Applied Soft Computing xxx (2016) xxx–xxx

Contents lists available at ScienceDirect

Applied Soft Computing

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

Stochastic partially optimized cyclic shift crossover for multi-objective genetic algorithms for the vehicle routing problem with time-windows

Q1 Djamalladine Mahamat Pierre*, Nordin Zakaria

Q2 *High Performance Computing Center, Universiti Teknologi PETRONAS, 32610 Tronoh, Perak, Malaysia*

ARTICLE INFO

Article history:
Received 15 March 2016
Received in revised form
22 September 2016
Accepted 23 September 2016
Available online xxx

Keywords:
Multi-objective vehicle routing problems
Multi-objective genetic algorithm
Crossover

ABSTRACT

This paper presents a stochastic partially optimized cyclic shift crossover operator for the optimization of the multi-objective vehicle routing problem with time windows using genetic algorithms. The aim of the paper is to show how the combination of simple stochastic rules and sequential appendage policies addresses a common limitation of the traditional genetic algorithm when optimizing complex combinatorial problems. The limitation, in question, is the inability of the traditional genetic algorithm to perform local optimization. A series of tests based on the Solomon benchmark instances show the level of competitiveness of the newly introduced crossover operator.

© 2016 Elsevier B.V. All rights reserved.

UNCORRECTED PROOF

1. Introduction

Q5 The vehicle routing problem with time windows (VRPTW) [1] is a combinatorial optimization problem which deals with time-constrained service provision. The problem consists of a set of trucks which leave a centralized depot, and service a set of geographically dispersed customers. Each customer has a demand of commodities which ought to be satisfied by the servicing truck within predefined time-windows. The problem is also subject to the restriction that each customer must be visited exactly once and that the cumulative demands of the serviced customers must not exceed the capacity of the servicing truck. The objective of the problem is to minimize the total travel cost (distance, time, number of trucks, etc.).

As an extension of the capacitated vehicle routing problem (CVRP), and a generalization of the travelling salesman problem with time windows (TSPTW), the applications of VRPTW range from modeling real-life logistical problems which involve time and capacity restrictions [2] to schedule sequential jobs when there exist dependencies between the jobs on each machine [3]. The multi-objective VRPTW (MOVRPTW) arise from the fact that many real life problems require the simultaneous optimization of two or more objectives [4]. The problem is of particular interest among researchers because of its classification as an NP-complete problem [5]. Thus, the problem is intractable; which forces researchers to resort to specialized heuristics and meta-heuristics to solve practical size problem instances.

Early attempts to solve VRPTW instances rely on heuristics based on deterministic methods. Such heuristics include the time-oriented nearest neighbor heuristics and the insertion heuristics presented in [1]. Aside from their inability to support multi-objective optimization (MO), deterministic methods are inefficient with large-scale problem instances. Multi-objective genetic algorithms (MOGAs) [6,7] are prime candidates for solving MOVRPTWs for two main reasons. Firstly, their ability to maintain a population of candidate solutions [7] makes them suitable to approximate the pareto-optimal set of the multi-objective problems. Secondly, as meta-heuristics, they are proven to provide near optimal solutions to complex optimization problems in acceptable time. Nevertheless, genetic algorithms, and by extensions MOGAs, are liable to the lack of accuracy and efficiency when optimizing complex problems [8]. In order to surmount those weaknesses, the traditional GA operators are modified to incorporate local optimization techniques. Among those operators is the crossover operator. The insertion based-crossover operators such as the Best Cost Route Crossover (BCRC) [2,7], or the construction algorithm presented in [9] have been successfully used in MOGAs for the optimization of VRPTWs. But, one must note that regardless of the quality of the solutions found, those crossover operators employ exhaustive techniques to find the best cost node positions.

The partially optimized cyclic-shift crossover (POCSX) incorporates the hill-climbing mechanism by sequentially appending the

Q3 * Corresponding author.
E-mail address: djamal2810@gmail.com (D.M. Pierre).

http://dx.doi.org/10.1016/j.asoc.2016.09.039
1568-4946/© 2016 Elsevier B.V. All rights reserved.

Please cite this article in press as: D.M. Pierre, N. Zakaria, Stochastic partially optimized cyclic shift crossover for multi-objective genetic algorithms for the vehicle routing problem with time-windows, Appl. Soft Comput. J. (2016), http://dx.doi.org/10.1016/j.asoc.2016.09.039

Nomenclature

Sets

C	set of customer nodes
N	set of all nodes
V	set of vehicles

Variables

b_{ik}	time at which the service begins at i
d_{ij}	distance between nodes i and j
d_i	demand of customer i
e_i	earliest allowable service time for customer i
i, j	indices of nodes
K	a very big number
k	index of vehicle
l_i	latest allowable service time for customer i
m_i	service completion time at customer i
q	uniform capacity of all the vehicles
s_{ik}	time needed for vehicle k to service customer i
t_{ij}	travel time between nodes i and j
w_{ik}	wait time of vehicle k at node i
x_{ijk}	variable set to 1 if the vehicle k travelled from node i to node j

