A Knowledge-Based Model Representation and On-Line Solution Method for Dynamic Vehicle Routing Problem

Lijun Sun, Xiangpei Hu, Zheng Wang, and Minfang Huang

School of Management, Dalian University of Technology,
Dalian 116023, P. R. China
{Lijun_Sun,yuqier2008}@163.com

Abstract. We propose a knowledge-based model representation and on-line solution method for dynamic vehicle routing problem (DVRP) in the paper in order to realize on-line modeling and solution process of the problem. This knowledge-based model representation is composed of six components—*B* (Basic Data collector), *R* (Restrictions), *I* (Initial state generator), *S* (State operator), *G* (Goal state), and *C* (Controller). We term the representation *BRISGC six-component model representation*. Based on the representation, an on-line solution approach to DVRP is presented. And a real-world DVRP in e-Commerce is solved by the representation and solution method in case study. The result proves that the proposed approach is effective for online and real-time vehicle routing.

Keywords: model representation, dynamic vehicle routing problem (DVRP), knowledge, algorithm.

1 Introduction

Vehicle routing problem (VRP) is a NP-hard and key problem in logistics distribution systems. It includes vehicle scheduling and routing. A distribution center that operates routing and scheduling scientifically could promote vehicle's utility, decrease travel distance and travel times, accordingly responds to the clients' demands promptly and increases customer satisfaction. VRP in practice frequently changes because of emergencies and environments, which needs a quick response mechanism to adapt routing process to new states. In order to adapt VRP to a dynamic environment, the model for it should own knowledge-based reasoning ability. Moreover, the final goal of modeling is to facilitate solution process. Therefore, the model should be represented based on knowledge and solution process's characteristics of VRP. So after reviewing literatures from two aspects: *categories of Dynamic Vehicle Routing Problem (DVRP) and their solution methods, knowledge representations for models*, we present an innovative method that combines reasoning knowledge with heuristic solution knowledge to represent a specified DVRP, which contains multiple dynamic changes consisting of three discrete events—new order arrival, old order withdrawnness, and demand change.

The definition of VRP and algorithms for this kind of problems can be found in reference [1]. Here, we unnecessarily waste pages to introduce them. We'll focus our

discussion on dynamic VRP, which is defined by Larsen [2] as: not all information relevant to the planning of the routes is known by the planner when the routing process begins, and information can change after the initial routes have been constructed. In recent years, with the development of web technology and information technology, dynamic VRP has drawn more attention.

Haghani and Jung have reviewed in reference [3] the existing literatures on dynamic vehicle routing problem. They found that most of the existing researches are concerning problems similar to DVRP, like dynamic traveling salesman problem, dynamic traveling repairman problem, dynamic dial-a-ride problem, and dynamic vehicle allocation problem. Existing researches focusing on pure dynamic vehicle routing can be categorized into three categories according to the different dynamic parameters. One is the *DVRP with Dynamic Travel Times*, whose dynamic parameter is the vehicle's travel time [3]~[6]. Another category is the *DVRP with Dynamic Customer Order Arrival*, in which new orders arrive when existing orders have been scheduled, dynamically changing the optimization problem. Reference [7]~[9] mainly concerns this kind of problems. The last category is the *DVRP with Stochastic Customer Demands*, which is also called stochastic vehicle routing problems. Secomandi [10] studied the applicability of neuro-dynamic programming algorithms to the single-vehicle routing problem with stochastic demands.

The above mentioned DVRP and solution methods for them are all exclusively suitable for a special defined problem which usually contains one parameter's one kind of change. However, in practice, multiple changes usually concur in a same case. How to handle this kind of problem is the mainly concerned topic of the paper.

Knowledge representation for models was firstly brought out by scholars in the field of model management in DSS in 1980s. As DSS is developed to intelligent decision support systems (IDSS), knowledge representation for models becomes a concerned issue in this field. Fay [11] presented a Fuzzy Petri Net notion that combines the graphical power of Petri Nets and the capabilities of Fuzzy Sets to model rule-based expert knowledge to represent models in a train traffic control decision support system. Oldenburg and Marquardt [12] proposed flatness and higher order differential model representations. This method improved the performance of dynamic optimization algorithms employed in on-line applications, and improved the efficiency of on-line decision support. Biswas and Narahari [13], Yeh and Qiao [14] used object-oriented method to represent models. The research of model representation in China started in late 1980s and has become flourishing in recent years. Many representations have been brought out. Hu et al. [15] brought out a six tuple structure $M = (I, G, O, T, B, S)$ to represent the model of discrete dynamic programming. Zhang et al. [16] came up with a model case representation method. Wang and Cai [17] presented an object-oriented reasoning model and its representation method.

