

Multiple Criteria Scatter Search

Ricardo P. Beausoleil*

* Optimización, ICIMAF
No.503, Vedado, C. Habana, Cuba
Email: rbeausol@cidet.icmf.inf.cu

1 Introduction

This paper reports an evolutionary approach called Scatter Search, that uses the concept of Pareto optimality to obtain a good approximate Pareto frontier. In order to designate a subset of strategies to be a reference solutions a choice function called Kramer Selection is used. A variant of measure of Kemen-Snell may be used, in our case study, in order to find a diverse set to complement the subset of high quality current Pareto solutions. Path Relinking and Extrapolated Path Relinking are used as a Combined Method. Several cases are studies to demonstrate the ability of our algorithm to find a diverse Pareto frontier.

2 Multiobjective Optimization

Multiobjective Optimization using evolutionary algorithms is a new area of research that has grown considerably in the last 15 years [1]. Evolutionary algorithms seem particularly desirable to solve multiobjective optimization problems because they deal simultaneously with a set of possible solutions or strategies. This paper investigates the problem of using a Tabu Search and Scatter Search approaches to solve a discrete multiobjective problems.

2.1 A Pareto-Based Approach

Formally, we can state a quantitative decision making problem as follows: Decisions have quantitative characters, strategy $s \in E_s$, where E_s denotes a space of solutions or strategies, $S \subseteq E_s$ is a set of admissible strategies. The set S is usually not fixed since a DM can change values of coefficients and/or bounding parameters. We have functions f_1, f_2, \dots, f_r defined over a set of situations $S \times V$, where V is a finite set of values uncertain factors (called elementary events or state of nature). Then for each situation (s,v) , where $s \in S$ and $v \in V$, we have a vector function $f(s, v) = (f_1(s, v), f_2(s, v), \dots, f_r(s, v))$. For deterministic problems the vector function $f(s)$ determines the quality of the strategy s . In our implementation in order to obtain a reference set of solutions that encourage the search toward the Pareto front, an optimality principle is used: " Selection by a number of a dominant criteria" [2]. For all $s, s' \in S$, let $q(s, s')$ the numbers of criteria for which the strategy s' improve the s strategy. $Q_S = \max_{s' \in S} q(s, s'), s \in S$. The function Q_S can be see as a discordance index if strategy s is assumed to be preferred to s' . Then the Kramer choice function is defined as follow: $C^K(S) = \{s' \in S | Q_S(s') = \min_{s \in S} Q_S(s)\}$ This choice function have the following properties [2]: Property 1. For $r = 3$, C^K consists of one element that improves the remaining elements in at least two criteria or $C^K = \Omega^P$, where Ω^P is the Pareto set. Property 2. For $r = 2$, $C^K = \Omega^P$. Property

3. $C^K \subseteq \Omega^P$. The selection function C^K can be utilized as a quality of optimality principle without information about importance of criteria.

2.2 Search by Goals

To implement our process, memory is maintained of selected attributes of recent moves and their associated solutions, where a current move is classified tabu if it reinstates an attribute (or attribute set) that was changed by earlier moves. Aspiration criteria are introduced in tabu search to determine when tabu restriction can be overridden. Aspirations are of two kinds: move aspirations and attribute aspirations. A move aspiration can remove the move's tabu classification. Our implementation uses as move attributes variables that changes their values as result of the move. We represent change by a difference of values $f_i(s') - f_i(s^*) \forall i = 1..r$, s^* , $s' \in S$ where s' was generated from s by a recent move, s is a current solution and s^* is a reference solution. A thresholding aspiration is used to obtain an initial set of solutions as follows: without lost generality, assume that every criteria is maximized. Notationally, let $\Delta f(s') = (\Delta f_1(s'), \dots, \Delta f_r(s'))$ where $\Delta f_i(s') = f_i(s') - f_i(s^*)$, $i \in \{1, \dots, r\}$. Set $U = (u_1, \dots, u_r)$ such that $u_i = \max \Delta f_i(s'_j)$ where s'_j is an element of the set of strategies generated S from s^* to s' , excluding s' , and U is a goal or aspiration threshold point. Let

$$\Delta f_i(s') = \begin{cases} \textit{preference} & \text{if } \Delta f_i(s') > u_i, \\ \textit{indifference} & \text{if } \Delta f_i(s') = u_i, \\ \textit{nonpreference} & \text{if } \Delta f_i(s') < u_i, \end{cases}$$

A goal is satisfied, permitting s' to be accepted and introduced in S if $(\exists \Delta f_i(s') = \textit{preference}) \textit{or} (\forall i \in \{1, \dots, r\} [\Delta f_i(s') = \textit{indifference}])$, in otherwise is rejected. The point s^* is updated if all attributes of s' are preference or indifference attributes, that is, $s^* = s' \textit{if} (\forall i \in \{1, \dots, r\}) [\Delta f_i(s') \geq u_i]$. In this case the goal U is updated too, setting all components $u_i = 0$.

