Automatica 47 (2011) 2094–2098

Contents lists available at SciVerse ScienceDirect

Automatica

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

Brief paper

Reachability determination in acyclic Petri nets by cell enumeration approach☆

Duan Li [a,*], Xiaoling Sun [b], Jianjun Gao [a], Shenshen Gu [c], Xiaojin Zheng [d]

[a] *Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Shatin, Hong Kong*
[b] *Department of Management Science, School of Management, Fudan University, Shanghai, China*
[c] *School of Mechatronics Engineering and Automation, Shanghai University, Shanghai, China*
[d] *School of Economics and Management, Tongji University, Shanghai 200092, China*

ARTICLE INFO

Article history:
Received 21 July 2009
Received in revised form
24 October 2010
Accepted 14 March 2011
Available online 20 July 2011

Keywords:
Petri nets
Reachability analysis
Cell enumeration
Linear Diophantine equations on bounded integer set

ABSTRACT

Reachability is one of the most important behavioral properties of Petri nets. We propose in this paper a novel approach for solving the fundamental equation in the reachability analysis of acyclic Petri nets, which has been known to be NP-complete. More specifically, by adopting a revised version of the cell enumeration method for an arrangement of hyperplanes in discrete geometry, we develop an efficient solution scheme to identify firing count vector solution(s) to the fundamental equation on a bounded integer set, with a complexity bound of $O((nu)^{n-m})$, where n is the number of transitions, m is the number of places and u is the upper bound of the number of firings for all individual transitions.

1. Introduction

Petri net, introduced by Petri in his seminal work "Communication with Automata" (Petri, 1962), has been a promising mathematical formalism for modeling, analyzing and designing discrete event systems, especially by its remarkable capability in modeling process synchronization, asynchronous events, concurrent and distributed operations and resource sharing. The past four decades have witnessed innumerable successful applications of Petri nets in various areas, including (i) modeling and analysis of communication networks, production systems and software development, and (ii) performance evaluation of the modeled systems.

The Petri net is a particular kind of bipartite directed graph consisting of three types of elements: places, transitions, and directed arcs connecting places and transitions, while each place may hold a nonnegative number of tokens. See Fig. 1 for an illustrative example of Petri nets with 4 places and 6 transitions. The distribution of tokens on places, called Petri net marking, defines the state of the modeled system. A marking for a Petri net with m places is represented by an $(m \times 1)$ vector M, where M_j, $j = 1, 2, \ldots, m$, are nonnegative integers representing the number of tokens in the corresponding places.

A general Petri net is characterized by a quintuple (P, T, I, O, M_0) (see Zurawski & Zhou, 1994), where $P = \{p_1, p_2, \ldots, p_m\}$ is a finite set of m places, $T = \{t_1, t_2, \ldots, t_n\}$ is a finite set of n transitions, $D_I : (P \times T) \mapsto Z_+^{m\times n}$ is an input function that defines weights associated with directed arcs from places to transitions, where $Z_+^{m\times n}(Z_+^m)$ is the set of $(m \times n)$ dimensional matrices (m dimensional vector) with all entries being in Z_+, $D_O : (P \times T) \mapsto Z_+^{m\times n}$ is an output function that defines weights associated with directed arcs from transitions to places, and $M_0 : P \mapsto Z_+^m$ is the initial marking.

The change of the distribution of the tokens represents the dynamics of the modeled system, while the distribution of tokens on places may change according to the following enabling rule and firing rule (see Zurawski & Zhou, 1994):

Enabling Rule: A transition t is said to be enabled if each input place p of t contains at least the number of tokens equal to the weight of the arc connecting p to t.

Firing Rule: (a) An enabled transition t may or may not fire depending on the additional interpretation; and (b) A firing of an enabled transition t removes from each input place p the number of tokens equal to the weight of the directed arc connecting p to t, and deposits in each output place p the number of tokens equal to the weight of the arc connecting t to p.

☆ Research supported in part by University Grants Council, Hong Kong, under CUHK414207 and 2050478; by the National Natural Science Foundation of China under Grants 10971034 and 70832002; and by the Joint NSFC/RGC Grants under Grant 71061160506. The material in this paper was partially presented at the 13th IFAC Symposium on Information Control Problems in Manufacturing, June 3–5, 2009, Moscow, Russia. This paper was recommended for publication in revised form under the direction of Editor Berç Rüstem.

* Corresponding author. Tel.: +852 2609 8323; fax: +852 2609 5505.
E-mail addresses: dli@se.cuhk.edu.hk (D. Li), xls@fudan.edu.cn (X. Sun), jjgao@se.cuhk.edu.hk (J. Gao), gushenshen@shu.edu.cn (S. Gu), xjzheng@fudan.edu.cn (X. Zheng).