Analyzing the development status of model representation, we can find that influenced by the structure of the multi-base system, conventional model representations in DSS usually separate data, model, inference and knowledge from each other and set up their bases and management systems respectively. In fact, in the process of decision making, all the acquired things that can provide help for decision makers, including facts, rules, methods and inference, are all the essential knowledge. Regarding all of

them as knowledge and managing them as a whole can facilitate the concision of the system structure and the consistent processing procedure, which has become the main research trend of knowledge representation in IDSS. We'll adopt the idea in the paper.

We present an innovative method that combines reasoning knowledge with heuristic solution knowledge to represent a specified DVRP, which contains multiple dynamic changes consisting of three discrete events—new order arrival, old order withdrawn-ness, and demand change. And a solution framework based on the knowledge representation is presented, too.

The remaining paper is organized as follows. Section 2 presents the knowledge representation for DVRP model. Section 3 proposes an on-line solution approach to DVRP based on the presented representation. Section 4 studies a case to demonstrate the approach's effect. The last section draws our conclusions.

2 Knowledge Representation for DVRP Model

2.1 Mathematical Model of VRP

The mathematical model of a VRP is usually an integer model as follows.

$$\left\{ \begin{array}{l} \min \quad z = \sum_{i=0}^n \sum_{j=0}^n \sum_{k=1}^m c_{ij} x_{ijk} \\ \sum_{i=1}^n g_i y_{ki} \leq Q \quad k = 1, \dots, m \\ \sum_{k=1}^m y_{ki} = 1 \quad i = 1, \dots, n \\ \sum_{i=0}^n x_{ijk} = y_{kj} \quad j = 0, 1, \dots, n; k = 1, \dots, m \\ \sum_{j=0}^n x_{ijk} = y_{ki} \quad i = 0, 1, \dots, n; k = 1, \dots, m \\ X = (x_{ijk}) \in S \quad (S \text{ is the constraint for the removal of branches}) \\ x_{ijk} = \begin{cases} 1 & \text{vehicle } k \text{ travels from } i \text{ to } j \\ 0 & \text{otherwise} \end{cases} \quad y_{ki} = \begin{cases} 1 & \text{vehicle } k \text{ visits node } i \\ 0 & \text{otherwise} \end{cases} \quad y_{k0} = 1 \end{array} \right.$$

This mathematical model is complicated and very difficult to be represented in a computer. And its fixed structures cannot be changed easily with the change of problems. What's worse, the pure mathematical symbols cannot support reasoning processes that is needed in a dynamic environment.

2.2 BRISGC Six-Component Representation

Generally, the solution process of heuristic algorithms for VRP needs four elements: initial state, neighborhood function, evaluation function, and goal state. An initial state is the necessity of the solution process of VRP, since every improving algorithm and meta-heuristic algorithm must start work on an initial solution. Every algorithm has its own neighborhood function, which is defined by specific algorithm, guiding the solution process to generate new neighborhood solutions of the current solutions. An evaluation function is also defined by specific algorithm. Different algorithms have different evaluation functions guiding the solution process to select better solutions in

current solution's neighborhood. A goal state, where the solution process must stop at, is preset by a real-world problem or founded by an algorithm.

Besides the four elements, there is another important element relevant to the working process of a heuristic algorithm for VRP. When a heuristic algorithm constructs or selects a state, it will test if the restrictions can be satisfied. Thus, restrictions should be separately represented. Moreover, an exclusive component is needed to dynamically collect basic information because of the unstable customer cluster in the on-line e-commerce. Finally, a coordinator is needed. Accordant with Du et al.'s conclusions [18], we believe that different phase of a solution process for DVRP should use different kinds of algorithms. A coordinator is needed to make these algorithms well cooperate with each other.

From the above analysis, we represent model of DVRP as follows:

$M_{DVRP} = (B, R, I, S, G, C)$. *B* is Basic data collector, a processor behind the man-machine interface. It is in charge of select route information associated with current planned customers from the digital map, and collect customers' demands. After collecting needed information, this component will transfer the information into a database. *R* is Restrictions that mainly contain the resource limit of the third logistics center, like vehicle's capacity limit, drivers' schedule limit, etc. *I* is Initial state generator, which contains several constructive heuristic algorithms. *S* is State operator, which consists of neighborhood function and evaluation function. This component mainly concerns how to generate new states and which state should be chosen as the next solution. Since the two functions are defined by specific improving heuristic algorithm and vice versa, the component accommodates different kind of improving heuristic algorithms. *G* is Goal state defining the stopping rules for improving algorithms' repeat process. *C* is Controller, a cooperative coordinator of the above five component. It has two functions. One is to connect the five components. The other is to dynamically select appropriate algorithm according to different problem goal and different phases of solution process. This component is realized by rules and it is the brain of solution process for DVRP. Because of the page limit, the detailed rules will be narrated in another paper. We term the representation *BRISGC Six-component representation* in the paper.

