A Novel Approach of Route Choice in Stochastic Time-varying Networks

Siliang Wang, Minghui Wang

Abstract**—**Many exist studies always use Markov decision processes (MDPs) in modeling optimal route choice in stochastic, time-varying networks. However, taking many variable traffic data and transforming them into optimal route decision is a computational challenge by employing MDPs in real transportation networks. In this paper we model finite horizon MDPs using directed hypergraphs. It is shown that the problem of route choice in stochastic, time-varying networks can be formulated as a minimum cost hyperpath problem, and it also can be solved in linear time. We finally demonstrate the significant computational advantages of the introduced methods.

Keywords**—**Markov decision processes (MDPs), stochastic time-varying networks, hypergraphs, route choice.

I. Introduction

RECENTLY, there has been growing interest in determining optimal path in stochastic time-varying networks (STV networks). In STV networks, travel times are modeled as random variables with time-dependent distributions, which often provide a better modeling tool in transportation applications [1, 2, 3].

Hall studies for the first time about STV networks [4]. It is shown that in a stochastic, time varying network, the standard shortest path algorithms (such as Dijkstra's algorithm) aren't able to find the shortest path. The best route from any given node to the goal node depends not only on the node, but also on the arrival time to the node. Thus, the optimal route choice is not simple path but a policy that describes which node should be visited once the arrival time to a node is realized. Hall suggested dynamic programming for finding optimal policy. Based on the Hall's work, many studies (Miller-Hooks [5], Pretolani [6], Gao [7], Nielsen [8]) are presented on how to compute the optimal route policy in STV networks. Among them, some paper search the policy based on computing the mean and variance of the travel time while others use the finite horizon MDPs to stimulate the procedure.

For the sake of deficient deterministic travel time distribution information of all links under stochastic networks, it is proved that the method using mean of link travel time is less applicable than the MDPs's in describing real transportation networks. However, the decision procedure of MDPs used in STV networks depends on a lot of state variables for its dynamicity and time dependency, which requires enormous computational times [9,10]. The difficulty of this type of problem has made find optimal route choice an area of intensive investigation.

In this paper we focus on the presence of variable traffic conditions on real transportation networks, where the information can greatly affect the outcomes of the planned schedule. We consider the finite horizon MDPs with finite state and action space and develop an efficient algorithm for finding optimal routing policies in STV networks. The main contribution of the paper is to apply finding minimum cost hyperpath to model MDPs used in finding stochastic shortest paths properly.

This paper is organized as follows: Section 2 is problem description. In section 3, a hypergraph model for the finite horizon MDPs. Section 4 presents an efficient solving method. The final section concludes the paper.

II. Problem description

We formulate the optimal route determination problem as a discrete time, finite horizon Markov decision processes in STV networks. Let $G=(N,A)$ be a stochastic time-varying network, N is the set of nodes and A is the set of links. The network has a single start node s and multiple destinations. Define the link travel time vector at time t to be $L(t)=\{L_1(t),...,L_m(t)\}$, where m denotes the number of links, and each link travel time is a random variable. Let $U=\{0,1,...T\}$ be the set of possible times that decisions are made. Define the state space of our decision problem to be

$$S=\{(n,t,l)\mid n\in N, t\in U, l\in L(t)\} \tag{1}$$

Therefore, the size of the state space is $l^{|N|\times|U|}$. Given the decision maker observes state, the vehicle may choose an action $a\in A_{s,st}$ from the set of actions and then generate cost $c_{st}(s,a)$. For simplicity, these kinds of actions omit staying and waiting at node, and only consider reroute or go direct next node. Moreover, the transition probabilities of obtaining next state $s'\in S_{st+1}$ at stage $st+1$ is $p_{st}(.\mid s,a)$.

A deterministic Markovian policy is defined a function $\pi:S\rightarrow N$ that prescribes which node should be visited next for each node. The characteristic of deterministic comes from its certainty and Markovian since it depends on

Siliang Wang is with the School of Computer Science, Sichuan University, Chengdu 610065, Sichuan, China (e-mail: hhren1980@126.com).

Minghui Wang is with the School of Computer Science, Sichuan University,Chengdu 610065,Sichuan,China.

