Expert Systems with Applications 39 (2012) 328–334

Contents lists available at ScienceDirect

Expert Systems with Applications

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

Pickup and delivery problem using metaheuristics techniques

Craig D'Souza [a], S.N. Omkar [b,*], J. Senthilnath [b]

[a] Department of Computer Engineering, NITK, Surathkal, India
[b] Department of Aerospace Engineering, Indian Institute of Science, Bangalore, India

ARTICLE INFO

Keywords:
Dial-a-ride problem
Metaheuristic
Simulated Annealing
Genetic Algorithm
Particle Swarm Optimization
Artificial Immune System

ABSTRACT

Dial-a-ride problem (DARP) is an optimization problem which deals with the minimization of the cost of the provided service where the customers are provided a door-to-door service based on their requests. This optimization model presented in earlier studies, is considered in this study. Due to the non-linear nature of the objective function the traditional optimization methods are plagued with the problem of converging to a local minima. To overcome this pitfall we use metaheuristics namely Simulated Annealing (SA), Particle Swarm Optimization (PSO), Genetic Algorithm (GA) and Artificial Immune System (AIS). From the results obtained, we conclude that Artificial Immune System method effectively tackles this optimization problem by providing us with optimal solutions.

1. Introduction

A lot of work has been done in the pickup and delivery problem (PDP) area which initially started in the 1980s (Psarafis, 1980). The work done in the pickup and delivery problem area can be found in some of the comprehensive literature surveys done over the years (Mitrovi'c-Mini'c, 1998; Parragh, Doerner, & Hartl, 2008a, Parragh, Doerner, & Hartl, 2008b). One of the main problems we face is scalability. When the number of requests are small it is very easy to analyse the problem. When the number of requests increase, the search space increases. The dial-a-ride problem (DARP) is NP-hard as proven by Baugh, Kakivaya, and Stone (1998). The pick-up and delivery problem is considered as an optimization problem where the total cost of the provided service is minimized. A lot of literature is available where optimization models are created to solve this problem. Initially dynamic programming was used to solve the PDP problem but did not yield good results due to the presence of local minima's (Psarafis, 1980). To overcome this drawback people employed metaheuristic approaches to solve this optimization problem (Baugh et al., 1998).

One of the first works published in this area using dynamic programming by Psarafis (1980, 1983), which could only solve problem instances where less than 10 customers are involved. After a few years another method was published by Dumas, Desrosiers, and Soumis (1991) which uses a column generation scheme with a constrained shortest path as a sub problem. The above methods could solve only small problem instances. Therefore researchers adopted metaheuristics to tackle large problems where optimal solutions could be achieved. Nanry and Barnes (2000) and Emmanouil, Christos, and Chris (2009) employed a metaheuristic approach to solve the general pickup and delivery problem (GPDP) using a reactive tabu search method. Similarly Cordeau and Laporte (2003) also used classical tabu search to solve the static multi-vehicle DARP. Potter and Bossomaier (1995) used Genetic Algorithms to solve the vehicle routing problem (VRP). Jih and Hsu (1999) used a hybrid method where they combined dynamic programming and Genetic Algorithms to solve the single-vehicle PDP with time windows.

The various challenges we face with respect to travel in Bangalore are the availability of regular, quick and cost effective transport. The cost effectiveness of a provided service is directly proportional to the distance travelled by the vehicle providing that service. Therefore in the service we provide, we not only reduce the distance travelled by the bus/van but also the distance travelled by each and every passenger. Here the bus/van service provided is of the Dial-a-ride type. The passengers decide where they want to be picked up and where they want to be dropped. This service reduces the distance travelled by every passenger as well as the total distance travelled by the bus/van.

In this paper, we consider the DARP optimization problem presented in earlier studies. Since the problem is NP-hard we use the metaheuristic techniques such as Simulated Annealing (SA), Genetic Algorithms (GA), Particle Swarm Optimization (PSO) and Artificial Immune System (AIS). Here we define $dist_{max}$ which is the maximum distance the bus is allowed to traverse. In Keiichi Uchimura, Takahashi, and Saitoh (2002) a Genetic Algorithm + 2-opt method is proposed. This is equivalent to using a Genetic Algorithm with both crossover and mutation operators and on giving them adaptive probabilities of occurring leads to better solutions. We also show that the AIS is good in giving optimal

* Corresponding author. Tel.: +91 080 229 32873; fax: +91 080 236 00134.
E-mail address: omkar@aero.iisc.ernet.in (S.N. Omkar).

0957-4174/$ - see front matter Crown Copyright © 2011 Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2011.07.022

solutions which are better than the results got using Genetic Algorithm. According to the authors knowledge there is no literature present where AIS (Clonal Selection Algorithm) has been used to solve this problem. On finding optimal solutions for the problem at hand we compare the results and find out which method successfully tackles the problem.

The paper is divided into the following sections. Section 2 contains the problem formulation and problem definition. Section 3 contains the methodologies used. Sections 4 and 5 contains the results and conclusion respectively.

