# Petri nets with small path property

M. Praveen and Kamal Lodaya

The Institute of Mathematical Sciences, Chennai

January 2009

# Petri nets - introduction

- Mathematical model.
- Widely used to model systems with concurrent processes.

## Example

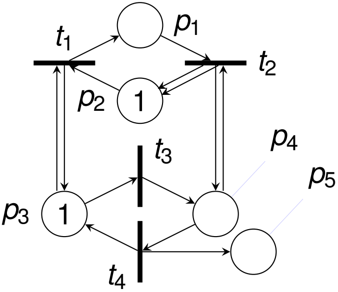



Figure: Hopcroft and Pansiot's example Petri net

# Petri nets - introduction

- Mathematical model.
- Widely used to model systems with concurrent processes.

## Example

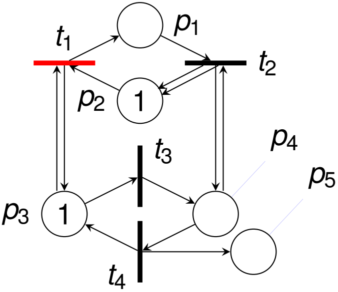



Figure: Hopcroft and Pansiot's example Petri net

# Petri nets - introduction

- Mathematical model.
- Widely used to model systems with concurrent processes.

## Example

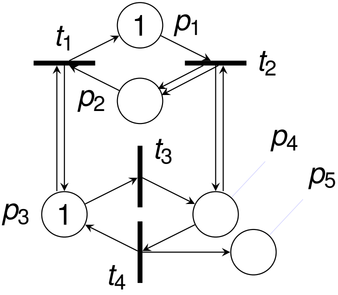



Figure: Hopcroft and Pansiot's example Petri net

# Petri nets - introduction

- Mathematical model.
- Widely used to model systems with concurrent processes.

## Example

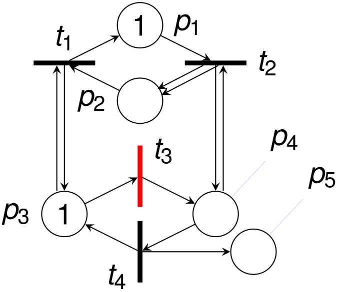



Figure: Hopcroft and Pansiot's example Petri net

# Petri nets - introduction

- Mathematical model.
- Widely used to model systems with concurrent processes.

## Example

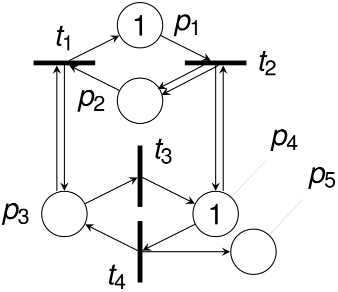



Figure: Hopcroft and Pansiot's example Petri net

# Petri nets - introduction

- Mathematical model.
- Widely used to model systems with concurrent processes.

## Example

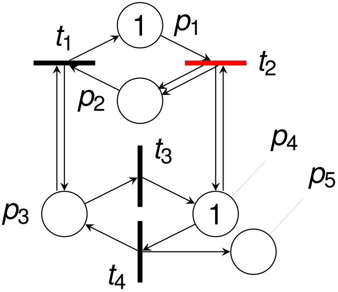



Figure: Hopcroft and Pansiot's example Petri net

# Petri nets - introduction

- Mathematical model.
- Widely used to model systems with concurrent processes.

## Example

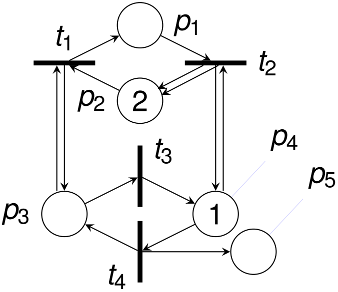



Figure: Hopcroft and Pansiot's example Petri net

# Petri nets - introduction

- Mathematical model.
- Widely used to model systems with concurrent processes.

