STUDIA UNIV. BABEŞ–BOLYAI, INFORMATICA, Volume **XLVI**, Number 2, 2001

A NEW EVOLUTIONARY ADAPTIVE REPRESENTATION PARADIGM

D. DUMITRESCU, CRINA GROŞAN, MIHAI OLTEAN

Abstract. In this paper a new evolutionary paradigm is proposed. A technique called Adaptive Representation Evolutionary Algorithm (AREA) based on this paradigm is designed. AREA involves dynamic alphabets for encoding solutions. Proposed adaptive representation is more compact than binary representation. Mutation is the unique variation operator. Mutations are usually more aggressive when higher alphabets are used. Therefore the proposed encoding ensures an efficient exploration of the search space.

Numerical experiments seem to indicate that APA process better than the best multiobjective evolutionary algorithms.

An AREA technique is used for solving multiobjective optimization problems. The resulting algorithm is called Adaptive Pareto Algorithm (APA).

Keywords: Evolutionary Computation, Evolutionary Multiobjective Optimization, Pareto Set, Higher Alphabet Encoding, Adaptive Representation.

1. Introduction

In this paper we propose a new evolutionary paradigm. An algorithm based on this paradigm and using a powerful adaptive representation is designed. The algorithm called Adaptive Representation Evolutionary Algorithm (AREA). AREA technique operators are mutation and selection for survival.

Many multiobjective optimization techniques using evolutionary algorithms have been proposed in recent years. Strength Pareto Evolutionary Algorithm (SPEA, [9]), Pareto Archived Evolution Strategy (PAES, [4]), Pareto Envelope – based Selection Algorithm (PESA, [1]), Nondominated Sorting Genetic Algorithm (NSGA II, [3]) and SPEA II ([10]) are the best present-day Multiobjective Evolutionary Algorithms (MOEAs).

Multi-alphabet representation proposed in this paper induces a powerful diversity maintaining mechanism. For this reason AREA technique seems to be very suitable for evolutionary multiobjective optimization purposes. Considered adaptive encoding allows solutions in the final population realizing a realistic picture of Pareto frontier.

2000 *Mathematics Subject Classification.* 68T05.

1998 *CR Categories and Descriptors.* I.2.8 [**Computing Methodologies**]: Artificial Intelligence – *Problem Solving, Control Methods, and Search.*

Numerical experiments with APA technique include several test functions reputated as difficult ([2], [7]) and comparisons with the best MOEAs.

The paper is structured as follows: Section 2 is a short resume of the principal recent evolutionary techniques for multiobjective optimization. Section 3 describes the proposed algorithm. In Section 4 a comparison of the proposed approach with some very efficient multiobjective evolutionary techniques is realized.

2. Recent MOEAs

In the last years a number of evolutionary algorithms for multiobjective optimization have been proposed. Some of them will be shortly reviewed here.

2.1. **Strength Pareto Evolutionary Algorithm.**

Zitzler and Thiele proposed an elitist evolutionary algorithm called Strength Pareto Evolutionary Algorithm (SPEA) ([9, 7]). The algorithm maintains an external population at every generation storing all nondominates solutions obtained so far. At each generation external population is mixed with the current population. All nondominated solutions in the mixed population are assigned fitness based on the number of solutions they dominate. Dominated solutions are assigned fitness worse than the worst fitness of any nondominates solutions. A deterministic clustering technique is used to ensure diversity among nondominates solutions.

Pareto Archived Evolution Strategy

Knowles and Corne [4] proposed a simple evolutionary algorithm called Pareto Archived Evolution Strategy (PAES). In PAES one parent generates by mutation one offspring. The offspring is compared with the parent. If the offspring dominates the parent, the offspring is accepted as the next parent and the iteration continues. If the parent dominates the offspring, the offspring is discarded and the new mutated solution (a new offspring) is generated. If the offspring and the parent do not dominate each other, a comparison set of previously nondominated individuals is used.

For maintaining population diversity along Pareto front, an archive of nondominated solutions is considered. A new generated offspring is compared with the archive to verify if it dominates any member of the archive. If yes, then the offspring enters the archive and is accepted as a new parent. The dominated solutions are eliminated from the archive. If the offspring does not dominate any member of the archive, both parent and offspring are checked for their nearness with the solution of the archive. If the offspring resides in the least crowded region in the parameter space among the members of the archive, it is accepted as a parent and a copy is added to the archive.

2.2. **Nondominated Sorting Genetic Algorithm.**

Deb and his students [3] suggested a fast elitist Nondominated Sorting Genetic Algorithm (NSGA II). In NSGA II, for each solution x the number of solutions that dominate solution x is calculated. The set of solutions dominated by x is also calculated. The first front (the current front) of the solutions that are nondominated is obtained.

