

Travel Routes Estimation in Transportation Systems Modeled by Petri Nets

Yizhi Qu, Lingxi Li, Yan Liu, Yaobin Chen, and Yaping Dai

Abstract—This paper develops an algorithm for estimating the route(s) with the least total travel time in transportation systems that are modeled as Petri nets. Each transition in the net is associated with a cost that is related to the travel time from a starting point to a destination. This cost can be computed from the traffic flow and vehicle speed information obtained from the traffic data via an approach called Adaptive Gray Threshold Traffic Parameters Measurement (AGTTPM). Given a transportation system modeled as a Petri net that has cost on each transition, we aim at finding the transition firing sequences (traffic routes) from an initial marking (a starting point) to a final marking (a destination) within a certain time period \mathbb{T} and have the least total cost (the least total travel time). In this paper we develop an algorithm that is able to systematically obtain these routes with the least total travel time.

Index Terms—Petri nets, transportation system, least-cost, transition firing sequences, AGTTPM.

I. INTRODUCTION

A discrete event system (DES) is a dynamic system that evolves in accordance with the asynchronous occurrence of physical events at possibly unknown and irregular intervals [1], [2]. Any large-scale dynamic system, such as a transportation system, a manufacturing system, a biological system, and a communication network could be modeled as a discrete event system after some level of abstraction [2].

Among discrete event system models, Petri nets (PNs) are widely used to model and analyze practical systems because of their modeling ability and flexibility. They are powerful models for describing dynamic systems that are characterized as asynchronous, concurrent, nondeterministic, and/or stochastic, and providing a structural framework for carrying out systematic analysis [3]–[5].

Recently, the constant increase on the number of vehicles has caused serious traffic congestion problems worldwide, especially in major cities. Researchers have proposed different approaches to alleviate this situation such as novel traffic signal lights control strategies, adaptive signal phase allocation, and traffic routing. These approaches mainly rely on traffic characteristics and a variety of traffic parameters

This material is based upon work supported in part by the Indiana University-Purdue University Indianapolis (IUPUI) RSFG Grant.

Yizhi Qu is with the Department of Electrical and Computer Engineering and the Transportation Active Safety Institute (TASI), Indiana University-Purdue University Indianapolis, Indianapolis, IN 46202, USA, and also with the School of Automation, Beijing Institute of Technology, Beijing, China. Lingxi Li and Yaobin Chen are with the Department of Electrical and Computer Engineering and the Transportation Active Safety Institute (TASI), Indiana University-Purdue University Indianapolis, Indianapolis, IN 46202, USA. Yan Liu and Yaping Dai are with the School of Automation, Beijing Institute of Technology, Beijing, China. Corresponding author: Lingxi Li, 723 West Michigan Street, Indianapolis, IN 46202, USA. Email: LL7@iupui.edu.

of interest. In [6], the authors proposed an approach called Adaptive Gray Threshold Traffic Parameters Measurement (AGTTPM) for extracting traffic parameters such as traffic flow and average vehicle speed from real traffic data. This approach was based on an image processing algorithm and was able to obtain the parameters in real time that could be used for traffic control and traffic condition evaluation.

In this paper, we focus on the problem of finding the routes with least travel time for a given transportation system that is modeled as a Petri net. Each transition in the net is associated with a cost that is related to the travel time computed from the traffic parameters obtained from the AGTTPM system. Given the starting location (the initial marking) and the destination (the final marking), we develop an algorithm that is able to obtain the routes (transition firing sequences) with the least travel time (the least total cost) from the starting point to the destination within a certain time period \mathbb{T} .

This paper is organized as follows. In Section II we introduce necessary Petri net notation. In Section III we give a brief review of the AGTTPM system. In Section IV we formulate the problem to be studied. The algorithm is developed and analyzed in Section V. We illustrate our algorithm using a transportation system example in Section VI. Conclusions and directions for future work are discussed in Section VII.