Example

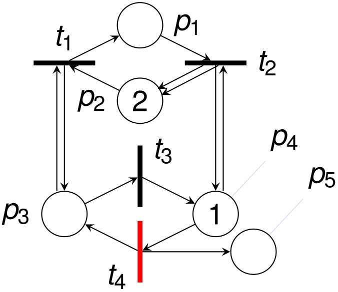



Figure: Hopcroft and Pansiot's example Petri net

# Petri nets - introduction

- Mathematical model.
- Widely used to model systems with concurrent processes.

## Example

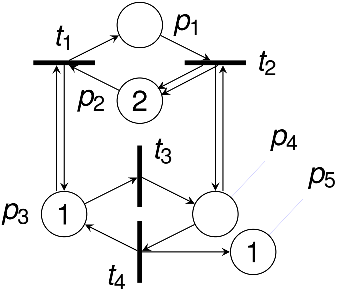



Figure: Hopcroft and Pansiot's example Petri net

# Petri nets - introduction

- Mathematical model.
- Widely used to model systems with concurrent processes.

## Example

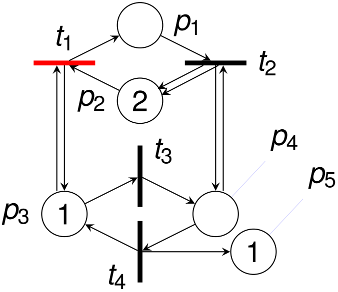



Figure: Hopcroft and Pansiot's example Petri net

# Petri nets - introduction

- Mathematical model.
- Widely used to model systems with concurrent processes.

## Example

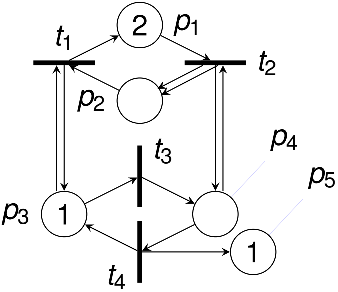



Figure: Hopcroft and Pansiot's example Petri net

# Petri nets - introduction

- Mathematical model.
- Widely used to model systems with concurrent processes.

## Example

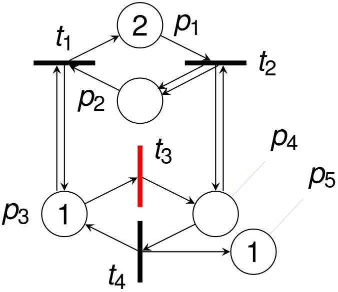



Figure: Hopcroft and Pansiot's example Petri net

# Petri nets - introduction

- Mathematical model.
- Widely used to model systems with concurrent processes.

## Example

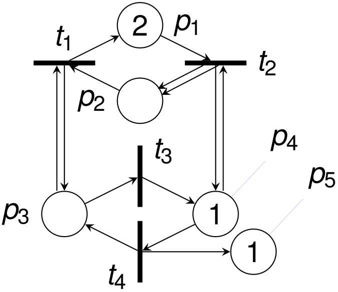



Figure: Hopcroft and Pansiot's example Petri net

# Petri nets - introduction

- Mathematical model.
- Widely used to model systems with concurrent processes.

## Example

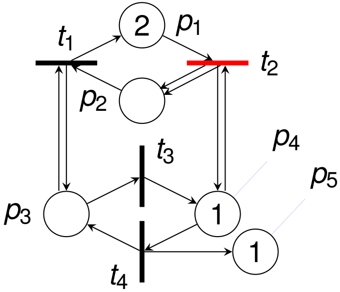



Figure: Hopcroft and Pansiot's example Petri net

# Petri nets - introduction

- Mathematical model.
- Widely used to model systems with concurrent processes.

## Example

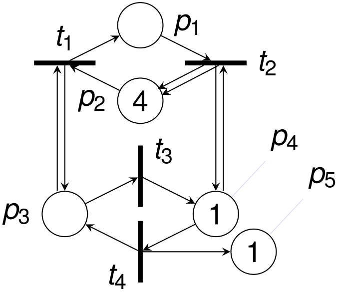



Figure: Hopcroft and Pansiot's example Petri net

# Petri nets - introduction

- Mathematical model.
- Widely used to model systems with concurrent processes.