Let us denote by S_i the set of solutions that are dominated by the solution x^i. For each solution x^i from the current front consider each solution x^q from the set S_i. The number of solutions that dominates $x^q$is reduced by one. The solutions that remain nondominates after this reduction will form a separate list. This process continues using the newly identified front as the current front.

Let $P(0)$ be the initial population of size N. An offspring population $Q(t)$ of size N is created from current population $P(t)$. Consider the combined population:

$R(t) = P(t) \quad \cup \quad Q(t)$.

Population $R(t)$ is ranked according nondomination. The fronts F_1, F_2, ... are obtained. New population $P(t+1)$ is formed by considering individuals from the fronts F_1, F_2, ..., until the population size exceeds N. Solutions of the last allowed front are ranked according to a crowded comparison relation.

NSGA II uses a parameter (called *crowding distance*) for density estimation for each individual. Crowded distance of a solution x is the average side-length of the cube enclosing the point without including any other point in the population. Solutions of the last accepted front are ranked according to the crowded comparison distance.

NSGA II works as follows. Initially a random population, which is sorted based on the nondomination, is created. Each solution is assigned a fitness equal to its nondomination level (1 is the best level). Binary tournament selection, recombination and mutation are used to create an offspring population. A combined population is formed from the parent and offspring population. The population is sorted according to the nondomination relation. The new parent population is formed by adding the solutions from the first front and the followings until exceed the population size. Crowding comparison procedure is used during the population reduction phase and in the tournament selection for deciding the winner.

3. AREA technique

In this paper we propose a new evolutionary paradigm, The main idea is to allow each solution be encoded on a different alphabet. Moreover representation of a particular solution is not fixed. Representation is adaptive and may be changed during the search process as an effect of mutation operator, An adaptive representation evolutionary algorithm (AREA) based on the new paradigm is designed.

AREA technique proposed in this paper uses a fixed population. Each AREA individual (chromosome) consists of a pair (x, B), where x is a string encoding object variables and B specifies the alphabet used for encoding x.

B is an integer number, $B \quad \geqslant 2$ and x is a string of symbols from the alphabet $\{0, 1, \ldots, B\text{-}1\}$. If $B = 2$, the standard binary encoding is obtained. The alphabet over which x is encoded may change during the search process.

Mutation is the unique variation operator. For mutation, a random number between 0 and 1 is uniformly generated for each position, including the last one, of the chromosom. Each position (gene) value is modified with a mutation probability (p_m).

Mutation can modify object variables as well as last position (fixing the representation alphabet). If the position giving B is changed, then the object variables will be represented using symbols over the new alphabet, corresponding to the mutated value of B. When the changing gene belongs to the object variable sub-string (x – part of the chromosome), the mutated gene is a symbol randomly chosen from the same alphabet.

4. APA method

In this section a new MOEA technique called Adaptive Pareto Algorithm (APA) is proposed. APA relies on the AREA method previously described.

AREA uses a unique population. No external or intermediary population is needed.

Initial population is randomly generated. Each individual is selected for mutation, which is the unique variation operator. The offspring and parent are compared. Dominance relation guides the survival.

If the offspring dominates the parent then the offspring enters the new population and the parent is removed. If the parent dominates the offspring obtained in $k$successive mutations then another alphabet is chosen and the parent is represented in symbols over this alphabet. In this case only representation is changed and the encoded solution does not change. Adaptive representation mechanism and the survival strategy is generates an effective and efficient diversity preserving mechanism.

APA algorithm

Proposed APA algorithm may be outlined as follows:

APA ALGORITHM:

```
begin
  Set $t = 0$;
  Random initializes chromosome population $P\ (0)$;
  Set to zero the number of harmful mutations for each individual in P(0);
  while ($t$ ¡ number of generations) do
  begin
    for k = 1 to PopSize do
      Mutate the k$^{th}$ chromosome from $P\ (t)$. An offspring is obtained.
      If the offspring dominates the parent then the parent is removed and the offspring is added to $P(t+1)$;
      else begin
        Increase the number of harmful mutations for current individual;
        If the number of harmful mutations = MAX_HARMFUL_MUTATIONS
        then begin
          Change the individual representation;
          Set to zero the number of harmful mutations for the current individual;
        endif
        Add individual to $P(t+1)$;
```

```
endif
endfor;
Set $t = t + 1$;
endwhile;
end.
```

Despite its simplicity APA is able to generate a population converging towards Pareto optimal set. Moreover, the diversity of the population is automatically maintained without any specialized mechanism.

