Gender-Specific Genetic Algorithms*

Tammy Drezner and Zvi Drezner
College of Business and Economics
California State University-Fullerton
Fullerton, CA 92834.

Abstract

In this paper we propose to incorporate gender in genetic algorithms. Existing genetic algorithms are all gender-neutral. Every genetic or hybrid genetic algorithm which is gender-neutral can be easily constructed as a gender-specific genetic algorithm. We compared the performance of the gender-neutral and its gender-specific counterpart on four optimization problems and the gender-specific algorithm exhibited superior performance. A statistical analysis allows this conclusion to be stated with 99.98% confidence.

1 Introduction

Genetic algorithms, first suggested by Holland (1975), have recently become a popular metaheuristic method in recent years for solving optimization problems. Borrowed from the natural sciences and Darwin's law of natural selection and survival of the fittest, genetic algorithms are based on the premise that, like in nature, successful matching of parents may produce better, improved offsprings. For a review see Goldberg (1989), Salhi (1998).

Unlike natural processes though, the basic tenet of genetic algorithms is that the "pool" of candidates for matching is gender-neutral. In nature, the pool of candidates is gender-specific. Matching of two males or two females will not produce an offspring. The "pool" of possible mates is therefore reduced by one half. In this paper we introduce a more realistic version of genetic algorithms, one that distinguishes between the sexes. A match can only take place between a male and a female.

*The idea of incorporating gender in genetic algorithms was proposed by the first author.

The gender principle can be used to convert any gender-neutral genetic algorithm to a gender-specific one. We tested the gender principle on gender-neutral hybrid genetic algorithms designed for the solution of four different optimization problems. The gender-specific counterpart algorithms performed extremely well.

2 Genetic Algorithms

The general framework of a genetic algorithm (whether gender-neutral or gender-specific) is:

1. An initial population of solutions is randomly generated.
2. Each generation, pairs of population members are selected and produce an offspring.
3. A new population for the next generation is formed by replacing some existing population members by some of the newly generated offsprings.
4. The procedure is stopped according to some rule.
5. The best solution found throughout the process is the result of the algorithm.

Mutations of population members or offsprings are also considered. A mutation of a solution may be either one perturbation of the solution or applying a heuristic algorithm on the solution.

3 The Gender Principle

A gender-specific algorithm is identical to a gender-neutral genetic algorithm with one exception, namely, the solutions (population members) are assigned a gender. Each population member is arbitrarily classified as either male or female and mating is allowed only between opposite sexes. The following three modifications are applied to a gender-neutral algorithm in order to convert it to a gender-specific one:

1. When generating the starting population, each population member is assigned a gender. Half of the starting population members are arbitrarily classified as males and the other half as females.

2. When selecting a pair for mating it is ensured that the mating is between a male and a female.

3. The gender of the offspring is randomly determined with equal probability for each gender.

The gender principle is simple to implement in conjunction with any gender-neutral genetic algorithm. It merely requires a vector of genders with cardinality of the population size, a simple modification in the selection process, and an assignment of a gender to the offspring. Every gender-neutral genetic algorithm has its gender-specific genetic algorithm counterpart.

3.1 On the Probability of Extinction

When the gender of an offspring is randomly generated with a probability of 50% for each gender, there exists a probability that at some point all population members will have the same gender and no offsprings can be generated. The procedure must be stopped prematurely and the best population member selected as the solution. In all our experiments we never encountered such a situation.

We calculated the probability of extinction following 100, 500, 1000, 2000, 5000 *changes in population*, not generations. We considered populations of 20, 50, and 100 members and assume that at the beginning we have equal number of males and females.