3 On-Line Solution Method for Dynamic Vehicle Routing Problem

3.1 Structure of the Knowledgebase

The relation among BRISGC six components and the structure of the knowledgebase composed of them is shown in figure 1.

In the knowledgebase, fact base stores the information from orders, including demand, customer information, and rout information associated with the customer, and information from the distribution center, like vehicle information, schedule information etc.. The rule base stores the modeling and solution rules for problem-solving process, including restrictions that the process should obey, constructive rules for initial routes,

rules for improving routes and process stopping rules. The dynamic knowledgebase is a special store to deposit interim solutions and temporarily to store the final solutions.

3.2 Problem-Solving Process Based on BRISGC Six-Component Knowledge Representation

The solution approach to DVRP based on BRISGC Six-component representation is shown in figure 2.

Fig. 1. Structure of the knowledgebase for BRISGC model representation of DVRP

Fig. 2. Problem-solving process of DVRP based on BRISGC Six-component knowledge representation

The flowchart in real line rectangle indicates the process that the BRISGC six-component representation solves the on-line DVRP. The five dashed rectangles represent the working realm of the other five components respectively. The whole process is realized under the control of Component C. The module “discrete event arrival” in the figure contains three kind of event—new order arrival, old order withdrawnness, and demand change. Component C can adopt different rules to adjust current solutions. For example, if suddenly a customer in one of routes in planned distribution scheme withdraws its order, component C will use “Delete Demand” rules, these rules are associated with how to rearrange customers in routes in existing distribution scheme after delete some customers. Details of some other rules, including “Add Demand”, “Change Demand”, aren’t given in the paper. We will exclusively draft another paper on the controller to narrate all rules. The arrows in the figure indicate the sequence when these components work and their transferring relations. The next part demonstrates the solution process in detail in a real-world case.

4 Case Study

A gas sale company in Dalian city sells several types of gas, including 97#, 93#, and 90#. According to historical sales experience, the company prepares vehicles with

different capacities respectively for delivering different kinds of gas, 8t capacity vehicle in charge of 97# gas, 10t in charge of 93#, and 16t in charge of 90#. Most of gas stations in the city adopt just-in-time policy, so they place orders through the gas company's website before 2-3 hours of their gas being sold up. In order to fulfill orders timely, the company process orders once an hour. However, after a batch of orders within an hour has been processed, some orders might be withdrawn because of unexpected market, or certain ordering quantity is changed. Therefore, a dynamic routing system which can promptly adjust distribution schemes to these emergencies is needed. Orders received on 11/5/2006 from 8:00a.m. to 9:00a.m. are shown in table 1.

Working process of the six components is as follows:

B component: the component will firstly abstract route information associated with the above mentioned 13 gas stations and the company's distribution center (results shown in table 2, the distribution center is represented by 0). And then, it aggregates demand information, including demand quantity of every type of gas, desired delivery time, etc. and transfers this information to basic database.

Table 1. Orders Received from Website of the Gas Company on 11/5/2006 from 8:00am. to 9:00am

Gas Station Id	Gas Station Name	Gas Type	Quantity	Expected Latest Arrival Time
1	Yinghua	90#	3	10:00
1	Yinghua	97#	6	10:00
3	Hanlin	90#	9	10:30
4	Qingming	90#	6	10:30
4	Huangming	93#	8	10:30
5	Taishang	90#	7	10:30
5	Taishang	97#	2	10:30
6	Suiyuan	90#	9	10:00
6	Suiyuan	97#	5	10:00
7	Lvshan	90#	9	10:30
8	Dafu	97#	2	10:00
9	Xinghaiwan	90#	4	10:00
9	Xinghaiwan	93#	6	10:00
10	Huaxia	93#	6	10:00
11	Tianying	90#	3	10:30
11	Tianying	93#	4	10:30
12	Tianlu	90#	9	10:00
13	Heishijiao	90#	3	10:30
15	Bobai	93#	3	10:00