Proposed APA algorithm realizes a realistic picture of Pareto optimal set. Numerical experiments emphasizes that for considered problems, APA technique is more effective then best present-day MOEAs. Moreover, APA's complexity is a reduced one with respect to the MOEAs techniques considered for comparison.

5. Comparison of Several Evolutionary Multiobjective Algorithms

In this section complexity of the proposed APA technique is compared with the complexity of several evolutionary multiobjective optimization algorithms (SPEA, PAES and NSGA II).

Let us denote by m the number of objectives and by N the population size.

SPEA uses an internal and an external population. The fitness is assigned differently to the individuals from these populations. A deterministic clustering technique is used to reduce external population size the population diversity. The complexity of this algorithm implementation is mN^2.

PAES uses a single parent, which generate an offspring. An archive, which maintains the nondominated solutions, is created. Let be a the archive size. The worst case complexity for the PAES is amN. Since the archive size is usually proportional to the population size N,the overall complexity of the algorithm is mN^2.

NSGA II computes for each individual x the number of solutions that dominates it and the number of solution, which x dominates. NSGA II uses for this N^2 computations. Identifying the fronts requires (in the worst case) N^2 computations. The overall complexity is $(mN^2 + N^2)$ or N^2. So, the complexity may increase from N to N^2 (the worst case). Computation complexity for density estimation is $mN \log N$. For sorting the combined population $2N \log(2N)$ computations are necessary. Overall NSGA II complexity is thus mN^2.

APA uses a unique fixed size population. Each individual is considered for mutation. This requires mN operations. The algorithm does not use a superposed mechanism for diversity maintaining. Overall complexity of SMEA algorithm is thus $O(mN)$.

6. Numerical experiments

In this section we compare the performance of APA algorithm with the performances of SPEA, NSGA II, PAES.

For this purpose by using six test functions introduced by Deb, Zitzler and Thiele [7] are used.

6.1. **Test functions.** Each test function considered in this section is built by using three functions f_1, g, h. Biobjective function T considered here is

$T(x) = (f_1(x),\ f_2(x))$.

The optimization problem is:

$$\begin{cases} Minimize\, T(x),\ where\, f_2(x) = g(x_2, \ldots, x_m)h(f_1(x_1),\, g(x_2, \ldots x_m), \\ \qquad\qquad x = (x_1, \ldots, x_m) \end{cases}$$

The five test functions used in this paper for comparison are:

Test function T_1 is defined using the following functions:

$$\begin{aligned} &f_1(x_1) = x_1, \\ &g(x_2, \ldots .x_m) = 1 + 9 \cdot \textstyle\sum_{i=2}^{m} x_i/(m-1), \\ &h(f_1, g) = 1 - \sqrt{f_1/g}, \end{aligned}$$

where m= 30 and $x_i \in$ [0,1] i = 1,2,. . . ,m.

Pareto optimal front for the problem T_1 is convex and is characterized by the equation

$$g(x) = 1.$$

Test function T_2 is defined by considering the following functions:

$$\begin{aligned} &f_1(x_1) = x_1 \\ &g(x_2, \ldots .x_m) = 1 + 9 \cdot \textstyle\sum_{i=2}^{m} x_i/(m-1) \\ &h(f_1, g) = 1 - (f_1/g)^2 \end{aligned}$$

where m = 30 and $x_i \in$ [0,1], i = 1,2,. . . ,m.

Pareto optimal front is characterized by the equation

$$g(x)=1.$$

T_2 is the nonconvex counterpart to T_1.

Pareto optimal set corresponding to the Test function T_3 presents a discreten feature. Pareto optimal front consists of several noncontiguous convex parts. The involved functions are:

$$\begin{aligned} &f_1(x_1) = x_1 \\ &g(x_2, \ldots .x_m) = 1 + 9 \cdot \textstyle\sum_{i=2}^{m} x_i/(m-1) \\ &h(f_1, g) = 1 - \sqrt{f_1/g} - (f_1/g)\sin(10\pi f_1) \end{aligned}$$

where m= 30 and $x_i \in$ [0,1], i = 1,2,. . . m.

Pareto optimal front is characterized by the equation

$$g(x) = 1.$$

The introduction of the function *sin* in the expression of function h causes discontinuity in the Pareto optimal front. However, there is no discontinuity in the parameter space.

The test function T_4 contains 21^9 local Pareto optimal fronts and, therefore, it tests the EA ability to deal with multimodality. The involved functions are defined by:

$$f_1(x_1) = x_1$$
$$g(x_2, \ldots .x_m) = 1 + 10(m-1) + \sum_{i=2}^{m} (x_i^2 - 10\cos(4\pi x_i))$$
$$h(f_1, g) = 1 - \sqrt{f_1/g}$$

where $m = 10$, $x_1 \quad \in$ [0,1] and $x_2,\ldots,x_m \quad \in$ [-5,5].