The calculation is done using Markov chains. For a population of P members there are $P+1$ states corresponding to the possible number of males in the population. States 0 and P are absorbing states, and the transition probabilities from state K, $1 \leq K \leq P-1$ are 0.5 to remain at that state, $\frac{K}{2P}$ to move to state $K-1$, and $\frac{P-K}{2P}$ to move to state $K+1$. To find the probability

Table 1: The Probability of Extinction

Number of Changes	Population Size		
	20	50	100
100	3.07×10^{-4}	1.41×10^{-13}	3.70×10^{-28}
500	2.08×10^{-3}	4.32×10^{-12}	2.95×10^{-26}
1000	4.30×10^{-3}	9.75×10^{-12}	6.81×10^{-26}
2000	8.72×10^{-3}	2.06×10^{-11}	1.45×10^{-25}
5000	2.19×10^{-2}	5.32×10^{-11}	3.77×10^{-25}

of extinction before a certain number of population changes, the transition matrix is raised to that power, and the sum of the probabilities from $K = \frac{P}{2}$ to 0 and P, is the probability of extinction. The results of these calculations are given in Table 1.

It is clear from Table 1 that for populations of size 50 or more, the possibility of extinction can be ignored. We used a population of 20 in one problem, and the number of population changes was only about 100. We never encountered even one case of extinction in all our experiments.

4 Computational Experiments

We selected four problems for our experiments.

The Golf Scramble Problem (Golf) (Dear and Drezner, 2000). There are 144 players, each with a given handicap. The players are stratified into four flights. Flight A includes 36 players with the lowest handicap, Flight B the next 36 players and so on. The objective is to construct 36 comparable teams of four members each. Each team has exactly one player from each flight. The goal is to minimize the difference between the team with the lowest total handicap, and the one with the highest.

In the hybrid-genetic algorithm for the golf problem (Dear and Drezner, 2000), each generation all pairs of parents are matched to produce offsprings. The best offspring is selected for possible inclusion in the population. By implementing the gender-specific algorithm, the number of possible offsprings is reduced by about one half. Run times are, therefore, much faster. To have a fair comparison between the gender-neutral and its gender-specific counterpart, we used a population of 20 for the original gender-neutral algorithm, and a population of 24 for the gender-specific one proposed in this paper. Run times for both algorithms are now about the same. Seven randomly generated problems given in Dear and Drezner (2000) were tested.

The Distance Dependent Unreliable Multifacility Location Problem (DDUMLP) (Berman et al., 2000). A network of n nodes is given with given demand generated at each node. m facilities are to be located on the network. The probability that a facility can provide service to a node is "1" if the distance is zero, and is decreasing exponentially with the distance. For each node, the probability that no service can be provided is the product of all probabilities that facilities are not able to provide service. The objective is to find m locations for the facilities such that the total expected unsatisfied demand is minimized. It is shown that the best locations for the facilities must be on nodes.

We compared the gender-neutral algorithm proposed in Berman et al. (2000) to its gender-specific counterpart on the 17 problems tested in Berman et al. (2000). A population of 50 and 500 generations were used.

The Network Design Problem (NDP). The problem was first suggested by Drezner and Wesolowsky (1997). A network with n nodes and m links is given. The traffic from every node to every other node and the lengths of the links are also given. The objective is to design a network where some links are converted to one-way links. Travel on a one-way link is faster. The length of a one-way link in the direction of the traffic flow is multiplied by a given constant

$0 \leq \alpha \leq 1$ (which represents the inverse of the speed increase on a one-way link). The objective is to minimize the total travel time of all users. A hybrid genetic algorithm for the solution of this problem is proposed in Drezner and Salhi (2000).

We tested the gender-specific algorithm on 12 problems which were tested in Drezner and Salhi (2000). Four of the problems are based on sparse networks, four on medium sparsity networks, and four on dense networks. A population of 50 and 500 generations were used.

The Quadratic Assignment Problem (QAP). The quadratic assignment problem is considered one of the most difficult combinatorial problems and is very well researched. For a review of quadratic assignment problem algorithms see Burkard (1990), Taillard (1995), Cela (1998). Genetic, or hybrid genetic, algorithms for the solution of the quadratic assignment problem are described in Ahuja et al. (2000), Drezner (2000), Fleurent and Ferland (1994), Tate and Smith (1995).

