Symbolic Unfolding of Time Petri Nets

Didier Lime

École Centrale de Nantes – IRCCyN

Réunion AFSEC Concurrence
Cachan
19 November 2013
Petri Nets, Sequentiality, and Concurrency

\[ \{p_1, p_2\} \]
Petri Nets, Sequentiality, and Concurrency

Preserve the structural relations between events

- run \rightarrow process;
- set of runs \rightarrow branching processes;
- marking graph \rightarrow unfolding.

Didier Lime
Petri Nets, Sequentiality, and Concurrency

Preserve the structural relations between events:

- run → process;
- set of runs → branching processes;
- marking graph → unfolding.

Didier Lime
Symbolic Unfoldings of Time Petri Nets
19/11/13
Preserve the structural relations between events ▶ run → process; ▶ set of runs → branching processes; ▶ marking graph → unfolding.
Preserve the structural relations between events

- run $\rightarrow$ process;
- set of runs $\rightarrow$ branching processes;
- marking graph $\rightarrow$ unfolding.
Petri Nets, Sequentiality, and Concurrency

Preserve the structural relations between events

\[ t_0 \rightarrow \text{process; } \quad \text{set of runs} \rightarrow \text{branching processes; } \quad \text{marking graph} \rightarrow \text{unfolding.} \]
Preserve the structural relations between events
▶ run → process;
▶ set of runs → branching processes;
▶ marking graph → unfolding.

Petri Nets, Sequentiality, and Concurrency
Petri Nets, Sequentiality, and Concurrency

- Are $t_1$ and $t_2$ concurrent?
- Can we avoid the *enumeration* of interleavings.
Petri Nets, Sequentiality, and Concurrency

- Are $t_1$ and $t_2$ concurrent?
- Can we avoid the enumeration of interleavings.

Preserve the structural relations between events

- run $\rightarrow$ process;
- set of runs $\rightarrow$ branching processes;
- marking graph $\rightarrow$ unfolding.
Processes and Branching Processes [McM92]

Definition (Cut-off event)

\( e \) is a cut-off event if there exists \( e' \) s.t.:

1. The marking created by the causal past of \( e \) and \( e' \) is the same.
2. The causal past of \( e' \) is less than that of \( e \) (adequate order).
Processes and Branching Processes [McM92]

Definition (Cut-off event)

\( e \) is a cut-off event if there exists \( e' \) s.t.:

- The marking created by the causal past of \( e \) and \( e' \) is the same.
- The causal past of \( e' \) is less than that of \( e \) (adequate order).
Processes and Branching Processes [McM92]

Definition (Cut-off event)

e is a cut-off event if there exists e′ s.t.:
▶ The marking created by the causal past of e and e′ is the same.
▶ The causal past of e′ is less than that of e (adequate order).
Processes and Branching Processes [McM92]

Definition (Cut-off event) $e$ is a cut-off event if there exists $e'$ such that:

1. The marking created by the causal past of $e$ and $e'$ is the same.
2. The causal past of $e'$ is less than that of $e$ (adequate order).

Petri Nets and Unfoldings
Processes and Branching Processes [McM92]

Definition (Cut-off event)

\(e\) is a cut-off event if there exists \(e'\) s.t.:

- The marking created by the causal past of \(e\) and \(e'\) is the same.
- The causal past of \(e'\) is less than that of \(e\) (adequate order).

Didier Lime
Processes and Branching Processes [McM92]

Definition (Cut-off event)

\( e \) is a cut-off event if there exists \( e' \) s.t.:

1. The marking created by the causal past of \( e \) and \( e' \) is the same.
2. The causal past of \( e' \) is less than that of \( e \) (adequate order).
Processes and Branching Processes [McM92]

Definition (Cut-off event)

\( e \) is a cut-off event if there exists \( e' \) s.t.:

1. The marking created by the causal past of \( e \) and \( e' \) is the same.
2. The causal past of \( e' \) is less than that of \( e \) (adequate order).
Processes and Branching Processes [McM92]