Open Science Index, Computer and Information Engineering Vol:5, No:2, 2011 publications.waset.org/7130.pdf

the current system state. A policy specifies the decision rules to be used at all stages and provides the decision maker with a plan of which action to take given stage and state. It means a policy is a sequence of decision rules, that is $\pi = (\pi_0, \pi_1, ..., \pi_{ST})$. Consequently, the mathematical model of the MDPs is

$$\begin{aligned} &\min\{E_s^{\pi}(\sum_{st=0}^{ST} c_{st}(X_{st}, \pi_{st}))\} \\ &s.t. M = \sum_{st \in ST, s \in S_{st}, a \in A_{s,st}} \left| p_{st}(. \mid s, a) \right| \end{aligned} \tag{2}$$

where X_{st} is a random variable taking values in S_{st} and denotes the state of the system at stage st. Here M is the total number of possible transition. In Equation (2), the policy maker seeks a policy π_{st} to minimize the expected total cost. It is well known that there exists a deterministic Markovian policy in above model [11]. According to the definition of MDPs, the cost to go $u_{st}^{\pi}(s)$ which denotes the expected total cost given policy π at stage st can be found using the following recursive equations

$$u_{st}^{\pi}(s) = \begin{cases} c_{st}(s,a) & st = ST \\ c_{st}(s,a) + \sum_{s' \in S_{st+1}} p(s' \mid s,a) u_{st+1}^{\pi}(s') & st < ST \end{cases} \tag{3}$$

Thus, the optimal route policy with the minimal total cost for all stages and states can be found using the Bellman equations

$$u_{st}^{*}(s) = \min_{a \in A_{s,st}} u_{st}^{\pi}(s) \tag{4}$$

The value of $u_{st}^{*}(s)$ is the optimal expected cost. The procedure is often referred to as value iteration or backward induction in MDPs.

III. A HYPERGRAPH MODEL FOR THE FINITE-HORIZON MDPS

The motivation that using hypergraph modeling the procedure of MDPs is the likeness between the recursive Eq. (3) and procedure of finding optimal path in hypergraph. The transition probabilities $p(s' \mid s, a)$ in Equation (3) determine possible nodes which will be visited next. That is, each arc has more than two nodes. Moreover, the definition of hypergraph just describes the special arc attribution.

A directed hypergraph is a pair $H(V, E)$, where V is the set of nodes, and E is the set of hyperarcs. A hyperarc $e \in E$ is a pair $e \in (T(e), h(e))$, where $T(e) \subset V$ denotes the set of tail nodes and $h(e) \subset V \setminus T(e)$ denotes the head node [12]. According to the discussion above, we redefine the hypergraph $H(V, E)$ with state space to describe the recursive Eq. (3).

Definition1: The definition of the node and hyperarc set of the revised hypergraph $H'(V, E)$ is

$$\begin{aligned} &V = \{v_{s,st} \mid st = 0, ..., ST, s \in S\} \\ &E = \{e_{a,s,st} \mid st = 0, ..., ST, s \in S, a \in A_{s,st}\} \\ &e_{a,s,st} = (\{v_{s',st+1} \mid s' \in S_{st+1}, p_{st}(s' \mid s, a)\}, v_{s,st}) \\ &A_{s,st} = \{\phi\}, if \ st = ST \end{aligned} \tag{5}$$

In view of the Definition1, it is easily proved that the structure of the revised hyperarc is the same as the standard hypergraph. The acyclic hypergraph $H'(V, E)$ can be built in $O(M)$ time and $size(H') = O(M)$ [13].

Consider a finite horizon MDP and its corresponding revised hypergraph, the policy $\pi = (\pi_0, \pi_1, ..., \pi_{ST})$ defines a unique hypertree for its acyclic feature. The weight of hyperpath is $W(v)$. A weighting function is defined by the recursive equation [14]

$$W(v) = \begin{cases} 0 & v = s \\ w(p(v)) + F(p(v)) & v \in V_{\pi} \setminus \{s\} \end{cases} \tag{6}$$

Here $p(v)$ is the unique hyperarc in the hypertree which has node v as the head, and s is the start node. Note that $F(e)$ denotes a non-decreasing function of the weights in the node of $T(e)$, and $w(e)$ is the nonnegative weight of arc e. The cost function is defined as follows:

$$F(e) = \sum\nolimits_{v \in T(e)} a_e(v) W(v), \exists p(v) = e \tag{7}$$

Compare with the (3) and (6), each hyperarc e can be assigned multipliers to replace the variable $a_e(u)$ in Equation (7).

$$a_e(v) = \begin{cases} 1 & v = v_{st+1} \\ p_{st}(s' \mid s, a) & e = e_{a,s,st}, v = v_{s',st+1} \in T(e) \end{cases} \tag{8}$$

Since the recursive definitions of W and $u_{st}^{\pi}(s)$ are identical, $W(v_{s,st}) = u_{st}^{\pi}(s)$, the problem of finding an optimal route choice in STV networks can be formulated as finding a minimum cost hyperpath problem in the revised hypergraph $H'(V, E)$.

