EMOCA: An Evolutionary Multi-Objective Crowding Algorithm

Ramesh Rajagopalan, Chilukuri K. Mohan, Kishan G. Mehrotra, and Pramod K. Varshney

Department of Electrical Engineering and Computer Science, 2-177, Center for Science and Technology, Syracuse University, Syracuse NY, 13244-4100, U.S.A

ABSTRACT

A new evolutionary algorithm is proposed for solving multi-objective optimization problems, focusing on the issue of developing a diverse population of non-dominated solutions. The key new approach in this algorithm is to use a diversity-emphasizing probabilistic approach in determining whether an off-spring individual is considered in the replacement selection phase, along with the use of a non-domination ranking scheme. This *evolutionary multi-objective crowding algorithm (EMOCA)* is evaluated using nine benchmark multi-objective optimization problems and shown to produce non-dominated solutions with significant diversity, outperforming three state-of-the-art multi-objective evolutionary algorithms on most ofthe test problems.

KEYWORDS

multi-objective optimization, evolutionary algorithms, benchmark test functions, Pareto-optimization, non-domination, diversity

---

Correspondence to: Professor Chilukuri K. Mohan, Dept. of Electrical Engineering and Computer Science, 2-177, Center for Science and Technology, Syracuse University, Syracuse, NY 13244-4100, USA; email: {mohan@ecs.syr.edu, kishan@ecs.syr.edu, varshney@ecs.syr.edu, rarajago@syr.edu}

1. INTRODUCTION

Considerable research efforts have recently been devoted to developing efficient algorithms for solving *Multi-Objective Optimization* (*MOO*) problems, for which multiple objectives must be simultaneously considered. Classical optimization methods suggest converting a *MOO* problem into a single objective optimization problem, e.g., by attempting to minimize a weighted sum of the various objective functions to yield a single solution. A vector ***a*** is non-dominated or Pareto-optimal if it is not dominated[1] by any other solution vector. The set of all non-dominated solutions is called the Pareto-optimal front. It is preferable to obtain a collection of *non-dominated* or *Pareto-optimal* solutions corresponding to different tradeoff points, but these cannot be obtained even by iterating the weighted approach with different weight values if the Pareto-optimal front is non-convex (Deb 2001).

This caveat is addressed by *Multi-Objective Evolutionary Algorithms (MOEAs*), which work with a population of candidate solutions, exploring multiple non-dominated solutions in parallel, whereas traditional approaches are crippled by focusing on one solution at a time. Dominance-based selection is used in *MOEA*s, along with density estimation methods, such as crowding distance and squeeze factor, which are used to preserve population diversity (Coello & Lamont 2004). Several *MOEA*s have been proposed in the literature, with early work in (Schaffer 1985, Hajela & Lin 1994, Fonseca & Fleming 1993, Horn et al. 1994, Veldhuizen 1999), and with the emergence of three state-of-the-art algorithms in recent years, with which our results are compared:

1. PAES (Knowles and Corne 1999) uses a (1+1)-ES, which is similar to a random mutation hill climbing strategy and performs well in problems having search space with non-uniformly dense solutions.
2. SPEA-II (Zitzler et al. 2001) is a generational algorithm with an elitist strategy which maintains an archive of non-dominated individuals at each generation.
3. NSGA –II (Deb et al. 2000) assigns a Pareto rank to each individual based

[1] If each objective function f_j is to be maximized, then a solution vector ***a*** *dominates* ***b***, written $\boldsymbol{a} \gg \boldsymbol{b}$, if and only if $\forall i \in \{1,\ldots,M\} : f_i(\boldsymbol{a}) \geq f_i(\boldsymbol{b})$, and $\exists j \in \{1,\ldots,M\} : f_j(\boldsymbol{a}) > f_j(\boldsymbol{b})$.

on a non-dominated sorting approach, and combines such attractive features as elitism, fast non-dominated sorting, and parameterless fitness sharing.

We propose a new evolutionary multi-objective crowding algorithm (*EMOCA*), distinguished from other *MOEA*s in the following features:

1. *Probabilistic replacement selection*: EMOCA allows some poor quality offspring to survive in the population if they improve population diversity.
2. *Mating selection*: Both non-domination and diversity are considered, so that EMOCA's mating pool consists of better as well as substantially different individuals, whereas most other MOEAs use only non-domination as the primary criterion to select the mating pool.