Table 2. Route information (symmetrical routes)

nodes \ nodes	0	1	3	4	5	6	7	8	9	10	11	12	13	15
0	0	9	14	9	16	6	14	7	8	8	17	9	6	15
1		0	7	13	12	7	8	6	12	9	10	16	6	13
3			0	5	7	7	12	17	9	9	7	8	8	6
4				0	7	13	10	11	8	7	5	13	16	15
5					0	5	13	10	16	13	15	14	11	5
6						0	11	13	6	14	11	15	14	14
7							0	16	16	16	17	15	15	15
8								0	8	9	16	15	15	5
9									0	10	11	9	17	7
10										0	9	7	14	14
11											0	7	13	7
12												0	11	13
13													0	15
15														0

R component: the content of the component in this case consists of capacities of different kinds of vehicle.

I component: after C component examines tradeoffs between basic information, it will command I component to use an appropriate constructive heuristic algorithm to generate initial routes under the constraint of R component. In this case, the component will use the Time-Oriented, Nearest-Neighbor heuristic algorithm to generate routes (consult reference [19] for details of the algorithm) according to commands from C component. The initial routes are as follows:

For delivering 90# gas:

Route 90#1: $0 \xrightarrow{6} 13 \xrightarrow{8} 3 \xrightarrow{7} 11 \xrightarrow{17} 0$, total distance (or travel time): 38, total load :15

Route 90#2: $0 \xrightarrow{6} 6 \xrightarrow{13} 4 \xrightarrow{9} 0$, total distance (or travel time): 28, total load :15

Route 90#3: $0 \xrightarrow{8} 9 \xrightarrow{9} 12 \xrightarrow{15} 1 \xrightarrow{9} 0$, total distance (or travel time): 41, total load :16

Route 90#4: $0 \xrightarrow{14} 7 \xrightarrow{13} 5 \xrightarrow{16} 0$, total distance (or travel time): 43, total load :16

For delivering 93# gas:

Route 93#1: $0 \xrightarrow{8} 9 \xrightarrow{11} 11 \xrightarrow{17} 0$, total distance (or travel time): 36, total load : 10

Route 93#2: $0 \xrightarrow{8} 10 \xrightarrow{14} 15 \xrightarrow{15} 0$, total distance (or travel time): 37, total load : 9

Route 93#3: $0 \xrightarrow{9} 4 \xrightarrow{9} 0$, total distance (or travel time): 18, total load : 8

For delivering 97# gas:

Route 97#1: $0 \xrightarrow{6} 6 \xrightarrow{5} 5 \xrightarrow{16} 0$, total distance (or travel time): 27, total load : 7

Route 97#2: $0 \xrightarrow{7} 8 \xrightarrow{6} 1 \xrightarrow{9} 0$, total distance (or travel time): 22, total load : 8

G component: the component will use time as the stop rule of improving process of S component according to the command from C component. In this case, the time is set to 30 seconds. That is the total work time of S component is 30 seconds.

S component: the 2-swap and 2-exchange algorithms in the component respectively improve initial routes within and between routes within 30 seconds under the constraint of R component. After 30 seconds, the initial routes are improved. Route 90#3 has been changed to Route 90#3':

Route 90#3': $0 \xrightarrow{9} 1 \xrightarrow{12} 9 \xrightarrow{9} 12 \xrightarrow{9} 0$, total distance (or travel time): 39, total load : 16

and other routes maintain original state. And after the goal is achieved, G component transfers the result to C component.

C component: the first function of the component is examining characteristics of the problem in order to command other components as we have mentioned in I and G components' work processes. The other function is commanding dynamic knowledgebase to hold the result according to G component for 5 minutes (the time is preset according to system characteristics). If within the 5 minutes there aren't changes, C component will command the knowledgebase to output the result. Otherwise, if some events occur within the 5 minutes, then C component will take appropriate actions to adjust the result.

For example, in the case, if gas station 3 additionally places an order at 9:02 a.m., which order 1t 93# gas, the component will trigger "Add Demand" rule module. The module will select the most appropriate insertion algorithm (in the case, it selected C-W savings algorithm) to insert the demand into the most suitable route:

Route 93#2: $0 \xrightarrow{8} 10 \xrightarrow{14} 15 \xrightarrow{15} 0$, total distance (or travel time): 37, total load : 9

and get Route 93#2':

Route 93#2': $0 \xrightarrow{8} 10 \xrightarrow{14} 15 \xrightarrow{6} 3 \xrightarrow{14} 0$, total distance (or travel time): 42, total load : 10.