II. PETRI NET NOTATION

This section provides some basic definitions and terminology that will be used throughout the paper. More details about Petri nets can be found in [2], [4], [5].

A Petri net structure is a weighted bipartite graph $N = (P, T, A, W)$ where $P = \{p_1, p_2, \dots, p_n\}$ is a finite set of places (drawn as circles), $T = \{t_1, t_2, \dots, t_m\}$ is a finite set of transitions (drawn as bars), $A \subseteq (P \times T) \cup (T \times P)$ is a set of arcs (from places to transitions and from transitions to places), and $W : A \rightarrow \{1, 2, 3, \dots\}$ is the weight function on the arcs.

A marking is a vector $M : P \rightarrow \mathcal{N}$ that assigns to each place in the Petri net a nonnegative integer number of tokens (drawn as black dots). We use M_0 to denote the initial marking of the Petri net. A Petri net is denoted by $\langle N, M_0 \rangle$, where M_0 denotes the initial marking of the Petri net.

Let b_{ij}^- denote the integer weight of the arc from place p_i to transition t_j , and b_{ij}^+ denote the integer weight of the arc from transition t_j to place p_i ($1 \leq i \leq n$, $1 \leq j \leq m$). Note that b_{ij}^- (b_{ij}^+) is taken to be zero if there is no arc from place p_i to transition t_j (or vice versa). We define the input incident matrix $B^- = [b_{ij}^-]$ (respectively the output incident matrix $B^+ = [b_{ij}^+]$) to be the $n \times m$ matrix with b_{ij}^- (respectively b_{ij}^+)

at its i -th row, j -th column position. The *incident matrix* of the Petri net is defined to be $B \equiv B^+ - B^-$.

A transition t is said to be *enabled* if each of its input places has at least $B^-(p, t)$ tokens, where $B^-(p, t)$ is the weight of the arc from place p to transition t . We use $M[t]$ to denote that t is enabled at marking M . An enabled transition t may *fire*. When it fires, it removes $B^-(p, t)$ tokens from each input place of t and deposits $B^+(p, t)$ tokens to each output place of t to yield a new marking $M' = M + B(:, t)$, where $B(:, t)$ denotes the column of B that corresponds to t . This is also denoted by $M[t]M'$.

Let $\sigma = t_{i_1}t_{i_2}\dots t_{i_k}$ ($t_{i_j} \in T$) be a transition firing sequence. We say σ is enabled with respect to M if $M[t_{i_1}]M_1[t_{i_2}]\dots[t_{i_k}]$; this is denoted by $M[\sigma]$. Let $M[\sigma]M'$ denote that the firing of σ from M yields M' and let $\bar{\sigma}(t)$ be the total number of occurrences of transition t in σ . More specifically, $\bar{\sigma} = [\bar{\sigma}(t_1)\dots\bar{\sigma}(t_m)]$ is the *firing vector* that corresponds to σ . Note that after firing a sequence σ from marking M , the new marking M' can also be computed as $M' = M + B\bar{\sigma}$.

A *cost function* $C : T \rightarrow \mathcal{N}^+$ assigns to each transition in the net a positive integer cost. Given a transition firing sequence $\sigma = t_{i_1}t_{i_2}\dots t_{i_k}$, its total cost is given by $C(\sigma) = \sum_{j=1}^k C(t_{i_j}) = \sum_{j=1}^m C(t_j) \cdot \bar{\sigma}(t_j)$.

