Applied Soft Computing 19 (2014) 290–311

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Adaptive directional local search strategy for hybrid evolutionary multiobjective optimization

Hyoungjin Kim[a,*], Meng-Sing Liou[b,1]

[a] Science Applications International Corporation, Cleveland, OH 44135, United States
[b] NASA Glenn Research Center, Cleveland, OH 44135, United States

ARTICLE INFO

Article history:
Received 27 November 2012
Received in revised form 11 February 2014
Accepted 13 February 2014
Available online 13 March 2014

Keywords:
Evolutionary multiobjective optimization
Memetic algorithms
Directional operator
Adaptive local search

ABSTRACT

A novel adaptive local search method is developed for hybrid evolutionary multiobjective algorithms (EMOA) to improve convergence to the Pareto front in multiobjective optimization. The concepts of local and global effectiveness of a local search operator are suggested for dynamic adjustment of adaptation parameters. Local effectiveness is measured by quantitative comparison of improvements in convergence made by local and genetic operators based on a composite objective. Global effectiveness is determined by the ratio of number of local search solutions to genetic search solutions in the nondominated solution set. To be consistent with the adaptation strategy, a new directional local search operator, eLS (efficient Local Search), minimizing the composite objective function is designed. The search direction is determined using a centroid solution of existing neighbor solutions without making explicit calculations of gradient information. The search distance of eLS decreases adaptively as the optimization process converges. Performances of hybrid methods NSGA-II + eLS are compared with the baseline NSGA-II and NSGA-II + HCS1 for multiobjective test problems, such as ZDT and DTLZ functions. The neighborhood radius and local search probability are selected as adaptation parameters. Results show that the present adaptive local search strategy can provide significant convergence enhancement from the baseline EMOA by dynamic adjustment of adaptation parameters monitoring the properties of multiobjective problems on the fly.

1. Introduction

Multiobjective optimization methods are getting increased attention, as consideration of trade-offs between conflicting objectives is becoming more critical in multidisciplinary design optimization (MDO) of modern engineering problems. Evolutionary algorithms (EAs) are well suited for multiobjective optimization problems (MOP), because they are based on a population rather than single solution, and therefore can more naturally adapt to generate distributed solutions on the Pareto front. During the last two decades, evolutionary multiobjective optimization algorithms (EMOAs) have shown great progress and success [1–4].

A well-known drawback of EAs is their slow convergence near the optimum solution. This behavior is still true for EMOAs. The convergence behavior of EMOAs can be graphically explained in a two-dimensional design space of a two objective minimization problem in Fig. 1 [5], in which the concentric ellipses are iso-contour lines of each objective function. The size of a descent cone is a set of search directions in which any of the objectives are not decreasing with properly selected step sizes. Since the descent cone angle decreases as design converges, a random search is expected to have slow convergence near the Pareto front. Another possible reason of slow convergence of EMOAs is that the relative size of the descent zone decreases exponentially as the number of objectives increases. To cope with the convergence issue, hybrid methods incorporating local search into EMOAs have been suggested. The hybrid methods are also referred to as memetic EMO algorithms [6].

Local search methods used in the hybrid EMO algorithms can be put into two categories: neighborhood-based local searches and directional local searches. For all the references in the introduction section, combinatorial optimization methods are explicitly mentioned as combinatorial, and unmarked optimization methods in the text are for continuous optimization.

Neighborhood-based local search schemes generate perturbed solutions around a baseline solution and try to find better solutions than the baseline in terms of either a composite objective or

* Corresponding author at: MS 5-10, 21000 Brookpark Road, United States. Tel.: +1 216 416 5648.
E-mail address: hyoungjinkim1@gmail.com (H. Kim).
1 Address: Aeropropulsion Division, MS 5-11, 21000 Brookpark Road, United States.

http://dx.doi.org/10.1016/j.asoc.2014.02.019
1568-4946/

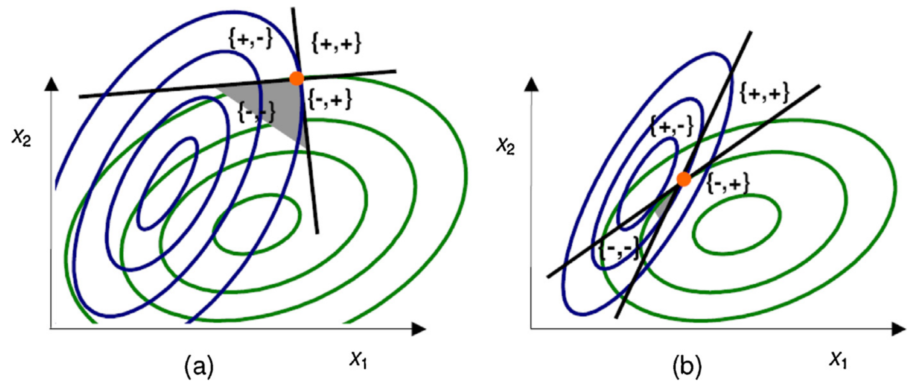

Fig. 1. Descent cones (shaded, {−,−}) for a two-parameter, two-objective minimization problem during initial (shown in (a)) and final (in (b)) stages of convergence. The concentric ellipses are iso-contour lines of each objective function. The cones are labeled with the sign of the corresponding change in objectives. The descent cone angle is relatively wide in the initial stages and decreases as the design converges.
This figure is taken from Ref. [5].