## Example

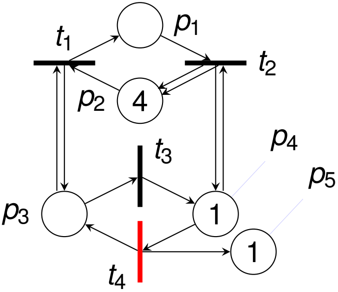



Figure: Hopcroft and Pansiot's example Petri net

# Petri nets - introduction

- Mathematical model.
- Widely used to model systems with concurrent processes.

## Example

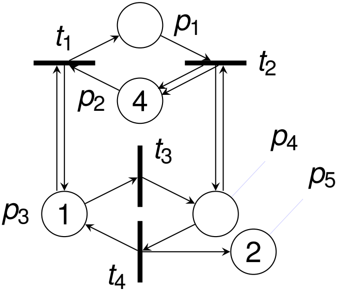
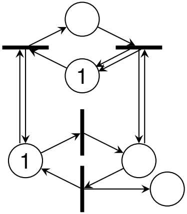


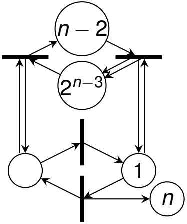

Figure: Hopcroft and Pansiot's example Petri net

# Reachability problem

Starting from




, can we reach




?

$$M_i = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

$$M_f = \begin{bmatrix} n-2 \\ 2^{n-3} \\ 0 \\ 1 \\ n \end{bmatrix}$$

# Previous work

- E.W.Mayr gave an algorithm for the general Petri net reachability problem in 1981/1984.
- S.R.Kosaraju and J.L.Lambert simplified the proofs in 1982 and 1992.
- Employing notions defined by Lambert, J.Leroux recently gave another approach for proving decidability.
- No upper bound known for above algorithms. They need non-primitive recursive space in the worst case.
- R.J.Lipton gave an exponential space lower bound for the general Petri net reachability problem.
- Exact complexity of the general reachability problem is not known.
- Better algorithms are known for Petri nets with special properties (1-safe nets, sinkless nets etc.).

# Formal definitions

- $N = (P, T, Pre, Post, M_i)$ is a Petri net system where
  - $P$ is the set of places,
  - $T$ is the set of transitions,
  - $Pre : (P \times T) \rightarrow \mathbb{N}$ and $Post : (P \times T) \rightarrow \mathbb{N}$ are the flow relations and
  - $M_i : P \rightarrow \mathbb{N}$ is the initial marking.
- A transition can be fired at marking $M$ provided there are enough tokens in all its input places. This firing results in the new marking $M'$: $M \xrightarrow{t} M'$.
  $M'(p) = M(p) - Pre(p, t) + Post(p, t)$ for all $p \in P$.
- A firing sequence $\sigma = t_1 \cdots t_r$ is enabled at marking $M_0$ if $M_0 \xrightarrow{t_1} M_1 \cdots M_{r-1} \xrightarrow{t_r} M_r$ and each $t_i$ is enabled at $M_{i-1}$.
- Given a Petri net system and a final marking $M_f$, the reachability problem is to determine if there exists a firing sequence $\sigma$ enabled at $M_i$ such that $M_i \xrightarrow{\sigma} M_f$.

# Definitions (Contd. . . )

- $\mathbf{N} = [c_{ij}]$ is the $|P| \times |T|$ incidence matrix where $c_{ij} = -Pre(p_i, t_j) + Post(p_i, t_j)$.
- For the firing sequence $\sigma$, its Parikh vector $\overline{\sigma}$ has as the $i^{\text{th}}$ component the number of times $t_i$ occurs in $\sigma$.
- A T-invariant $\mathbf{J}$ is an integral solution to $\mathbf{Nj} = \mathbf{0}$.

# $k$-safe nets

- Every place will have at most $k$ tokens in any reachable marking.
- If there are $m$ places, there are only $(k+1)^m$ distinct possible markings.
- Any reachable marking can be reached by a firing sequence of length at most $(k+1)^m$.
- A non-deterministic algorithm will take polynomial space to guess a firing sequence and verify that it reaches the final marking.
- Lower bound given by Cheng, Esparza and Palsberg (1993): reduction from QBF-SAT to reachability in 1-safe nets.

