A Multi-Objective Meta-Model Assisted Memetic Algorithm with Non Gradient-Based Local Search

Saúl Zapotecas Martínez*
Computer Science Department
CINVESTAV-IPN
México, DF
saul.zapotecas@gmail.com

Carlos A. Coello Coello†
Computer Science Department
CINVESTAV-IPN
México, DF
ccoello@cs.cinvestav.mx

ABSTRACT

In this paper, we present an approach in which a local search mechanism is coupled to a multi-objective evolutionary algorithm. The local search mechanism is assisted by a meta-model based on support vector machines. Such a mechanism consists of two phases: the first one involves the use of an aggregating function which is defined by different weighted vectors. For the (scalar) optimization task involved, we adopt a non-gradient mathematical programming technique: the Hooke-Jeeves method. The second phase computes new solutions departing from those obtained in the first phase. The local search engine generates a set of solutions which are used in the evolutionary process of our algorithm. The preliminary results indicate that our proposed approach is quite promising.

Categories and Subject Descriptors

I.2.8 [**Computing Methodologies**]: Artificial Intelligence—*Problem Solving, Control Methods, and Search.*

General Terms

Algorithms, Performance, Theory.

Keywords

Multi-objective optimization, support vector machines, hybrid algorithms, Hooke-Jeeves algorithm.

1. INTRODUCTION

Multi-objective evolutionary algorithms (MOEAs) have been successfully applied in a wide variety of engineering and scientific problems. However, in the real world there exist problems in which the objective functions are very expensive (computationally speaking) to evaluate (e.g., in aeronautical engineering). For such problems, the use of MOEAs becomes inefficient and even impractical, because of the high computational costs involved. In recent years, the development of MOEAs hybridized with mathematical programming techniques as well as the use of function approximation models has significantly increased. These sort of hibridizations are not a new task. The use of MOEAs assisted by surrogate models has been adopted by several researchers, mainly for solving engineering optimization problems. A surrogate is a function which can be modelled by means of a simple linear regression, by a polynomial regression or by more elaborate models such as Artificial Neural Networks (ANNs), Radial Basis Functions (RBFs), Support Vector Machines (SVMs), Gaussian processes (also known as Kriging), among others.

In this work, we present a strategy which employs a surrogate model (based on SVMs) as a function predictor. The meta model is used for approximating solutions to the Pareto front by means of a local search engine. The knowledge provided by the local search engine is incorporated into a MOEA by recombining the solutions returned by the local search engine with the individuals from the current population. Our main goal is to speed up convergence towards the Pareto optimal set while reducing the total number of objective function evaluations.

2. OUR PROPOSED APPROACH

As indicated before, our proposal is characterized by using an approximation model based on SVMs. Additionally, it adopts an external archive A and a solutions set R (obtained by the local search engine) which are used to create the offspring population in the MOEA. The meta-model is trained with the set D, which consists of all the solutions evaluated with the real fitness function values obtained up to the current generation. The general scheme of this approach is described next.

2.1 The Algorithm

Initially, we defined a set of solution S of size $2N$ (where N is the population size) which is randomly distributed in the search space using the Latin hypercube sampling method. The initial population P_0 is defined by N solutions randomly chosen from S. Our algorithm uses the current population P_t, the set of solutions R_t provided by the local search mechanism and the external archive A_t (defined by all the non-dominated solutions found throughout the evolutionary process) to create the offspring population Q_t at generation t.

2.1.1 Archiving solutions

Our proposed approach uses an external archive A which stores all the non-dominated solutions found at each generation of the MOEA. The archive is bounded according to the population size. At each generation, the archive is updated with the new non-dominated solutions found in the popu-

*The first author acknowledges support from CINVESTAV-IPN and CONACyT.
†The second author acknowledges support from CONACyT project no. 103570.

Copyright is held by the author/owner(s).

GECCO'10, July 7–11, 2010, Portland, Oregon, USA.
ACM 978-1-4503-0072-8/10/07.

lation P. If the number of solutions is greater than N (i.e. more greater than the population size), then the archive is pruning using k-means algorithm (with $k = N$).