0005-1098/$ – see front matter
doi:10.1016/j.automatica.2011.06.017

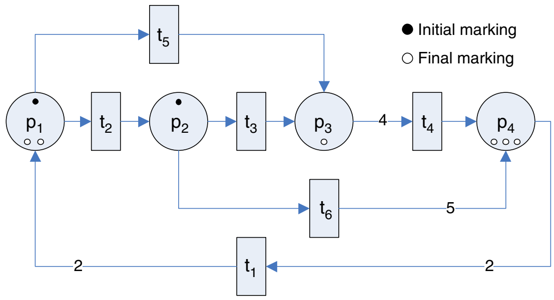

Fig. 1. Example of a Petri net.

Reachability is one of the most important behavioral and structural properties of Petri nets. Given both an initial state M_0 and a target terminal state M, a natural question to ask is whether or not we have a sequence of firing rules such that the system can reach the specific target state within finite steps. There are two primary approaches in investigating reachability: reachability graph analysis and reachability algebraic analysis. Reachability graph analysis is based on the creation and investigation of a reachability graph or a reduced reachability graph. As this approach relies on exhaustively generating all the reachable markings from a given initial marking, it suffers the state explosion phenomenon, while transferring the reachability graph to a reduced counterpart is NP-hard (see Kostin, 2003). The second approach is based on methods of linear algebra. It is well known (see Murata, 1977; Yen, 2006) that a necessary condition for reachability of marking M from an initial marking M_0 of a Petri net is that there exists a nonnegative *integer* vector solution of the following system of linear equations,

$$Ax = b, \tag{1}$$

where $b = M - M_0, x = (x_1, x_2, \ldots, x_n)^T$ is a firing count vector and A is an $(m \times n)$ dimensional incidence matrix given by $A = [a(p_i, t_j)] = [D_O(p_i, t_j) - D_I(p_i, t_j)]$. Without loss of generality, we assume that $n > m$ and A is of rank m.

When the Petri net is acyclic, the above condition becomes both necessary and sufficient (see Matsumoto, Miyano & Jiang, 1997). We confine ourselves to study acyclic Petri nets in this paper. Eq. (1), termed the *fundamental equation* in Petri nets, reveals that the investigation of reachability can be achieved, to certain degree, by finding out nonnegative integer solution(s) to a system of linear equations. The first step in the reachability algebraic analysis is to find firing count vectors, which can be performed either by solving the fundamental equation directly, or by solving the corresponding integer programming problem. Unfortunately, finding nonnegative integer solutions to the fundamental equation in Petri nets is NP-complete (Matsumoto, Takata & Moro, 2002). The second step in the reachability algebraic analysis is to translate a firing count vector into a firing sequence, if there is a reachable path, while such an algorithm (see, e.g., Kostin, 2003) is of a combinatorial nature.

In essence, solving the fundamental equation is an integer programming problem (see, for example, Li & Sun, 2006; Nemhauser & Wolsey, 1988; Schrijver, 1986). Generally speaking, the capability of the existing methods for finding the nonnegative integer solution to the fundamental equation (1) is still very limited. In particular, the Petri nets community has been largely dependent on some existing softwares to obtain nonnegative integer solutions to the fundamental equation. More efficient solution schemes that invoke the state-of-the-art from some other related subjects, especially from discrete optimization and integer programming, deserve further investigation. There have been a few attempts in the Petri net communities in this direction. For example, the Groebner basis method from integer programming is applied in Matsumoto, Takata & Moro (2002) to solve reachability problems in Petri nets. The solution approach using Groebner bases depends heavily on symbolic computing power, and its capability in solving large-scale problems thus seems not very promising.

Although the original analytical reachability analysis is to find nonnegative integer solution(s) to the fundamental equation, a more practical problem is to find integer solution(s) to the fundamental equation on a bounded integer set, $X = \{x \in Z^n \mid 0 \le x_i \le u_i,\ i = 1, \ldots, n\}$, where Z^n denotes the set of all integer points in R^n. This confinement is justifiable as there always exist resource, e,g., time and cost, constraints in real-world operations which correspond to transitions in associated Petri nets. For example, the operation of a reactor in chemical processes is corresponding to a transition in a corresponding Petri net and such an operation can be considered to be expensive. Thus the number of such operations (or equivalently, the number of firings) should be as small as possible, or best to be bounded by some given upper bound. In any case, we are always most interested in the minimum-time or minimum-energy transition to a target terminal state. Furthermore, the size of the bounded integer set can be adjusted when needed. Although linear Diophantine equations, $Ax = b$, $x \in Z^n$, are known to be *polynomially solvable* (see, e.g., Nemhauser & Wolsey, 1988; Schrijver, 1986), linear Diophantine equations on a bounded integer set, $Ax = b, x \in X$, where A is an $m \times n$ integral matrix with rank$(A) = m$, and $b \in R^m$ being integral, are *NP-complete*, as the special case of linear Diophantine equations with $m = 1, x \in \{-1, 1\}^n$ and $b = 0$ is NP-complete (Garey & Johnson, 1979).