On several benchmark functions suggested by (Deb 2001) and (Zitzler et al. 2000), we compared the performance of EMOCA with NSGA-II, SPEA-II and PAES using metrics that evaluate convergence and diversity of solutions. Simulation results show that EMOCA outperforms the other algorithms in most of the test problems.

The new algorithm (EMOCA) is described in Section 2. The benchmark test problems used in our simulations are given in Section 3. Simulation results and conclusions are presented in Sections 4 and 5, respectively.

2. EVOLUTIONARY MULTI-OBJECTIVE CROWDING ALGORITHM (EMOCA)

The *MOEA*s have two main goals, viz., finding non-dominated solutions, and obtaining a uniformly spaced diverse set of solutions. Most existing *MOEA*s emphasize the first goal at the expense of the second, yielding populations of poor diversity in which a few regions of the Paretofront might be crowded whereas other regions are relatively unexplored. EMOCA addresses this issue by emphasizing diversity in multiple stages of the algorithm.

1. Initialize;
2. For the number of iterations determined by computational bounds, do:
    2.1. Generate Mating Population;
    2.2. Generate offspring by crossover followed by mutation;
    2.3. Create a new pool consisting of parents and some offspring;
    2.4. Trim new pool to generate the population for the next iteration;
    2.5. Update archive to contain all non-dominated solutions

Fig. 1: EMOCA algorithm overview

2.1 Description of EMOCA

Figure 1 shows the general overview of EMOCA. The individual steps of EMOCA are then described below.

2.1.1 *Mating population generation.* EMOCA employs binary tournament selection to fill the mating population. Each solution is assigned a fitness value equal to the sum of its *non-domination rank* and *diversity rank*, defined below.

Non-domination rank: As in (Deb et al. 2000), rank(x)=j in a population X *iff* $x \in F_j$, where fronts $F_1, F_2 \ldots F_{i+1}$ are inductively defined as follows:

$$F1 = \{x \in X | \forall y \in X : \neg(y >> x)\};$$
$$Fi+1 = \{x \in X | \forall y \in X - (F_1 \cup F_2 \cup \ldots \cup F_i) : \neg(y >> x)\}; \qquad (1)$$

Diversity rank: The crowding distance of a solution x_i of front F (Deb et al. 2000), is defined as

$$\psi(x_i) = \sum_{m=1}^{M} \Psi_m(x_i) \qquad (2)$$

where

$$\psi(x_i) = \begin{cases} \infty & if \quad \forall x_j, f_m(x_i) < f_m(x_j) \quad or \quad \forall x_k, f_m(x_k) < f_m(x_i) \\ \frac{g}{h} & otherwise \end{cases} \qquad (3)$$

$$g = \min_{j,k} \left(f_m(x_j) - f_m(x_k)\right) \text{ where } x_i \neq x_j \neq x_k \in F \text{ and } f_m(x_j) > f_m(x_i) > f_m(x_k)$$

and

$$h = \max_{j,k} \left(f_m(x_j) - f_m(x_k)\right) \text{ where } x_j \neq x_k \in X \text{ and } f_m(x_j) > f_m(x_k)$$

The solutions are sorted and ranked based on the crowding distance. The solution with the highest crowding distance is assigned the best (lowest) diversity rank.

2.2.2 *New pool generation.* A "New Pool" is generated consisting of all the parents and some of the offspring, following a comparison of each offspring O with a randomly chosen parent P. The result of the comparison governs the probability with which O is inserted into the new pool as summarized in Table 1.

TABLE 1

Acceptance probabilities of offspring for different scenarios. $P >> O$ indicates P dominates O. $P \approx O$ indicates that P and O are mutually non-dominating

Non-domination comparison	Crowding distance comparison	Prob(O is accepted)
$P>>O$	$\psi(O) > \psi(P)$	$1 - e^{\psi(P)-\psi(O)}$
$P>>O$	$\psi(P) \geq \psi(O)$	0
$O>>P$		1
$P \approx O$	$\psi(O) \geq \psi(P)$	1
$P \approx O$	$\psi(P) > \psi(O)$	0

2.2.3 *Trimming new pool.* The new pool is sorted based on the primary criterion of non-domination rank and the secondary criterion of diversity rank. In other words, solutions with the same non-domination rank are compared based on diversity rank. The new population will consist of the first μ elements of the sorted list containing solutions grouped into different fronts: F1, F2,..., Fn as defined in (1).

