Proceedings of KFIS Spring Conference 2007 Vol. 17, No. 1.

Evolutionary Approach for Traveling Salesperson Problem with Precedence Constraints

Chiung Moon[1] YoungSu Yun[2]

[1] Department of Information and Industrial Engineering, Yonsei University
(*e-mail*: cumoon@yonsei.ac.kr)
[2] Division of Business Administration, Chosun University (e-*mail*: ysyun@chosun.ac.kr)

Abstract

In this paper we suggest an efficient evolutionary approach based on topological sort techniques for precedence constrained TSPs. The determination of optimal sequence has much to offer to downstream project management and opens up new opportunities for supply chains and logistics. Experimental results show that the suggested approach is a good alternative to locate optimal solution for complicated precedence constrained sequencing as in optimization method for instance.

Keywords: Precedence constrained TSP, mixed integer programming, evolutionary algorithm, topological sort.

1. INTRODUCTION

Traveling salesperson problem with precedence constraints (TSPPC) can be applied to a number of economic problems such as project management, logistics, routing, assembly flow, scheduling, and networking. Finding out an optimal sequence for this problem would yield great economic effect because inefficient sequence affects all downstream stages of manufacturing, logistics, and networking. The TSPPC is about finding the shortest travel time through all nodes, visiting each node only once on a directed graph $G(C, A)$, where $C = \{1, 2, \cdots, I\}$ is a set of nodes, and $A = \{\langle i, j\rangle, i, j \;\; C$ and $i \neq j\}$ is a set of precedence relations. Precedence relation $\langle i, j\rangle$ determines the sequence of traveling between given pair of nodes.

A number of industrial models are built around TSPPC. Chen [1], He and Kusick [2], Savelsbergh and Sol [3], Renaud et al. [4] proposed heuristic models to solve TSPPC based problems.

This paper aims at formulating a mathematical model as well as at developing an efficient constrained evolutionary approach (CEA) by adapting evolutionary algorithm and topological sort (TS) techniques to the TSPPC. TS is used to generate a set of feasible solutions to the model within a reasonable computing time. The objective of the model is to find an optimal solution having minimal total travel time among the feasible sequences.

2. Model Formulation

TSPPC requires that there is no cycle in precedence relations. It is clear that a linear sequence is not possible if the graph has a loop, since for two nodes i and j on the cycle there is a sequence from i to j and from j to i. A feasible sequence passes through each node in the given directed graph only once and cannot visit any two nodes at the same time.

Thus, the TSPPC is a type of directed acyclic graph. Since a directed graph may have several feasible sequences, it's necessary to decide an optimal sequence among the feasible ones. The optimal solution, which would minimize the total processing time, can be decided by comparing all sequences. In order to avoid computational difficulty and ambiguity, the proposed model is formulated by modifying the standard integer programming TSP [2, 8, 9]. The discussion below uses the following notation is used to describe the TSPPC model:

i, j	node index, $i, j = 1, \cdots, I$, where I is the number of nodes.
t_{ij}	travel time from node i to j.
A_i	arrival time at node i.
P	set of nodes (i, j), where node i is visited before node j.
R	set of nodes (i, j), where nodes i and j can be visited in any feasible sequence.
L	an arbitrary large positive number.

The variable is introduced to adapt the TSPPC model as follows:

$$y_{ij} = \begin{cases} 1, & \text{if node } i \text{ is visited before } j, \\ 0, & \text{otherwise.} \end{cases}$$

Let E be the total processing time for visiting all nodes defined by Equation (1).

$$E = \max_{\forall i}\{A_i\} \qquad (1)$$

A mixed-integer programming model of the TSPPC for minimizing the total processing time is presented below.

$$\begin{array}{llr} & \textbf{Minimize } E & \\ & \textbf{subject to} & \\ A_i - E \le 0 & \forall\, i, & (2) \\ A_j - A_i \ge t_{ij} & \forall\, (i, j) \in P \text{ and } i \ne j, & (3) \\ A_i - A_j + L\, y_{ij} \ge t_{ij}\, y_{ij} + t_{ji}\, y_{ji} & \forall\, (i, j) \in R \text{ and } i \ne j, & (4) \\ y_{ij} + y_{ji} = 1 & \forall\, (i, j) \in R \text{ and } i \ne j, & (5) \\ y_{ij} = 0, 1 & \forall\, (i, j) \in R \text{ and } i \ne j. & (6) \end{array}$$

Constraint (2) means that the arrival time at each node is not greater than the total travel time for visiting all nodes. Constraints (3) and (4) ensure that two nodes cannot be visited at the same time. Constraint (5) represents that any two nodes are to be visited in only one sequence. This constraint represents the technical restriction on visiting sequence as the existence of a partial sequencing among the nodes based on precedence relations. Also, this constraint represents that there is no cycles in precedence relations. Constraint (6) restricts variables to integer.