Example 1 Consider the Petri net model shown in Fig. 1(a). The net has three places p_1, p_2, p_3 and three transitions t_1, t_2, t_3 . The initial marking is given by $M_0 = [2 \ 1 \ 1]^T$ and the cost vector C is given by $C = [5 \ 2 \ 4]$. The input incident matrix B^- (respectively the output incident matrix B^+) is given by

$$B^- = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad B^+ = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}.$$

The incident matrix B is obtained as

$$B \equiv B^+ - B^- = \begin{bmatrix} -1 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}.$$

After transition t_1 fires under the initial marking, one token is removed from its input place p_1 and one token is deposited into its output place p_2 to result a new marking $M = [1 \ 2 \ 1]^T$ as shown in Fig. 1(b). Note that marking M can also be obtained through

$$M = M_0 + B\bar{\sigma} = \begin{bmatrix} 2 \\ 1 \\ 1 \end{bmatrix} + \begin{bmatrix} -1 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}$$

where $\bar{\sigma} = [1 \ 0 \ 0]^T$ is the firing vector of transition firing sequence t_1 (it has only one nonzero entry "1" corresponding to transition t_1). Clearly, the cost of transition firing sequence t_1 is 5. \square

Definition 1 Given an initial marking M_0 , the set of reachable markings with respect to transition firing sequences of length L is given by $Z(L) = \{M' \mid \exists \sigma : M_0[\sigma]M' \text{ and } |\sigma| = L\}$, where $|\sigma|$ is the length of transition firing sequence σ .

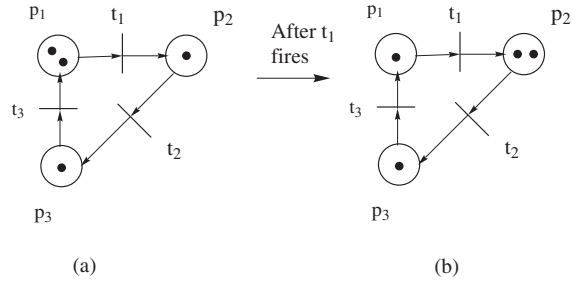


Fig. 1. A simple Petri net (a) and the net after transition t_1 fires (b).

III. AGPPTM SYSTEM

In this section we briefly introduce the AGPPTM system, more details can be found in [6].

In AGTTPM, TMS320DM642 (named DM642 for short) is the core processing unit which is connected with PC (personal computer) and SCM (Single Chip Mickey). The output of DM642 can be monitored by PC and sent to SCM for the traffic lights control. The system hardware structure is shown in Fig. 2.

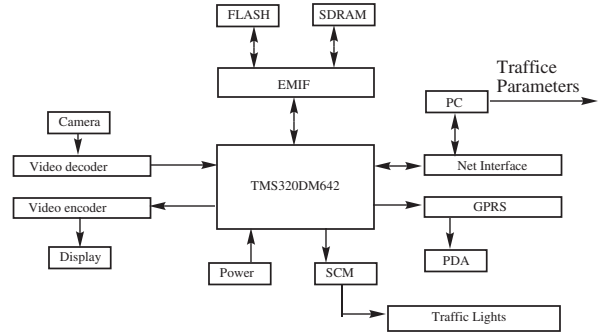


Fig. 2. AGTTPM system hardware structure.

In AGTTPM, the video analog are obtained from CCD camera and sent to the DM642 after analog to digital conversion. The video is processed via software and traffic parameters such as vehicle flow and average vehicle speed are extracted by the RAGT (a real-time adaptive gray threshold) measurement and sent to the central controller. The working flow of AGTTPM is shown in Fig. 3. The RAGT measurement is presented to overcome the sensitivity of vision based system to illumination changing. The traffic parameters can be obtained in real time and the corresponding errors can be inhibited.

IV. PROBLEM FORMULATION

The problem we deal with in this paper is motivated by the following scenario. Consider a vehicle at a starting point and can reach the destination via different travel routes. In general, the driver will take the route with less traffic so that to get the destination in shorter time compared with the congested route. As shown in the Petri net model in Fig. 4, a token in place p_1 models a vehicle at an intersection and the

