INSERTING PROBLEM-SPECIFIC KNOWLEDGE IN MULTIRECOMBINED EVOLUTIONARY ALGORITHMS

PANDOLFI D., DE SAN PEDRO M., VILLAGRA A. , VILANOVA G.
Proyecto UNPA-29/B032[1]
División Tecnología
Unidad Académica Caleta Olivia
Universidad Nacional de La Patagonia Austral
Ruta 3 Acceso Norte s/n
(9011) Caleta Olivia – Santa Cruz - Argentina
e-mail: {dpandolfi,gvilanov,edesanpedro,avillagra}@uaco.unpa.edu.ar
Phone/Fax : +54 0297 4854888

GALLARD R.
Laboratorio de Investigación y Desarrollo en Inteligencia Computacional (LIDIC)[2]
Departamento de Informática
Universidad Nacional de San Luis
Ejército de los Andes 950 - Local 106
(5700) - San Luis -Argentina
e-mail: rgallard@unsl.edu.ar
Phone: +54 2652 420823
Fax : +54 2652 430224

ABSTRACT

In the restricted single-machine common due date problem the goal is to find a schedule for the *n* jobs which jointly minimizes the sum of earliness and tardiness penalties, while for the weighted tardiness problem the goal is to find a schedule that minimizes the tardiness penalties. Both problems, even in theirs simplest formulations, are an NP-Hard optimization problem [4, 14]. This presentation discusses how problem specific knowledge is inserted into the evolutionary algorithm to enhance its performance.

INTRODUCTION

New trends to enhance evolutionary algorithms introduced *multiple-crossovers-on-multiple-parents* (MCMP) [5, 6, 7] a multirecombinative approach allowing multiple crossovers on the selected pool of (more than two) parents. MCMP-SRI [13] is a novel MCMP variant, which considers the inclusion of a stud-breeding individual in a pool of random immigrant parents. Members of this mating pool subsequently undergo multiple crossover operations.

The main objective of this new recombinative method is to find an equilibrium between exploration and exploitation in the search process. An individual of the old population is selected as the stud and subsequently mated with a set of new generated individuals (immigrants). The presence of the stud ensures to retain good features of previous solutions while the immigrants, as continuous source of genetic diversity, avoid premature convergence and make unnecessary to apply mutation.

Recent results are promising and showed its potential by finding good solutions, and in some instances improving the upper bound published by the OR-Library [2].

---

[1] The Research Group is supported by the Universidad Nacional de La Patagonia Austral.

[2] The LIDIC is supported by the Universidad Nacional de San Luis and the ANPCYT (National Agency to Promote Science and Technology).

This presentation shows different multirecombined approaches applied on two problems of single machine scheduling. In these approaches we use two different ways to introduce problem-specific knowledge to the algorithms. Both problems and details of implementation are discussed.

1. THE SINGLE-MACHINE COMMON DUE DATE PROBLEM

In the restricted single-machine common due date problem (GWET), *n* jobs with deterministic processing times must be processed attempting to conform a common due date, thus minimizing penalties imposed to early and tardy jobs.

The problem can be stated as follows: A set of *n* jobs with deterministic processing times p_i and a common due date *d* is given. The jobs have to be processed on one machine. For each of the jobs an individual earliness α_i and tardiness β_i penalties are given. The goal is to find a schedule for the *n* jobs which jointly minimizes the sum of earliness and tardiness penalties.

Even simple in the formulation, this model leads to an optimization problem that is NP-Hard [4], and can be precisely stated as defined in [11]:

$$\min \sum_{i=1}^{n} (\alpha_i E_i + \beta_i T_i) \text{where}$$

$$E_i = \max\{0, d - c_i\}, \; T_i = \max\{0, c_i - d\}$$

$$\text{and } c_i \text{ is the completion time of job } j_i$$

An optimal schedule for GWET problem [1] is V-shaped around the due date. A chromosome contains *n* binary bits, each of which indicates one of two sets (the tardy job set T and the nontardy job set E) a corresponding job belongs to.