# Measuring progress

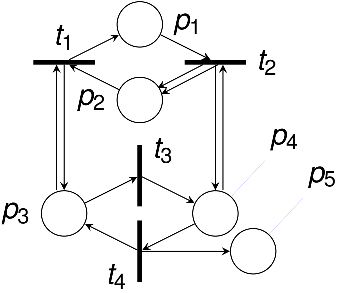



Figure: Hopcroft and Pansiot's example net

$$3x_1 \qquad + \qquad 2x_2 \qquad + \qquad x_4 \qquad + \qquad 2x_5$$

# Measuring progress

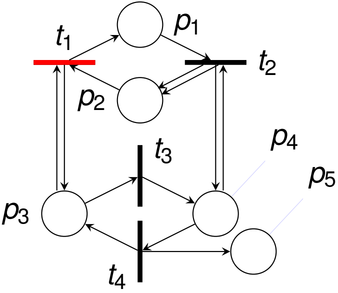



Figure: Hopcroft and Pansiot's example net

$$\begin{array}{ccccccc} 3x_1 & + & 2x_2 & + & x_4 & + & 2x_5 \\ 3(x_1+1) & + & 2(x_2-1) & + & x_4 & + & 2x_5 \end{array}$$

# Measuring progress

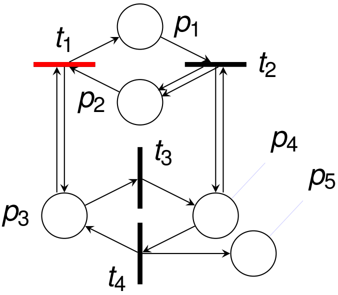



Figure: Hopcroft and Pansiot's example net

$$
\begin{array}{ccccccc}
3x_1 & + & 2x_2 & + & x_4 & + & 2x_5 \\
3(x_1+1) & + & 2(x_2-1) & + & x_4 & + & 2x_5 \\
\hline
+3 & & -2 & & & &
\end{array}
$$

# Measuring progress

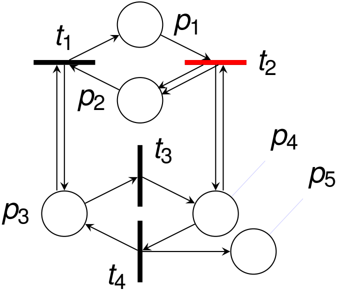



Figure: Hopcroft and Pansiot's example net

$$
\begin{array}{ccccccccc}
3x_1 & + & 2x_2 & + & x_4 & + & 2x_5 \\
3(x_1 - 1) & + & 2(x_2 + 2) & + & x_4 & + & x_5 \\
\hline
-3 & & +4 & & & &
\end{array}
$$

# Measuring progress

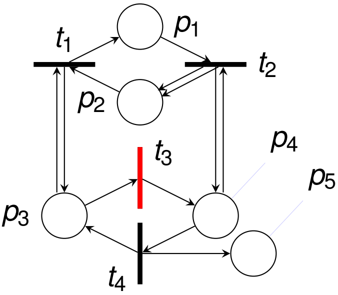



Figure: Hopcroft and Pansiot's example net

$$
\begin{array}{ccccccc}
3x_1 & + & 2x_2 & + & x_4 & + & 2x_5 \\
3x_1 & + & 2x_2 & + & x_4 + 1 & + & 2x_5 \\
\hline
 & & & & +1 & &
\end{array}
$$

# Measuring progress

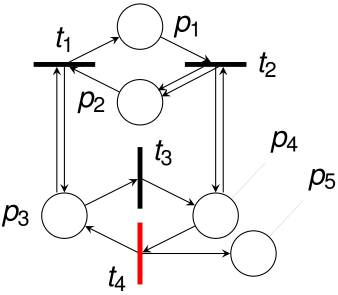



Figure: Hopcroft and Pansiot's example net

| | | | | | | |
|---|---|---|---|---|---|---|
| $3x_1$ | $+$ | $2x_2$ | $+$ | $x_4$ | $+$ | $2x_5$ |
| $3x_1$ | $+$ | $2x_2$ | $+$ | $x_4 - 1$ | $+$ | $2(x_5 + 1)$ |
| | | | | $-1$ | | $+2$ |