2.2.4 *Archiving strategy.* EMOCA maintains an archive (with fixed size) of non-dominated solutions at every generation. When the archive is full, a a new non-dominated solution can replace an existing archive element with lower crowding distance.

TABLE 2

Summary of test problems and the associated Pareto optimal fronts: 1-convex, $\bar{1}$ - non-convex, 2- connected, $\bar{2}$-disconnected, 3-numerous Pareto-optimal solution

proble m	n, range	Objective functions and their parameters	PO Front
KUR	3, [-5,5]	$f_1(x) = \sum_{i=1}^{n-1} -10\,exp(-0.2\sqrt{x_i^2 + x_{i+1}^2}\,)$, $f_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert^8 + 5\,sin\,x_i^3\,)$	$\bar{1},\bar{2}$
FON	3, [-4,4]	$f_i(x) = 1 - exp(-\sum_{j=1}^{3}(x_j + (-1)^i \times 1/\sqrt{3})^2\,)$, i=1,2	$\bar{1}$,2
SCH	$1,[10^3,10^3]$	$f_1(x) = x^2,\; f_2(x) = (x-2)^2$	1,2
POL	$2,[-\pi,\pi]$	$f_1(x) = [1 + (g_1 - h_1)^2 + (g_2 - h_2)^2\,]$ $f_2(x) = [(x_1+3)^2 + (x_2+1)^2\,]$ $h_1 = 0.5\,sin\,x_1 - 2\,cos\,x_1 + sin\,x_2 - 1.5\,cos\,x_2, g$ $h_2 = 1.5\,sin\,x_1 - cos\,x_1 + 2\,sin\,x_2 - 0.5\,cos\,x_2,$	$\bar{1},\bar{2}$
ZDT1 ZDT2 ZDT3	30, [0,1] 30, [0,1] 30, [0,1]	$f_1(x) = x_1, f_2(x) = g(x)[1 - \sqrt{x_1/g(x)}\,]$ $f_1(x) = x_1, f_2(x) = g(x)[1 - (x_1/g(x))^2\,]$ $f_1(x) = x_1$, $f_2(x) = g(x)[1 - \sqrt{x\;\;/g(x)} - (x_1/g(x)) \times si$ $g(x) = 1 + 9(\sum_{i=2}^{n} x_i)/(n-1)$ for ZDT1, ZDT2 and ZDT3.	1,2 $\bar{1}$,2 1,$\bar{2}$
ZDT4	$x_1 \in [0,1]$ $x_i \in [-5,5$	$f_1(x) = x_1, f_2(x) = g(x)[1 - \sqrt{x_1/g(x)}\,]$ $g(x) = 1 + 10(n-1) + \sum_{i=2}^{n}[x_i^2\;\; -10\,cos(4\pi x_i$	$\bar{1}$,2,3
ZDT6	10,[0,1]	$f_1(x) = 1 - exp(-4x_1)\,sin^6(4\pi x_1)$ $f_2(x) = g(x)[1 - (f_1(x)/g(x))^2\,]$, $g(x) = 1 + 9[(\sum_{i=2}^{n} x_i)/(n-1)]^{0.25}$	$\bar{1}$,2

3. TEST PROBLEMS

We evaluated the performance of EMOCA on four widely used benchmark multi-objective problems (FON, POL, KUR and SCH) from (Veldhuizen 1999), and six test problems (ZDT1, ZDT2, ZDT3, ZDT4, ZDT5 and ZDT6) from (Zitzler et al. 2000). These bi-objective problems are summarized in Table 2.

We have used a real coded representation for individuals. In the implementation of NSGA-II (Deb et al. 2000), simulated binary crossover (*SBX*) and parameter based mutation have been used. The studies in (Deb & Agarwal 1995) showed that *SBX* outperforms binary coded genetic algorithms for continuous search space problems. Since the test problems in this study have continuous search space, we have employed *SBX* and parameter based mutation for EMOCA, NSGA-II, SPEA-II and PAES.

