Time Petri Nets

Miriam Zia
School of Computer Science
McGill University
Timing Specifications

Why is time introduced in Petri nets?
- To model interaction between activities taking into account their start and end times.
Time Associated with Tokens

- Each token is associated with a time-stamp $\theta$ that indicates when the token is available to fire a transition.
Time Associated with Arcs

- Each arc is associated with a traveling delay $t$.
- Tokens are available for firing only when they reach the transition.
Time Associated with Places

- Timed Place Petri Nets (TPPN)
  - Each place $p$ is associated with a delay attribute, say $t$.
  - Tokens generated in $p$ only become available to fire a transition after the delay $t$ has elapsed.

![Diagram showing Timed Place Petri Nets](image-url)
Time Associated with Transitions

- Timed Transition Petri Net (TTPN)
  - Each transition represents an activity.
    - Transition Enabling: start of activity.
    - Transition Firing: end of activity.
  - Two basic PN-based models were developed for handling time.
Ramchandani’s Timed PN \[\text{[Ram74]}\]

- A firing duration $t$ is associated with each transition of a PN.
- Firing rule:
  - Transitions are fired as soon as they are enabled.
  - Transitions take time $t$ to fire.
- Used mainly for performance evaluation.

Miriam Zia  
mzia2@cs.mcgill.ca  
Modeling and Simulation-based design: Time Petri Nets
Merlin’s Time PN [Mer74] (1/2)

- More general than Timed PN.
- TPN used to investigate recoverability problems in computer systems and in communications protocols.
- Two real numbers $a, b$ are associated with each transition of a PN, with $0 \leq a \leq b \leq \infty$.
  - $a$: time that must elapse between the ENABLING and the FIRING of a transition.
  - $b$: maximum time during which transition can be enabled without being fired.
Assume $t_1$ has been enabled at time $r$:
- $t_1$ cannot fire before time $r+a$.
- $t_1$ must fire before or at time $r+b$.

Times $a$ and $b$ for transition $t_1$ are relative to the moment at which transition $t_1$ is enabled.
Research conducted at the LAAS of CNRS, Toulouse, France.

Motivation: Specifying and proving correctness of time-dependent systems.

Research:
- Propose for TPN a technique for modeling the behaviour and analyzing the properties of timed systems.
  - Similar to the reachability analysis for PN.
- Develop a software tool for analyzing TPN.
  - Time Petri Net Analyzer (TINA)
Two main papers:

Outline of Paper Presentation

1. Time Petri nets.
   - States in a TPN.
   - Enabledness and firability condition of a set of transitions.
   - Firing rule between states.
   - Behaviour of TPN.
2. Method for analyzing TPN.
   - State classes.
   - Firing rule between state classes.
   - Reachability tree.
4. TINA : TIme petri Net Analyzer.
Time Petri Net is a Tuple (1/2)

\[ TPN = \langle P, T, B, F, M_0, \text{SIM} \rangle \]

\[ P: \text{finite nonempty set of places;} \]

\[ T: \text{finite nonempty set of transitions } t_i \text{ can be viewed as an ordered set } \{t_1, t_2, \ldots, t_i, \ldots, \}; \]

\[ B: \text{backward incidence function} \]
\[ B: T \times P \rightarrow N \text{ (where } N \text{ is the set of nonnegative integers);} \]

\[ F: \text{forward incidence function} \]
\[ F: T \times P \rightarrow N; \]

\[ M_0: \text{initial marking function} \]
\[ M_0: P \rightarrow N; \]
Time Petri Net is a Tuple (2/2)

TPN = \langle P, T, B, F, M_0, \text{SIM} \rangle

\textbf{SIM:} static interval mapping

SIM: T \to Q^* \times (Q^* \cup \infty) (where N is the set of positive rational numbers)

- A static interval is associated with transitions:
  \text{SIM}(t_i) = (\alpha_i^s, \beta_i^s)

- \alpha_i^s, \beta_i^s are rationals such that:
  \begin{align*}
  0 &\leq \alpha_i^s \leq \beta_i^s \leq \infty
  \end{align*}

- \((\alpha_i^s, \beta_i^s)\) is called the static firing interval of transition \(t_i\).