2.1.2 *Creating the offspring population*

We consider D to be the set of all solutions obtained by the MOEA. Since we assume that the MOEA is able to converge to the true Pareto optimal set, we also assume that in the last generations of the MOEA, the predictor function generates good approximations of the true objective function values (this is because all the non-dominated solutions are kept in the set D). Therefore, the set of solutions R obtained by the local search algorithm (within the meta-model) will have a low approximation error. Thus, we assume that both the R set and the A set have solutions of similar quality. Based on the previous, crossover takes place between each individual of the population P (the current population) and an individual which can be chosen from either R or A. Thus, we define the parents for the crossover operator according to the following rule:

$$\begin{aligned} parent^1 &= x_i \in P \quad \forall i = 1, \ldots, N \\ parent^2 &= \begin{cases} y \in R, & \text{if } \left(g < 1 - \frac{|A|}{2N}\right) \\ y \in A, & \text{otherwise} \end{cases} \end{aligned} \tag{1}$$

where g is a uniformly distributed random number within $(0, 1)$ and y is a solution randomly chosen from A or R. Clearly, when the archive pool $\mathcal{A}$ is full, $|A| = N$ and equation (1) guarantees to choose a solution from either R or A, both with the same probability.

The mutation operator is applied (based on certain probability) to each new child generated by the crossover operator. In this work, we adopted the genetic operators from the NSGA-II [1] (Simulated Binary Crossover (SBX) and Parameter-Based Mutation (PBM)), but other operators can be employed. Below, we describe the local search mechanism employed in our proposal.

2.2 Local Search

The local search mechanism incorporated into our meta-model has as its main goal to find new solutions nearby the solutions provided by the MOEA (which should be at least nondominated with respect to the current and previous populations). In this way, while the local search engine explores promising areas into the meta-model, the MOEA performs a broader exploration of the search space.

2.2.1 *Approximating solutions*

In order to generate approximate solutions of the Pareto optimal set, we solve n_w different scalarization functions defined by a Tchebycheff problem (it is worth noting that it is possible to use any other scalarization function). Initially, a set of n_w weighted vectors $W \subset \mathbb{R}^k$ is defined. In this way, the approximate solutions to the Pareto optimal set are obtained by solving the n_w different Tchebycheff problems defined by each weighted vector. For each weighted vector $w_j \in W$, a set of solutions λ_j is found, which consists of all the solutions evaluated so far into the meta-model by solving the Tchebycheff problem.

Here, we use the Hooke and Jeeves algorithm method [2] in order to solve each Tchebycheff problem of our interest. Clearly, the candidate solutions are evaluated into the surrogate model.

The initial search point for solving the first problem corresponding to the w_1 vector, is given by the solution $x_s \in \{P_t \cup A\}$ which minimizes the Tchebycheff problem with respect to the weighted vector w_1.

The remaining sets λ_j $(j = 2, \ldots, n_w)$ are obtained by solving the Tchebycheff problem for the weighted vector w_j. Thus, the we define the set Λ as the union of all the sets λ_j found by solving the n_w Tchebycheff problems.

2.2.2 *Generating New Solutions*

Our proposal uses a heuristic method for generating more approximate solutions of the Pareto optimal set. Here, we adopt the well-known Differential Evolution (DE) meta-heuristic [3]. First, we consider Λ to be the set of solutions found by the above process. The initial population in DE is given by $G_0 = \Lambda$. Each new individual $x_{i,g+1}$ is stored (or not) in an external archive L according to a dominance rule. The archiving strategy can make that the set of solutions L increases or decreases its size. Thus, we generate more non-dominated solutions from archive L. In this way, the next population for the DE algorithm is defined by the set of solutions L (that is: $G_{g+1} = L$).

Since all the solutions in the archive are non-dominated, we can say that the algorithm has converged when it has obtained N different non-dominated solutions from the evolutionary process. However, this stopping criterion is not always satisfied. Thus, if we do not have N different solutions in a certain number of iterations, then we can use Pareto ranking for selecting N individuals from $\Lambda \cup L$. Therefore, the final set of solutions R given by the local search mechanism is defined by the set of solutions L (that is: $R = L$).

3. CONCLUSIONS

We have proposed a multi-objective memetic algorithm assisted by support vector machines, with the aim of performing an efficient exploration of the search space in multi-objective optimization problems of moderate dimensionality (10 to 30 decision variables). For the local search, we have used a weighted Tchebycheff function and the Hooke-Jeeves method as a minimizer for each problem defined by the weighted vectors. Our proposed approach was found to be competitive with respect to the NSGA-II over a set of test functions taken from the specialized literature, when performing only 1,000 fitness function evaluations.

We consider that our strategies adopted to approximate solutions to the Pareto optimal set and to incorporate knowledge into our memetic algorithm, make our approach competitive with respect to state-of-the-art MOEAs.

4. REFERENCES

[1] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA–II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, April 2002.

[2] R. Hooke and T. A. Jeeves. “direct search” solution of numerical and statistical problems. *J. ACM*, 8(2):212–229, 1961.

[3] K. V. Price, R. M. Storn, and J. A. Lampinen. *Differential Evolution. A Practical Approach to Global Optimization*. Springer, Berlin, 2005. ISBN 3-540-20950-6.