The unfolding is the union of all branching processes.
The exists of a finite complete prefix
**Processes and Branching Processes [McM92]**

**Definition (Cut-off event)**

*e* is a **cut-off** event if there exists *e'* s.t.:

- The marking created by the **causal past** of *e* and *e'* is the same.
- The causal past of *e'* is less than that of *e* (**adequate order**);
Time Processes [AL00]
Time Processes [AL00]

\[ t_1[0; \infty[ \quad t_0[0; 0] \quad t_2[1; 2] \]

\[ t_3[2; 2] \]

\[ e_1; 0 \quad t_1 \]

\[ p_1 \quad p_2 \quad p_3 \quad p_4 \quad p_5 \]
Time Processes [AL00]

- $t_1[0; \infty[$
- $t_0[0; 0]$
- $t_2[1; 2]$
- $t_3[2; 2]$

- $p_1$
- $p_2$
- $p_3$
- $p_4$
- $p_5$

- $e_1; 0$
- $e_2; 2$

Time branching process?
Time Processes [AL00]
Time Processes [AL00]
Time Processes [AL00]

\[ t_1[0; \infty[, \quad t_0[0; 0[, \quad t_2[1; 2]] \]

\[ t_3[2; 2], \quad e_1; 0, \quad e_2; 2, \quad e_3; 2, \quad e_4; 2 \]
Time Processes [AL00]

Time branching process?
Time Branching Process

The symbolic unfolding is the union of all valid time branching processes.

\[
\begin{align*}
\theta(e_1) & = 0, \\
\theta(e_2) & = 2, \\
\theta(e_3) & = 2, \\
\theta(e_4) & = +\infty.
\end{align*}
\]
Time Branching Process

\[ t_1[0; \infty] \]

\[ t_0[0; 0] \]

\[ t_2[1; 2] \]

\[ e_1; 8.1 \]

\[ t_3[2; 2] \]

\[ t_5 \]

\[ p_1 \]

\[ p_2 \]

\[ p_3 \]

\[ p_4 \]

\[ p_5 \]

The symbolic unfolding is the union of all valid time branching processes.

\[ \theta(e_1) = 0, \theta(e_2) = 2, \theta(e_3) = 2, \theta(e_4) = +\infty \]

\[ 0 \leq \theta(e_1), 1 \leq \theta(e_2) \leq 2, \theta(e_3) = +\infty, \theta(e_4) = \max(\theta(e_1), \theta(e_2)) \]
Time Branching Process

The symbolic unfolding is the union of all valid time branching processes.

\[ \begin{align*}
\theta(e_1) &= 0, \\
\theta(e_2) &= 2, \\
\theta(e_3) &= 2, \\
\theta(e_4) &= +\infty.
\end{align*} \]
The symbolic unfolding is the union of all valid time branching processes.

\[
\begin{align*}
\theta(e_1) &= 0, \\
\theta(e_2) &= 2, \\
\theta(e_3) &= 2, \\
\theta(e_4) &= +\infty,
\end{align*}
\]
The symbolic unfolding is the union of all valid time branching processes.

\[
\begin{align*}
\theta(e_1) &= 0, \\
\theta(e_2) &= 2, \\
\theta(e_3) &= 2, \\
\theta(e_4) &= +\infty.
\end{align*}
\]
Time Petri Nets

Time Branching Process

$t_1[0; \infty]$  
$t_0[0; 0]$  
$t_2[1; 2]$  
$t_3[2; 2]$  
$t_4[0; \infty]$  

$p_1 \rightarrow p_2$  
$p_3 \rightarrow p_4$  
$p_5 \rightarrow p_1, p_2$  

Didier Lime
Time Branching Process

\[ t_1[0; \infty[ \quad t_0[0; 0] \quad t_2[1; 2] \quad e_1; \theta(e_1) \quad t_3[2; 2] \quad e_3; \theta(e_3) \quad e_4; \theta(e_4) \]

\[ p_1 \quad p_2 \quad p_3 \quad p_4 \quad p_5 \quad p_1 \quad p_2 \quad p_3 \quad p_4 \quad p_5 \quad p_1 \quad p_2 \]

