Knowledge-Based Systems 15 (2002) 13–25

Knowledge-Based SYSTEMS

www.elsevier.com/locate/knosys

Enhanced evolutionary algorithms for single and multiobjective optimization in the job shop scheduling problem

S. Esquivel[a,*], S. Ferrero[a], R. Gallard[a], C. Salto[b], H. Alfonso[b], M. Schütz[c]

[a]Laboratorio de Investigación y Desarrollo en Inteligencia Computacional (LIDIC), Universidad Nacional de San Luis, Ejército de los Andes 950, Local 106, 5700 San Luis, Argentina

[b]Proyecto UNLPAM-09/F009, Departamento de Informática, Universidad Nacional de La Pampa, Calle 110 esq. 9, Local 14, 6360, General Pico, La Pampa, Argentina

[c]Center for Applied Systems Analysis (CASA), Informatik Centrum Dortmund (ICD), Joseph-von-Fraunhofer, Str 20, D-44227 Dortmund, Germany

Abstract

Over the past few years, a continually increasing number of research efforts have investigated the application of evolutionary computation techniques for the solution of scheduling problems. Scheduling can pose extremely complex combinatorial optimization problems, which belong to the NP-hard family. Last enhancements on evolutionary algorithms include new multirecombinative approaches. *Multiple Crossovers Per Couple* (MCPC) allows multiple crossovers on the couple selected for mating and *Multiple Crossovers on Multiple Parents* (MCMP) do this but on a set of more than two parents. Techniques for preventing incest also help to avoid premature convergence. Issues on representation and operators influence efficiency and efficacy of the algorithm. The present paper shows how enhanced evolutionary approaches, can solve the Job Shop Scheduling Problem (JSSP) in single and multiobjective optimization.

Keywords: Evolutionary algorithms; Multiplicity of crosoovers and parents; Scheduling; Single and multiobjective optimization

1. Introduction

Due to its complexity [1] and reflecting the industrial relevance of this application domain, a variety of evolutionary schedulers based on genetic algorithms have been reported in the literature in the past [2–11]. In general, the task of scheduling is the allocation of jobs over time when limited resources are available, where a number of objectives should be optimized, and several constraints must be satisfied. A job is determined by a predefined set of operations, and the result of a scheduling algorithm is a schedule that contains the start times and allocation of resources to each operation [12]. Improvements in evolutionary algorithms (EAs) have been recently found by using a multiplicity feature, which allows multiple recombination on a couple of parents or on multiple parents [13–16]. The method was successfully applied to multimodal optimization problems. As a consequence of this approach, it was detected that all individuals of the final population are much more centered on the optimum. This is an important issue when the application requires provision of multiple alternative near-optimal solutions confronting system dynamics as in production planning.

The idea of *incest prevention*, was initially proposed by Eshelman and Shaffer [17] and it showed its benefits to avoid premature convergence. The method avoided mating of pairs showing similarities based on the parents' Hamming distance. Incest prevention was extended in an earlier work by maintaining information about ancestors within the chromosome and modifying the selection for reproduction in order to prevent mating of individuals belonging to the same 'family', for a predefined number of generations. This novel approach was also tested on a set of multimodal functions. Description of experiments and analyses of improved results can be seen in Ref. [18]. Current trends in chromosome representation are problem-dependent and genetic operators are closely related to representation.

In scheduling the quality of a schedule is measured by means of an objective function, which assigns a numerical value to a schedule. In our case, for single objective optimization the completion time of the last job abandoning the system (makespan) is optimized. For multiobjective optimization, an aggregative approach with three objectives

* Corresponding author.

E-mail addresses: esquivel@unsl.edu (S. Esquivel), swf@unsl.edu (S. Ferrero), rgallard@unsl.edu (R. Gallard), saltoc@ing.unlpam.edu.ar (C. Salto), alfonsoh@ing.unlpam.edu.ar (H. Alfonso), schuetz@icd.de (M. Schütz).