We created a gender-specific counterpart to the gender-neutral genetic algorithm proposed in Drezner (2000). Ten large QAP problems with $72 \leq n \leq 100$ were used for comparison. All problems are given in the data base on http://www.mim.du.dk/~sk/qaplib. A population of 100 and 1500 generations were used. Run times for the $n = 100$ problems were about 30 minutes on a 600MHz PC using Microsoft FORTRAN PowerStation 4.0.

The Golf problems were solved 100 times each whereas the other problems were solved 10 times each. The results are summarized in Tables 2-5. In Tables 3-5 the results are expressed as percentages over the best known solution. All the results are very good. The highest average over the best known solution is 0.387% for DDUMLP, 0.184% for NDP, and 0.047% for QAP.

Table 2: Comparison for Golf

Problem	Gender-Neutral			Gender-Specific		
	Min	Times†	Aver.	Min	Times†	Aver.
8000	19	1	29.56	17	1	28.08
4000	8	1	14.90	8	1	14.83
2000	5	4	7.46	5	6	7.43
1000	3	27	3.95	3	25	3.94
500	1	1	2.10	1	2	2.08
250	1	78	1.22	1	78	1.22
125	1	100	1.00	1	100	1.00

†Number of times out of 100 that the minimum was obtained

5 Discussion of Results

We solved a total of 46 different problems. We first compare the algorithms' performance in regards to finding the "best known solution". In two cases (one for Golf and one for DDUMLP) the gender-specific algorithm found a new "best known" solution. There were 7 cases (3 for DDUMLP, 3 for NDP, and one for the QAP) where both algorithms failed to find the best known solution. The gender-neutral algorithm failed to find the best known solution in 3 additional cases (one for DDUMLP 2 for QAP). In no case did the gender-neutral algorithm find the best known solution while the gender-specific did not.

For subsequent comparisons, two measures were used to compare the two algorithms:

1. The minimal value of the objective function obtained in all runs. If there is a tie in the minimal value, the tie is broken by the number of times this minimum is obtained.

2. The percentage of the average value of the objective function over the best known value of the objective function.

The results of Tables 2-5 are summarized in Table 6.

The comparison between the two algorithms clearly shows that the gender-specific algorithm is

Table 3: Comparison for DDUMLP

n	m	Best Known	Gender-Neutral			Gender-Specific		
			Min	Times†	Aver.	Min	Times†	Aver.
50	5	31.628	0	10	0	0	10	0
50	10	20.005	0	10	0	0	10	0
50	20	8.0948	0	10	0	0	10	0
100	5	70.205	0	10	0	0	10	0
100	10	48.809	0	10	0	0	10	0
100	20	24.654	0	10	0	0	10	0
100	50	2.5471	0	2	0.220	0	3	0.185
200	5	149.20	0	10	0	0	10	0
200	10	110.26	0	10	0	0	10	0
200	20	63.304	0	9	0.003	0	10	0
200	50	11.101	0.018	3	0.036	0.009	2	0.027
200	100	.51862	0.002	4	0.129	0	1	0.179
500	5	365.84	0	10	0	0	10	0
500	10	275.83	0	10	0	-0.004	4	-0.004
500	20	161.18	0	9	0.006	0	7	0.006
500	50	33.847	0.003	3	0.050	0.009	1	0.047
500	100	2.5349	0.276	1	0.387	0.110	1	0.335