2.3 Scatter Search and Path Relinking Methods

Multiple Criteria Optimization applications are conveniently suited to the use of a population-based approaches. We use scatter search and its generalized form path relinking, that have demonstrated the practical advantages for solving combinatorial problems. An overall view of our procedure is given in this subsection. The procedure starts with the generation of $|S|$ distinct strategies. These strategies are generated by a search by goals approach explained above within a long term taboo search (I phase). The reference set RefSet is constructed with C^K and $P \setminus C^K$ where P is a Pareto set. Scatter Search/Path Relinking Phase has three main loops: 1) a "for loop" that control the maximum number of iterations, 2) a "while loop" that monitors the presence of new elements in the reference set and 3) a "for loop" that controls the examination of all the subsets with at least one new element. Generate subsets of the reference set as a basis for creating combined solutions and for each subset X , use a Solution Combined Method. We propose in our algorithm a Path Relinking Approach, to produce a set $\Omega(X)$. The variable MaxSubset takes the value of the number of subsets with at least one new element. Avoiding the duplicated strategies already generated can be a significant factor in producing an effective overall procedure. The control is limited to these solutions that hold the condition to be Pareto. Our algorithm is provided by a "critical event design" that monitors the current solutions in the RefSet and in the Combined Set. The elements of this critical event are the values of the objectives. The new solutions are put in the set $\Omega(X)$ if they do not belong to the set of critical events. If all subsets have been examined then, the algorithm subtracts the Pareto set $\Omega^P \subseteq \cup \Omega(X) \setminus C$, where C is the critical set. New elements are incorporated into RefSet if $|P \cap \textit{RefSet}| < |P|$ where $P \subseteq \Omega^P \cup \textit{RefSet}$. Then add to RefSet1 the strategies pertaining to $C^K(P)$, RefSet1 contain the best solution so far, and set $\textit{RefSet2} = P \setminus C^K(P)$. Note that our reference set is a dynamic reference set.

3 A Case Study

Multiobjective optimization in a one-machine problem with weighted flowtime is a very popular objective, it is easy to use and very intuitive, also it is robust in the sense that schedules that are optimal for it often produce good schedules for problems with somewhat different objectives. Reducing the amount and frequency by which individual flowtime exceed due dates will often be the primary objective when customers desire reliable time delivery. For this we take the weighted tardiness objective. Then we have two objectives, the total weighted flowtime denoted by Fwf, desired in order to minimize WIP inventories, and the total weighted tardiness to penalize tardy jobs, denoted Twt. Flowtime is defined as $F_j =$ amount of time activity j spends in the system

$$\text{Fwf} = \sum_j w f_j F_j$$

where $F_j = (C_j - r_j)$

Tardiness is defined as $T_j =$ amount of time by which the completion of activity j exceeds its due date
 $T_j = \max\{0, L_j\}$

where $L_j = C_j - d_j$ and $C_j =$ completion time activity j.

The Total Weighted Tardiness is

$$\text{Twt} = \sum_j w t_j T_j$$

Here we consider one-machine problem, J is the set of jobs, the resource is available over the scheduling interval t_s to t_e , n-single operations jobs arrive over the interval, the job j has a processing time p_j , a ready time r_j , a due date d_j , nonpreemptive, objective functions Flow and Total Tardiness times.

3.1 Neighborhoods and Diversification Strategies

In our Tabu Search Phase to generate an initial solutions set we use a general pairwise interchange operation that considers the neighborhood generated by every possible pairwise interchange. In order to explain our diversification scheme we use the following notation [4]. Let $s_j = p$ denote the statement "job j is assigned to position p", $N_o S(s_j = p)$ the number of times the attribute $s_j = p$ resides in the solution of S, where S is a full solution. We use a matrix freqconfg where the entries of this matrix are the number of times that the jobs occupies the positions in a full solution sequence (in our algorithm the full solution sequence is a set of solutions that was admitted in the S), that is, $freqconfig(j, p) = N_o S(s_j = p)$. Let $\max N_o S(s_{j \in J} = p)$ denote "the job that more time has occupied a particular position p". Let $s\text{-to}[p]$ denote a to-attribute of a move that change one job to a position p. To obtain a diversification scheme we classifies a move tabu if it contains $s_j - \text{to}[p] = j : \max N_o S(s_{j \in J} = p) \cap s_j^{best} \neq \emptyset$, where $s_j^{best} \in \text{bestsolution}$. An additive utility function defined by $Utility(s) = \lambda_1 Fwf(s) + \lambda_2 Twt(s)$ is used in order to measure the quality of the solution.