PII: S0950-7051(01)00117-4

(makespan, global earliness and weighted completion time) was first considered, then a Pareto optimality problem with two objectives (makespan, mean absolute deviation from a common due date) was studied. Section 2 discusses multi-recombination and incest prevention as means to enhance EAs. Section 3 is related to single objective JSSP optimization. Section 4 discusses aggregative and Pareto multiobjective JSSP optimizations. Section 5 discusses the conclusions.

2. The mechanisms for improvements

In this section multirecombination and incest prevention, as means to improve EAs, are discussed.

2.1. Multirecombination

In EAs the common approach is to operate once on each mating pair after selection. Such procedure is known as the *Single Crossover Per Couple* (SCPC) approach. But in nature, when the mating process is carried out, crossover is applied many times and the consequence is a multiple and variable number of offspring.

Multiple Crossover Per Couple (MCPC) [13] is a novel crossover method. It was applied to optimize classic testing functions and some harder (non-linear, non-separable) functions. For each mating pair MCPC allows a variable number of children. It is possible to choose for insertion in the next generation the best, a randomly selected or all of the generated offspring. In previous works, it was noticed that in some cases MCPC found better results than those provided by SCPC. Also a reduced running time resulted when the number of crossovers per couple increased, and best quality results were obtained allowing between 2 and 4 crossover per couple. Moreover, seeking for exploitation of a greater sample of the problem space, an extended multi-recombination can be applied to a set of more than two parents. In Eiben's *multiparent* (MP) approach [19,20], offspring creation is based on a larger sample from the search space and consequently larger diversity is supplied. This can help to avoid premature convergence. Eiben used, three Scanning Crossover (SX) methods; *Uniform Scanning Crossover* (USX), *Occurrence Based Scanning* (OBSX) and **Fitness** *Based Scanning* (FBSX) generating a single offspring. In USX, each gene in the child is provided from any of the corresponding genes in the parents with equal probability. OBSX selects that gene value which occurs more frequently in a particular position of the parent's chromosomes. FBSX chooses the value to inherit being proportional to the fitness value of the parents. On different function optimization different versions of scanning crossover showed different behavior. Following this idea and to improve performance, *Multiple Crossovers on Multiple Parents* (MCMP) allows multiple recombination of multiple parents under *scanning crossover* (SX), expecting that exploitation and exploration of the problem space be adequately balanced [16].

2.2. Extended incest prevention (EIP)

The extension of the concept of incest is strongly related to the mating members of the same family. To prevent incest EIP allows only recombination of individuals without common ancestors. To build the new population in EIP, individuals are randomly chosen from the previous one according to the conventional *fitness proportional selection*, but they are allowed to crossover if no common ancestors are detected in earlier generations. In this way, exchange of similar genetic material is reduced and population diversity is maintained up to some convenient degree. Consequently, each individual maintains information about their ancestors. Persistent high population diversity has also a detrimental effect: to slow down the search process. To make this point clearer we have to note that by allowing crossover only on some *non-relative* individuals, we modify the effect of the selection mechanism on the population.

In the evolutionary process two important, closely related, issues exist: population diversity and selective pressure enforced by the mechanism. They are the sides of the same coin: exploration of the searching space versus exploitation of information gathered so far. Selection plays an important role here because strong selective pressure can lead to premature convergence and weak selective pressure can make the search ineffective. In this work we addressed the issue by fixing the number of generations to determine the ancestry relationship between individuals.

2.3. Multiplicity and incest prevention (MCMPIP)

The following pseudo-code delineates a procedure to prevent incest between members of the same or consecutive generations (brother–sister and parent–offspring), when a number of $n_2 > 2$ parents are used

procedure MCMPIP (multiple crossovers, multiple parents, incest prevention)
begin
  int mating_pool[number_of_parents]//array to store selected parents
  int children_pool[number_of_cross]//array to store created offspring
  for 1 to popsize
    select indiv-1 C(t)//C(t) is the current population
    mating_pool[1] = indiv-1
    $i = 2$
  while ($i \leq$ number_of_parents)
    repeat
    select indiv-*i* $C(t)$
    until(is_not_relative(mating_pool, indiv-*i*))//control of no common ancestry and
    uniqueness of parents in the mating pool
    matting_pool[*i*] = indiv-*i*
    $i = i + 1$
  end while