The symbolic unfolding is the union of all valid time branching processes.

\begin{align*}
\theta(e_1) &= 0 \\
\theta(e_2) &= 2 \\
\theta(e_3) &= 2 \\
\theta(e_4) &= +\infty \\
\end{align*}
the symbolic unfolding is the union of all valid time branching processes.

\[
\begin{align*}
\theta(e_1) &= 0, \\
\theta(e_2) &= 2, \\
\theta(e_3) &= 2, \\
\theta(e_4) &= +\infty
\end{align*}
\]
Locality

Conflicts and urgency break the locality of the firing rule.

[Diagram of a Petri net with places labeled p1, p2, p3, p4, and p5, and transitions labeled t1, t0, t2, and t3.]
Conflicts and urgency break the locality of the firing rule.
Locality: A solution from Chatain and Jard [CJ06]

Add read arcs to the unfolding.

\[
\begin{align*}
0 & \leq \theta(e_1), \\
1 & \leq \theta(e_2) \leq 2, \\
\theta(e_3) & = 2, \\
\theta(e'_3) & = \theta(e_1) + 2, \\
\theta(e_4) & = \max(\theta(e_1), \theta(e_2))
\end{align*}
\]
Locality: our approach – structural analysis of conflicts

\[ e_1 \text{ and } e_2 \text{ are in direct conflict } (e_1 \text{conf} e_2) \text{ iff they share a precondition and all their preconditions are concurrent.} \]
conf-complete Time Branching Processes

Add all the events that are in direct conflict.
conf-complete Time Branching Processes

Add all the events that are in direct conflict
Valid Time Branching Process

\(\text{valid} = \text{conf-complete} + \text{constraints on the occurrence dates of events:}\)

\[\forall e \in E:\]

\[\begin{cases} 
[\theta(e) \neq +\infty \text{ and } \theta(e) - \text{TOE}(\bullet e, l(e)) \in I_s(l(e)) ] \\
\text{and } \forall e' \in E \text{ s.t. } e \text{ conf } e', \theta(e') = +\infty \\
\text{or } [\theta(e) = +\infty \text{ and } \exists b \in \bullet e, \theta(\bullet b) = +\infty] \\
\text{or } [\theta(e) = +\infty \text{ and } \exists e' \in E \text{ s.t. } e \text{ conf } e' \text{ and } \theta(e') \neq +\infty \\
\text{and } \theta(e') - \text{TOE}(\bullet e, l(e)) \in I_s(l(e)) \downarrow] \end{cases}\] (1)

\(\text{TOE} = \text{enabling date of an event}\)

\(I_s = \text{static firing interval of a transition}\)
Valid Time Branching Process: Example

\[\begin{align*}
\text{Valid Time Branching Process: Example} & \\
\[t_1[4, 5] & \\
\[t_2[2, 3] & \\
\[t_3[0, \infty] & \\
\end{align*}\]
Valid Time Branching Process: Example

\[
\begin{align*}
4 \leq \theta(e_1) &\leq 5 \\
\text{and } \theta(e_2) &= +\infty \\
\text{or } (\theta(e_1) &= \infty \text{ and } \theta(e_2) \leq 5)
\end{align*}
\]
Valid Time Branching Process: Example

\[ t_1[4, 5] \quad t_2[2, 3] \quad t_3[0, \infty[ \]

\[
\begin{cases}
4 \leq \theta(e_1) \leq 5 \\
\text{and } \theta(e_2) = +\infty \\
or \ (\theta(e_1) = \infty \text{ and } \theta(e_2) \leq 5)
\end{cases}
\]

\[
\begin{cases}
2 \leq \theta(e_2) \leq 3 \\
\text{and } \theta(e_1) = +\infty \\
\text{and } \theta(e_3) = +\infty \\
or \begin{cases}
\theta(e_2) = +\infty \\
\text{and } (\theta(e_1) \leq 3 \text{ or } \theta(e_3) \leq 3)
\end{cases}
\end{cases}
\]
Valid Time Branching Process: Example