2. THE WEIGHTED TARDINESS SCHEDULING PROBLEM

The single-machine total weighted tardiness problem [3] can be stated as follows: n jobs are to be processed without interruption on a single machine that can handle no more than one job at a time. Job j (j = 1,...,n) becomes available for processing at time zero, requires an uninterrupted positive processing time p_j on the machine, has a positive weight w_j, and a due date d_j by which it should ideally be finished. For a given processing order of the jobs, the earliest completion time Cj and the tardiness Tj = max{Cj-dj,0} of job j can readily be computed. The problem is to find a processing order of the jobs with minimum total weighted tardiness,

$$\sum_{j=1}^{n} w_j T_j$$

Even with this simple formulation, this model also leads to an optimization problem that is NP-Hard [3]. For this problem there are heuristics which provide good solutions. We decided to use three of the best ones [14]. These heuristics are described below.

- Rachamadagu and Morton Heuristic (R&M). This heuristic provides a schedule according to the following calculation.

$$\pi_j = (w_j / p_j)[\exp\{-(S_j)^+ / kp_{av}\}]$$

where Sj = [dj – (pj + Ch)] is the slack of job j at time Ch, and Ch is the total processing time of the jobs already scheduled, k is a parameter of the method (usually k =2.0) and p_{av} is the average processing time of jobs competing for top priority. In the R&M heuristic, also called the Apparent Tardiness Cost heuristic, jobs are scheduled one at a time and every time a machine becomes free a ranking index is computed for each remaining job. The job with the highest ranking index is then selected to be processed next.

- The Covert rule. This heuristic provides a schedule according to the following calculation.

$$\pi_j = (w_j / p_j)\{1-(S_j)^+ / kp_j\}^+$$

Under this heuristic the WSPT (Weighted Shortest Processing Time first) rule is modified by an slack factor, and processing times different than the job being considered are not taking into account. The Covert rule works, in the case of a single resource similar to R&M.

- Modified R&M heuristic. Here a logarithmic function of the slack of the job considered and the average processing times of remaining jobs is used,

$$\pi_j = (w_j / p_j)[\ln\{-(S_j)^+ / kp_{av}\}]$$

Note that k is a free parameter and it was experimentally determined for each problem class. The seeds (schedules) provided by each heuristic were determined for each instance before the EA begins running.

3. MULTIRECOMBINED APPROACHES

In our attempt to achieve a better balance between exploration and exploitation we devised MCMP-SRI[13, 10], where a permutation based representation was adopted. Here, the process for creating offspring is performed as follows. From the old population an individual, designated the stud, is selected by means of proportional selection. The number of n2 parents in the mating pool is completed with randomly created individuals (random immigrants). The stud mates every other parent, the couples undergo partial mapped crossover (PMX) and 2*n2 offspring are created. After crossover the best of these 2*n2 offspring is stored in a temporary children pool. The crossover operation is repeated n1 times, until the children pool is completed. Finally, the best offspring created from n2 parents and n1 crossover is inserted in the new population. Children are not exposed to mutation because the random immigrants provide the necessary genetic diversity to the mating pool.

The idea of inserting seeds (good solutions) in evolutionary processes was originally implemented by Reeves [15] as an alternative way to introduce problem-specific knowledge to the algorithm. In his approach Reeves, inserted one seed provided by a non evolutionary heuristic, only once in the initial population, expecting that its genetic material were occasionally exchanged by means of the selection mechanism. In the weighted tardiness problem the seeds are always present in the mating pool for each new generation and consequently, because they mate the stud, their contribution is guaranteed. The latter we called MCMP-SRSI. The three seed immigrants are obtained at the beginning of the process by means of the conventional heuristics described above in section 2.

Attempting to insert problem-specific-knowledge in the *single-machine common due date problem* we decided to follow Lee and Kim [12] proposal giving rise to MCMP-V[9] and MCMP-SRI-V [8]. Regarding representation, a chromosome can suitably be represented by a binary string of n bits. Here, bit indicates one of two sets a corresponding job belongs to (1 represents the tardy job set T, and 0 represents the non-tardy job set E). Afterwards the jobs are sequenced in set E in non-increasing order of pi /αi and jobs in set T in non-decreasing order of pi/βi to form a V-shaped schedule. It is important to note that using the conventional permutation representation the algorithm is compelled to search in a problem space of size n!. In MCMP-V, the process for creating the new population from the old population is performed as follows. Each time an offspring is to be inserted in the new population, n2 parents selected from the old population undergo n1 crossover operations (Uniform Scanning Crossover). Each crossover operation generates a single offspring, which eventually undergoes mutation. The Uniform Scanning Crossover (USX) can be safely used under this representation and is performed as follows: each gene in the child is provided from any of the corresponding alleles in the parents with equal probability.