We propose in this paper a novel approach for solving the fundamental equation in the reachability analysis of acyclic Petri nets over a bounded integer set X of firing count vectors. Our method is based on a recognition that whether or not there exists a solution to the fundamental solution is equivalent to whether or not the distance from X to the affine solution set of $Ax = b$ in R^n attains zero. Furthermore, from recent results in Sun, Liu, Li, and Gao (in press), finding the distance from an integer set to an affine set can be efficiently achieved by the cell enumeration method for an arrangement of hyperplanes in discrete geometry. This cell enumeration approach provides a promising platform for designing an efficient method with a complexity of $O((nu)^{n-m})$, where $u = \max_{i=1,\ldots,n} u_i$, to find integer solutions to the fundamental equation on a bounded integer set.

2. Solving the fundamental equation by cell enumeration

Let $U = (U_1, \ldots, U_{n-m})$ be an orthogonal basis for the null space of A and x_0 be a special solution to $Ax = b$ in R^n, then $C = \{x \in R^n \mid Ax = b\}$ can be expressed by

$$C = \left\{x \in R^n \mid x = x_0 + \sum_{i=1}^{n-m} z_i U_i,\ z_i \in R, \imath = 1, \ldots, n - m\right\}. \tag{2}$$

Consider the Euclidean distance from C to $X = \{x \in Z^n \mid 0 \le x_i \le u_i,\ i = 1, \ldots, n\}$:

$$\delta = \text{dist}(C, X) = \min\{\|x - y\| \mid x \in X,\ y \in C\}. \tag{3}$$

It is obvious that $Ax = b$ has a solution in X if and only if $\delta = 0$. Furthermore, when $\delta = 0$, any $x^* \in X$ that achieves the zero distance is an integer solution to the linear equations $Ax = b$ in X.

Combining (2) and (3) yields

$$\begin{aligned}\delta &= \text{dist}(C, X) \\ &= \min_{y \in C} \min\{\|y - x\| \mid 0 \le x_i \le u_i,\ x_i \in Z,\ i = 1, \ldots, n\} \\ &= \min_{y \in C} \|y - \varphi(y)\|,\end{aligned}$$

where, for $j = 1, \ldots, n$, $\varphi(y)_j$ is determined by the following equation:

$$\varphi(y)_j = \begin{cases} 0, & y_j \in \left(-\infty, \frac{1}{2}\right], \\ i, & y_j \in \left(i - \frac{1}{2}, i + \frac{1}{2}\right),\ i \in \{1, \ldots, u_j - 1\}, \\ u_j, & y_j \in \left[u_j - \frac{1}{2}, \infty\right). \end{cases}$$

Let us consider the hyperplane arrangement generated by the following $\sum_{j=1}^{n} u_j$ hyperplanes in R^{n-m}:

$$h_{ij} = \left\{ z \in R^{n-m} \mid g_{ij}(z) := (x_0)_j + \sum_{l=1}^{n-m} z_l U_{lj} - \left(i - \frac{1}{2}\right) = 0 \right\} \quad (4)$$

for $i = 1, \ldots, u_j$, $j = 1, \ldots, n$. Note that for each j, $h_{ij}(i = 1, \ldots, u_j)$ are parallel hyperplanes. A cell E of the hyperplane arrangement corresponding to h_{ij}'s is an $(n-m)$-dimensional polyhedral set formed by the half-spaces induced by h_{ij}'s and can be characterized by a $\sum_{j=1}^{n} u_j$-dimensional sign vector, $\text{sign}(E) = (w^1, \ldots, w^n)$, where $w^j = (w_{1,j}, \ldots, w_{u_j,j})$ is specified by

$$w_{ij} = \begin{cases} +, & \text{if } g_{ij}(\pi) > 0 \\ -, & \text{if } g_{ij}(\pi) < 0, \end{cases} \quad i = 1, \ldots, u_j, \quad (5)$$

with π being an interior point of E. For fixed j, since $h_{ij}(i = 1, \ldots, u_j)$ are parallel hyperplanes, $w^j = (w_{1,j}, \ldots, w_{u_j,j})$ must take the following form:

$$(\overbrace{+, \ldots, +}^{i}, -, \ldots, -), \quad i = 0, 1, \ldots, u_j.$$

The one-to-one map between all distinct $\varphi(y)$ for $y \in C$ and the sign vectors of all the cells of the hyperplane arrangement can be established as follows:

$$w = (w^1, \ldots, w^n) \longleftrightarrow \varphi = (\varphi_1, \ldots, \varphi_n),$$

where for $j = 1, \ldots, n$,

$$w^j = (-, \ldots, -) \longleftrightarrow \varphi_j = 0,$$

$$w^j = (+, \ldots, +) \longleftrightarrow \varphi_j = u_j,$$

$$w^j = (\overbrace{+, \ldots, +}^{i}, -, \ldots, -) \longleftrightarrow \varphi_j = i,$$

where $i \in \{1, \ldots, u_j - 1\}$.

It has been known that the number of cells of the hyperplane arrangement generated by (4) is bounded by $O((nu)^{n-m})$, where $u = \max_{i=1,\ldots,n} u_i$ (see Edelsbrunner, 1987; Zaslavsky, 1975). Moreover, using the cell enumeration methods in Avis & Fukuda (1996) and Sleumer (1999), we can find all the cells of the hyperplane arrangement generated by (4) in $O((nu)^{n-m})$ time. Listing all the distinct $\varphi(y)$ for $y \in C$ as $\varphi^1, \ldots, \varphi^k$, the distance $\delta = \text{dist}(C, X)$ can then be calculated by using the projection theorem as follows:

$$\delta = \min_{j=1,\ldots,k} \delta_j \equiv \min_{j=1,\ldots,k} \text{dist}(C, \varphi^j)$$

$$= \min_{j=1,\ldots,k} \|(UU^T - I_n)(\varphi^j - x_0)\|. \quad (6)$$

Therefore, finding a solution of the linear system $Ax = b$ over the integer set X or checking its infeasibility can be done in $O((nu)^{n-m})$ time. Especially, when $r = n-m$ is fixed $(0 \le r \le n-1)$, the linear system $Ax = b$ over X is polynomially solvable.

The cell enumeration method in Avis & Fukuda (1996) starts from the root cell with all w_{ij} being $+$ and involves only computation of a set of linear programming at each iteration. For any given current cell with $\omega = \{w_{ij} \mid w_{ij} = + \text{ when } ij \in I^+ \text{ and } w_{ij} = - \text{ when } ij \in I^-\}$, three types of calculations are performed: (i) Finding an interior point of the cell by solving the following linear programming problem, $\{\max t \mid g_{ij} - t \ge 0,\ \forall ij \in I^+ \text{ and } g_{ij} + t \le 0,\ \forall ij \in I^-, t \le \bar{t}\}$, where $\bar{t}$ is an upper bound imposed on t; (ii) Finding out in the subtree all the neighboring cells of the current cell that have one less component in I^+ by solving the following set of linear programming problems, $\{\min g_{\hat{i}\hat{j}} \mid g_{ij} \ge 0,\ \forall ij \in I^+ \text{ and } g_{ij} \le 0,\ \forall ij \in I^-\}, \forall \hat{i}\hat{j} \in I^+$ (Any $\hat{i}\hat{j}$ with zero objective value gives rise to a neighboring cell with one less component in I^+); and (iii) For any cell w^* generated in (ii), identifying the parent cell of w^* by pinpointing out the hyperplane h_{ij} which intersects with the straight line connecting a pair of interior points of cell w^* and the root cell and has the minimum distance to the interior point of cell w^*. If such a hyperplane boards between w and w^*, then we regard w as the parent cell of w^*, take w^* as the current cell and repeat the above steps. Otherwise, we stop generating cells rooted by cell w^*. The reason to define a parent cell is to avoid a repeat search of cells already identified.

Let us consider now the example in Fig. 1 to illustrate the solution algorithm for the fundamental equation using the cell enumeration method. For the Petri net given in Fig. 1, we have

$$D_I = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 4 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$D_O = \begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 5 \end{pmatrix},$$

and $M_0 = (1, 1, 0, 0)^T$. If we set in Fig. 1 the final marking to be $M = (2, 0, 1, 3)^T$, the fundamental equation for this example is then given by

$$\begin{pmatrix} 2 & -1 & 0 & 0 & -1 & 0 \\ 0 & 1 & -1 & 0 & 0 & -1 \\ 0 & 0 & 1 & -4 & 1 & 0 \\ -2 & 0 & 0 & 1 & 0 & 5 \end{pmatrix} x = \begin{pmatrix} 1 \\ -1 \\ 1 \\ 3 \end{pmatrix}.$$