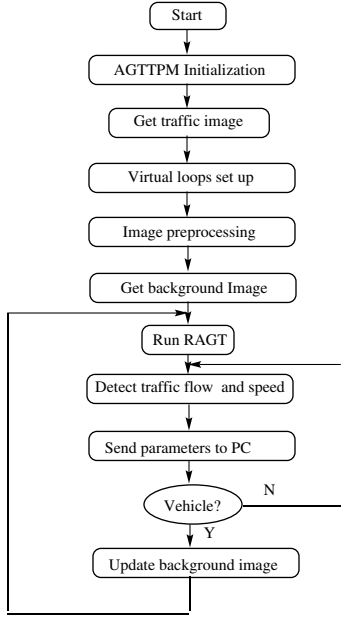


Fig. 3. AGTTPM system working flow.

set of transitions $T = \{t_1, t_2, t_3\}$ denotes the possible behavior of the vehicle as

- t_1 : turn left
- t_2 : go straight
- t_3 : turn right

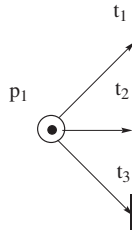


Fig. 4. Vehicle behavior at an intersection.

Consider the transportation system shown in Fig. 5, a vehicle is ready to move from location A to B. It is not difficult to see that there are three different travel routes that can lead the vehicle to the destination. However, the traffic flow situation of each route is different so that the travel time needed is different. The AGTTPM system could efficiently obtain average vehicle speed of each road section, which is used to compute the average travel time (cost) given the fixed distance of each road section.

Assume that the distance of each road section is given by $d_{AC} = 100$ meters, $d_{AE} = 120$ meters, $d_{AF} = 100$ meters, $d_{CD} = 120$ meters, $d_{DE} = 100$ meters, $d_{BE} = 100$ meters, $d_{BF} = 120$ meters; and the average vehicle speed of each road section is estimated by an AGTTPM system as $v_{AC} = 10$ m/s, $v_{AE} = 6$ m/s, $v_{AF} = 5$ m/s, $v_{CD} = 12$ m/s, $v_{DE} = 10$ m/s, $v_{BE} = 5$ m/s, $d_{BF} = 6$ m/s (where m/s stands for meters per second), the estimated travel time can be computed as the ratio of the distance to the average vehicle speed as $t_{AC} =$

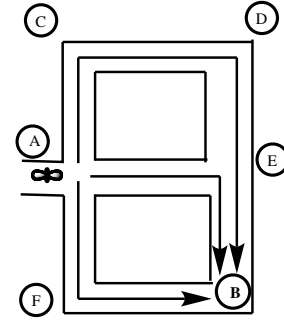


Fig. 5. An example of a transportation system.

$10s$, $t_{AE} = 20s$, $t_{AF} = 20s$, $t_{CD} = 10s$, $v_{DE} = 10s$, $v_{BE} = 20s$, $d_{BF} = 20s$ (where s stands for seconds).

These estimated travel times can be incorporated into Petri net models as transition costs. The Petri net model of the transportation system in Fig. 5 is shown in Fig. 6. This net has six places (i.e. intersections) and seven transitions. The initial marking is given by $[1\ 0\ 0\ 0\ 0\ 0]^T$ and the final marking is $[0\ 0\ 0\ 0\ 0\ 1]^T$. From the previous discussion, the cost of each transition is defined as $C(t_1) = 10$, $C(t_2) = 10$, $C(t_3) = 10$, $C(t_4) = 20$, $C(t_5) = 20$, $C(t_6) = 20$ and $C(t_7) = 20$. In order to go from the initial marking to the final marking (to drive from point A to point B), there are three different transition firing sequences (travel routes) $t_1t_2t_3t_6$, t_4t_6 , and t_5t_7 with the total cost 50, 40, and 40 respectively. Clearly, the least-cost transition firing sequences are given as t_4t_6 and t_5t_7 .

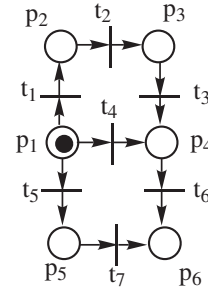


Fig. 6. The Petri net model for the system shown in Fig. 5.

Remark 1 Note that the average vehicle speed of each road section changes dynamically, the estimated routes (transition firing sequences) with the least total travel time (the least cost) are not fixed during the time of the day in general.