†Number of times out of 10 that the minimum was obtained

Table 4: Comparison for NDP

α	Best Known	Gender-Neutral			Gender-Specific		
		Min	Times†	Aver.	Min	Times†	Aver.
Sparse Problems							
0.8	188621	0	10	0	0	10	0
0.7	131356	0	10	0	0	10	0
0.6	165481	0	8	0.038	0	8	0.008
0.5	143259	0	9	0.022	0	8	0.012
Medium Sparsity Problems							
0.8	160192	0	4	0.011	0	4	0.010
0.7	146431	0	7	0.036	0	8	0.041
0.6	127950	0	1	0.054	0	4	0.053
0.5	108046	0	2	0.071	0	3	0.047
Dense Problems							
0.8	144676	0	2	0.023	0	3	0.031
0.7	128935	0.038	1	0.184	0.060	1	0.155
0.6	110525	0.029	1	0.145	0.040	1	0.159
0.5	92104	0.075	1	0.181	0.110	1	0.181

†Number of times out of 10 that the minimum was obtained

Table 5: Comparison for QAP

Name	Best Known	Gender-Neutral			Gender-Specific		
		Min	Times†	Aver.	Min	Times†	Aver.
Sko72	66256	0	2	0.019	0	5	0.011
Sko81	90998	0	1	0.021	0	3	0.019
Sko90	115534	0.007	4	0.023	0.007	5	0.037
Sko100a	152002	0.016	1	0.039	0	3	0.029
Sko100b	153890	0	3	0.023	0	3	0.014
Sko100c	147862	0	1	0.013	0	1	0.005
Sko100d	149576	0	1	0.030	0	2	0.031
Sko100e	149150	0	7	0.008	0	4	0.009
Sko100f	149036	0	2	0.047	0	1	0.043
Wil100	273044	0.002	4	0.022	0	2	0.019

†Number of times out of 10 that the minimum was obtained

Table 6: Summary of Results

Problem Name	Gender-Neutral is Better	Gender-Specific is Better
Golf	1	8
DDUMLP	3	12
NDP	7	10
QAP	5	13

superior to its gender-neutral counterpart. This is proven statistically. A difference in performance between the two algorithms was observed for 59 measures. In 43 out of 59 (72.9%) cases the gender-specific algorithm performed better. The null hypothesis is that the two algorithms perform equally well, and the alternative hypothesis is that the gender-specific algorithm performs better than its gender-neutral counterpart. This is stated as:

$$\begin{aligned} H_0 &: \quad p = 0.5 \\ H_1 &: \quad p > 0.5 \end{aligned}$$

where p is the proportion of cases in which the gender-specific algorithm performed better than its gender-neutral counterpart. This test leads to $z = 3.52$ which gives a p-value of 0.0002. In conclusion, we claim that the gender-specific algorithm is superior to its gender-neutral counterpart with 99.98% confidence.

5.1 Why Does the Gender-Specific Algorithm Work?

The gender-specific algorithm simulates natural evolutionary processes more closely and more accurately. Having two genders in nature has been proven to be advantageous. It promotes diversity and reduces in-breeding. We believe that our gender principle has similar effects. We observed that in gender-specific algorithms the homogenizing process of the population is slower and populations remain more diverse for a longer time. Diverse populations enhance the chance for obtaining better offsprings.

In order to test this assertion we calculated the standard deviation of the values of the objective function of all population members and plotted it as a function of the generation. The standard deviation is a measure of the diversity of the population. In Figure 1 we depict the standard deviation for the QAP first Sko72 run for both the gender-neutral and gender-specific algorithms. The standard deviation is expressed as a percentage of the best known solution. Both algorithms start with the same population of 100 solutions (thus equal standard deviations) and were run for 1500 generations. We note that these graphs are representative of many other graphs for other problems.

In Figure 2 we depict the *difference* between the standard deviations of the gender-specific algorithm and the gender-neutral one. Figure 2 provides a clearer picture of the different behavior of the two algorithms. The same behavior is observed in almost all graphs that we constructed for other problems as well. The difference between the two standard deviations is erratic for the first 100 or so generations. Then a clear difference is observed up to 500-600 generations with a higher gender-specific standard deviation. After 600 generations the difference is rather small and may go either way. In many cases, like for the Sko72 problem, there is a range at which the standard deviation of the gender-neutral algorithm is higher. After about 1000 iterations the gender-specific standard deviation is consistently higher but it is quite small for both algorithms. This happens at the final stage of the genetic algorithms when the population is rather homogeneous and stagnant. A feature common to all graphs is that at the final generations the standard deviation of the gender-specific algorithm is higher than that of the gender-neutral one, showing a higher degree of diversity.

