Article

Solving a class of facility location problems using genetic algorithms

Sohail S. Chaudhry,[1] Shiwei He[2] and Peggy E. Chaudhry[3]

(1) Department of Decision and Information Technologies, College of Commerce and Finance, Villanova University, Villanova, PA 19085, USA
E-mail: sohail.chaudhry@villanova.edu
(2) Department of Transportation Management Engineering, Northern Jiaotong University, Beijing 100044, China
E-mail: shwhe@center.njtu.edu.cn
(3) Department of Management, College of Commerce and Finance, Villanova University, Villanova, PA 19085, USA
E-mail: peggy.chaudhry@villanova.edu

Abstract: *Locating p facilities to serve a number of customers is a problem in many areas of business. The problem is to determine p facility locations such that the weighted average distance traveled from all the demand points to their nearest facility sites is minimized. A variant of the p-median problem is one in which a maximum distance constraint is imposed between the demand point and its nearest facility location, also known as the p-median problem with maximum distance constraint. In this paper, we apply a fairly new methodology known as genetic algorithms to solve a relatively large sized constrained version of the p-median problem. We present our computational experience on the use of genetic algorithms for solving the constrained version of the p-median problem using two different data sets. Our comparative experimental experience shows that this solution procedure performs quite well compared with the results obtained from existing techniques.*

Keywords: facility location, *p*-median problem, genetic algorithms

1. Introduction

In this paper, a location problem known as the *p*-median problem will be studied. The *p*-median problem is to locate *p* facilities on a network such that the average distance traveled from the demand points to their nearest facility sites is minimized. The *p*-median problem was first introduced by Hakimi (1965). It is *np* hard, and so optimal solutions to large sized problems are difficult to obtain. Optimally locating public and private facilities such as schools, parks and distribution centers are typical examples of this problem. Practical applications of the *p*-median problem are reported in the literature and include locating audit offices for a comptroller (Fitzsimmons & Austin, 1983) and locating service teams for a utility company serving a large geographic region (Erkut *et al.*, 2000). Because of such diverse applications, the problem has received a considerable amount of attention from researchers who have developed a variety of solution procedures to solve it (e.g. see Teitz & Bart, 1968; Narula *et al.*, 1977; Captivo, 1991; Densham & Rushton, 1992; Choi & Chaudhry, 1993; Galvão, 1993; Chaudhry *et al.*, 1995; Murray & Church, 1996; Rolland *et al.*, 1996; Canós *et al.*, 1999; and textbooks by Daskin, 1995; Mirchandani & Francis, 1990).

During the past decade, there has been tremendous advancement in the field of artificial intelligence. Furthermore, in today's highly competitive business environment, it has become extremely important to have a knowledge-based view of the firm. Due to this evolution, there has been an increasing integration of artificial intelligence with business disciplines. These knowledge modeling techniques of artificial intelligence consist of expert systems, fuzzy systems, neural networks and genetic algorithms, among others (Hayes-Roth & Jacobstein, 1994).

Genetic algorithms have been successfully applied in solving a variety of business problems (Shin & Han, 1999; Li & Aggarwal, 2000; Manolas *et al.*, 2001; Chiu, 2002; Shin & Lee, 2002). The purpose of this paper is to demonstrate that the methodology known as genetic algorithms can be applied to solve the relatively large sized constrained version of the *p*-median problem. There has been some research on the application of genetic algorithms to solve the unconstrained version of the *p*-median problem with varied results (Hosage & Goodchild, 1986; Perez *et al.*, 1994; Estivill-Castro & Torres-Velazquez, 1999; Bozkaya *et al.*, 2000; Jaramillo *et al.*, 2002). However, to the best of our knowledge, no research paper has been published that explicitly considers a variant of the *p*-median problem in which an upper bound is imposed between a demand point and its nearest facility site. This problem is known as the constrained p-median problem (Moon & Chaudhry, 1984). In this paper, the genetic algorithm approach will be tested on the constrained version of the p-median problems using two different data sets. These two data sets were selected since they have been used in past location studies for which optimal solutions are readily available. Our computational experience will also be reported.