In this paper we consider the problem of estimating the least-cost transition firing sequences within a certain time period \mathbb{T} . Since the travel time can be seen as the cost, the time period \mathbb{T} is essentially the upper bound on the total cost of these transition firing sequences. This quantity can also be converted to the upper bound on the length of these transition firing sequences as follows.

Definition 2 Consider a cost vector $C = [C(t_1) \dots C(t_m)]^T$ with positive integer cost on each transition, we define C_{min}

as the smallest cost in this vector. Then the upper bound on the length of the transition firing sequences L is given by $L = \frac{T}{C_{min}}$.

Now the problem we consider reduces to the problem of finding the transition firing sequence(s) with length less than or equal to L which: (i) is (are) consistent with the structure of the Petri net, (ii) leads (lead) us from the initial marking to the final marking, and (iii) has (have) the least total cost.

Clearly, the set of least-cost transition firing sequence(s) $\{\sigma_{min}\}$ is the solution to the following problem:

$$\arg \min_{\sigma} C(\sigma) \text{ s.t. } M_0[\sigma]M \ \& \ |\sigma| \leq L. \quad (1)$$

In the next section, we will develop a recursive algorithm to obtain these least-cost transition firing sequences efficiently. The recursion is in terms of the length of the transition firing sequences.

V. LEAST-COST TRANSITION FIRING SEQUENCE(S)

A. Algorithm

Data structure $\mathcal{C} = (M_{current}, leastcost, (t_{in}, M_{previous}))$ is used to capture the information we need to store. More specifically, at length j : $M_{current}$ denotes the marking that is under consideration; $leastcost$ is the least cost among all valid firing sequences from M_0 to $M_{current}$; $(t_{in}, M_{previous})$ denotes that input transition t_{in} that is enabled at $M_{previous}$ such that the least cost firing sequence goes through $M_{previous}$ at length $j-1$ and leads to $M_{current}$ by firing transition t_{in} .

Note that if M can be reached from M_0 via one or more transition firing sequences, the algorithm outputs the one(s) that has (have) the least total cost; if no such firing sequence exists, the algorithm does not provide any output.

We describe the algorithm in detail below.

Algorithm 1

Input: A Petri net N with positive costs C associated with each transition, a given initial marking M_0 , a target final marking M , and a maximum time cost T .

1. $\mathcal{C}(0) = \{(M_0, 0, (\emptyset, \emptyset))\}$. Compute maximum length $L = T/C_{min}$ as shown in Definition 2.
2. Let $j = 1$.
3. Set $\mathcal{C}(j) = \emptyset$.
4. For all $R \in \mathcal{C}(j-1)$ and $R.M_{current} \neq M$ do
 - For all t such that $R.M_{current}[t]$
 - compute $M' = R.M_{current} + B(\cdot, t)$
 - If M' is a new marking that has not appeared in $\mathcal{C}(j)$
 - $\mathcal{C}(j) = \mathcal{C}(j) \cup \{(M', R.leastcost + C(t), (t, R.M_{current}))\}$
 - Else
 - M' has appeared in $R' \in \mathcal{C}(j)$
 - If $R.leastcost + C(t) < R'.leastcost$
 - $R' = (M', R.leastcost + C(t), (t, R.M_{current}))$
 - Else If $R.leastcost + C(t) = R'.leastcost$
 - $R' = (M', R'.leastcost, R'.(t_{in}, M_{current}) \cup (t, R.M_{previous}))$
 - End IF
- End IF
- End For

End For

5. $j = j + 1$.