- Left bound \(\alpha_i^s\) is the static Earliest Firing Time (static EFT) for \(t_i\).

- Right bound \(\beta_i^s\) is the static Latest Firing Time (static LFT) for \(t_i\).
A Couple of Comments

- Times $\alpha_i^s$ and $\beta_i^s$ are relative to the moment at which $t_i$ is enabled.
- If a pair $(\alpha_i^s, \beta_i^s)$ is not defined for $t_i$, it has the pair $(0, \infty)$ – classic PN transition.
- In [BM91]: TPNs considered are such that none of their transitions may become enabled more than once “simultaneously” by any marking $M$: for any enable transition $t_i$ $(\exists p)(M(p) < 2 \cdot B(t_i, p))$ there is at least 1 place which prevents $t_i$ from being firable twice.
States in a TPN Are a Pair (1/2)

- $S = (M, I)$ consisting of:
  - A marking $M$.
  - A Firing Interval vector $I$
    - Associates with each transition enabled by $M$ the time interval in which the transition is allowed to fire.
States in a TPN Are a Pair (2/2)

- $S_0 = (M_0, I_0)$, with
  - $M_0: p_1(1), p_2(2)$
  - $I_0: \{(4,9)\}$

- $S_1 = (M_1, I_1)$, with
  - $M_1: p_3(1), p_4(1), p_5(1)$
  - $I_1: \{(0,2),(1,3),(0,2),(0,3)\}$

- $S_2 = (M_2, I_2)$, with
  - $M_2: p_2(1), p_3(1), p_5(1)$
  - $I_2: \{(1,3),(0,2),(0,3)\}$

if transition $t_2$ fires
Enabledness Condition of a Set of Transitions

- Transition $t_i$ becomes enabled at time $r$ in state $S = (M,I)$ in the usual PN sense:
  \[ M(p) \geq B(t_i,p) \text{ for all } p \text{ in the incident set } I(t_i) \]
Firability Condition of a Set of Transitions

Formally expressed by 2 conditions:

- **Condition 1**: \( t_i \) is enabled by marking \( M \) at time \( r \) (absolute enabling time).
- **Condition 2**: the relative firing time \( \theta \) (relative to \( r \)) is not smaller than the EFT of \( t_i \) and not greater than the smallest of the LFTs of all the transitions enabled by \( M \):
  \[
  \text{EFT of } t_i \leq \theta \leq \min\{\text{LFT of } t_k\} \quad \text{(where } k \text{ ranges over the set transitions enabled by } M)\.
  \]
Firing Rule Between States (1/2)

- State $S' = (M', I')$ can be reached by firing $t_i$ at relative time $\theta$ from state $S = (M, I)$.
- $S'$ is computed in 2 steps:
  - $M'$ is computed, for all places $p$, as:
    $$(\text{for all } p)M'(p) = M(p) - B(t_i, p) + F(t_i, p)$$
  - $I'$ is computed in 3 steps:
    - Remove from $I$ those intervals disabled when $t_i$ is fired.
    - Shift by $\theta$ towards the origin of times all intervals of $I$ that remained enabled; time is always nonnegative: $I' = (\max(0, \text{EFT}_k - \theta), \text{LFT}_k - \theta)$
    - Introduce in $I'$ the static intervals of the new transitions enabled.
Firing Rule Between States (2/2)

- $S_0 = (M_0, I_0)$, with
  - $M_0$: $p_1(1), p_2(2)$
  - $I_0$: $\{(4,9)\}$
- $t_1$ fires at $\theta_1$
  - $S_1 = (M_1, I_1)$, with
    - $M_1$: $p_3(1), p_4(1), p_5(1)$
    - $I_1$: $\{(0,2), (1,3), (0,2), (0,3)\}$
- If $t_2$ fires at $\theta_2$
  - $S_2 = (M_2, I_2)$, with
    - $M_2$: $p_2(1), p_3(1), p_5(1)$
    - $I_2$: $(\max(0, 1 - \theta_2), 3 - \theta_2)$, $(0, 2 - \theta_2)$, $(0, 3 - \theta_2)$
Behaviour of a TPN (1/2)