Analyzing Figure 2 we conclude that the population is more diverse in the gender-specific algorithm at the crucial phase of the algorithm (up to about 500 iterations). Beyond this point, any improvement in the population is marginal for both algorithms.

6 Conclusions

Genetic algorithms attempt to solve optimization problems by simulating natural selection and survival of the fittest processes. Most known species are gender-specific. It is plausible that gender

Figure 1: The Standard Deviation Throughout the Generations as Percentage of the Best Known Value of the Objective Function

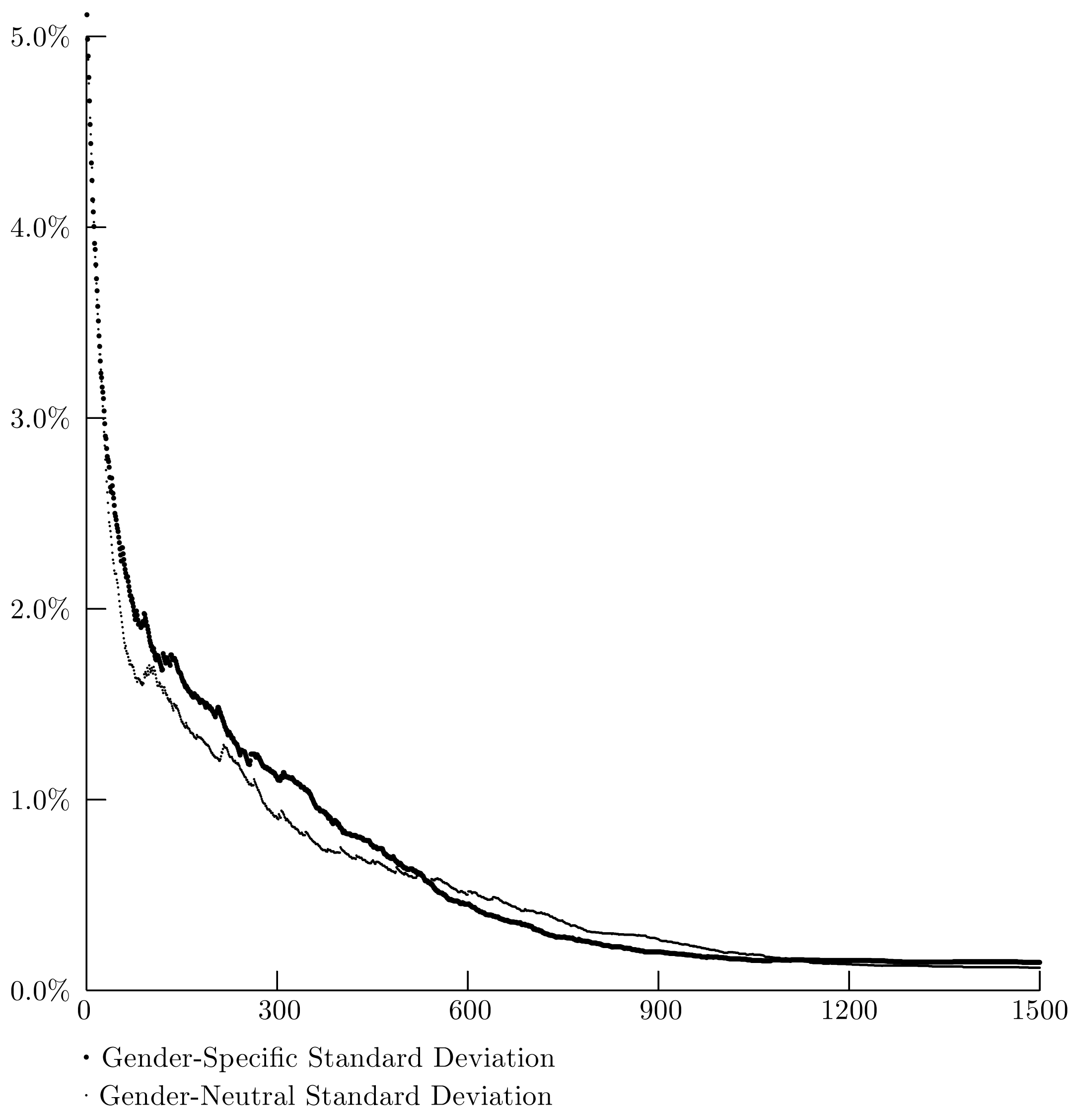
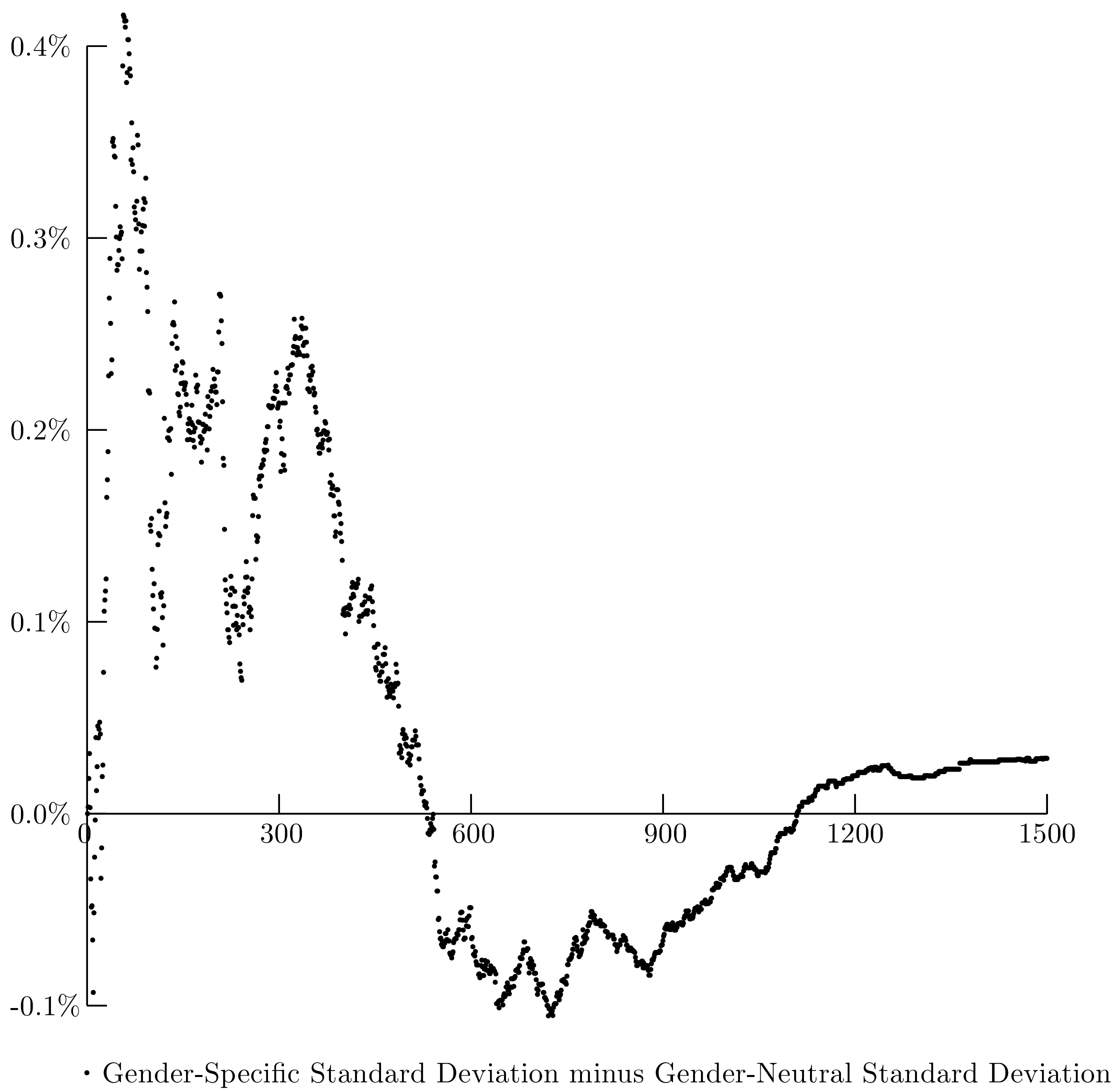