4. SIMULATION RESULTS

In this section, we present the simulation results of EMOCA, NSGA-II, SPEA-II, and PAES for various test problems[2]. For fairness of comparison with the work in (Deb et al. 2000), we have used a population size of 100. The algorithms were simulated for 250 generations over 30 trials. No further improvements in the performance of the algorithms were observed after 250 generations. The crossover probability was 0.9 and mutation probability was chosen to be $1/n$ where n is the number of decision variables. The algorithms exhibited similar performances for small variations in population size, cross-over, and mutation probabilities. We have used distribution indices of 20 for the simulated binary crossover and mutation (Deb et al. 2000). For PAES, we have used a depth value of 4. We have used an archive size of 100 for all approaches. We performed experiments on an Intel Pentium 4 processor (3.2 GHz, 2 GB RAM). All approaches required an average computational time of 1 second per trial. Similar to NSGA-II, the computational complexity of

[2] For our simulations, we used the code for PAES, SPEA-II and NSGA-II from the following websites respectively: http://www.rdg.ac.uk/~ssr97jdk/multi/PAES.html, http://www.tik.ee.ethz.ch/pisa/, http://www.iitk.ac.in/kangal/soft.htm.

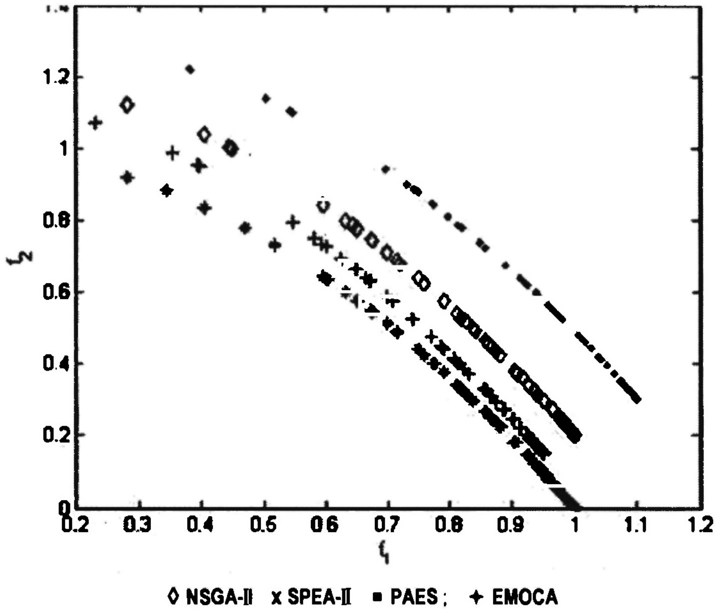

Fig. 2: Performance comparison of EMOCA, NSGE-II, AND PAES ON ZDTB

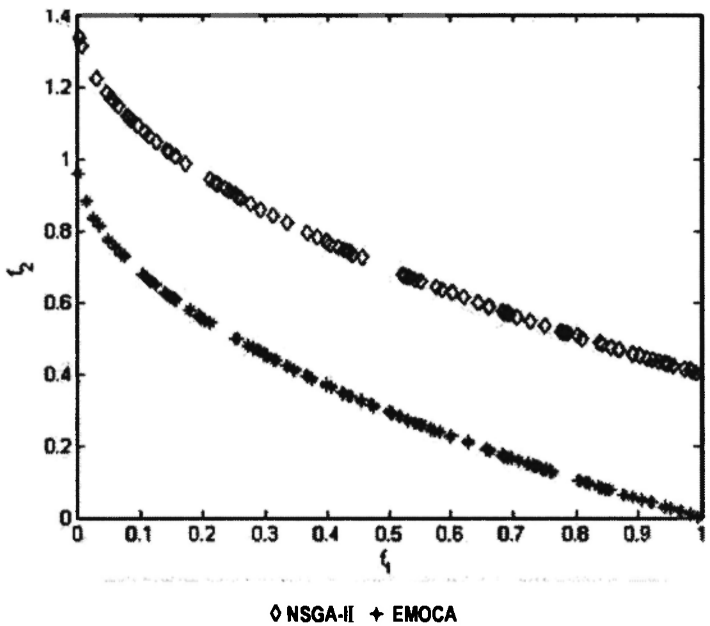

Fig. 3: Performance comparison of EMOCA and NSGA=II on ZDT4

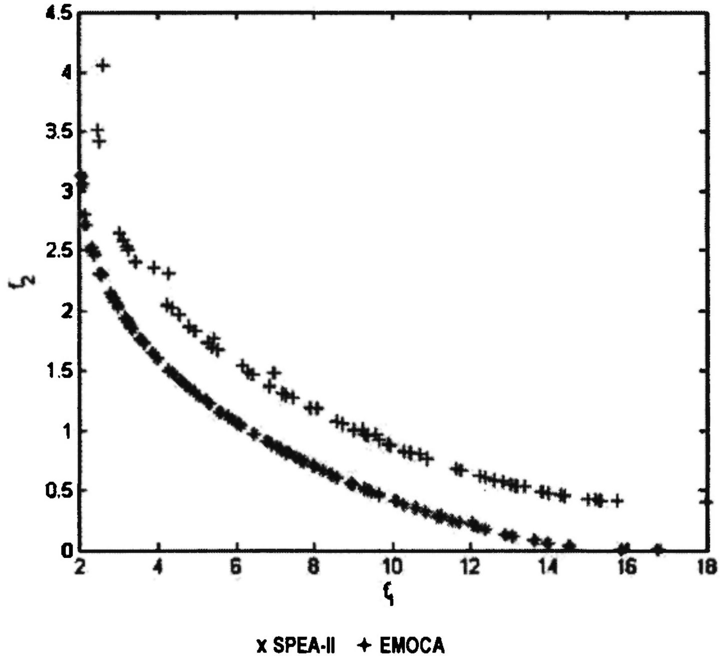

Fig. 4: Performance comparison of EMOCA and SPEA-II on POL