Finally, MCMP-SRI-V is combination of both previously explained approaches, using binary representation and Uniform Crossover(UX). From the recombination point of view this method is similar to MCMP-SRSI and from the representation point of view this method is similar to MCMP-V.

4. CONCLUSIONS

In the single machine common due date problem inserting problem-specific-knowledge through this representation the problem space size is reduced from n! to 2n and consequently obtained better results than other approaches with representation based in permutation.
In weighted tardiness problem inserting seeds in the mating pool combined with random immigrants also best results are obtained.
Inserting problem specific knowledge in evolutionary algorithms, not only improve the quality of results, but also with less computational effort leads the algorithm to promising search sub-spaces avoiding a large search.

5. ACKNOWLEDGEMENTS

We acknowledge the co-operation of the project group for providing new ideas and constructive criticisms. Also to the Universidad Nacional de San Luis, the Universidad Nacional de La Patagonia Austral, and the ANPCYT from which we receive continuous support.

6. REFERENCES

[1] Baker K. and Scudder G., Sequencing with earliness and tardiness penalties: A review, Operations Research, Vol. 38, pp 22-36, 1990.
[2] Beasley J. E. Scheduling Problems, OR Library, http://mscmga.ms.ic.ac.uk/jeb/orlib.
[3] M. Pinedo, "Scheduling: Theory, Algorithms and System." First edition Prentice Hall, 1995.
[4] Chen T. and Gupta M., Survey of scheduling research involving due date determination decision, European Journal of Operational Research, vol 38, pp. 156-166, 1989.
[5] Eiben A.E., Raué P-E., and Ruttkay Zs., *Genetic algorithms with multi-parent recombination*. In Davidor, H.-P. Schwefel, and R. Männer, editors, Proceedings of the 3rd Conference on Parallel Problem Solving from Nature, number 866 in LNCS, pages 78-87. Springer-Verlag, 1994
[6] Eiben A.E., van Kemenade C.H.M., and Kok J.N., *Orgy in the computer: Multi-parent reproduction in genetic algorithms*. In F. Moran, A. Moreno, J.J. Merelo, and P. Chacon, editors, Proceedings of the 3rd European Conference on Artificial Life, number 929 in LNAI, pages 934-945. Springer-Verlag, 1995.
[7] Eiben A.E. and. Bäck Th., An empirical investigation of multi-parent recombination operators in evolution strategies. Evolutionary Computation, 5(3):347-365, 1997.
[8] Pandolfi D.; Vilanova G.; De San Pedro M.; Villagra A. Gallard R.: Solving the Single-Machine Common Due Date Problem Via Stud and Inmigrants in Evolutionary Computation vol III pp 409-413 World Multiconference on Systemics, Cybernetics and Informatics Orlando 2001.
[9] Pandolfi D.; Vilanova G.; De San Pedro M.; Villagra A. Gallard R.: Evolutionary algorithms to minimize earliness-tardiness penalties from a common due date vol III pp 409-413 World Multiconference on Systemics, Cybernetics and Informatics Orlando 2001
[10] Pandolfi D., De San Pedro M., Villagra A, Vilanova G., Gallard R., Studs and Immigrants in Multirecombined Evolutionary Algorithm to face weighted tardiness Scheduling Problems. vol II pp.1251 Proceedings of the 7th Argentine Congress of Computation Sciences 2001, El Calafate 2001.
[11] Hall N. and Posner M., Earliness Tardiness Scheduling problems: weighted deviation of completion times about a common due date. Operations Research, vol. 39, pp. 836-846, 1991.

[12] Lee C. and Kim S., Parallel Genetic Algorithms for the tardiness job scheduling problem with general penaltiy weights, International Journal of Computers and Industrial Engineering, vol. 28, pp. 231-243, 1995

[13] Pandolfi D., Vilanova G., De San Pedro M., Villagra A. Multirecombining studs and immigrants in evolutionary algorithm to face earliness-tardiness scheduling problems. Proceedings of the International Conference in Soft Computing. University of Paisley, Scotland, U.K., June2001.

[14] Morton T., Pentico D., Heuristic scheduling systems, Wiley series in Engineering and technology management. John Wiley and Sons, INC, 1993.

[15] Reeves C., A genetic algorithm for flow shop sequencing, Computers and Operations Research, vol 22, pp5-13, 1995.