\[
\begin{align*}
\{ & 4 \leq \theta(e_1) \leq 5 \\
& \text{and } \theta(e_2) = +\infty \\
& \text{or } (\theta(e_1) = \infty \text{ and } \theta(e_2) \leq 5) \} \\
\{ & 2 \leq \theta(e_2) \leq 3 \\
& \text{and } \theta(e_1) = +\infty \\
& \text{and } \theta(e_3) = +\infty \\
& \text{or } \{ \theta(e_2) = +\infty \\
& \text{and } (\theta(e_1) \leq 3 \text{ or } \theta(e_3) \leq 3) \} \\
\end{align*}
\]
Temporally Complete Time Branching Process

The firing date of all events in the TBP should be less or equal than the latest firing date of all possible extensions.
The firing date of all events in the TBP should be less or equal than the latest firing date of all possible extensions.
Temporally Complete Time Branching Process

The firing date of all events in the TBP should be less or equal than the latest firing date of all possible extensions.
Soundness, Completeness

Theorem (Soundness)

For all valid time branching processes (temporally complete), its events with a finite occurrence date form a valid time process.

Theorem (Completeness)

By adding all the events in direct conflict to a valid time process, we obtain a valid time branching complete (which is temporally complete).
Finite Complete Prefix

Theorem (Finite Complete Prefix)

For any time Petri net, there exists a finite prefix of its symbolic unfolding such that: for all valid time process in the unfolding, there exists a valid time process in the prefix that has the same future behaviors.

Definition (Cut-off event)

e is a cut-off event if there exists e' s.t.:

- The marking created by the causal past of e and e' is the same.
- The causal past of e' is less than that of e (adequate order);
- Equality between the (reduced) ages of the tokens in the markings of all the smallest valid processes containing the causal past of e and e' respectively.

finite unions of zones
Valid Time Branching Process again

Two processes containing the causal past of $e_3$.

Valid

Not valid
Valid Time Branching Process again

Two processes containing the causal past of $e_3$. 
### Parametric Time Petri Nets

Consider a time Petri net with parametric transitions.

- **Initial marking**: $V_0 = \{ a \geq 0, b \geq 0 \}$

- **Transitions**:
  - $t_0[0,0]$ with parameters $a, b$
  - $t_1[a, a+b]$ with parameters $a, b$
  - $t_2[3,4]$ with parameters $a, b$
  - $t_3[2,2]$ with parameters $a, b$

- **Places**:
  - $p_1$
  - $p_2$
  - $p_3$
  - $p_4$
  - $p_5$

The behavior of this net is governed by the following constraints:

1. $\nu(a) \geq 1$
2. $\nu(b) > 0$
3. $\nu(a) + \nu(b) < 2$
4. $\theta(e_1) \leq \nu(a) + \nu(b)$
5. $3 \leq \theta(e_2) < 4$
6. $\theta(e_3) = \theta(e_1) + 2$
7. $\theta(e_4) \leq \max\{\theta(e_1), \theta(e_2)\}$
8. $\theta(e_4) = +\infty$

**Example Constraints**:

- $\nu(a) > 0$
- $\nu(b) > 0$
- $\nu(a) + \nu(b) < 2$
- $\theta(e_1) \leq \nu(a) + \nu(b)$
- $3 \leq \theta(e_2) < 4$
- $\theta(e_3) = \theta(e_1) + 2$
- $\theta(e_4) \leq \max\{\theta(e_1), \theta(e_2)\}$
- $\theta(e_4) = +\infty$
Stopwatch Petri Nets

Didier Lime
Symbolic Unfoldings of Time Petri Nets
19/11/13 17 / 21
Stopwatch Petri Nets

$e_1$ and $e_4$ correspond to “missed opportunities” of firing $t_1$ because it was not active for long enough:

- $\text{DOA}({b_1}, t_1, \theta(e_2)) = 3.2 < 10,$
- $\text{DOA}({b_1}, t_1, \theta(e_5)) = (11.3 - 8.2) + (3.2 - 0) = 6.3 < 10.$

$\text{DOA} = \text{sum of durations during which the transition is active}$
SwPNs: Valid Time Branching Processes

\[ \forall E' \in \text{RCycles}(E), \exists e' \in E' \text{ s.t. } \theta(e') = +\infty \]

and \( \forall e \in E : \)

\[
\left[ \begin{array}{l}
\theta(e) \neq +\infty \text{ and } \theta(e) \geq \max_{b \in \cdot e \cup \circ e} \theta(\cdot b) \\
\text{and } \text{DOA}(\star e, l(e), \theta(e)) \in v(l_s(l(e)))) \\
\text{and } \forall e' \in E \text{ s.t. } e \text{ conf } e', \theta(e') = +\infty \\
\text{and } \forall e' \in E \text{ s.t. } e \xrightarrow{} e', \theta(e) \leq \theta(e')
\end{array} \right] \\
\text{or } \left[ \begin{array}{l}
\theta(e) = +\infty \text{ and } \exists b \in \cdot e \cup \circ e, \theta(\cdot b) = +\infty \\
\text{or } \left[ \begin{array}{l}
\theta(e) = +\infty \text{ and } \exists e' \in E \text{ s.t. } (e \text{ conf } e' \text{ or } e \xrightarrow{} e') \text{ and } \theta(e') \neq +\infty \\
\text{and } \text{DOA}(\star e, l(e), \theta(e')) \in v(l_s(l(e))))
\end{array} \right]
\end{array} \right]
\]

conf and \( \xrightarrow{} \) are interpreted on the underlying Petri net w/ additional read arcs for the activator arcs
SwPNs: Valid TBPs

Constraints for \( e_4 \):

\[
\begin{align*}
\theta(e_4) &\geq \theta(e_3) \\
\theta(e_4) - \theta(e_3) + \theta(e_2) &\in [10, 20] \\
\theta(e_1) &\equiv +\infty \\
\theta(e_8) &\equiv +\infty \\
\theta(e_4) &\leq \theta(e_5)
\end{align*}
\]
SwPNs: Valid TBPs

Constraints for $e_4$:

\[
\begin{align*}
\theta(e_4) &\geq \theta(e_3) \\
\text{and } \theta(e_4) - \theta(e_3) + \theta(e_2) &\in [10, 20] \\
\text{and } \theta(e_1) &= +\infty \\
\text{and } \theta(e_8) &= +\infty \\
\text{and } \theta(e_4) &\leq \theta(e_5) \\
\text{or} \\
\theta(e_4) &= +\infty \\
\text{and } (\theta(e_8) - \theta(e_3) + \theta(e_2) \leq 20 \text{ or } \theta(e_5) - \theta(e_3) + \theta(e_2) \leq 20)
\end{align*}
\]
Conclusion

- Summary:
Conclusion

- Summary:
  - Symbolic unfoldings for time Petri nets
Conclusion

Summary:
- Symbolic unfoldings for time Petri nets
- Can be extended to:
  - Read arcs;
  - Parameters;
  - Stopwatches.
Conclusion

Summary:
- Symbolic unfoldings for time Petri nets
- Can be extended to:
  - Read arcs;
  - Parameters;
  - Stopwatches.
- Existence of a finite complete prefix for TPNs with read arcs
Conclusion

Summary:
- Symbolic unfoldings for time Petri nets
- Can be extended to:
  - Read arcs;
  - Parameters;
  - Stopwatches.
- Existence of a finite complete prefix for TPNs with read arcs

Perspectives:
- Efficient implementation;
- Extension to non-safe bounded models.
References

T. Aura and J. Lilius.
A causal semantics for time Petri nets.

T. Chatain and C. Jard.
Complete finite prefixes of symbolic unfoldings of safe time Petri nets.

K. L. McMillan.
Using unfolding to avoid the state space explosion problem in the verification of asynchronous circuits.