4 Combination Method with Path Relinking

Path Relinking has been suggested as an approach to integrate intensification and diversification strategies[4]. This approach generates new solutions by exploring trajectories that "connect" high-quality solutions - by starting from one of the solutions, called an initiating solution, and generating a path in neighborhood space that leads toward the other solutions, called guiding solutions. In our implementation, the path relinking strategy is used in conjunction with extrapolated path relinking [see [4]] in order to integrate intensification and diversification strategies. The relinking process implemented in our search may be summarized as follows: Let, s be a permutation of numbers of jobs, that is, $s = (s_1, s_2, \dots, s_n)$ then, we define $Insert(s_j, i)$ to consist of deleting s_j from its current position j to be inserted in position i. $s' = (s_1, \dots, s_{i-1}, s_j, s_i, \dots, s_{j-1}, s_{j+1}, \dots, s_n)$ for $i < j$, $s' = (s_1, \dots, s_{j-1}, s_{j+1}, \dots, s_i, s_j, s_{i+1}, \dots, s_n)$ for $i > j$. and $Exchange(s_i, s_j)$ as was described in pairwise interchange, that is: $s' = (s_1, \dots, s_{i-1}, s_j, s_{i+1}, \dots, s_{j-1}, s_i, s_{j+1}, \dots, s_n)$. .. Neighborhood for Path Relinking: two neighborhoods were considered in our path relinking approach, if i is the

position that s_j occupies in the guide solution, then $N_1 = \{s' : Insert(s_j, i), i, j \in \{1, 2, \dots, n\}\}$
 $N_2 = \{s' : Exchange(s_i, s_j), i, j \in \{1, 2, \dots, n\}\}$. In conjunction with these neighborhoods, an oscillation strategy that alternates among these neighborhoods is used to obtain news points, extrapolated relinking is applied and then, the guide points are inverted and this process is repeated.

4.1 Measure of dissimilarity "D"

For more than three objective we can use a variant of Kemen-Snell measure [2] in order to find a diverse elements to complement the subset of high quality solutions . We propose to apply this measure to permutation problems and define the distance between two solutions as follows:

$$d(A, B) = 1/2 \sum_{i,j=1}^n |a_{ij} - b_{ij}|$$

where A,B are matrices defined in the following form

$$A = (a_{ij}) = 1 \Leftrightarrow s_i > s_j$$

$$= 0 \Leftrightarrow s_i = s_j$$

$$= -1 \Leftrightarrow s_j = s_i$$

additionally $a_{ii} = 0$, s_i is a component of s then, we selecting strategies with A_0 , such that $A_0(s) \in \operatorname{argmax}_{s \in RefSet2} \{\sum_{i=1}^{|RefSet2|} d(A(s), A^i(s^i))\}$

5 Conclusion

We conclude that, in our experiment the first TS phase was useful to generate an initial good Pareto frontier. The combined method using path relinking and extrapolated path relinking as an intensification-diversification method seem a good mechanism to generate newly Pareto points. A good approximation Pareto frontier was obtained in a few iteration. A Kramer Choice Function seem a good optimality principle to obtain a first reference set of high quality current Pareto solutions. Further, we will investigate with more than three objective function and the utility of the Kemen-Snell function in the search. In the figure 1 we show the evolution to obtain the Pareto frontier of our experiment, one can see in it, that in the first iteration a few points distributed throughout the frontier were obtained. These solutions were generated with our TS in a very short computational time. In a few runs the algorithm obtained an improved approximation to pareto frontier compared with the initial points. The diversity was maintained over the apparent front.

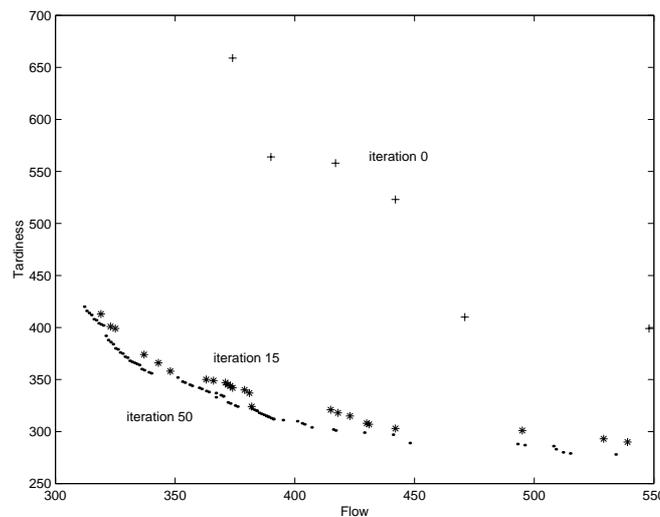


Figure 1: Evolution to Pareto frontier

6 Acknowledgements

Thanks are due to Prof. Fred Glover for useful remarks on a first draft of the manuscript.

References

- [1] Coello Coello, C.A., (1999), A Comprehensive Survey of Evolutionary-Based Multiobjective Optimization Techniques, R.I; LANIA, Xalapa, Veracruz, Mexico.
- [2] Makarov, I. M.; Vinigradskaia, T. M; Rubinski, A. A; Sokolov. V. B; (1982), "Choice Theory and Decision Making", Publisher Nauka, Moscu, pp 228-234, (in Russian).
- [3] , Glover, F., (1998), A Template for Scatter Search and Path Relinking , in Artificial Evolution, Lecture Notes in Computer Science 1363, J.-K,Hao, E.Lutton, E. Ronald, M. Schoenauer and D. Snyers (Eds), Springer, pp.13-54.
- [4] F. Glover and M. Laguna, (1993), Modern Heuristic Techniques for Combinatorial Problems, published in the Halsted Press, John Wiley & Sons, Inc., chapter 3, 70-147.