6. If $j = L + 1$, Goto 7; else Goto 3.

7. Recover all firing sequences from M_0 to M using the information stored and output the one(s) that has (have) the least total cost. \square

Given the Petri net structure N (with n places and m transitions) and its initial marking M_0 , Algorithm 1 recursively computes markings that are reachable from the initial marking along with the least cost of transition sequences that lead from the initial marking to each of these markings. When the final marking M appears at a particular step, the algorithm does not consider transitions that emanate from M at later steps (because, even if they revisit M , they are guaranteed to have higher cost). After considering all valid transition firing sequences with length equal to or less than L , the algorithm recovers all transition firing sequences from the initial marking M_0 to the final marking M , and outputs the one(s) that has (have) the least total cost.

VI. AN ILLUSTRATIVE EXAMPLE

In this section, we illustrate the algorithm via a more complicated example in a transportation system.

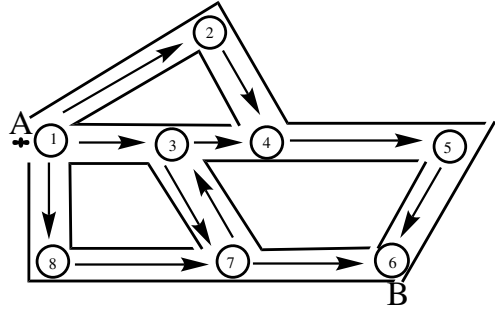


Fig. 7. A transportation system example.

As shown in Fig. 7, there are seven intersections with distances in between given as $d_{12} = 40\text{m}$, $d_{24} = 30\text{m}$, $d_{13} = 30\text{m}$, $d_{34} = 20\text{m}$, $d_{45} = 50\text{m}$, $d_{65} = 30\text{m}$, $d_{76} = 30\text{m}$, $d_{73} = 40\text{m}$, $d_{37} = 40\text{m}$, $d_{87} = 40\text{m}$, $d_{18} = 30\text{m}$, where d_{ij} is the distance from intersection i to j . A vehicle at current location A will move to destination B within time $T = 18\text{s}$.

Fig. 8 is the Petri net model for the transportation system shown in Fig. 7, where intersections are modeled as places and possible vehicle movements are modeled as transitions. The net has eight places and eleven transitions. The initial marking is given by $M_0 = [1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$ and the final marking is given by $M = [0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0]^T$. According to AGTTPM systems employed at each road section, we obtain the average vehicle speed at each road section as $v_{12} = 10\text{m/s}$, $v_{24} = 5\text{m/s}$, $v_{13} = 5\text{m/s}$, $v_{34} = 5\text{m/s}$, $v_{45} = 10\text{m/s}$, $v_{65} = 10\text{m/s}$, $v_{76} = 3.33\text{m/s}$, $v_{73} = 10\text{m/s}$, $v_{37} = 5\text{m/s}$, $v_{87} = 10\text{m/s}$, $v_{18} = 10\text{m/s}$. Clearly, the cost associated with each transition could be computed as the ratio of the distance to the average vehicle speed for each section as $C = [4 \ 6 \ 6 \ 4 \ 5 \ 3 \ 9 \ 4 \ 8 \ 4 \ 3]$.

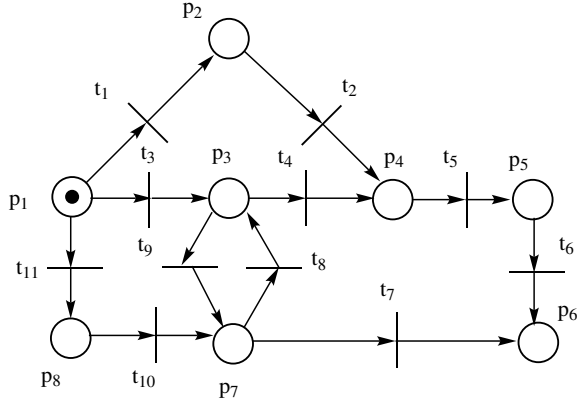


Fig. 8. Petri net model for the transportation system shown in Fig. 7.

Given the time period $\mathbb{T} = 18$ and minimum cost in C as $C_{min} = 3$, we could compute the upper bound on the length of the possible transition firing sequences as $L = \frac{\mathbb{T}}{C_{min}} = 6$. Our goal is to find the least-cost transition firing sequence(s) within length 6 that could lead us from the initial marking to the final marking.