Global Pareto optimal front is characterized by the equation

$$g(x) = 1.$$

The best local Pareto optimal front is described by the equation

$$g(x) = 1.25.$$

Note that not all local Pareto optimal sets are distinguishable in the objective space.

$$f_1(x_1) = 1 - \exp(-4x_1)\sin^6(6\pi x_1)$$
$$g(x_2, \ldots .x_m) = 1 + 9 \cdot \left(\sum_{i=2}^{m} x_i/(m-1)\right)^{0.25}$$
$$h(f_1, g) = 1 - (f_1/g)^2$$

The test function T_5 includes two difficulties caused by the nonuniformity of the search space. First, the Pareto optimal solutions are nonuniformly distributed along the global Pareto optimal front (the front is biased for solutions for which $f_1(x)$ is neat one). Second, the density of the solutions is lowest near the Pareto optimal front and highest away from the front.

This test function is defined by using:
where m= 10, $x_i \quad \in$ [0,1], i = 1,2,... m.

The Pareto optimal front is characterized by the equation

$$g(x) = 1$$

and is nonconvex.

6.1.1. *Numerical comparisons.* Several numerical experiments were performed with APA. According to these experiments APA gives a good approximation of the Pareto front for all considered test functions.

For both test functions T_1 and T_2, the differences between the four considered algorithms are very small (see Figure 1 and Figure 2).

The difference between APA and the other algorithms is significant for test function $T4$. APA gives the best arrangement on the front. Good solution distribution is obtained also by NSGA II and SPEA. Solutions distribution realized by NSGA II and SPEA are close to Pareto front. Moreover solution only distribution supplied by APA is covers the real front.

For test function T_5, APA also gives the best solution arrangement on the Pareto front. PAES also gives distribution.

In these comparisons 25.000.000 function evaluations have been considered for each algorithm. This ensures a realistic comparison of the algorithm outputs.

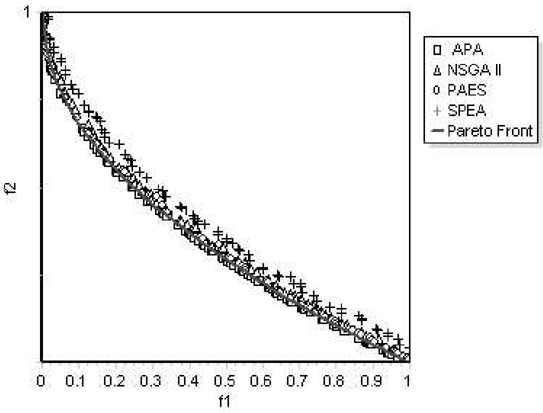

Figure 1. Results for test function T_1. Pareto optimal front is convex

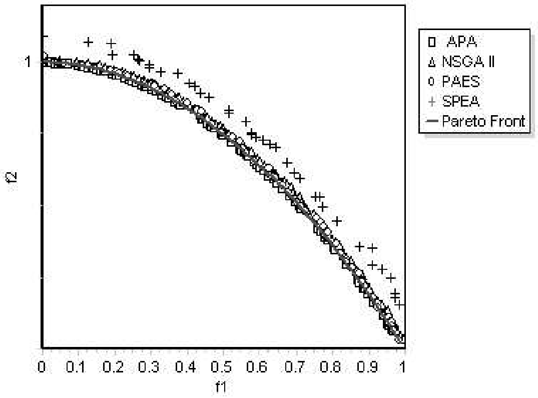

Figure 2. Results for test function $T2$. Pareto optimal front is nonconvex

Conclusions

In this paper a new evolutionary paradigm is proposed. An evolutionary algorithm (AREA) based on the new paradigm is designed.

A new evolutionary algorithm (called APA) for multiobjective optimization is also proposed. AREA uses a new, dynamic solution representation.

APA technique is compared with four well-known evolutionary multiobjective optimization algorithms. The results show that APA performs better than considered algorithms.