2. Problem formulation

To state the maximum distance constrained p-median problem mathematically we use the following notation and definitions: $M=\{1, 2, \ldots, m\}$ is the index set for facility sites; $N=\{1, 2, \ldots, n\}$ is the index set for demand points; d_{ij} is the shortest path distance from demand point i to a nearest facility site j; w_i is the demand at demand point i; x_{ij} is the fraction of w_i assigned to a facility site j; p is the number of facilities to be established; s is the constant maximum distance limit between a new facility site and its closest demand point; M_i is the set of sites which are within s distance units from demand point i, $\forall i \in N$, $M_i \subseteq M$.

The formulation of the maximum distance constrained p-median problem in which demand points and facility sites are restricted to finite sets N and M, respectively, is based on Toregas *et al.* (1971) and can be stated as follows.

$$\text{minimize}\, Z = \sum_{i \in N} \sum_{j \in M_i} w_i d_{ij} x_{ij}$$

subject to

$$\sum_{j \in M_i} x_{ij} = 1 \qquad \forall i \in N \tag{1}$$

$$x_{jj} \geq x_{ij} \qquad \forall i \in N,\, i \neq j,\, \forall j \in M_i \tag{2}$$

$$\sum_{j \in M} x_{jj} = p \tag{3}$$

$$x_{ij} \geq 0 \qquad \forall i \in N,\, \forall j \in M \tag{4}$$

$$x_{jj} = \{0, 1\} \qquad \forall j \in M \tag{5}$$

In the above formulation, the objective function is to minimize the weighted average distance traveled from demand points to facility sites. Constraint (1) ensures that the demand of each demand point is met by the facility sites within the maximum distance limit s. The constraint (2) guarantees that the demand of each node is allocated only to those vertices that are in the median set and the constraint (3) ensures that there are exactly p-median vertices.

3. Genetic algorithm approach

A genetic algorithm is a heuristic search procedure that is based on the natural process of evolution as in biological sciences and was first introduced by Holland (1975). As this highly adaptive evolutionary process progresses, the population genetics evolve in a given environment according to the natural behavior in which the fittest survive and the weakest are destroyed. Thus, the genes from the adept donor will propagate to other recipients during each successive generation, hence creating more apt offspring suitable for the defined environment. In optimization terms, the search algorithm improves the solution over generations as it progresses towards the optimum. Genetic algorithms have been successfully applied in solving a variety of optimization problems that are difficult to solve, including the traveling salesperson problem, job-shop scheduling problems, vehicle routing problems, airline crew scheduling problems, optimizing the sequence of advertisements within a commercial break at a British television station, and painting trucks at a General Motors production facility, among others (Chaudhry *et al.*, 2000). For a more thorough coverage of genetic algorithms, the reader is referred to the excellent textbook by Goldberg (1989).

In terms of an optimization problem, the genetic algorithm approach is summarized as follows. At any given point in time, the genetic algorithm generates a population of possible candidate solutions. Initially, the population size is chosen at random. However, this choice typically depends on the characteristics of the problem. Each population component is a string entity of chromosomes that represents a possible solution to the problem. The population components are evaluated based on a given objective function. Highly fit population components are given the chance to reproduce through a crossover process with other highly fit population elements by exchanging pieces of their genetic information. This process produces 'offspring' or new solutions to the optimization problem based upon the high performance characteristics of the parents. A mutation process prevents premature loss of important information by randomly altering bits within a chromosome. This procedure continues until a satisfactory solution is achieved.

It has been shown (Michalewicz, 1994) that it is important to design genetic algorithms based on the problem specified. We propose a direct encoding method and a hybrid approach of a genetic algorithm with local search techniques to enhance the power of conventional genetic algorithms.

3.1. *Encoding scheme (representation)*

As the traditional binary representation is not well suited for the p-median problem (Booker, 1987; Bozkaya *et al.*, 2000), a direct encoding method is adopted where the location facilities are represented as chromosomes:

$$S = \{s_1, s_2, \ldots, s_p\}$$

where s_i denotes the ith facility points, $s_i \in L$ and $L = \{l_1, l_2, \ldots, l_n\}$.