For this case, Algorithm 1 finds the set of least-cost firing sequences that could lead us from the initial marking to the final marking within length 6 to be $\sigma_{min} = \{t_{11}t_{10}t_7\}$ with total cost 16. We provide the following table where *Length* denotes the length of the transition firing sequences considered so far, *Leastcost* captures the least total cost of sequence(s) that satisfies (satisfy) the length constraint and leads (lead) from M_0 to M (we use ∞ to denote that M cannot be reached via transition sequences of the given length from M_0), and σ_{min} is the set of least-cost transition sequence(s) of the length considered.

<i>Length</i>	<i>Leastcost</i>	σ_{min}
1	∞	\emptyset
2	∞	\emptyset
3	16	$t_{11}t_{10}t_7$
4	18	$t_3t_4t_5t_6$ $t_1t_2t_5t_6$
5	28	$t_{11}t_{10}t_8t_9t_7$
6	23	$t_{11}t_{10}t_8t_4t_5t_6$

Remark 2 Note that the final marking could be reached via different transition firing sequences. In addition, the sequence with smaller length is not guaranteed to give us the least total cost. For instance, the final marking could be reached via sequence $t_{11}t_{10}t_8t_9t_7$ of length 5 with cost 28; but could

also be reached via sequence $t_{11}t_{10}t_8t_4t_5t_6$ of longer length 6 with smaller cost 23. Therefore, in order to capture the least-cost transition firing sequences of length less than or equal to L from the initial marking to the final marking, we need to consider all markings that could be reached from the initial marking within L steps.

We do not discuss the complexity of the algorithm in this paper as it could be analyzed using the similar techniques employed in [7].

VII. CONCLUSIONS AND FUTURE WORK

In this paper, we developed an algorithm for estimating the route(s) with the least total travel time in transportation systems that are modeled as Petri nets. The intersections are modeled as places and possible vehicle movements are modeled as transitions. Each transition in the net is associated with a cost that is related to the travel time computed from the average vehicle speed information via the AGTPM approach. Given the Petri net with the initial marking, the final marking, and costs associated with transitions, we aim at finding the transition firing sequences from an initial marking to the final marking within a certain time period \mathbb{T} and have the least total cost. We have shown that this time upper bound could be converted to the upper bound on the length of the transition firing sequences and have proposed a recursive algorithm to obtain these sequences efficiently.

One direction for future work is to study the estimation problem in nets with both observable transitions and unobservable transitions. Another interesting extension is to address the problem in a setting where the marking of the net is only partially known.

REFERENCES

- [1] P. J. Ramadge and W. M. Wonham, "The control of discrete event systems," *Proc. of the IEEE*, vol. 77, no. 1, pp. 81–98, January 1989.
- [2] C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*, Boston, MA: Kluwer, 1999.
- [3] M. C. Zhou and F. DiCesare, *Petri Net Synthesis for Discrete Event Control of Manufacturing System*, Norwell, MA: Kluwer, 1993.
- [4] T. Murata, "Petri nets: properties, analysis and applications," *Proc. of the IEEE*, vol. 77, no. 4, pp. 541–580, April 1989.
- [5] J. L. Peterson, *Petri Net Theory and the Modeling of Systems*, Englewood Cliffs, NJ: Prentice-Hall, 1981.
- [6] Y. Liu, Y. Dai, and Z. Dai, "Real-time adaptive gray threshold measurement in extracting traffic parameters," in *Proc. of the 3rd Intl. Symp. Computational Intelligence and Industrial Applications*, pp. 255–265, 2008.
- [7] L. Li, Y. Ru, and C. N. Hadjicostis, "Least-cost transition firing sequence estimation in labeled Petri nets," in *Proc. 45th IEEE Intl. Conf. Decision and Control*, pp. 416–421, December 2006.