IV. OPTIMAL ROUTE CHOICE IN STV NETWORKS

In order to get efficient method to find a minimum cost hyperpath, two major problems are addressed in the research. The first one that strongly influences the computation efficiencies is the big size of state space. The second one is how to develop an algorithm to solve the minimum cost hyperpath problem efficiently based on the former work.

Example1. A directed hypergraph $H(V, E)$ is shown in Fig. 1,where $E = \{e_1, e_2, e_3, e_4, e_5\}$, $V = \{v_s, v_1, v_2, v_3, v_4, v_5, v_d\}$. Assume each node (except node v_s, v_d) with n state and one stage number is m in its corresponding revised hypergraph $H'(V, E)$, the number of nodes in the revised hypergraph $H'(V, E)$ is enlarged to $5nm$. Furthermore, according to the discussion, we know the size of the revised hypergraph is mainly decided by the size of state space and stage number. As a result, researches on reduction of the size of state space are urgent in its application in real transportation networks.

In this section, we develop a method on reduction of state space. The main idea relies on the definition of earliest start time and latest arrival time. The variables offer us the heuristic method that defines the bound of the time space. As a result, each node/time pair would be defined in a concrete time and position space.

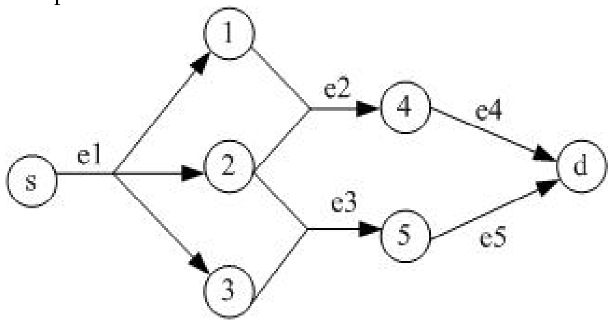

Fig. 1 A directed hypergraph $H(V,E)$

Proposition 1. Generally, given $G(N,E)$, we denote the discrete time horizon as $U=\{L,L+1,\cdots,T\}$, where L is the earliest start time while T is the latest arrival time of any vehicle. Let Sp_i be the shortest path value from source node s to node i using static link travel cost $cs_{ij},\forall(i,j)\in E$, and let $L_i \equiv L+Sp_i,\forall i\in V$. The final stochastic shortest path solution $\{s=n_1,\cdots,n_u=d\}$ that represents the nodes visiting at respective times $T_1,...,T_u$ must meet $T_k \ge L_{n_k},\forall k=1,...,u.$

Proof. Since the definition of L, we have $T_1 \ge L_{n_k}=L$. Thus, for each $k=2,...,u,$ since $T_k-T_1\ge Sp_{n_k}$, we have that following conclusion

$$T_k \ge L_{n_k} \tag{9}$$

*Remark 1.*In a directed hypergraph, we denote by

$$FS(v)=\{e\in E \mid v\in T(e)\}, BS(v)=\{e\in E\mid v=h(e)\} \tag{10}$$

forward links and the backward links of node v, respectively. According to equation (1), the state space S is defined by (n,t), where state (n,t) denotes the status of being at node n at time t such that it is possible to transition along arc $(n,succ(n))$ en route to reach node d at the specified time, and here $succ(n)$ is the successive node of the node n.

The set of states at stage $st\in\{0,1,...,ST\}$, is denoted S_{st}. The stage denotes an epoch when the traveler is located at a node from which it is possible to reach node d in finite steps. Since the backward induction used in MDPs, we have that $S_0=\{(d,T)\}$. Moreover, we compute S_k as follows:

$$S_k=\bigcup_{(j,t')\in S_{k-1}}\{(i,t)\mid i\in BS(j),\, t=t'-cs_{ij},\, t\ge L_i\} \tag{11}$$

Hence, we reduce the set of state space according to

$$S_k=S_k-\{(i,t)\} \tag{12}$$

where $(i,t')\in S, t'=t$. The principle of the reduction is to delete the state as it is not on the way, in which the time consuming descends in the procedure of backward recursion from the destination node to start node.