3.2. *Fitness evaluation*

Let Z_{best} denote the weighted average distance traveled from demand points to facility sites of the best chromosome, Z_k denote the weighted average distance traveled from demand points to facility sites of the kth chromosome, and λ is a constant. When the distance between a demand point and a facility site exceeds the maximum distance limit s, a large number M will be added to the objective function. The fitness ε_k for the kth chromosome is calculated as follows:

$$\varepsilon_k = \lambda Z_k / Z_{\text{best}}$$

3.3. *Genetic operators*

3.3.1. Selection strategy We give more reproductive chances to the populations that are the most fit. The M highest-rated chromosomes will be preserved to the next generation.

3.3.2. *Crossover operator*

(1) Two parents are selected randomly from the population. For example, the two parents are

$$P_1: \ 1\,2\,5\,6\,7\,9$$
$$P_2: \ 3\,4\,7\,8\,9\,10$$

(2) The same segments from two parents are copied into the front of P_1 (P_2). The remaining genes (separated by |) are chosen randomly to get the crossover subsegments (marked by underlining). We have

$$C_1': 7\,9|1\,\underline{2\,5}\,6$$
$$C_2': 7\,9|3\,\underline{4\,8}\,10$$

(3) Exchange the crossover subsegments of two parents to get two offspring:

$$C_1'': 7\,9|1\,\underline{4\,8}\,6$$
$$C_2'': 7\,9|3\,\underline{2\,5}\,10$$

(4) After adjusting the sequence, the final offspring are

$$C_1: 1\,4\,6\,7\,8\,9$$
$$C_2: 2\,3\,5\,7\,9\,10$$

3.3.3. Mutation operator The mutation operator adopts a single node exchanging method. First, the going out index s_i in the solution $S = \{s_1, s_2, \ldots, s_p\}$ is selected randomly. Then, an index j in node set L is also selected randomly. If $j \notin S$ and $j \neq s_i$, exchange two nodes. The operation will avoid the indices already in the solution. For example,

$$O_2: \ 4\,2\,7\,5\,6$$

After randomly selecting the mutation point number 3, i.e. 7, we select another mutation point $j = 9$ in $L = \{1\,2\,3\,4\,5\,6\,7\,8\,9\}$. The final chromosome is mutated into

$$O_2: 4\,2\,9\,5\,6$$

3.3.4. Termination condition A predefined maximum number of generations or time limit is reached.

3.4. *Genetic algorithm*

The detailed algorithm is described as follows:

```
PROCEDURE GA
begin
  k := 0;
  initialize P(k);
  calculate the fitness value ε of chromosomes in P(k);
  while(not termination-condition) do
    begin
    k := k + 1;
    select P(k) from P(k−1);
    crossover;
     if (not mutation-condition) then
       {mutation;}
    endif
  calculate the fitness value ε of chromosomes in P(k);
  end
end
```

4. Computational study

The genetic algorithm was coded in C and run on a Pentium III microcomputer. For the computational performance testing of the genetic algorithm, two data sets were used. The first was the 30-community data set based on intercity distances in New York State, USA (Toregas *et al.*, 1971) with population weights taken from Khumawala (1972). The second data set was generated based on the distances between the 150 largest US cities, in terms of population size (Choi & Chaudhry, 1993).

For the 30-community data set, 15 problems were generated for which optimal solutions were available (Chaudhry *et al.*, 1995). The number of facilities to be opened, p, was set to 10, 12, 15, 20 and 25, whereas the