- “transition $t_i$ is firable from state $S$ at time $\theta$ and its firing leads to state $S'$”

$S \xrightarrow{(t_i, \theta_i)} S'$

- A firing schedule will be a sequence of pairs (transition $t$, relative time $\theta$):
  - $(t_i, \theta_1) \cdot (t_2, \theta_2) \cdot \ldots \cdot (t_n, \theta_n)$
  - This schedule is feasible from a state $S$ iff there exist states $S_1, S_2, \ldots, S_n$ such that:

$S \xrightarrow{(t_1, \theta_1)} S_1 \xrightarrow{(t_2, \theta_2)} S_2 \ldots \ldots \xrightarrow{(t_n, \theta_n)} S_n$
Behaviour of a TPN (2/2)

- The firing rule permits one to compute states and a reachability relation among them.
- The set of states that are reachable from the initial state, through a firing sequence $\omega$, characterize the behaviour of the TPN.
  - Much like with reachable markings in PN.
- Problem: firing sequences can be defined but enumerating this set of states is not possible.
  - Why? Because there are infinite time values which can be selected to fire a transition from a given marking.
Recap:
A state is a set of all possible firing intervals, defined as the product set of the firing intervals of the transitions enabled by M.

Now we consider the following:
The set of all states reached from the initial state by firing all feasible firing values corresponding to the same firing sequence \( \omega \).
This set will be called the state class associated with the firing sequence \( \omega \).
Class $C = (M,D)$, associated with a firing sequence $\omega$ from the initial state, consisting of
- A marking $M$ of the class: all states in the class have the same marking.
- A firing domain $D$ of the class
  - Finitely represents the infinite number of firing domains of states possible from a marking $M$ by firing schedules with firing sequence $\omega$.
  - $D$ may be expressed as the solution set of some system of linear inequalities:
    $$ D = \{ t \mid A \cdot t \geq b \} $$
    where $A$ a matrix, $b$ is a vector of constants, and variable $t_i$ corresponds to the $i^{th}$ transition enabled by $M$.

*Note:* $t$ is an ordered set, and $t(i)$ will refer to the $i^{th}$ enabled transition.
Enabledness of Transitions from Classes

- Assuming \( t(i) \) is the \( i^{th} \) transition enabled by marking \( M \), \( t(i) \) becomes enabled if:

\[
M(p) \geq B(t(i), p) \quad \text{for all } p \text{ in the incident set } I(t(i))
\]
Firability of Transitions from Classes

Transition \( t(i) \) is firable from class \( C = (M,D) \) iff:

- Condition 1: \( t(i) \) is enabled by marking \( M \).
- Condition 2: the firing interval related to transition \( t(i) \) must satisfy the following augmented system of inequalities:

\[
A \cdot t \geq b
\]

\[
t(i) \leq t(j) \quad \text{for all } j, j \neq i \quad (\text{where } t(j) \text{ also denotes the firing interval related to the } j^{\text{th}} \text{ component of vector } t)
\]
C0 = (M0,D0), with
M0: p1(1),p2(2)
D0: Solution set of
4 ≤ θ₁ ≤ 9

t1 fires at θ₁
C1 = (M1,D1), with
M1: p3(1),p4(1),p5(1)
D1: Solution set of
0 ≤ θ₂ ≤ 2
1 ≤ θ₃ ≤ 3
0 ≤ θ₄ ≤ 2
0 ≤ θ₅ ≤ 3

Simple case: When firing t1, no transition already enabled remained enabled after the firing.
A complex case occurs when some transitions remain enabled. 