For this instance, we are interested in finding out integer solutions to the above fundamental equation within the bounded integer set given by $X = \{x \in Z^6 \mid 0 \le x_i \le 2,\ i = 1, \ldots, 6\}$. Since the incidence matrix is of rank 4 in this example, solving the fundamental equation in R^6 yields

$$C = \{x \in R^6 \mid Ax = b\}$$
$$= \{x \in R^6 \mid x = x_0 + z_1 U_1 + z_2 U_2, z_i \in R, i = 1, 2\},$$

where

$$x_0 = (0.2912, -0.2138, 0.0100, -0.2984, -0.2038, 0.776)^T$$

and

$$U = (U_1, U_2) = \begin{pmatrix} 0.3508 & 0.3944 \\ -0.1605 & 0.6882 \\ -0.2713 & 0.5636 \\ 0.1477 & 0.1661 \\ 0.8622 & 0.1006 \\ 0.1108 & 0.1245 \end{pmatrix}.$$

Note that $u_j = 2, j = 1, \ldots, 6$. Setting each $x_j, j = 1, \ldots, 6$, equal to $\frac{1}{2}$ and $\frac{3}{2}$, respectively, generates the following 12 hyperplanes on C defined in (4),

$$j = 1 : \begin{cases} g_{11}(z) = 0.3508z_1 + 0.3944z_2 = 0.2088, \\ g_{21}(z) = 0.3508z_1 + 0.3944z_2 = 1.2088, \end{cases}$$

$$j = 2 : \begin{cases} g_{12}(z) = -0.1605z_1 + 0.6882z_2 = 0.7138, \\ g_{22}(z) = -0.1605z_1 + 0.6882z_2 = 1.7138, \end{cases}$$

$$j = 3 : \begin{cases} g_{13}(z) = -0.2713z_1 + 0.5636z_2 = 0.4900, \\ g_{23}(z) = -0.2713z_1 + 0.5636z_2 = 1.4900, \end{cases}$$

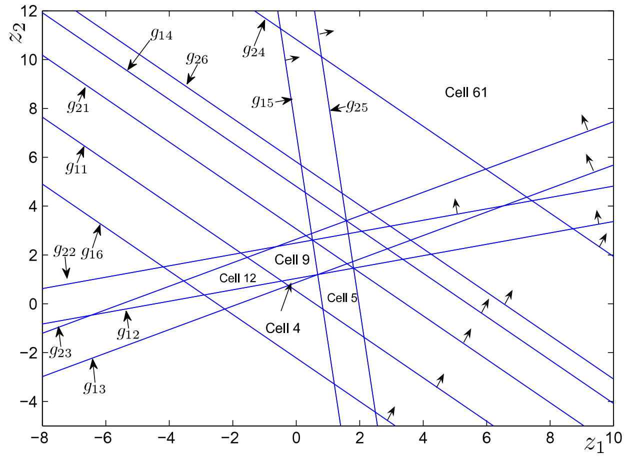

Fig. 2. Cell arrangement for the example problem.

$$j = 4: \begin{cases} g_{14}(z) = 0.1477z_1 + 0.1661z_2 = 0.7984, \\ g_{24}(z) = 0.1477z_1 + 0.1661z_2 = 1.7984, \end{cases}$$

$$j = 5: \begin{cases} g_{15}(z) = 0.8622z_1 + 0.1006z_2 = 0.7038, \\ g_{25}(z) = 0.8622z_1 + 0.1006z_2 = 1.7038, \end{cases}$$

$$j = 6: \begin{cases} g_{16}(z) = 0.1108z_1 + 0.1245z_2 = -0.2762, \\ g_{26}(z) = 0.1108z_1 + 0.1245z_2 = 0.7238. \end{cases}$$

As illustrated in Fig. 2, these 12 hyperplanes partition set C into 61 cells, each of which corresponds to a unique sign vector defined in (5), $\mathrm{sign}(E) = (w^1, \ldots, w^6)$, where $w^j = (w_{1j}, w_{2j})$. Note that the arrow attached to each hyperplane points to the positive half-space generated by this hyperplane.

Let us consider Cells 5 and 61 labeled in Fig. 2. As any interior point π of Cell 5 satisfies $g_{11}(\pi) > 0$, $g_{21}(\pi) < 0$, $g_{12}(\pi) < 0$, $g_{22}(\pi) < 0$, $g_{13}(\pi) < 0$, $g_{23}(\pi) < 0$, $g_{14}(\pi) < 0$, $g_{24}(\pi) < 0$, $g_{15}(\pi) > 0$, $g_{25}(\pi) < 0$, $g_{16}(\pi) > 0$, and $g_{26}(\pi) < 0$, the sign vector for any interior point of Cell 5 is thus specified by $((+-)(--)(--)(--)(+-)(+-))$, which implies that any interior point of Cell 5 satisfies $\frac{1}{2} < x_1 < \frac{3}{2}$, $x_2 < \frac{1}{2}$, $x_3 < \frac{1}{2}$, $x_4 < \frac{1}{2}$, $\frac{1}{2} < x_5 < \frac{3}{2}$, and $\frac{1}{2} < x_6 < \frac{3}{2}$. Thus, for all interior points in Cell 5, the closest integer point in $X = \{0, 1, 2\}^6$ is clearly given by $\varphi^5 = (1, 0, 0, 0, 1, 1)$. Furthermore, the distance between X and Cell 5 is given by

