of selected points. In this work, the candidate location points are set as same as the points of considered customers. In the instances, the numbers of candidate location points (m), which are 50, 100, 200, 300, and 402 and the numbers of selected points (p), which are 5, 10, 15, 20, 25, and 30, are considered.

Two methods of initial population generation are applied. The population size is set depending on the initial population generation method. The size of offspring sets from selection, crossover, and mutation

are equal to 0.3, 0.4, and 0.3 of the population size, respectively. The probability of nondominated element inheritance for crossover operation is set to 0.7. In each iteration, the hypervolume of the nondominated solutions set are computed. A reference point is the sum of the distances from each customer to the farthest facility for the first objective and the sum of p maximum costs to install the facilities for the second objective. For this work, the number of iterations since the last hypervolume values improvement is chosen to be a stopping criterion, and it is set to 500 iterations. It means that the algorithm will terminate, if hypervolume value improvement is stable for more than 500 iterations. The algorithm is implemented in C++, and thirty runs per instance are tested. Two methods which are a random generation and a distributed generation to generate the initial population are applied in NSGA-II. The set of testing instances consists of thirty instances, which consider different numbers of candidate location points (m) and different numbers of selected points (p). The results which are obtained from two methods of the initial population generation for the NSGA-II process, are compared. The hypervolume values of the approximate Pareto front are computed. Box plots of some instances, which illustrate the maximum value, the median value, the minimum value, and the interquartile range of the hypervolume values, and the average computation times, are presented in Fig. 2. The results of the random generation and distributed generation are compared. They show that the first spends more computation time than the second one for all instances. However, it obtains better hypervolume values, both on median value and standard deviation. Therefore, the random generation with the fixed population size can obtain better approximate Pareto front and the best approximate Pareto fronts of some tested instances are illustrated in Fig. 3. The figure shows that NSGA-II obtains many nondominated solutions on the approximate Pareto front. It depends on the decision makers to choose one preferred solution from the approximate Pareto front to be the best solution of this problem.

5. Conclusions and future works

A nondominated sorting genetic algorithms or NSGA-II is applied to solve a multi-objective p-Median problem using many chromosomes in the population. Each chromosome consists of several genes. Each gene represents one selected location point, which is assigned in the solution. For the initial population generation, two methods are used. In the process of NSGA-II, the selection, crossover, and mutation operators are applied. A binary tournament is used in the selection process; a biased gene value assignment is applied for the crossover; and the random generation is used in the mutation step. The experiments are conducted on randomly generated instances. The objectives of this problem are to minimize the sum of the distances from

each customer to the nearest facility and to minimize the sum of the costs to install each facility in the selected location points. The results, which are obtained from the two methods of initial population generation, are compared. The random generation method gives better solution than the distributed generation method, but it spends more computation times. Therefore in this paper, we show that NSGA-II can be used for solving the multi-objective p-Median problem.

In future works, this proposed algorithm may be applied to the location analysis problems with real GIS data, which can be applied to the future services (such as urban planning) in our organization, GISTDA. Moreover, other methods for genetic algorithm operators will be tested and the obtained results will be compared. We also plan to apply some evolutionary multi-objective optimization algorithms for solving the multi-objective p-Median problem.

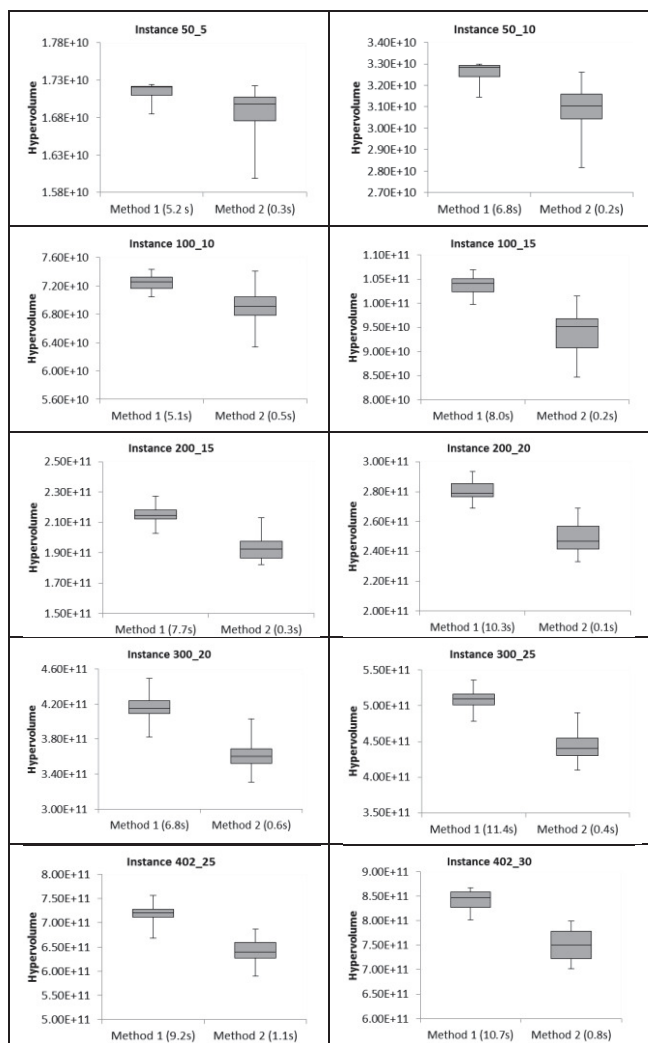


Fig. 2 Comparison of the hypervolume values of the approximate Pareto front and the average computation times between two methods of initial population generation