3. Evolutionary Approach

3.1 TS-based approach

The TS procedure used to produce a travel sequence on a directed graph is described in Figure 1.

```
Procedure:
  Input: A precedence matrix and its directed
    graph. Let I be the number of nodes to
    process
  For i = 1 to I do

  Begin
    If every node has a predecessor
    Then the directed graph is not feasible, stop;
    Else randomly select a node i which has no
      predecessors;
    Print i
      Remove i and its arcs from the graph
  End;
```

Figure 1. TS procedure

The TS is a role of generating a set of feasible sequences on the directed graph.

3.2 Representation and initialization

In order to generate a feasible sequence, the representation scheme mustbe able to account for the constraints of the given TSPPC. Therefore, the development of the representation scheme is an important issue in designing evolutionary approach. The most common representation scheme for directed acyclic graph is priority-based representation [5, 6, 10, 11]. This scheme has been widely used to represent feasible solutions. It has, however, a number of serious limitations such as idling. Let's consider the directed graph in Figure 3 and the corresponding node priorities be (4, 1, 5, 2, 3). The value of each gene indicates the priority for candidate selection where nodes with no incoming arcs exist. The value of a gene is generated at random within (1, I)exclusively. The TS producing process for travel sequence is shown in Figure 4.

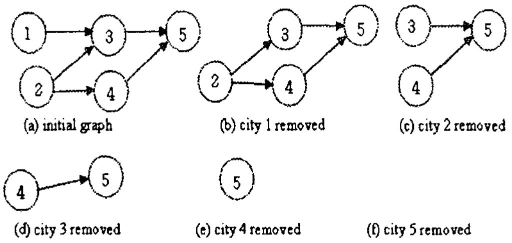

Figure 4. Producing travel sequence

In this example, nodes 1 and 2 have no predecessors and their corresponding priority values are 4 and 1. Node 1 is selected first for travel sequence because its priority is higher. After selecting node 1, we can remove it along with all outgoing arcs. Now node 2 has no predecessor and it is selected next for visiting. Accordingly, the travel sequence can be obtained for all nodes, 1-2-3-4-5.

By the way, even though node 2 has a higher priority among 1, 2 or 3 through evolutionary operations, the travel sequence is not changed. Likewise, the same is for nodes 3 and 4. These cases are called idling and restrict search efficiency of the evolutionary process. Because of idling, priority-based strategy cannot guarantee optimal solution within reasonable generations.

In order to cope with the limitations, an efficient and simple representation scheme using TS procedure is proposed. A chromosome is defined as a string of integer numbers, representing a sequence to be followed by the salesman. The value of each gene indicates the node number to be visited sequentially, and is generated at random within (1, I) exclusively. If the directed graph has I nodes, the corresponding chromosome has I genes. We call this a sequence representation. In the example in Figure 3 the string expressed by TS procedure is as follows:

< 2 4 1 3 5 >

Now, we can obtain a feasible sequence from the chromosome: 2-4-1-3-5.

The next step is to initialize solution that involves generating initial solutions for the problem. The initialization of the population using chromosomes can be made by the TS procedure as many as the population size, pop_size. Unlike other search methods, evolutionary algorithm not only performs with one point in the solution space, but also with multiple directional search with a group of points simultaneously. The number of multiple points was already determined by the pop_size.

3.3 Fitness score and selection

The fitness value is computed for each chromosome and the objective is to find a chromosome with the shortest travel time. We consider the objective function of mixed-integer model which is to minimize total travel time. Let $f(k)$ be the fitness function of k-th chromosome, then $f(k)$ can be expressed as follows:

$$f(k) = \min\{\max_{\forall i}\{A_i\}\} \tag{7}$$

Selection strategy is about choosing chromosomes from the population space. It can create a new population for the next generation based on either parent and offspring or their part. Roulette wheel strategy [10] was suggested for selection procedure.