EMOCA in the worst case is $O(MN^2)$ per iteration where M is the number of objectives and N is the population size.

To compare the nature of solutions obtained by EMOCA, we plot the Pareto surfaces obtained by different algorithms for a few selective examples. Figure 2 shows the Pareto plots obtained by EMOCA, NSGA-II, SPEA-II, and PAES on ZDT6. ZDT6 has a non-convex Pareto-optimal front with non-uniform spacing between the solutions. The plot shows that EMOCA obtains a better spread and convergence compared to the other algorithms. Figure 3 shows the non-dominated solutions obtained by EMOCA and NSGA-II on ZDT4. The problem ZDT4 has 21^9 different local Pareto-optimal fronts (Deb 2001). This is a challenging problem for *MOEA*s. The plots clearly show that EMOCA has a better convergence and diversity compared to NSGA-II.

Figure 4 shows the Pareto plots obtained by EMOCA and SPEA-II on POL. The function POL has a non-convex and disconnected Pareto-optimal front. We observe that EMOCA obtains solutions with better convergence and spacing compared to the solutions obtained by SPEA-II.

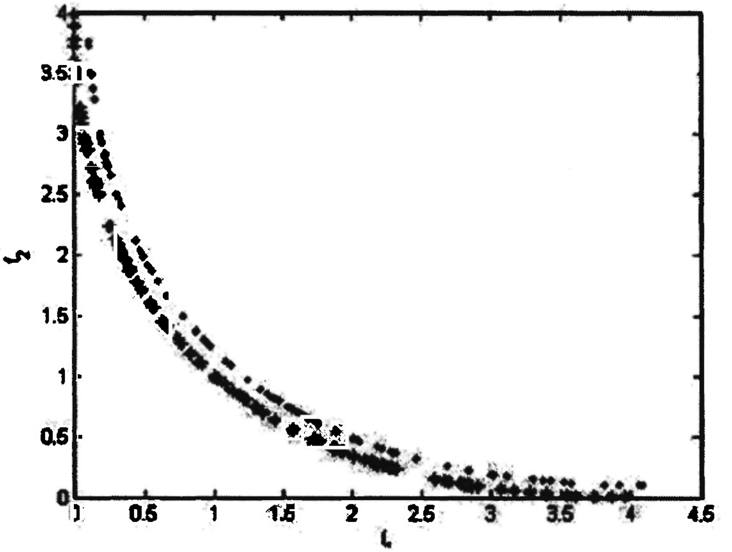

Fig. 5: Performance comparison of EMOCA and PAES on SCH

Figure 5 shows the Pareto plots obtained by EMOCA and PAES on SCH. The plot shows that EMOCA obtains a better convergence toward the Pareto-optimal front than PAES.

We now present the performance comparison between different algorithms in terms of different performance metrics. While evaluating the performance of *MOEA*s, we consider both convergence to the Pareto optimal front and diversity, using four metrics: generational distance, set coverage metric, domination metric and spread metric.