Table 1: *Results for the 30-community data set*

Number of facilities p	*Maximum distance s*	*Optimal solution*	*p facility set*
10	55	—	No feasible solution
	61	17116	{9, 17, 1, 3, 28, 12, 29, 25, 11, 16}
	65	16166	{11, 3, 16, 12, 15, 25, 28, 18, 1, 9}
12	55	—	No feasible solution
	61	12439	{9, 17, 1, 3, 28, 12, 29, 25, 11, 16}
	65	12439	{28, 25, 11, 12, 3, 15, 18, 17, 6, 5, 16, 4}
15	55	12252[a]	{24, 3, 29, 9, 26, 6, 8, 11, 15, 17, 21, 10, 13, 25, 4}
	61	8300	{15, 17, 12, 3, 28, 6, 11, 2, 18, 5, 8, 10, 27, 7, 4}
	65	8300	{10, 6, 15, 28, 5, 12, 11, 17, 18, 3, 7, 2, 4, 27, 8}
20	55	3893	{20, 27, 8, 17, 7, 2, 3, 1, 15, 11, 12, 6, 29, 13, 16, 21, 14, 18, 28, 5}
	61	3833	{12, 5, 3, 17, 15, 28, 18, 6, 7, 11, 8, 2, 14, 21, 13, 10, 29, 23, 1, 27}
	65	3833	{3, 11, 28, 29, 8, 17, 6, 12, 2, 7, 18, 13, 21, 14, 15, 5, 1, 10, 23, 27}
25	55	1098	{28, 13, 12, 9, 2, 7, 14, 1, 24, 10, 8, 23, 29, 18, 11, 6, 17, 21, 3, 15, 5, 20, 19, 16, 25}
	61	1098	{28, 13, 12, 9, 2, 7, 14, 1, 24, 10, 8, 23, 29, 18, 11, 6, 17, 21, 3, 15, 5, 20, 19, 16, 25}
	65	1098	{28, 13, 12, 9, 2, 7, 14, 1, 24, 10, 8, 23, 29, 18, 11, 6, 17, 21, 3, 15, 5, 20, 19, 16, 25}

[a]Optimal solution for this instance, 10,854.

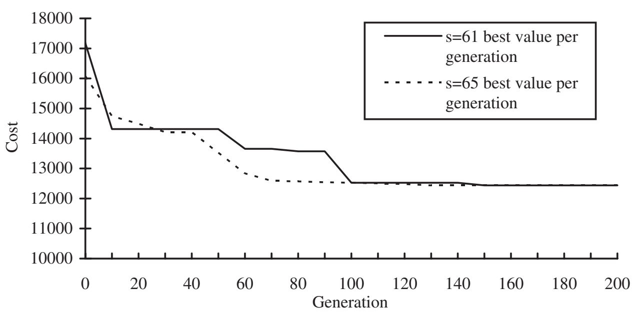

Figure 1: *Evaluation curves.*

maximum distance s was set to 55, 61 and 65. In all but one instance, the genetic algorithm was able to obtain the optimal solution. In fact, the instant in which the genetic algorithm failed to provide the optimal solution gave the same objective function value as was obtained in Rahman and Smith (1991) using the Teitz and Bart (1968) heuristic. In most cases, the computational burden for the 15 problems was less than 180 s per problem. When the population size is 100, the crossover rate is 0.85 and the mutation rate is 0.05, the genetic algorithm obtains the solution within 60 s when the generation is 200. For example, when $p = 12$ and the maximum distance is 61, the optimal solution is achieved at 150 generations. When $p = 15$ and the maximum distance is 65, the optimal solution is attained at 130 generations. The evaluation curve is shown in Figure 1. The remaining test results for the 30-community data set are shown in Table 1.

For the 150-cities data set, 16 problems were generated for which optimal solutions were available (Choi & Chaudhry, 1993). The number of facilities to be opened, p, was set to 20, 40, 60, 80 and 100, whereas the maximum distance parameter s was set to 200, 300 and 400 plus one additional problem instance where p was set at 20 and s was set at 250. Again, in all but one problem, the genetic algorithm was able to obtain the optimal solution. In almost 90% of the problem instances for the 150-cities data set, the genetic algorithm was able to obtain the optimal solution within 300 s. However, in a few instances, the computational burden was in excess of an hour when the population size was 500, the crossover rate was 0.85 and the mutation rate was 0.05. Table 2 shows the test results for the 150-cities data set.

5. Conclusion

In this paper we addressed the p-median problem with maximum distance constraint. Our computational experience, using two different data sets for which optimal solutions were available, showed that the genetic algorithm approach works well on the p-median problem with maximum distance constraint. This is contrary to Hosage and Goodchild (1986), who reported non-promising results for the unconstrained p-median problem, but supports the most recent findings of Bozkaya *et al.* (2000). However, additional computational experiments will be conducted after the genetic algorithm is further refined in order to enhance the computational burden.