# Measuring progress

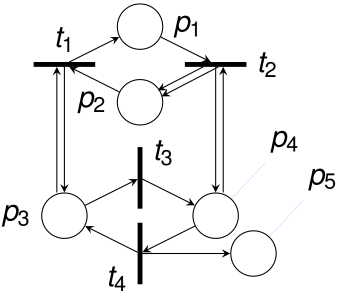



Figure: Hopcroft and Pansiot's example net

- $3x_1 + 2x_2 + x_4 + 2x_5$ is an S-variant for this net.
- If starting at marking $(i_1, i_2, i_3, i_4, i_5)$, a firing sequence $\sigma$ reaches the marking $(f_1, f_2, f_3, f_4, f_5)$, length of $\sigma$ is at most $3f_1 + 2f_2 + f_4 + 2f_5 - 3i_1 - 2i_2 - i_4 - 2i_5$.

# Measuring progress using S-variants

- An S-variant **V** is an integral solution to the inequality $\mathbf{N}^T\mathbf{v} \geq \mathbf{1}$.
- The *value* of S-variant **V** at a marking $M$ is $\mathbf{V}(M) = \sum_{p \in P} \mathbf{V}(p)M(p)$.
- $M \xrightarrow{t} M' \Rightarrow \mathbf{V}(M') \geq \mathbf{V}(M) + 1$.
- The existence of an S-variant is equivalent to finding rational solutions to $\mathbf{N}^T\mathbf{v} \geq \mathbf{1}$ and can be checked in polynomial time.
- An application of the Farkas lemma shows that S-variants exist for a net iff it doesn't have semi-positive T-invariants. Such *T-invariant-less* nets were studied by Kostin (2000).

# Using progress measure in reachability algorithm

- If S-variants exist for a net and **V** is an S-variant, $\mathbf{V}(M_f) - \mathbf{V}(M_0)$ is an upper bound on length of $\sigma$ where $M_0 \xrightarrow{\sigma} M_f$.
- Values in **V** can be bounded by **N** using bounds on solutions of linear Diophantine equations.
- Length of firing sequences bounded by input size. A non-deterministic algorithm takes polynomial space to guess a firing sequence and verify it.
- The 1-safe net to which QBF-SAT is reduced to also happens to have S-variants. So, the reachability problem for this subclass of Petri nets is PSPACE -complete.

# Partial S-variants

- Partial S-variants are those whose value increases strictly for *certain* transitions and doesn't change for others.
- Another application of Farkas lemma shows that the "certain" transitions are exactly those that are not part of any semi-positive T-invariant.

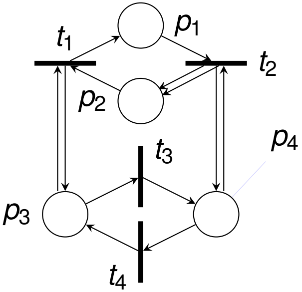



Figure: Example of partial S-variants

# Structurally partially bounded nets

- Structurally partially bounded nets are those satisfying the following (polytime-checkable) property:
  *If all progressive transitions are removed, what remains is a structurally bounded net.*

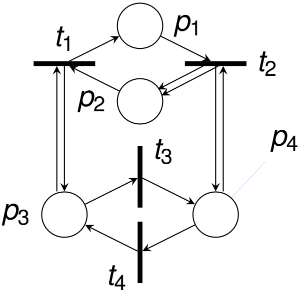



Figure: Example of a structurally partially bounded net

# Structurally partially bounded nets

- Suppose $M_i \xrightarrow{\sigma} M_f$ in a structurally partially bounded net.
- The number of progressive transitions occurring in $\sigma$ can be bounded using partial S-variant.
- The number of other transitions is also bounded since they form a structurally bounded net.
- Total number of transition occurrences in $\sigma$ is again bounded by input size, resulting in a PSPACE reachability algorithm.

# Summary

- We looked at $k$-safe Petri nets where all reachable markings are reachable via “short” firing sequences.
- With S-variants, we saw that we can analyze Petri nets where length of firing sequences are bounded by size of the net, initial and final markings.
- Above two properties ensure that if a marking is reachable, it is reachable by a firing sequence whose length is at most some exponential function of the input size.
- Is it possible to combine the above two classes to form a subclass that is bigger than the union of these two subclasses and has the small paths property?

Thank you.

Questions?