4.1 Generational Distance

The generational distance

$$GD = (\sum_{i=1}^{|Q|} d_i^2)^{0.5} \text{ (Valdhuizen 1999)}$$

evaluates the convergence of the non-dominated solutions obtained by the algorithm to the true Pareto-optimal front where Q indicates the set of non-dominated solutions obtained by the algorithm and d_i is the Euclidean distance between the solution $i \in Q$ and the nearest solution of the Pareto-optimal front[3] P.

4.2 Set Coverage Metric (C – metric)

The set coverage metric (Zitzler 1999) calculates the fraction of solutions in one non-dominated set (obtained by one algorithm) that are dominated by those obtained by the other algorithm, and is defined as

$$C(A,B) = \frac{|\{b \in B \mid \exists a \in A : a >> b\}|}{|B|} \quad (4)$$

$C(A,B)=1$ indicates that every solution in B is dominated by solutions in A and $C(A,B)=0$ means that none of the solutions in B is dominated by any element in A. Since the C metric is asymmetric, it is necessary to examine both $C(A,B)$ and $C(B,A)$.

4.3 Domination Metric

The domination metric (*Dom* metric) (Rajagopalan et al. 2004) is based on the number of solutions (obtained by one algorithm) dominated by each solution obtained by the other algorithm. The *Dom* metric is defined as:

$$Dom(A,B) = d(A,B)/(d(A,B)+d(B,A)) \quad (5)$$

where $d(X,Y) = \sum_{x} |\{ y \in Y \mid x >> y \}|$

Mutually non-dominating solutions are ignored while evaluating the dominance factor $d(A,B)$ so that $Dom(B,A)=1-Dom(A,B)$. If each solution of algorithm A dominates every solution of algorithm B, then $Dom(A,B)=1$ and $Dom(B,A)=0$.

[3] Since the Pareto-optimal solutions for the test functions are known, we obtain a set of 500 uniformly spaced solutions from the true Pareto-optimal front in the objective space, as in (Deb et al. 2000).

4.4 Spread Metric

The spread metric (Deb et al. 2001) evaluates the diversity of the solutions obtained and is defined as:

$$\Delta = \frac{\left(\sum_{m=1}^{M} d_m^e + \sum_{i=1}^{|Q|} | d_i - \bar{d} | \right)}{\left(\sum_{m=1}^{M} d_m^e + |Q| \bar{d} \right)} \qquad (6)$$

where d_i is the Euclidean distance between neighboring solutions and $\bar{d}$ is the mean value of the distances. The parameter d_m^e is the distance between extreme solutions of the Pareto-optimal front and the nearest solution of Q corresponding to the m^{th} objective function. Δ approaches zero when the non-dominated solutions obtained by the algorithm are near uniformly spaced.

4.5 Performance Comparison of Different Algorithms Using the Metrics

Tables 3, 4 and 5 show the mean and variance of the generational distance, C metric and *Dom* metric values obtained over 30 trials. The results clearly indicate that EMOCA outperforms NSGA-II in all test problems except in FON. EMOCA outperforms SPEA-II and PAES in all test problems except ZDT3. The performance of EMOCA is identical to SPEA-II and PAES in ZDT3. These results confirm that EMOCA performs the best in a majority of the test problems. In particular, for problems ZDT4 and ZDT6, EMOCA obtains much lower generational distance values as compared to the other algorithms. Table 6 shows the mean and variance of the spread metric obtained over 30 trials. The results indicate that EMOCA outperforms NSGA-II and PAES in all test problems in terms of the spread. In problem ZDT4, SPEA-II performs the best in terms of spread. On all other test problems EMOCA outperforms SPEA-II.

5. CONCLUSIONS

We have proposed a new MOEA called EMOCA that employs a stochastic replacement selection strategy that considers both non-domination

TABLE 3

Problem	EMOCA		NSGA-II		SPEA-II	
	Mean	variance	Mean	variance	Mean	variance
KUR	**0.59**	0.05	**0.97**	0.008	**0.99**	0.0003
FON	0.40	0.021	**0.98**	0.0001	**0.73**	0.046
SCH	**0.63**	0.06	**1**	0	**0.75**	0.026
POL	**0.65**	0.061	**0.91**	0.009	**0.73**	0.037
ZDT1	**0.78**	0.035	**1**	0	**0.82**	0.03
ZDT2	**0.8**	0.037	**0.99**	0	**0.76**	0.041
ZDT3	**0.83**	0.009	0.42	0.0003	0.46	0.054
ZDT4	**0.80**	0.082	**0.99**	0.002	**0.70**	0.008
ZDT6	**0.96**	0.0332	**0.99**	0.0001	**0.85**	0.049