$$\delta_5 = \mathrm{dist}(X, \text{Cell } 5) = \|(UU^T - I_6)(\varphi^5 - x_0)\| = 0.$$

For Cell 61 labeled in Fig. 2, all points are above the 12 hyperplanes. Thus, the sign vector for Cell 61 is $((++)(++)(++)(++)(++)(++))$ and the closest integer point in $\{0, 1, 2\}^6$ to Cell 61 is $\varphi^{61} = (2, 2, 2, 2, 2, 2)$. Furthermore, the distance between X and Cell 61 is given by

$$\delta_{61} = \mathrm{dist}(X, \text{Cell } 61) = \|(UU^T - I_6)(\varphi^{61} - x_0)\| = 1.578.$$

Now we take Cell 9 with $\omega = ((+-), (+-), (+-), (--), (--), (+-))$ as the current cell to illustrate the process of cell enumeration. In such a situation, $I^+ = \{11, 12, 13, 16\}$. Solving the four corresponding linear programming problems, we find two neighboring cells with one less component in I^+, Cell 12 with $\omega = ((--), (+-), (+-), (--), (--), (+-))$ and Cell 4 with $\omega = ((+-), (--), (+-), (--), (--), (+-))$. Setting $\bar{t}$ at 1 and solving two associated linear programming problems yield two interior points of Cell 12 and Cell 4, respectively, $(z_1^*, z_2^*) = (-1.376, 1.0941)$ and $(z_1^*, z_2^*) = (-0.2102, 0.8892)$. Checking the hyperplanes intersecting with the line between an interior point of the root cell (Cell 61), $(z_1^*, z_2^*) = (6, 8)$ and the interior point of Cell 12, $(z_1^*, z_2^*) = (-1.376, 1.0941)$, identifies g_{11} to be the closest hyperplane to $(z_1^*, z_2^*) = (-1.376, 1.0941)$, which implies that Cell 9 is the parent of Cell 12. Similarly, we conclude that Cell 9 is the parent of Cell 4. The enumeration process continues then from both Cell 12 and Cell 4.

Enumerating all 61 cells figures out two solutions, Cell 5 with $\varphi = (1, 0, 0, 0, 1, 1)^T$ and Cell 9 with $\varphi = (1, 1, 1, 0, 0, 1)^T$, to the fundamental equation of this example, as they both achieve the zero distance. We need to emphasize that our solution scheme enables us to identify all solutions to the fundamental equation on a finite integer set. Note also that the total number of cells in this example is 61, which is much smaller than the upper bound of the cell numbers $O((6 \times 2)^{(6-4)})$. This conclusion applies for general situations as observed from randomly generated numerical experiments presented in the next section.

Table 1
Numerical results for randomly generated problems.

m	n	u	$\lvert X\rvert$	Number of cells	CPU time
19	20	5	$O(5^{20})$	123	27.2
18	20	4	$O(4^{20})$	690	12.2
17	19	5	$O(5^{19})$	513	10.9
16	18	5	$O(7^{16})$	244	8.2
15	18	5	$O(5^{18})$	6 126	109.8
14	17	5	$O(5^{17})$	6 268	100.1
8	11	6	$O(6^{11})$	2 410	20.1
12	16	3	$O(3^{16})$	20 681	352.3
11	15	5	$O(5^{15})$	23 029	541.4

3. Numerical results

We present some numerical results for the solution method using cell enumeration developed in Section 2. The solution algorithm was coded by C^{++} and ran on a Pentium PC (2.6 GHz and 1G RAM). Our implementation of the cell enumeration method is based on the reverse search procedure proposed in Avis & Fukuda (1996) and Sleumer (1999). Note that for any fixed j, $j = 1, \ldots, n$, the hyperplanes of h_{ij} are parallel for i, $i = 1, \ldots, u_j$. Such a special structure in the problem formulation facilitates a significant reduction in the number of constraints in linear programming (LP), which is used in finding the active bounds of individual cells (Sleumer, 1999). In our implementation, we use CPLEX 9.0 as the LP solver in the reverse search procedure. For each triple of (n, m, u), 20 test problems are randomly generated. In particular, the elements of A are integers uniformly drawn from $\{-20, \ldots, 20\}$ and u_j's are uniformly generated from $\{1, \ldots, u\}$.