- $t_2$ can fire from time $\theta=0$ to $\theta=\theta_{\text{max}}$, e.g.: $t_2$ can fire at any $\theta_2$ in the interval $0 \leq \theta_2 \leq 2$.
- Firing $t_2$ is possible if the following system has a solution:
  
  \[
  \begin{align*}
  0 & \leq \theta_2 \leq 2 \\
  1 & \leq \theta_3 \leq 3 \\
  0 & \leq \theta_4 \leq 2 \\
  0 & \leq \theta_5 \leq 3 \\
  \theta_2 & \leq \theta_3 \\
  \theta_2 & \leq \theta_4 \\
  \theta_2 & \leq \theta_5 
  \end{align*}
  \]

- Computation of all possible firing times for transitions can be handled by an adequate change of variables: $\theta_{2F}$ denotes the relative time at which $t_2$ is fired.
- After the firing of $t_2$, transitions $t_3$, $t_4$, $t_5$ remain enabled while a time $\theta_{2F}$ has elapsed. Their new time values $\theta'_3$, $\theta'_4$, $\theta'_5$ can be defined by $\theta_i = \theta'_i + \theta_{2F}$.
- Firing $t_2$ is possible if the following system has a solution:
  
  \[
  \begin{align*}
  1 & \leq \theta'_3 + \theta_{2F} \leq 3 \\
  0 & \leq \theta'_4 + \theta_{2F} \leq 2 \\
  0 & \leq \theta'_5 + \theta_{2F} \leq 3 
  \end{align*}
  \]
State Classes of a TPN (2/2)

or

\[ 1 - \theta_2 F \leq \theta_3^\prime \leq 3 - \theta_2 F \]  
\[ 0 - \theta_2 F \leq \theta_4^\prime \leq 2 - \theta_2 F \]  
\[ 0 - \theta_2 F \leq \theta_5^\prime \leq 3 - \theta_2 F \]

with

\[ 0 \leq \theta_2 F \leq 2 \]

(8), (9) and (10) can be rewritten:

\[ 1 - \theta_3^\prime \leq \theta_2 F \leq 3 - \theta_3^\prime \]  
\[ 0 - \theta_4^\prime \leq \theta_2 F \leq 2 - \theta_4^\prime \]  
\[ 0 - \theta_5^\prime \leq \theta_2 F \leq 3 - \theta_5^\prime \]

Eliminating \( \theta_2 F \) gives:

\[ 0 \leq \theta_3^\prime \leq 3 \] from (11) and (14)
\[ 0 \leq \theta_4^\prime \leq 2 \] from (12) and (14)
\[ 0 \leq \theta_5^\prime \leq 3 \] from (13) and (14)
\[ \theta_3^\prime - \theta_4^\prime \leq 3 \] from (15), (16) and (17)
\[ \theta_3^\prime - \theta_5^\prime \leq 3 \] from (15), (16) and (17)
\[ \theta_4^\prime - \theta_3^\prime \leq 1 \] from (15), (16) and (17)
\[ \theta_4^\prime - \theta_5^\prime \leq 2 \] from (15), (16) and (17)
\[ \theta_5^\prime - \theta_3^\prime \leq 2 \] from (15), (16) and (17)
\[ \theta_5^\prime - \theta_4^\prime \leq 3 \] from (15), (16) and (17)

The state class reached after firing \( t2 \) is:

\[ C2 = (M2, D2), \] with:

\[ M2: \text{p2(1),p3(1),p5(1)} \] and \[ D2: \text{solution set to the inequalities defined above.} \]
Firing Rule Between State Classes

- Class $C' = (M', D')$ can be reached by firing $t(f)$ from class $C = (M, D)$.
- $C'$ is computed in 2 steps:
  - $M'$ is computed, for all places $p$, as:
    \[(\text{for all } p)M'(p) = M(p) - B(t_i, p) + F(t_i, p)\]
  - $D'$ is computed in 3 steps:
    - Add to the system $A \cdot t \geq b$ the firability condition for $t(f)$, leading to the augmented system:
      $A \cdot t \geq b ; t(f) \leq t(j)$ for all $j, j \neq f$
      Make the change of variable: $t(j) = t(f) + t''(j)$ and eliminate from the system the variable $t(f)$.
    - Remove from the system obtained above all variables corresponding to transitions disabled when $t(f)$ is fired.
    - Augment the system with new variables associated with each new transition enabled. These variables belong to their static firing interval.
Formal Definition of D