2. Problem formulation

Pickup and delivery problems with timing constraints are referred to as pickup and delivery problems with time windows (PDPTW). Here we find optimal routes for the given transportation requests keeping in mind the capacity, time window and precedence constraints. The vehicles start from a depot. Each request has an origin and a destination. The vehicle which is carrying out a certain request has to first go to the origin and then destination. The various constraints are priority, time and capacity. Since in this problem we only deal with single vehicle pickup and delivery we ignore the capacity and time constraints. The objective here is to not only minimize the total distance travelled by the bus but also minimize the distance travelled by every passenger.

In our objective function we first specify the maximum distance ($dist_{max}$) the bus is allowed to traverse. Any distance above this value is taken as it is by the objective function as S. If it is lesser than the maximum value for any particular route the ratio R is calculated, where R is the maximum ratio of distance between pickup point and drop point for passenger i for that particular route to shortest distance between the two points. Therefore the objective function not only minimizes the total distance travelled by the bus but also the individual distance travelled by each passenger.

Mathematically the objective function f is defined as follows:

$$f = \begin{cases} S, & \text{if } S \geqslant \text{dist}_{max} \\ R, & \text{if } S < \text{dist}_{max} \end{cases} \tag{1}$$

$$R = max(P_iD_i/)\overline{P_lD_l} \tag{2}$$

where,

S = total distance travelled by the bus for a particular route
$dist_{max}$ = maximum distance the bus is allowed to travel
P_iD_i = distance between pickup and drop point for passenger i for a given route
$\overline{P_lD_l}$ = shortest distance between pickup and drop point for passenger i

We shall now develop a graph and address the above problem by defining the constraints in a mathematical framework.

We first assume one vehicle is carrying out all the pickup and delivery requests. Consider n pickup points and n drop points for n passengers. The pickup and drop points can be taken as the vertices of an undirected graph $G(V, E)$. Each edge has a particular weight equivalent to the distance between the two vertices joining the edge. We denote c_{ij} as the distance connecting the two vertices i and j. If i is a pickup point then $n + i$ is the drop point for that particular passenger. Then $P_i = \{1,2,\ldots,n\}$, and $D_i = \{n + 1,n + 2,\ldots,2n\}$. The bus starts from the depot labelled as 0 and ends at $2n + 1$. In total the graph $G(V,E)$ has $2n + 2$ vertices and is strongly connected in nature.

To evaluate total distance travelled by the bus we use the Boolean variable x_{ij} which is either true or false depending whether the bus travels through the points i and j. Then S is evaluated as follows:

$$S = \sum_{i=0}^{2n+1} \sum_{j=0}^{2n+1} x_{ij}c_{ij} \tag{3}$$

where,

x_{ij} is the Boolean variable
c_{ij} is the distance between the ith and jth point.

We then check whether priority constraints are satisfied by a particular route. For a particular route we should see that the pickup point for a passenger is visited before the drop point. Consider a route of $2n + 1$ points. Then the priority constraint can be defined as follows:

$$pos(P_i) < pos(D_i) \text{ for passenger } i1 \leqslant i \leqslant n \tag{4}$$

where,
$pos(x)$ gives the position of the point x in the route.

We define fitness of a particular path as f^{-1}. The various steps in evaluating the fitness of a route can be summarised as follows:

1. Choose dist_{max}.
2. Select path to traverse.
3. Check whether path satisfies priority constraints.
4. Calculate f^{-1} for that path.

Evaluate fitness of the path based on the defined objective function. The aim is to find a path having maximum fitness.

3. Methodology

We use four metaheuristic methods to solve the above problem namely Simulated Annealing, Genetic Algorithms, Particle Swarm Optimization and Artificial Immune System. Out of the four metaheuristic methods three are population based algorithms.

3.1. Simulated Annealing (SA)

Simulated Annealing is a metaheuristic used to find maximum or minimum in a given search space (Baugh et al., 1998; Rutenbar, 1989). The cooling function is dependent on the number of iterations. The cooling starts at an initial random state. The initial temperature is taken as T_1K and the final temperature is T_2K where $T_2 < T_1$. The cooling may terminate for two reasons. The first reason is when convergence is achieved and the second reason is when a fixed number of iterations are completed. The main aim of simulated annealing is to go from an initial state having higher energy to a final state having lower energy. The change of state depends on the probability P and T. The probability P depends on the energy of the current state as well as the energy of the neighbouring state.

The initial state is taken as a random path which satisfies the various constraints. The neighbouring function N contains a swap operator which swaps any two points of the current evaluated path. The probability T determines whether the algorithm is greedy or not. As the temperature decreases the probability of choosing a state with lower energy increases i.e. the algorithm tends to be greedy. If the algorithm behaves in a non greedy manner then depending on the probability P the current state changes to the new state or, the old state is retained. P is dependent on the difference between the energy of the current state and the neighbouring state. If there is a large gap between the two states the chance of changing state decreases where as if there is a small difference between the energy of the two states there is a high probability that