Table 1 summarizes the average numerical results for randomly generated problems with different choices of m, n and u. In Table 1, m is the number of places, n is the number of transitions, u is the upper bound of the number of firings for each transition and $|X|$ denotes the number of integer points in X, a naive bound specified by u^n. It can be seen from Table 1 that the number of cells generated in the cell numeration process and consequently the CPU time increase rapidly when $n - m$ increases. This confirms with the complexity analysis of the cell numeration method.

4. Conclusion

An integer solution to the fundamental equation must be also a real solution at the same time. Therefore, we confine our search efforts on set C, the set of all real solutions to the fundamental equation, while set C can be partitioned into a finite number of cells by a family of hyperplanes, h_{ij}'s. Furthermore, for each cell, e.g., cell i, it is straightforward to identify its closest integer point, φ^i, in X. Thus, the existence of an integer solution to the fundamental solution on a bounded integer set X reduces to a problem whether or not there is a φ^i that is exactly falling on C (or its distance to C is zero). The efficiency of the solution algorithm developed in this paper therefore depends entirely on the efficiency of the cell enumeration scheme which is of a complexity bound $O((nu)^{n-m})$. Preliminary numerical results have revealed that the proposed solution method is promising when $n - m$ is relatively small.

Our solution method developed in this paper for solving the fundamental equation of Petri nets is applicable in finding integer solution(s) to general linear equations on a finite integer set. In contrast to the rich literature on general linear Diophantine equations in Z^n, there exist only a few methods developed for linear Diophantine equations on a bounded integer set. Based on Lovász lattice basis reduction, Aardal, Hurkens & Lenstra (2000) proposed an algorithm to solve bounded linear Diophantine equations. Utilizing the results of Kertzner (1981) to express explicitly integer solutions to a linear equation of two variables, Ramachandran (2006) developed a two-phase algorithm for bounded linear Diophantine equations based on the Euclidean algorithm. Compared to the existing literature, our proposed approach is the first to achieve a fixed-rank complexity of $O((nu)^{n-m})$, i.e., for fixed $(n - m)$, our proposed approach is polynomial in n. Furthermore, the existing literature only applies to situations with integral matrix A, while our method proposed in this paper is applicable to situations with real-valued matrix A, thus enabling a wider range of applications.

Preliminary numerical results have revealed that the proposed solution method is promising when $n - m$ is relatively small. One of our future research topics is to explore possible ways to improve the efficiency of the algorithm when $n - m$ is large. More specifically, we plan to investigate a possible integration of the results in Ramachandran (2006) with cell enumeration algorithm under a two-phase framework: (i) By aggregation of the variables, we can reduce the difference between the numbers of the equations and the unknowns to a situation where the cell enumeration can be successfully applied. (ii) We then perform decomposition to recover the solutions of the primary variables step by step.

References

Aardal, K., Hurkens, C. A. J., & Lenstra, A. K. (2000). Solving a system of linear diophantine equations with lower and upper bounds on the variables. *Mathematics of Operations Research*, *25*, 427–442.

Avis, D., & Fukuda, K. (1996). Reverse search for enumeration. *Discrete Applied Mathematics*, *65*, 21–46.

Edelsbrunner, H. (1987). *Algorithms in Combinatorial Geometry*. Springer.

Garey, M. R., & Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-completeness*. San Francisco: Freeman.

Kertzner, S. (1981). The linear diophantine equation. *American Mathematical Monthly*, *88*, 200–203.

Kostin, A. E. (2003). Reachability analysis in T-invariant-less Petri nets. *IEEE Transactions on Automatic Control*, *48*, 1019–1024.

Li, D., & Sun, X. L. (2006). *Nonlinear Integer Programming*. New York: Springer.

Matsumoto, T., Miyano, Y., & Jiang, Y. (1997). Some useful sufficient criteria for the basic reachability problem in general Petri nets, In *Proceedings of the 36th IEEE Conference on Decision and Control* (pp. 4104–4109), San Diego, CA, USA.

Matsumoto, T., Takata, M., & Moro, S. (2002). Reachability analyses in Petri nets by groebner bases. In *Proceeding of SICE* (pp. 841–846), Osaka.

Murata, T. (1977). State equation, controllability, and maximal matchings of Petri nets. *IEEE Transactions on Automatic Control*, *22*, 412–416.

Nemhauser, G. L., & Wolsey, L. (1988). *Integer and Combinatorial Optimization*. New York: John Wiley & Sons.