References

Booker, L. B. (1987) Improving search in genetic algorithms, in *Genetic Algorithms and Simulated Annealing*, L. Davis (ed.), Los Altos, CA: Morgan Kauffmann, 61–73.

Table 2: *Results for the 150-community data set*

Number of facilities p	*Maximum distance s*	*Optimal solution*	*p facility set*
100	200	122094	{80 38 28 21 70 57 146 24 74 17 119 1 129 56 130 120 53 36 95 60 134 122 107 11 4 14 54 76 94 18 116 23 85 88 40 104 102 145 103 113 132 7 27 90 33 141 15 75 142 72 67 59 91 13 41 84 2 12 112 10 126 115 73 136 109 37 123 79 68 26 133 48 19 31 16 30 131 98 49 32 110 139 128 25 9 29 44 148 71 63 82 111 137 97 124 69 64 114 81 43}
	300	122094	{64 146 12 94 27 21 116 53 80 4 120 17 63 95 40 129 115 123 76 59 104 28 102 130 43 54 33 14 7 36 23 9 57 139 11 90 1 60 107 141 85 26 72 18 82 119 69 48 142 132 67 84 13 19 24 136 32 148 75 73 124 122 29 81 2 70 37 68 10 134 74 111 128 15 88 133 113 103 31 97 126 79 56 137 25 41 16 110 38 145 131 114 30 49 109 91 44 98 71 112}
	400	122094	{104 31 73 21 109 28 94 130 53 18 85 41 69 120 15 43 107 2 88 19 128 103 123 1 4 26 9 10 136 102 80 142 7 91 70 139 79 112 115 17 113 23 14 11 29 36 54 84 146 57 90 60 27 129 137 33 68 63 30 97 145 134 122 16 38 67 82 76 141 119 132 95 24 48 131 98 110 124 40 32 64 75 74 116 72 13 56 25 12 133 59 49 114 126 71 37 44 148 111 81}
80	200	355618	{79 81 123 36 102 1 9 130 60 119 21 4 53 67 57 94 104 132 14 13 11 27 54 2 76 122 95 49 23 120 116 129 73 43 107 124 28 90 32 59 115 142 24 88 51 70 145 113 15 141 48 17 91 18 109 74 64 75 25 126 72 85 146 128 137 139 103 33 136 80 84 10 56 82 29 7 3 63 111 31}
	300	355618	{53 1 102 113 54 130 4 36 120 146 73 82 24 15 72 141 21 119 85 116 32 11 7 80 33 18 76 142 57 137 9 104 60 81 122 90 107 128 94 2 14 27 95 132 29 88 123 79 43 74 13 59 23 145 103 136 31 63 49 3 67 124 139 115 51 91 126 70 84 10 28 109 64 111 48 56 129 17 25 75}
	400	355618	{57 116 9 4 32 29 90 60 102 76 36 53 72 1 120 54 27 130 129 107 142 73 80 141 33 146 59 79 124 119 24 95 49 17 82 88 63 13 122 2 28 137 145 81 91 25 31 48 84 56 21 11 15 85 115 103 128 94 104 14 109 7 18 64 43 70 123 139 23 132 3 75 74 111 51 136 126 113 67 10}
60	200	771652	{60 116 88 21 24 113 74 27 36 94 53 76 14 120 128 9 7 1 103 4 54 31 90 102 85 145 141 146 80 59 72 13 129 142 115 122 28 95 79 123 91 119 130 43 29 137 57 18 3 73 67 15 124 104 81 33 107 34 23 63}
	300	771652	{4 119 53 54 74 102 81 21 7 36 1 85 57 129 145 80 128 67 141 104 142 59 31 13 3 18 43 73 130 115 122 23 137 79 90 116 146 15 120 27 60 72 94 88 107 24 76 63 33 28 113 124 9 14 103 95 91 123 34 29}
	400	771652	{67 34 1 113 21 7 57 119 90 142 54 53 4 14 122 63 102 85 94 60 145 120 95 128 116 27 107 43 88 130 129 36 81 76 3 79 141 124 28 103 74 18 146 115 91 59 24 73 15 13 104 23 29 9 123 33 80 72 137 31}
40	200	1696328[a]	{135 60 57 53 1 36 4 103 119 130 14 21 88 54 15 33 81 42 113 76 116 19 0 31 74 90 95 49 63 67 145 102 72 122 120 29 79 91 137 142}
	300	1604999	{53 57 7 135 4 76 90 120 29 1 88 63 42 72 21 119 54 81 36 85 79 142 122 91 102 15 113 103 145 60 116 95 74 33 0 27 67 49 137 31}
	400	1604999	{21 91 53 4 103 54 102 33 72 90 0 119 29 15 113 120 95 27 7 36 85 88 57 1 49 63 60 42 76 122 116 137 31 74 79 81 142 67 135 145}
20	250	5244499	{42 90 135 53 63 120 33 4 23 31 54 111 29 56 81 21 97 88 72 138} (no feasible solution for distance $s = 200$)
	300	4318959	{90 63 81 1 53 42 119 31 97 88 135 21 54 29 56 4 72 103 111 24}
	400	4072860	{42 90 135 53 63 120 33 4 23 31 54 111 29 56 81 21 97 88 72 138}