TABLE 4

Mean and variance of C metric values for various algorithms over 30 trials

Problem	C(NSGA-II,EMOCA) C(EMOCA,NSGA-II)		C(SPEA-II, EMOCA) C(EMOCA, SPEA-II)		C(PAES,EMOCA) C(EMOCA,PAES)	
	mean	variance	mean	variance	mean	variance
KUR	**0.12** 0.40	0.0017 0.0004	**0.09** 0.72	0.0001 0.016	**0.13** 0.69	0.18 0.0012
FON	0.63 **0.104**	0.0012 0.0003	**0.01** 0.68	0.0002 0.008	**0.12** 0.89	0.001 0.001
SCH	**0.01** 0.521	0.0052 0.003	**0.02** 0.9	0 0.02	**0.017** 0.83	0.005 0.00003
POL	**0.07** 0.627	0.0017 0.0002	**0.06** 0.6	0.02 0.004	**0.08** 0.85	0.014 0
ZDT1	**0.03** 0.619	0.0091 0.0006	**0.00** 0.64	0 0.01	**0.058** 0.72	0.0031 0.0017
ZDT2	**0.02** 0.41	0.0005 0.001	**0.001** 0.6	0 0.01	**0.05** 0.75	0.003 0.001
ZDT3	**0.05** 0.85	0.0026 0.0006	**0.003** 0.12	0 0.01	**0.053** 0.16	0.01 0.003
ZDT4	**0.04** 0.91	0.032 0.001	**0.003** 0.66	0 0.02	**0.043** 0.79	0.003 0.014
ZDT6	**0.01** 0.93	0.0073 0.0211	**0.01** 0.65	0.009 0.12	**0.1** 0.83	0.005 0.0003

TABLE 5

Mean and variance of *Dom* metric values over 30 trials. The values in bold indicate *Dom(A,B)*>0.5

Problem	*Dom* (EMOCA,NSGA-II)		*Dom*(EMOCA,SPEA-II)		*Dom* (EMOCA,PAES)	
	Mean	variance	mean	variance	Mean	variance
KUR	**0.59**	0.05	**0.97**	0.008	**0.99**	0.0003
FON	0.40	0.021	**0.98**	0.0001	**0.73**	0.046
SCH	**0.63**	0.06	**1**	0	**0.75**	0.026
POL	**0.65**	0.061	**0.91**	0.009	**0.73**	0.037
ZDT1	**0.78**	0.035	**1**	0	**0.82**	0.03
ZDT2	**0.8**	0.037	**0.99**	0	**0.76**	0.041
ZDT3	**0.83**	0.009	0.42	0.0003	0.46	0.054
ZDT4	**0.80**	0.082	**0.99**	0.002	**0.70**	0.008
ZDT6	**0.96**	0.0332	**0.99**	0.0001	**0.85**	0.049

and diversity. We have compared the performance of EMOCA with NSGA-II, SPEA-II and PAES on nine difficult test problems with distinct features. Several performance measures were used to compare the algorithms. The simulation results show that EMOCA outperforms the other algorithms in eight out of the nine test problems in terms of convergence and diversity, consistently discovering a widely spread set of non-dominated solutions.

We have also employed EMOCA for several real world applications such as path planning, sensor placement and mobile agent routing in wireless sensor networks (Rajagopalan et al. 2004, Rajagopalan et al. 2005). For example, in path planning, simulations with three objectives (path length, risk and reward) on a graph with 10,000 nodes showed that EMOCA outperformed NSGA-II and PAES. The successful performance of EMOCA in the real world applications and test problems show that EMOCA is an efficient multi-objective optimization algorithm that should find extensive applications in optimization problems spanning a wide variety of areas from path planning to wireless networks.