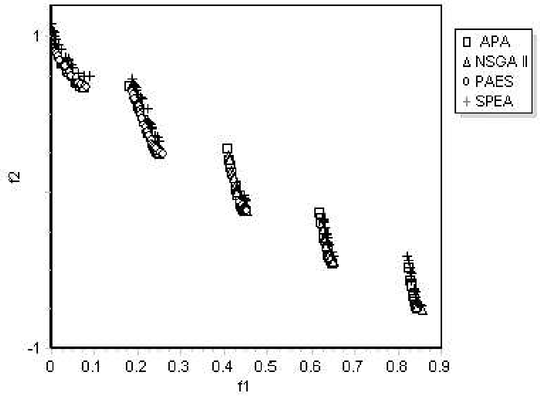

FIGURE 3. Results for test function $T3$. All considered algorithms give a good approximation of the Pareto front

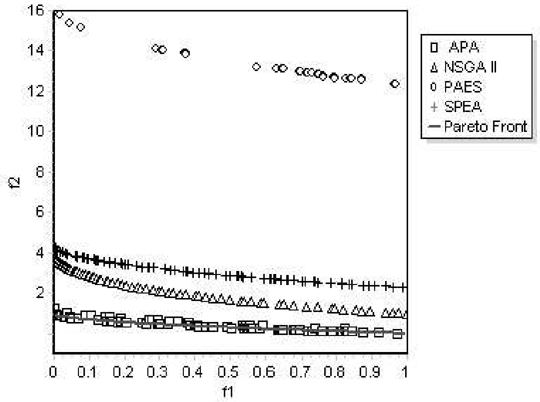

FIGURE 4. Results for test function T_4. For test function T_4, APA gives the best arrangement on the Pareto front. NSGA II and SPEA converge toward global Pareto front. PAES did not converge to the global Pareto front

REFERENCES

[1] Deb, K., Multi-objective genetic algorithms: Problem difficulties and construction of test functions. *Evolutionary Computation*, 7(3), (1999), pp. 205-230.

[2] Deb, K., S. Agrawal, Amrit Pratap and T. Meyarivan, A fast elitist non – dominated sorting genetic algorithm for multi-objective optimization: NSGA II. In M. S. et al. (Ed), *Parallel Problem Solving From Nature – PPSN VI*, Berlin, (2000), pp. 849 – 858. Springer.

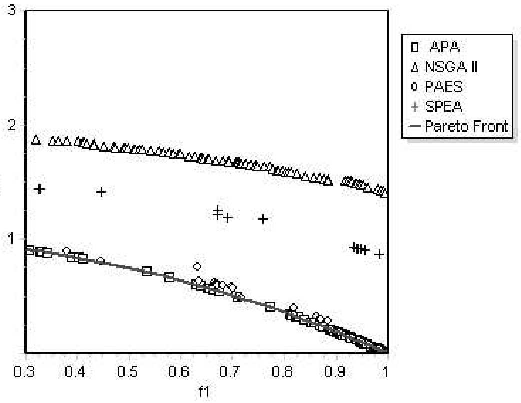

FIGURE 5. Results for test function T_5

[3] Dumitrescu, D., Groşan, C., Oltean, M., Simple Multiobjective Evolutionary Algorithm, Seminars on Computer Science, Faculty of Mathematics and Computer Science, Babeş-Bolyai University of Cluj-Napoca, 2001, pp. 3-12.

[4] Knowles, J. D. and Corne, D. W., The Pareto archived evolution strategy: A new baseline algorithm for Pareto multiobjective optimization. In *Congress on Evolutionary Computation (CEC 99)*, Volume 1, Piscataway , NJ, (1999), pp. 98 – 105. IEEE Press.

[5] Rudolph, G., On a multi – objective evolutionary algorithm and its convergence to the Pareto set. Technical Report No. CI – 17/98, Department of Computer Science/XI, university of Dortmund, (1998).

[6] Zitzler, E., Evolutionary algorithms for multiobjective optimization: Methods and Applications. Ph. D. thesis, Swiss Federal Institute of Technology (ETH) Zurich, Switzerland. TIK – Schriftenreiche Nr. 30, Diss ETH No. 13398, (1999), Shaker Verlag, Aachen, Germany.

[7] Zitzler, E., Deb, K. and Thiele, L., Comparison of multiobjective evolutionary algorithms: empirical results. Technical report 70, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH), (1999), Zurich.

[8] Zitzler, E. and Thiele, L., Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Transaction on Evolutionary Computation 3*(4), (1999), pp. 257 – 271.

[9] Zitzler, E. and Thiele, L., An evolutionary algorithm for multiobjective optimization: The strength Pareto approach. Technical report 43, Computer engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH), (1999), Zurich.

"BABEŞ-BOLYAI" UNIVERSITY, FACULTY OF MATHEMATICS AND COMPUTER SCIENCE, 1 M. KOGĂLNICEANU STREET, RO-3400 CLUJ-NAPOCA, ROMANIA

E-mail address: `ddumitr|cgrosan|moltean@cs.ubbcluj.ro`