Figure 2: The Difference Between the Standard Deviations between the Gender-Specific and Gender-Neutral Algorithms



is conducive to the evolution of species. However, gender is not currently incorporated in existing genetic algorithms, hence they are gender-neutral. We propose to assign a gender to population members making them gender-specific in order to more closely emulate nature.

The proposed gender principle is simple and can be easily incorporated into any existing gender-neutral genetic algorithm yielding a gender-specific counterpart. A comparison of the gender-specific algorithm to its gender-neutral counterpart for the solution of four optimization problems yielded a clear advantage to the gender-specific algorithm. This is proven statistically with a 99.98% confidence. Run times are comparable for both algorithms.

References

[1] Ahuja R.K., J.B. Orlin, and A. Tiwari (2000) "A Descent Genetic Algorithm for the Quadratic Assignment Problem," *Computers and Operations Research*, to appear.

[2] Berman O., Z. Drezner and G.O. Wesolowsky (2000) "Locating Service Facilities Whose Reliability is Distance Dependent," submitted for publication.

[3] Burkard R.E. (1990) "Locations with Spatial Interactions: The Quadratic Assignment Problem," In P.B. Mirchandani and R.L. Francis, editors, *Discrete Location Theory*, Wiley, Berlin.

[4] Cela E. (1998) *The Quadratic Assignment Problem: Theory and Algorithms*, Kluwer Academic Publishers, Dordrecht.

[5] Dear R. and Z. Drezner (2000) "Applying Combinatorial Optimization Metaheuristics to the Golf Scramble Problem," *International Transactions of Operations Research*, 331-347

[6] Drezner Z. (2000) "A Hybrid Genetic Algorithm for the Quadratic Assignment Problem," Working paper, College of Business Administration, California State University-Fullerton.

[7] Drezner Z. and S. Salhi (2000) "Using Metaheuristics for the One-Way and Two-Way Network Design Problem," *Naval Research Logistics*, forthcoming.

[8] Fleurent C., and J.A. Ferland (1994) "Genetic Hybrids for the Quadratic Assignment Problem," in P. Pardalos and H. Wolkowicz, editors, *Quadratic Assignment and Related Problems*, volume 16, pages 173-187. DIMACS Series in Discrete Mathematics and Theoretical Computer Science.

[9] Goldberg D.E. (1989) *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, Wokingham, England.

[10] Holland J.H. (1975) *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor.

[11] Salhi S. (1998) "Heuristic Search Methods," in G. Marcoulides (Ed.) *Modern Methods for Business Research*, Lawrence Erlbaum Associates, Mahwah, NJ.

[12] Taillard E.D. (1995) "Comparison of Iterative Searches for the Quadratic Assignment Problem," *Location Science*, 3, 87-105.

[13] Tate D.M. and A.E. Smith (1995) "A Genetic Approach to the Quadratic Assignment Problem," *Computers and Operations Research*, 22, 73-83.