a Pareto-dominance relation. Deb and Goel [7] applied a hill climbing local search method to only final solutions obtained by EMO algorithms. A composite objective function is used with weighting factors determined by the location of each solution in the objective space so that a solution movement by the local search results in a better spread (diversity) of solutions. Ishibuchi and Murata [8,9] suggested the multiobjective genetic local search (MOGLS) algorithm for combinatorial optimization, in which a local search is applied to each offspring in every generation using a composite objective function with the same random weights for parental selection and evaluation of local search solutions. Jaszkiewicz [10] improved the MOGLS by adding mating restriction to its parental selection mechanism. Knowles and Corne [11] suggested a memetic Pareto archived evolution strategy (M-PAES) by including a crossover operator in PAES [12]. In M-PAES, all new solutions generated by genetic and local searches are accepted or rejected based on the Pareto-dominance relation and the grid-type partition of the objective space. In Ishibuchi and Narukawa [13], it was reported that the composite function approach gives better results than the Pareto-dominance relation approach for combinatorial optimization problems. Minella et al. [14] gave a comprehensive review of multi-objective algorithms for flowshop scheduling problems and identified two state of the art algorithms: MOSA (multi-objective simulated annealing) [15] and MOGALS [16]. Later, Dubois-Lacoste et al. [17] devised a new multi-objective local search method for flowshop scheduling problems hybridizing two-phase (TP) and Pareto local search (PLS) algorithms that is superior to the algorithms discussed by Minella et al. [14]. Liefooghe et al. [18] presented a more recent review of dominance-based multiobjective local search algorithms that are adopting a neighborhood structure and Pareto dominance relation in the context of combinatorial optimization.

In continuous optimization, the covariance matrix adaptation evolutionary strategy (CMA-ES) [48] is the state of the art algorithm for single-objective problems. A multi-objective variant of CMA-ES (MO-CMA-ES) was proposed by Igel et al. [53]. CMA-ES and its variants can also be considered as a neighborhood-based local search method.

Unlike the neighborhood-based methods, directional local search methods conduct a local search along a search direction, which can be determined either by sensitivity information of objective functions with respect to decision variables or by approximation of an improving search direction using neighborhood solutions. They will allow a very accurate local search if objective functions are differentiable and accurate gradient information can be obtained. An important issue for directional local search methods is whether to calculate the objective function gradient vector explicitly or not. In general cases without any efficient sensitivity analysis code available, an explicit calculation of the gradient vector of an objective function by a first-order finite differencing requires N_{dv} additional function calls, where N_{dv} is the number of decision variables. The computational cost would easily become prohibitive if N_{dv} is large and objective function evaluations are expensive.

Bosman and de Jong [19] tested three different techniques to use gradient information. They reported that using nondominated improving directions is superior to using gradients of objectives separately. Harada et al. [20] proposed a new gradient-based local search method called the Pareto Descent Method, which finds Pareto descent directions and moves solutions in such directions. In both Refs. [19,20], hybrid methods using EMO algorithms and gradient-based local searches are adopted with an explicit calculation of gradient information by finite differencing. Search directions are randomly selected among descent directions in the objective space. Both references, however, show difficulties in improving convergences especially for problems of a large number of decision variables N_{dv}, mainly because of excessive computational burden for gradient calculation by finite differencing and very expensive line searches. Since modern engineering MDO problems tend to have a large N_{dv} and expensive objective functions, we prefer not to explicitly calculate gradient information for directional local searches.

As an alternative to explicit calculation of gradient information, search directions can be approximated by utilizing neighboring individuals. Brown and Smith [5] suggested an approximation method for the local Jacobian matrix using the difference between a point of interest and local neighbors. Wanner et al. [21] proposed a quadratic polynomial approximation of all objective functions around an initial solution for local search with sample solutions gathered in the optimization process. Recently, Lara et al. [22] compared hybrid methods with and without explicit gradient information and showed that using the gradient information was advantageous for the problems they tested, assuming that an explicit gradient calculation is equivalent to five function calls. In Schutze et al. [51], however, the authors reported that using the gradient information was *not* advantageous for the test problems. The effectiveness of using explicitly calculated gradient information depends heavily on how to take the cost of sensitivity analysis into account in the computational budget.

Another major issue for the hybrid methods is that we do not know a priori if conducting a local search would be