- The firing domains \( D \) of state classes for a T-Safe TPN can be expressed as solution sets of systems of inequalities of the following form:
  \[
  \alpha_i \leq t(i) \leq \beta_i \quad \text{for all } i \\
  t(j) - t(k) \leq \gamma_{jk} \quad \text{for all } j, k \neq j
  \]
Reachability Tree (1/2)

- Using the firing rule, a tree of classes can be built.
  - The root is the initial class C, and there is an arc labelled $t_i$ from C to C' if $t_i$ is firable from class C, and if its firing leads to C'.
  - Each class will have a finite number of successors, at most one for each transition enabled by the marking of the class.
  - Any sequence of transitions firable in the TPN will be a path in this tree.
A finite graph will be associated to the TPN when the tree of classes will have a bounded number of distinct nodes.

- The graph is obtained by grouping equal classes of the tree into the same class.
  - Two classes are defined to be equal if their markings are equal and their firing domains are equal.
  - A method to achieve this is to define the domains into some canonical form, and then compare these forms.

- This will be called the reachability graph of the TPN.
Some Properties of TPN (1/2)

- The set of markings a TPN can reach from its initial marking $M_0$ is denoted $R(M_0)$.
- The **reachability** problem is whether or not a given marking belongs to $R(M_0)$.
- The **boundedness** problem is whether or not all markings in $R(M_0)$ are bounded:
  - For all markings in $R(M_0)$ and for all places in $P$: $M(p) \leq k$, for some $k$ in $\mathbb{N}$
A TPN is said **T-bounded** if there exists a natural number $k$ s.t. none of its transitions may be enabled more than $k$ times simultaneously by any reachable marking.

- for all $t_i$ in $T$ there exists $p$ in $P$ such that: $M(p) < (k+1) \cdot B(t_i,p)$
- When $k = 1$, the TPN is said to be **T-safe**.

The reachability and boundedness problems for TPNs are undecidable.
So, What Do We Have Here?

- An approach for analyzing TPNs:
  - Permits one to check the properties of systems in the presence of timing specifications.
Possible Extensions

- No necessary or sufficient condition can be stated for the boundedness property
  - Must develop strong conditions!
- More specific and semantic checks could be developed
  - We could stop enumeration early on if the behaviour is not as expected.
- Develop alternative analysis techniques.
TINA

- Experimental toolbox for editing and analyzing PNs and TPNs.
  - **tina:**
    - Builds various state space abstractions for PN and TPN: reachability and coverability graphs (Karp & Miller technique), and efficiently checks the boundedness property.
    - Builds a linear state class graph of a TPN (Berthomieu & Menasche technique).
    - Takes as input descriptions of PN/TPN in textual or graphical form.
  - **struct:**
    - Computes generator sets for semi-flows and flows.
    - Determines the invariance and consistence properties.
  - **nd (NetDraw):**
    - PN, TPN and Automata editor.
    - Allows one to create TPN in graphical or textual form.
    - Interfaced with the above tools.

Miriam Zia  
mzia2@cs.mcgill.ca  
Modeling and Simulation-based design: Time Petri Nets
TINA is not a Model-Checker

- It can’t be used to check satisfaction of a concrete property (except reachability properties): no design verification performed.
- It can be used as a front-end for a model-checker.
  - It provides a reduced state space on which the properties can be checked more efficiently than on the original state space.
What Do I Intend to do with TPN?

Traffic MM

Traffic

Timed Traffic MM

Timed Traffic

TPN MM

TPN

Textual

Input

TINA

Miriam Zia

mzia2@cs.mcgill.ca

Modeling and Simulation-based design: Time Petri Nets
References (1)

  
  link: [http://ieeexplore.ieee.org/xpl/tocresult.jsp?isNumber=2506&puNumber=32

  
  link: [http://www.laas.fr/tina/papers.php

  
  link: [www.laas.fr/~poribet/PUBLICATIONS/ribet_2004_ijpr.ps

Miriam Zia
mzia2@cs.mcgill.ca
Modeling and Simulation-based design: Time Petri Nets
References (2)

link: http://home.postech.ac.kr/~mkjang/Resources/TimePetriIntro.pdf