After that, if there are no other emergencies within the 5 minutes, G component will output final schemes consist of Route 90#1, Route 90#2, Route 90#3', Route 90#4, Route 93#1, Route 93#2', Route 93#3, Route 97#1, Route 97#2.

5 Conclusions

In this paper, we propose a knowledge-based model representation—BRISGC six-component model representation for DVRP and a problem-solving approach to it based on deep analysis of existing solutions to VRP. The structure of the representation is flexible enough to support on-line and real-time modeling and solution process. Moreover, the representation can make fully use of conventional algorithms and extend their application field from static VRP to dynamic VRP, and support on-line and real-time modeling and solution process.

Future work includes how to adapt the representation structure to integrate meta-heuristic algorithms and to solve DVRP with more parameters' more changes.

Acknowledgement. This work is supported by National Natural Science Foundation of China (Grant No. 70571009, 70171040 and 70031020 (key project)), Key Project of the Chinese Ministry of Education (No. 03052), and Ph.D. Program Foundation of Ministry of Education of China (No. 20060141013).

References

1. Sun, L. J., Hu, X. P., Wang, Z.: Reviews on Vehicle Routing Problem and Its Solution Methods. *Systems Engineering*, no.11 (2006) 35-41
2. Larsen, A.: The Dynamic Vehicle Routing Problem. PhD Thesis, IMM-PHD-2000-73, ISSN 0909-3192, LYNGBY (2001) 5
3. Haghani, A. and Jung, S.: A dynamic vehicle routing problem with time-dependent travel times. *Computers & Operations Research* 32 (2005) 2959-2986
4. Taniguchi, E. and Shimamoto, H.: Intelligent transportation system based dynamic vehicle routing and scheduling with variable travel times, *Transportation Research Part C* 12 (2004) 235-250
5. Ichoua, S., Gendreau, M., Potvin, JY.: Vehicle dispatching with time dependent travel times. *European Journal of Operations Research* 144 (2003) 379-96
6. Potvina, J. Y., Xua, Y., Benyahia, I.: A dynamic vehicle routing problem with time-dependent travel times. *Computers & Operations Research* 32 (2005) 2959-2986
7. Fleischmann, B., Gnutzmann, S., Sandvoß, E.: Dynamic Vehicle Routing Based on Online Traffic Information, *Transportation science*, Vol. 38, No. 4 (2004) 420-433
8. Branke, J., Middendorf, M., Noeth, G., Dessouky, M.: Waiting Strategies for Dynamic Vehicle Routing, *Transportation science*, Vol. 39, No. 3 (2005) 298-312
9. Montemanni, R., Gambardella, L. M., Rizzoli, A. E., Donati, A. V.: Ant Colony System for a Dynamic Vehicle Routing Problem, *Journal of Combinatorial Optimization*, Vol. 10, No. 4 (2005) 327-343
10. Secomandi, N.: Comparing neuro-dynamic programming algorithms for the vehicle routing problem with stochastic demands. *Computers & Operations Research* 27 (2000) 1201-1225
11. Fay, A.: A fuzzy knowledge-based system for railway traffic control. *Engineering Applications of Artificial Intelligence* 13 (2000) 719-729
12. Oldenburg, J., and Marquardt, W.: Flatness and higher order differential model representations in dynamic optimization. *Computers and Chemical Engineering* 26 (2002) 385-400

13. Biswas, S., and Narahari, Y: Object oriented modeling and decision support for supply chains. *European Journal of Operational Research* 153 (2004) 704-726
14. Yeh, A. G., and Qiao, J. J.: ModelObjects—a model management component for the development of planning support systems. *Computers, Environment and Urban Systems* 29 (2005) 133-157
15. Hu, X. P., Xu, Z. C., and Yang, D. L.: Intelligent operations research and real-time optimal control for dynamic systems, *Journal of Management Sciences In China*, vol. 5, no. 4 (2002) 13-21
16. Zhang, J., Pan, Q. S., and Hu, Y. Q.: Model Case Representation. *Journal of Management Science in China*, vol. 3, no. 2 (2000) 90-94
17. Wang, H. Y. and Cai, R. Y.: An Object-oriented Inference Model and Its Knowledge Representation. *Journal of Nanjing University of Technology*, vol. 24, no. 3 (2002) 20-24
18. Du, T. C., Li, E. Y., and Chou, D: Dynamic vehicle routing for online B2C delivery. *Omega* 33 (2005) 33-45
19. Solomon, M. M.: Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research* 35 (1987) 254-265