[a]Optimal solution for this instance, 1,692,948.

Bozkaya, B., J. Zhang and E. Erkut (2000) An effective genetic algorithm for the *p*-median problem, Working Paper, Faculty of Business, University of Alberta.

Canós, M.J., C. Ivorra and V. Liern (1999) An exact algorithm for the fuzzy *p*-median problem, *European Journal of Operational Research*, **116** (1), 80–86.

Captivo, M.E. (1991) Fast primal and dual heuristic for the *p*-median location problem, *European Journal of Operational Research*, **52**, 65–74.

Chaudhry, S.S., I.-C. Choi and D.K. Smith (1995) Facility location with and without maximum distance constraints through the *p*-median problem, *International Journal of Operations and Production Management*, **15** (10), 75–81.

Chaudhry, S.S., M.W. Varano and L. Xu (2000) Systems research, genetic algorithms, and information systems, *Systems Research and Behavior Science*, **17**, 149–162.

Chiu, C. (2002) A case-based customer classification approach for direct marketing, *Expert Systems with Applications*, **22** (2), 163–168.

Choi, I.-C. and S.S. Chaudhry (1993) The *p*-median problem with maximum distance constraints: a direct approach, *Location Science*, **1** (3), 235–243.

Daskin, M. (1995) *Network and Discrete Location Analysis*, New York: Wiley.

Densham, P.J. and G. Rushton (1992) A more efficient heuristic for solving large *p*-median problems, *Papers in Regional Science*, **71**, 307–329.

Erkut, E., T. Myroon and K. Strangway (2000) TransAlta redesigns its service delivery network, *Interfaces*, **30** (2), 54–69.

Estivill-Castro, V. and R. Torres-Velazquez (1999) Hybrid genetic algorithm for solving the *p*-median problem, in *SEAL'98, LNCS 1585*, X. Yao *et al.* (eds), New York: Springer, 18–25.

Fitzsimmons, J.A. and A.L. Austin (1983) A warehouse location model helps Texas comptroller select out-of-state audit offices, *Interfaces*, **13** (5), 40–46.

Galvão, R.D. (1993) Use of Lagrangean relaxation in the solution of uncapacitated facility location problems, *Location Science*, **1** (1), 57–70.

GOLDBERG, D.E. (1989) *Genetic Algorithms in Search, Optimization, and Machine Learning*, Reading, MA: Addison-Wesley.

HAKIMI, S.L. (1965) Optimum distribution of switching centers in a communication network and some related graph theoretic problems, *Operations Research*, **13**, 462–475.

HAYES-ROTH, F. and N. JACOBSTEIN (1994) The state of knowledge-based systems, *Communications of the ACM*, **37** (3), 27–39.

HOLLAND, J. (1975) *Adaptation in Natural and Artificial Systems*, Ann Arbor, MI: University of Michigan Press.

HOSAGE, C.M. and M.F. GOODCHILD (1986) Discrete space location–allocation solutions from genetic algorithms, *Annals of Operations Research*, **6**, 35–46.

JARAMILLO, J.H., J. BHADURY and R. BATTA (2002) On the use of genetic algorithms to solve location problems, *Computers and Operations Research*, **29** (6), 761–779.