TABLE 6

Mean and variance of the spread metric over 30 trials

Problem	EMOCA		NSGA-II		SPEA-II		PAES	
	mean	variance	mean	variance	mean	variance	mean	variance
KUR	0.1032	0.0008	0.4347	0.00092	0.3034	0.041	0.371	0.001
FON	0.1593	0.0041	0.3978	0.0073	0.6634	0.0207	0.562	0.054
SCH	0.2321	0.003	0.4891	0.0047	0.981	0.002	0.286	0.012
POL	0.1077	0.0029	0.4561	0.00294	0.2359	0.059	0.251	0.032
ZDT1	0.4024	0.0047	0.4109	0.0018	0.6404	0.014	0.531	0.038
ZDT2	0.2482	0.0023	0.4476	0.0052	0.6437	0.019	0.581	0.066
ZDT3	0.4853	0.0016	0.6898	0.0096	0.7282	0.034	0.634	0.037
ZDT4	0.3072	0.0086	0.7451	0.00178	0.195	0.113	0.412	0.07
ZDT6	0.5399	0.0016	0.6981	0.0029	0.6543	0.056	0.831	0.054

REFERENCES

Coello Coello, C.A. and Lamont, G.B. 2004. Application of multi-objective evolutionary algorithms, *World Scientific*, Singapore.

Deb, K. and Agarwal, R.B. 1995. Simulated binary crossover for continuous search space, *Complex Systems*, 9, 115-148.

Deb, K. 1999. Multi-objective genetic algorithms: Problem difficulties and construction of test functions, *Evolutionary Computation*, 7, 205-230.

Deb, K. Pratap, A. Agarwal, S. and Meyarivan, T. 2000. A fast and elitist multi-objective genetic algorithm: NSGA-II, *In Proceedings Parallel Problem Solving from Nature VI*, 849-858.

Deb, K. 2001. *Multi-objective optimization using evolutionary algorithms*, Wiley and Sons, England.

Fonseca, C.M. and Fleming, P.J. 1993. Genetic algorithms for multi-objective optimization: Formulation, discussion and generalization, *In Proceedings of the fifth international conference on Genetic Algorithms*, 416-423.

Hajela, P. and Lin, C.Y. 1992. Genetic strategies in multicriterion optimal design, *Structural Optimization*, 4, 99-207.

Horn, J., Nafpliotis, N., and Goldberg, D.E. 1994. A niched pareto genetic algorithm for multiobjective optimization, *In Proceedings of the first IEEE conference on Evolutionary Computation*, 1, 82-87.

Knowles, J.D. and Corne, D.W. 1999. The Pareto archived evolution strategy: A new baseline algorithm for multi-objective optimization, *In Proceedings of the 1999 congress on Evolutionary Computation*, 98-105.

Knowles, J.D and Corne, D.W. 2002. On metrics for comparing non-dominated sets, *In Proceedings of IEEE Congress on Evolutionary Computation*, 711-16.

Rajagopalan, R., Mohan, C.K., Mehrotra, K.G., and Varshney, P.K. 2004. Evolutionary multi-objective crowding algorithm for path computations, *In Proceedings of Fifth International Conference on Knowledge based Computer Systems*, 46-65.

Rajagopalan, R., Mohan, C.K., Varshney, P.K., and Mehrotra, K.G. 2005. Sensor placement for energy efficient target detection in wireless sensor networks: A multi-objective optimization approach, *In Proceedings of the 39th Annual Conference on Information Sciences and Systems, Baltimore, MD, USA*.

Rajagopalan, R., Mohan, C.K., Varshney, P.K., and Mehrotra, K.G. 2005. Multi-objective mobile agent routing in wireless sensor networks, *In Proceedings of the IEEE Congress on Evolutionary Computation.*

Schaffer, J.D. 1985. Multiobjective optimization with vector evaluated genetic algorithms, *In Proceedings of an International Conference on Genetic Algorithms and their Applications*, 93-100.

Veldhuizen, D.V. 19999. *Multi-objective evolutionary algorithms: Classifications, analyses and new innovations*, PhD thesis, Air Force Institute of Technology, Ohio, Technical Report AFIT/DS/ENG/99-01.

Zitzler, E. 1999. *Evolutionary algorithms for multi-objective optimization: Methods and applications*, PhD thesis, Swiss Federal Institute of Technology (ETH).

Zitzler, E., Deb, K. and Thiele, L. 2000. Comparison of multi-objective evolutionary algorithm: Empirical results, *Evolutionary Computation*, **8**, 173-95.

Zitzler, E., Laumanns, M. and Thiele, L. 2001. *SPEA2: Improving the strength Pareto evolutionary algorithm,* Swiss Federal Institute of Technology, Tech. Rep. TIK-Rep. 103.