Other symbols

α_{ij}	cost of edge (i, j)
p_av_1	sequence of available customer nodes in parent 1
$p\_u\,nas$	a boolean array which keeps track of nodes not yet appended
p_1, p_2	parent chromosomes
$p_1[i]$	ith gene of parent 1

low-cost gene (nodes) of the two mating parents at each gene position of the child chromosome. Though the crossover has low time complexity (compared to the insertion based crossovers), its application to VRPTW have yielded disappointing solutions in [10]. In this paper, we introduce the stochastic POCSX (SPOCXS) which is the improvement of the POCSX. The contributions in this paper include:

- A new child construction algorithm based on the sequential appendage of genes.
- A mathematical model which defines the viability of the potential node to ensure that the resulting child encodes a feasible solution.
- A mathematical model for the appendage cost to ensure that the resulting child is an improvement of the parent chromosomes.
- A stochastic appendage rule to overcome potential traps to local minima.

In the following lines, we cover the literature review in Section 2. Section 3 gives a formal definition of the problem, and Section 4 provides an in-deft description of our MOGA, including the newly introduced SPOCSX. Section 5 describes the experimental set-up. The results are discussed in Section 6. Finally, the conclusion is given in Section 6.

2. Background

Over the years, many attempts have been made to efficiently solve instances of the VRPTW. Early attempts to solve the problems include problem solving strategies which account for a single objective (mainly minimizing the distance cost). Solomon's early works on time-oriented nearest neighbor heuristics and on the insertion-based heuristics [1] are but a few examples of approaches used to solve single objective VRPTW instances. The time-oriented nearest neighbor heuristics sequentially constructs a solution based on the lowest cost of neighboring nodes. The insertion-based heuristics, on the other hand, rely on two cost functions to determine the best insertion positions and the best node to be inserted among a series of unrouted nodes. Solomon's works have proven that the insertion heuristics outperforms the appendage-based heuristics in most cases. Solomon's findings inspired Potvin and Rousseou to further improve the insertion heuristics which resulted on the conception of the parallel construction algorithm described in [11]. Although the insertion-based heuristics have proven effective in finding optimal results, they rely on exhaustive search to identify the best insertion positions.

Particle Swarm Optimization (PSO) is a population-based metaheuristic which arises from the simulation of the social behaviour of swarms (flock of birds, fish schooling, etc.) [12]. PSO is an iterative method which bears a lot of similarities with other evolutionary algorithms (namely genetic algorithm). PSO maintains a population of independent entities or particles which go through changes to achieve a common goal: reaching the optimal or near optimal point of a given objective function. Each particle is represented by an n-dimensional position vector and an n-dimensional velocity vector. In addition, each particle has the ability to remember its last recorded best position deemed personal best. Moreover, the local best position of selected neighbouring particles is also known to the particle. The best position of all the local best is archived as the global best. Over the course of the iterations, the particle's position vector and velocity vector are updated in such a way that the particle's next position coincide with the best or near best positions with the minimum possible fluctuations over the search space. Originally conceived for continuous optimization, PSO has seen its scope broaden and expanded to various integer programming problems. PSO has been successfully applied to solve small VRPTW instances in [13,14]. It is noteworthy that PSO, as a population-based search method, can be extended to support multi-objective optimization. In [15], Coello and Lechuga laid down the foundations of approximating the pareto-optimal front with PSO by keeping track of all the non-dominated global best in an external archive. The method has been applied to multi-objective VRPTW in [16]. However, the investigations in [17] found that such approach can be detrimental to the performance of PSO as the complexity to update the archive of the global best can, at each iteration, rise up to $\mathcal{O}(KN^2)$ with K the number of objectives and N the number of particles. The past few years have seen the multi-objective PSO (MOPSO) reach a high level of maturity in the multi-objective optimization of integer programs. The work by Torabi et al. on multi-objective optimization of unrelated parallel machines scheduling problem [18] is a worthy example of the application of MPSO to a multi-objective integer problem. One of the challenges to approach VRTPW (an integer program) with MOPSO (a continuous method) is the representation of solutions which, in most cases, are left to the researcher's interpretation of the problem resulting in a wide variety of indirect representations such as the $n+2\times m$-dimensional representation of the particle position in [13,14], the n-dimensional particle position shown in [16], or the $2\times n$-dimensional representation in [19]. Along with each representation, a decoding scheme is deployed prior to the evaluation process.

Genetic algorithm (GA) [20] is a global optimization technique which comprises of interrelated operations which include initialization, evaluation, selection, crossover (mating), and/or the mutation, and truncation (or replacement strategy) [21]. The combination of the evaluation, selection, crossover and/or mutation is subject to an iterative process (evolution) until a stop condition is met. Like PSO, GA maintains a population of potential solutions in the process of finding the optimal or pseudo optimal solutions. Unlike PSO, during the optimization process, GA's search procedure

UNCORRECTED PROOF

Please cite this article in press as: D.M. Pierre, N. Zakaria, Stochastic partially optimized cyclic shift crossover for multi-objective genetic algorithms for the vehicle routing problem with time-windows, Appl. Soft Comput. J. (2016), http://dx.doi.org/10.1016/j.asoc.2016.09.039

Download English Version:

https://daneshyari.com/en/article/4963507

Download Persian Version:

https://daneshyari.com/article/4963507

Daneshyari.com