3.4 Evolutionary operators

A new crossover procedure is suggested as follows:

Step 0. Use roulette wheel strategy to select a chromosome from the current population.

Step 1. Select in the chromosome two positions b and c at random.

Step 2. Generate a random number d from (0, 2). If d = 0 then select the left subsequence of the first cut point, if d = 1 then select the center cut, if d = 2 then select the right subsequence of the second cut point for crossover.

Step 3. Use TS procedure to produce an alternative subsequence for the segment selected in Step 2.

Step 4. Replace the subsequence with a new one.

Mutation operator is as follows:

Step 0. Use roulette wheel strategy to select a chromosome from the current population.

Step 1. Select at random a gene in the chromosome.

Step 2. Use TS procedure to insert the gene in a position of its immediate predecessor or immediate successor.

4. Experiments

In this section, we discuss five TSPPCs, which were tested to prove the efficiency of the proposed CEA. We make a comparison between topological sort-based encoding used in the CEA and priority-based encoding [5] in conventional GA procedure.

For various TSPPCs with 25, 35, 45, and 70 nodes were considered. In the CEA, the proposed topological sort-based encoding was used so we could compare performance between topological sort-based encoding and priority-based encoding used in conventional GA approaches and determine that the CEA performance is better. Table 1 shows performance results for comparison of topological sort-based encoding and priority-based encoding for using in CEA.

Table 1. Computing result

	Priority-based encoding method			Proposed method		
No. of nodes	CPU time (Sec.)	Optimal solution	NGS	CPU time (Sec.)	Optimal solution	NGS
25	6.20	134	0	6.15	134	0
35	8.77	185	1	8.20	185	0
45	11.00	219	2	9.68	219	0
70	34.57	372	7	30.70	372	0

5. CONCLUSION

This paper has presented an efficient CEA based on combination of evolutionary algorithm and TS technique for solving various types of TSPPCs. The suggested CEA produced an optimal solution with minimal total travel time among all feasible sequences.

References

[1] C. Chen, 1990, " AND/OR precedence constraint traveling salesman problem and its application to assembly schedule generation," Proceedings of 1990 *IEEE International Conference on Systems, Man and Cybernetics*, 1990, 560 - 562.

[2] W. He and A. Kusiak, 1992, "Scheduling manufacturing systems," *Computers in Industry*, 1992, Vol. 20, 163 - 175.

[3] M. Savelsbergh and M. Sol, "The general pickup and delivery problem," *Transportation Science*, 1995, Vol. 29, 17-29.

[4] J. Renaud, F. F. Boctor and J. Ouenniche, "A heuristic for the pickup and delivery traveling salesman problem," *Computers & Operations Research*, 2000, Vol. 27, 905-916.

[5] C. Moon, J. Kim, G. Choi and Y. Seo, "An efficient genetic algorithm for the traveling salesman problem with precedence constraints," *European J. of Operational Research*, 2002, Vol. 140, No. 3, 606 - 617.

[6] C. Moon and Y. Seo, "Advanced planning for minimizing makespan with load balancing in multi-plants chain, " *International J. of Production Research*, 2005, Vol. 43, No. 20, 4381 - 4396.

[7] E. Duman and I. Or, "Precedence constrained TSP arising printed circuit board assembly," *International J. of Production Research*, 2004, Vol. 42, No. 1, 67 - 78.

[8] K. Baker, *Introduction to Sequencing and Scheduling,* John Wiley & Sons, 1974.

[9] M. Pinedo and X. Chao, *Operations Scheduling*, McGraw-Hill, 1999.

[10] M. Gen and R. Cheng, *Genetic Algorithms and Engineering Optimization*, John-Wiley & Sons, 2000.

[11] M. Gen, F. Altiparmak and L. Lin, "A genetic algorithm for two-stage transportation problem using priority-based encoding," *OR Spectrum*, 2006, Vol. 28, 337-354.

[12] K. Shahookar and P. Mazumder, "A genetic approach to standard cell placement using meta-genetic parameter optimization," *IEEE Trans. On Computer-Aided Design*, 1990, Vol. 9, No. 5, 500-511.