KHUMAWALA, B.M. (1972) Branch and bound algorithms for locating emergency service facilities, Working Paper 355, Krannert Graduate School of Industrial Administration, Purdue University, Lafayette, Indiana.

LI, F. and R.K. AGGARWAL (2000) Fast and accurate power dispatch using a relaxed genetic algorithm and a local gradient technique, *Expert Systems with Applications*, **19** (3), 159–165.

MANOLAS, D., G. EFTHIMEROS and D. TSAHALIS (2001) Development of an expert system shell based on genetic algorithms for the selection of the energy best available technologies and their optimal operating conditions for the process industry, *Expert Systems*, **18** (3), 124–130.

MICHALEWICZ, Z. (1994) *Genetic Algorithms + Data Structures = Evolution Programs*, 2nd edn, Berlin: Springer.

MIRCHANDANI, P.B. and R.L. FRANCIS (1990) *Discrete Location Theory*, New York: Wiley.

MOON, I.D. and S.S. CHAUDHRY (1984) An analysis of network location problems with distance constraints, *Management Science*, **30** (3), 290–307.

MURRAY, A.T. and R.L. CHURCH (1996) Applying simulated annealing to location-planning models, *Journal of Heuristics*, **2**, 31–53.

NARULA, S.C., U.I. OGBU and H.M. SAMUELSSON (1977) An algorithm for the *p*-median problem, *Operations Research*, **25**, 709–712.

PEREZ, J.A.M., J.L.R. GARCIA and J.M. MORENO-VEGA (1994) A parallel genetic algorithm for the discrete *p*-median problem, *Studies in Locational Analysis*, **7**, 131–141.

RAHMAN, S. and D.K. SMITH (1991) A comparison of two heuristic methods for the *p*-median problem with and without maximum distance constraints, *International Journal of Operations and Production Management*, **11** (6), 76–84.

ROLLAND, E., D.A. SCHILLING and J.R. CURRENT (1996) An efficient tabu search procedure for the *p*-median problem, *European Journal of Operational Research*, **96**, 329–342.

SHIN, K.-S. and I. HAN (1999) Case-based reasoning supported by genetic algorithms for corporate bond rating, *Expert Systems with Applications*, **16** (2), 85–95.

SHIN, K.-S. and Y.-J. LEE (2002) A genetic algorithm application in bankruptcy prediction modeling, *Expert Systems with Applications*, **23** (3), 321–328.

TEITZ, M.B. and P. BART (1968) Heuristic methods for estimating the vertex median of a weighted graph, *Operations Research*, **12** (3), 955–961.

TOREGAS, C., R. SWAIN, C. REVELLE and L. BERGMAN (1971) The location of emergency service facilities, *Operations Research*, **19**, 1363–1373.

The authors

Sohail S. Chaudhry

Sohail S. Chaudhry received his PhD in industrial engineering and operations research from Columbia University. His research interests include the areas of network facility location, management and control of quality, and vendor selection. He has published numerous papers in journals such as *Decision Sciences*, *European Journal of Operational Research*, *Journal of the Operational Society*, *Journal of Quality and Reliability Management*, *Management Science*, *Omega* and *Systems Research and Behavioral Science*.

Shiwei He

Shiwei He received a PhD in transportation management engineering and operations research from Southwest Jiaotong University, Chengdu, China. His research interests include the areas of transportation planning and management, operations research and the application of management science techniques to transportation problems. He has published many papers in journals such as *European Journal of Operational Research*, *Computers and Operations Research* and *Journal of Systems Science and Systems Engineering* as well as in a variety of national and international conference proceedings.

Peggy E. Chaudhry

Peggy E. Chaudhry received her PhD in international business from the University of Wisconsin at Madison. She has taught in the EMBA Program of Temple University in Tokyo, Japan, and the European Business School in Oestrich-Winkel, Germany. Her current research interests include protection of intellectual property rights and gray markets in the global environment. She has published several papers in the *Columbia Journal of World Business*, *European Management Journal*, *Journal of Health Care Marketing*, *Journal of International Marketing* and *Production and Inventory Management Journal*.