Petri, C. A. (1962). *Kommunikation mit automaten*. Ph.D. thesis Schriften des Rheinisch–Westfalischen Instituts fur Instrumentelle Mathematik, Bonn, Germany.

Ramachandran, P. (2006). Use of extended euclidean algorithm in solving a system of linear diophantine equations with bounded variables. *Lecture Notes in Computer Science*, *4076*, 182–192.

Schrijver, A. (1986). *Theory of Linear and Integer Programming*. New York: John Wiley & Sons.

Sleumer, N. (1999). Output-sensitive cell enumeration in hyperplane arrangements. *Nordic Journal of Computing*, *6*, 137–161.

Sun, X. L., Liu, C., Li, D., & Gao, J. J. On duality gap in binary quadratic optimization, *Journal of Global Optimization* (in press). Available at: http://www.optimization-online.org/DB_HTML/2010/01/2512.html.

Yen, H. C. (2006). Introduction to Petri net theory. In *Recent advances in formal languages and applications Studies in Computational Intelligence*, *25*, 343–373.

Zaslavsky, T. (1975). Facing up to arrangements: face-count formulas for partitions of space by hyperplanes. *Memoirs of the American Mathematical Society*, *1*, 1–101.

Zurawski, R., & Zhou, M. C. (1994). Petri nets and industrial applications: a tutorial. *IEEE Transactions on Industrial Electronics*, *41*, 567–583.

Duan Li was born in Shanghai, China. He graduated from Fudan University in 1977 and received the M.E. degree in automatic control from Shanghai Jiaotong University in 1982, and the Ph.D. degree in systems engineering from Case Western Reserve University in 1987. From 1987 to 1994, he was a Faculty Member at the University of Virginia. He joined the Department of Systems Engineering and Engineering Management, the Chinese University of Hong Kong, in December 1994, where he is currently a Chair Professor and Department Chairman. Duan Li's research interests include optimization, optimal control and financial engineering. Duan Li was an Associate Editor of IEEE Transactions on Automatic Control from 2003 to 2004 and has been a member of the editorial board or a guest editor for many other journals, including Journal of Global Optimization, IIE Transactions on Operations Engineering and Control Theory and Advanced Technology.

Xiaoling Sun was born in Jiangxi, China, in 1963. He received his Ph.D. degree in Operations Research from Shanghai University in 1995. From 1995 to 2006, he worked at the Department of Mathematics, Shanghai University, as a lecturer, associate professor and full professor, respectively. Since 2007, he has been a full professor at the School of Management, Fudan University. From 1997 to 1998, he was a visiting scholar at the Department of Systems Engineering and Engineering Management, the Chinese University of Hong Kong. From 1999 to 2000, and 2002, he was a visiting scholar at the School of Mathematics and Statistics, Edinburgh University. Professor Sun's research interests include integer programming, global optimization and financial optimization. He has co-authored two books published by Springer and Science Press China and published over 40 papers in referred international journals. He was awarded the Fok Ying Tung Distinguished Young Teachers Prize in 1998 and the Bao Steel Excellent Teachers Prize in 2001, both by the Ministry of Education of China. He is a Vice President of the Mathematical Programming Society of China and a Council Member of the Operations Research Society of China.

Jianjun Gao was born in Xinjiang, China. He received the B.E. degree in Automatic Control from the University of Science and Technology of China, Hefei, China, in 2003, and the M. Phil. and Ph. D. degrees in Systems Engineering and Engineering Management from the Chinese University of Hong Kong, Hong Kong, in 2005 and 2009, respectively. He is currently a Visiting Scholar in the Department of Systems Engineering and Engineering Management, the Chinese University of Hong Kong. His research interests include optimal control theory and optimization techniques with applications in management science.

Shenshen Gu was born in Shanghai, China. He received his B.E degree in Computer Science from Shanghai University of Engineering Science in 2002, his M.E. degree in Computer Science from Shanghai University in 2005, and his Ph.D. degree in Automation and Computer-Aided Engineering from the Chinese University of Hong Kong in 2009. He then joined the School of Mechatronics Engineering and Automation at Shanghai University, where he is currently a lecturer. Dr. Shenshen Gu's research interests include optimization, optimal control and neural networks.

Xiaojin Zheng was born in Zhengjiang, China, in 1983. She received her M.S. degree in Mathematics from Zhejiang Normal University in 2007, and her Ph.D. degree in Operations Research from Shanghai University in 2009. From 2008 to 2009, she worked at the Department of Systems Engineering and Engineering Management, the Chinese University of Hong Kong, as a research assistant. In 2010, she was a postdoctoral research fellow at the School of Management, Fudan University. She is currently a Lecturer in School of Economics and Management, Tongji University. Xiaojin Zheng's research interests include integer optimization, semidefinite optimization and financial optimization.