Proposition 2. The hypergraph method is more efficient than the MDPs in solving the optimal route choice in STV networks.

Proof. According to Equation (1), the complexity of the standard MDPs method is $\Theta(l^{|N||U|})$. It grows exponentially as large state space and will make computing difficult in transportation networks. As regards the Algorithm 1, its computation complexity is $\Theta(m\log n+size(H))$, where n is the number of nodes and m is the number of arcs in the network.

The comparison of complexity between MDPs and hypergraph proves that the method of hypergraph is more efficient than the MDPs.

Algorithm 1 Optimal route choice in STV networks based on state space reduction

step1: Initialization. Set $S_0=\{(d,T)\}$, and compute $L_i,\forall i\in V$,as defined in Proposition 1. Check whether the time constraints ($T_i\ge L_{n_i}$) satisfied, if exists, then set $k=1$, else stop.

step2: 1.Given S_{k-1}, compute S_k using Eq. (11),(12).
2. Build a hypergraph $H(V,E)$ based on the reduced state space and $G(N,E)$.
3. Main loop
For ($e\in BS(v_i)$) do
If ($W(v_i)>w(e)+F(e)$) then
$W(v_i)=w(e)+F(e)$;
$p(v_i)=e$;
End for

step3: If $S_k=\phi$ or $k=T$, go to step4. Else, $k=k+1$ and repeat step2.

step4: Output the optimal route choice according to the set of vertices through $p(v_i)$.

V. Conclusions

A novel approach is proposed to finding optimal route choice in STV networks. The approach models finite horizon MDPs using directed hypergraphs. With the help of efficient shortest path algorithm, the computation complexity is obviously decreased. The results are proved clearly in third section. Future work is to apply the method in real transportation networks.

References

[1] Miller-Hooks, E.. Adaptive least expected time paths in stochastic, time-varying transportation and data networks. Networks 37(1), 35–52,2001.

[2] Miller-Hooks, E., Mahmassani, H.,. Least possible time paths in stochastic, time-varying networks. Computers and Operations Research 25, 1107–1125.,1998.

[3] Miller-Hooks. Optimal routing in time-varying, stochastic networks:Algorithms and implementation. Ph.D. Dissertation, The University of Texas at Austin..,1997

[4] R.W. Hall. The fastest path through a network with random time-dependent travel times. Transportation Science, 20(3):182-188, 1986..

[5] Miller-Hooks, E., Mahmassani, H.. Least expected time paths in stochastic, time-varying transportation networks.Transportation Science 34, 198-215.,2000.

[6] D. Pretolani. A directed hypergraph model for random time-dependent shortest paths. European Journal of Operational Research, 123:315, 2000..

[7] S.Gao et al.Best routing policy problem in stochastictime-dependent networks, Transp. Res. Rec., vol.1783, pp.188–196,2002..

[8] L.R. Nielsen, K.A. Andersen, and D. Pretolani. Bicriterion shortest hyperpaths in random time-dependent networks. IMA Journal of Management Mathematics, 2003.

[9] S. Kim, Optimal vehicle routing and scheduling with real-time traffic information, Ph.D. thesis, Dept. Ind. Syst. Eng., Univ. Michigan, AnnArbor, MI, 2003..

[10] G.H.Polychronopoulos and J. N. Tsitsiklis, Stochastic shortest path problems with recourse, Networks, vol. 27, no. 2, pp. 133–143, 1996..

[11] L.R. Nielsen. Route Choice in Stochastic Time-Dependent Networks. PhD thesis, Department of Operations Research, University of Aarhus, 2004..

[12] L.R. Nielsen, K.A. Andersen, and D. Pretolani. Finding the K shortest hyperpaths.Computers & Operations Research, 32(6):1477–1497, 2005.....

[13] L.R. Nielsen, D. Pretolani, and K.A. Andersen. Finding the K shortest hyperpaths using reoptimization. Operations Research Letters.doi:10.1016/j.orl.2005.04.008, 2005....

[14] G. Gallo, G. Longo, S. Pallottino, and S. Nguyen. Directed hypergraphs and applications. Discrete Applied Mathematics, 42:177–201, 1993...