Petri Net Analysis (Conserved Properties)

Sokhom Pheng
School of Computer Science
McGill University
April 2\textsuperscript{nd} 2004
Overview

- Intro to conservation properties
- Steps to analyze conservation prop.
- Examples
- Problems/constraints with implementation
- Demo
Intro to Conservation Properties

What is conservation?
- Property to maintain a fixed number of tokens ∀ states reached in a sample path

Why is it useful?
- Sometimes tokens represent resources

We will look at relaxed conservation
- Not the whole Petri net satisfies conservation property
Conservation Definition

Def’n: A Petri net with a given initial state $x_0$ is said to be *conservative with respect to* $\gamma = [\gamma_1, \gamma_2, \ldots, \gamma_n]$ if

$$\sum_{i=1}^{n} \gamma_i x(p_i) = \text{constant}$$

For all states contained in all possible sample paths [1]

Steps to Analyze Cons. Prop.

1. Petri Net
2. Coverability Graph
3. Matrix
4. Linear Programming
Example

Step 1

Step 2

Step 3

\[
\begin{align*}
\begin{bmatrix} 1, 0, 1 \end{bmatrix} & \rightarrow \begin{bmatrix} 1, 1, 0 \end{bmatrix} \\
\gamma_1 + \gamma_3 &= C \\
\gamma_1 + \gamma_2 &= C \\
\begin{bmatrix} 1 & 0 & 1 & 1 \\
1 & 1 & 0 & 1 \\
\gamma_1 & \gamma_2 & \gamma_3 & C
\end{bmatrix}
\end{align*}
\]
Example (cont’d)

Step 4: reduce matrix using Gauss-Jordan elimination

\[
\begin{bmatrix}
1 & 0 & 1 & 1 \\
1 & 1 & 0 & 1 \\
\end{bmatrix} \quad \rightarrow \quad \begin{bmatrix}
1 & 0 & 1 & 1 \\
0 & 1 & -1 & 0 \\
\end{bmatrix}
\]

Alternative: put C in first column

\[
\begin{bmatrix}
-1 & 1 & 0 & 1 & 0 \\
-1 & 1 & 1 & 0 & 0 \\
\end{bmatrix} \quad \rightarrow \quad \begin{bmatrix}
1 & -1 & 0 & -1 & 0 \\
0 & 0 & 1 & -1 & 0 \\
\end{bmatrix}
\]

\[C = \gamma_1 + \gamma_3\]
\[\gamma_2 = \gamma_3\]
Example (cont’d)

Step 5: solve linear programming problem using lp_solve [2]

\[
\begin{align*}
C &= \gamma_1 + \gamma_3 \\
\gamma_2 &= \gamma_3 \\
\gamma_1, \gamma_3 &\geq 0 \\
\gamma_1 + \gamma_3 &\geq 1 \\
\text{int } \gamma_1, \gamma_3
\end{align*}
\]

C = \gamma_1 + \gamma_3 \\
\gamma_2 = \gamma_3 \\
min: \gamma_1 + \gamma_3 \\
\gamma_3 \geq 0 \\
\gamma_1 + \gamma_3 \geq 1 \\
\gamma_1 \geq 0 \\
\gamma_3 \geq 0 \\
\text{int } \gamma_1, \gamma_3

Example (cont’d)

\[
\begin{align*}
\text{min: } & \gamma_1 + \gamma_3 \\
\gamma_3 & \geq 0 \\
\gamma_1 + \gamma_3 & \geq 1 \\
\gamma_1 & \geq 0 \\
\gamma_3 & \geq 0 \\
\text{int } & \gamma_1, \gamma_3 \\
C & = \gamma_1 + \gamma_3 \\
\gamma_2 & = \gamma_3 \\
\end{align*}
\]

Set \( \gamma_1 = 1 : \) \( \gamma_3 = 0 \)
\( C = 1, \gamma_2 = 0 \)

Set \( \gamma_3 = 1 : \) \( \gamma_1 = 0 \)
\( C = 1, \gamma_2 = 1 \)

\[\sum_{i=1}^{n} \gamma_i x(p_i) = \text{constant}\]

\[
\begin{align*}
x[p1] & = 1 \\
x[p2] + x[p3] & = 1 \\
\end{align*}
\]
Example (end)

\[ \sum_{i=1}^{n} \gamma_i x(p_i) = \text{constant} \]

\[ x[p1] = 1 \]

\[ x[p2] + x[p3] = 1 \]
Example with $\infty$ Capacity

\[ [p_1, p_2, p_3, p_4] \]
\[ [1, 0, 1, \omega] \]
\[ [1, 1, 0, \omega] \]

\[ \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \end{bmatrix} \]

\[ \gamma_1 \gamma_2 \gamma_3 \gamma_4 \text{ C} \]

\[ \sum_{i=1}^{n} \gamma_i x(p_i) = \text{constant} \]
\[ x[p1] = 1 \]
\[ x[p2] + x[p3] = 1 \]
Problems/Constraints

- Not guaranteed to find solution
  - Integer linear programming problem
- Get less constraint equations than number of independent variables
  - Permute matrix columns & redo reduction
- Might not be unique solution
References


[2] lp_solve home page, 
http://elib.zib.de/pub/Packages/mathprog/linprog/lp-solve/

[3] Petri Net conservation analysis assignment, 
http://studwww.ugent.be/~dsooms/ftw/syssim/project2/

[4] Petri Net boundedness analysis assignment, 
http://moncs.cs.mcgill.ca/people/hv/teaching/MS/assignments/assignment2/
Demo