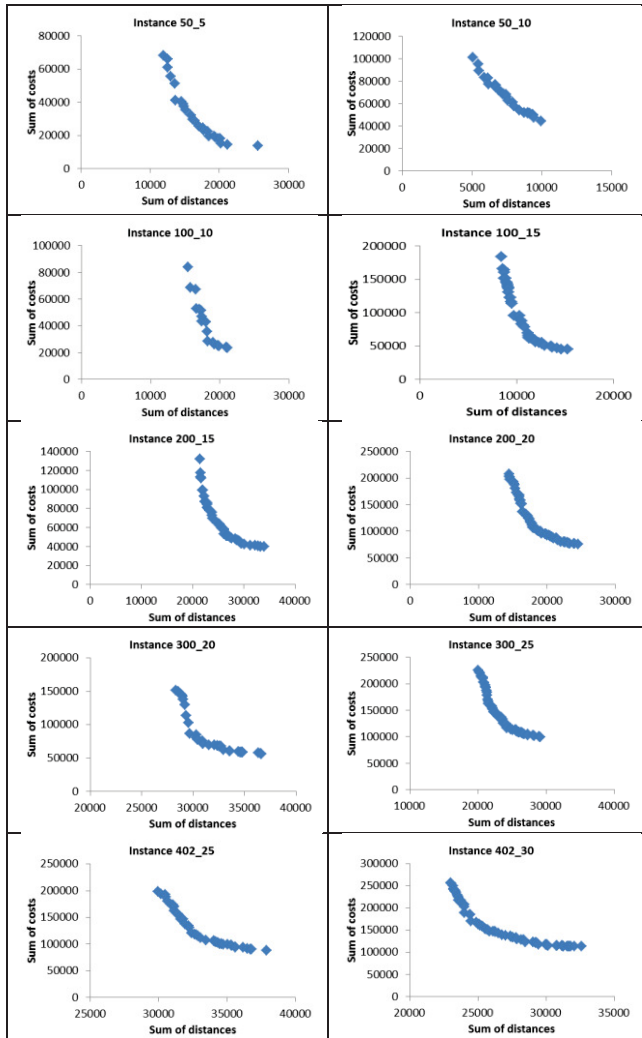


Fig. 3 The best approximate Pareto front of each instance

References

- [1] C.S. ReVelle and H.A. Eiselt, "Location analysis: A synthesis and survey," *European Journal of Operational Research*, vol. 165, pp. 1 – 19, 2005.
- [2] M.T. Melo et al., "Facility location and supply chain management – A review," *European Journal of Operational Research*, vol. 196, pp. 401 – 412, 2009.
- [3] N. Mladenovic et al., "The p-median problem: A survey of metaheuristic approaches," *European Journal of Operational Research*, vol. 179, pp. 927 – 939, 2007.
- [4] O. ALP et al., "An efficient genetic algorithm for the p-Median problem," *Annals of Operations Research*, vol. 122, pp. 21 – 42, 2003.
- [5] E.S. Correa et al., "A genetic algorithm for the p-median problem," in *Proceedings of 2001 Genetic and Evolutionary Computation Conf. (GECCO-2001)*, 2001, pp. 1268 – 1275.
- [6] J.E.C. Arroyo et al., "A multi-objective genetic algorithm with path relinking for the p-Median problem," In A. Kuri-Morales and G.R. Simari

(Eds.), IBERAMIA 2010, volume 6433 of *Lecture Notes in Artificial Intelligence (LNAI)*, page 70 – 79. Springer Berlin Heidelberg, 2010.

- [7] J.E.C. Arroyo et al., "A bi-objective iterated local search heuristic with path relinking for the p-Median problem," In R.H.C. Takahashi et al. (Eds.), *EMO 2011*, volume 6576 of *Lecture Notes in Computer Science (LNCS)*, page 492 – 504. Springer Berlin Heidelberg, 2011.
- [8] K. Deb et al., "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182 – 197, 2002.
- [9] J. F. Gonçalves and M. G. C. Resende, "Biased random-key genetic algorithms for combinatorial optimization," *Journal of Heuristics*, vol. 17, no. 5, pp. 487 – 525, 2011.



Panwadee Tangpattanakul obtained her bachelor's degree (first class honors) and master's degree in electrical engineering from Khon Kaen university, Thailand in 2005 and 2009, respectively. She received her Ph.D. in computer science from Institut National des Sciences Appliquées de Toulouse, France, in 2013. Since November 2013, she has been a researcher at Geo-Informatics and Space Technology Development Agency (Public organization), Thailand. Her research interests are focused on multi-objective optimization, operational research, metaheuristics, and location analysis.