Embedding network calculus and event stream theory in a common model

21st IEEE Int. Conf. on Emerging Technologies and Factory Automation [ETFA 2016]
Track 3: Real-Time and (Networked) Embedded Systems [RTNES]

Marc Boyer    Pierre Roux

Wednesday, September 7th, 2016
Outline

Context and goal

The unifying model

Model evaluation

Conclusion
Outline

Context and goal

The unifying model

Model evaluation

Conclusion
## Global context

### Kind of systems

- real-time system

### Kind of property

- worst case response time
Global context

Kind of systems
Distributed real-time system

Kind of property
worst case response time
Global context

Kind of systems: flow/component

Distributed real-time system
- Components (computation node, bus, switch, etc.)
- Event flows between components
- Event reception triggers a local workload (computation, data forwarding...)

Kind of property
worst case response time
Global context

Kind of systems: flow/component

Distributed real-time system

- Components (computation node, bus, switch, etc.)
- Event flows between components
- Event reception triggers a local workload (computation, data forwarding...)

Kind of property

Bounds on worst case response time
Global context

Kind of systems: flow/component

Distributed real-time system
- Components (computation node, bus, switch, etc.)
- Event flows between components
- Event reception triggers a local workload (computation, data forwarding...)

Kind of property

Bounds on worst case response time
- local latency
- end-to-end latency
Two flow/component models

<table>
<thead>
<tr>
<th>Event Stream/CPA</th>
<th>Network Calculus</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E \xrightarrow{C} E'$</td>
<td>$A \xrightarrow{C} A'$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Flow model</th>
<th>$E(t)$: number of events up to time $t$</th>
<th>$A(t)$: amount of data up to time $t$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contract</td>
<td>$\eta^+, \eta^-$: event arrival functions</td>
<td>$\alpha$: arrival curve</td>
</tr>
<tr>
<td>$\forall t, d \geq 0$</td>
<td>$E(t + d) - E(t) \leq \eta^+(d)$</td>
<td>$A(t + d) - A(t) \leq \alpha(d)$</td>
</tr>
<tr>
<td>$E(t + d) - E(t) \geq \eta^-(d)$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Busy window
- Residual service

- Two very close models
- No best method (depends on the system)
<table>
<thead>
<tr>
<th>Goals</th>
</tr>
</thead>
<tbody>
<tr>
<td>more accurate results</td>
</tr>
<tr>
<td>better understanding of each theory</td>
</tr>
<tr>
<td>modelling of new kind of components: CAN/AFDX gateway, per block memory allocation...</td>
</tr>
</tbody>
</table>
Toward unifying model

Goals

- more accurate results
- better understanding of each theory
- modelling of new kind of components: CAN/AFDX gateway, per block memory allocation...

Success criteria

- accurate
- easy to use
  - modelling
  - proofs
# Toward unifying model

## Goals
- more accurate results
- better understanding of each theory
- modelling of new kind of components: CAN/AFDX gateway, per block memory allocation...

## Success criteria
- accurate
- easy to use
  - modelling
  - proofs

## Guidelines
- a compositional model
- an algebraic model
Outline

Context and goal

The unifying model

Model evaluation

Conclusion
The global picture

Real World System  \[\xrightarrow{\text{Input data flow}}\]  Network Calculus  \[\xrightarrow{\text{Output data flow}}\]  Event Stream / CPA

New model

\[\xrightarrow{\text{A}} S \xrightarrow{\text{A'}}\]  \[\xrightarrow{\text{E}} C \xrightarrow{\text{E'}}\]
### Definition of the new model

<table>
<thead>
<tr>
<th>Arrival curve</th>
<th>Packet count</th>
<th>Event count</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A : \mathbb{R}^+ \rightarrow \mathbb{R}^+$</td>
<td>$P : \mathbb{R}^+ \rightarrow \mathbb{N}$</td>
<td>$E : \mathbb{R}^+ \rightarrow \mathbb{N}$</td>
</tr>
<tr>
<td>$A(t)$: amount of data up to $t$</td>
<td>$P(d)$: number of full packets in the $d$ first “bits”</td>
<td>$E(t)$: number of full packets up to $t$</td>
</tr>
</tbody>
</table>

\[
P(A) = E \quad \text{NC/CPA}
\]

---

Scenario:

- First packet: size 1, throughput 1
- Second packet: size 1, throughput 1/2
- Third packet: size 2, throughput 2
- Fourth packet: size 1, throughput 1
Scenario:

- **First packet**: size 1, throughput 1
- **Second packet**: size 1, throughput 1/2
- **Third packet**: size 2, throughput 2
- **Fourth packet**: size 1, throughput 1
Scenario:

- First packet: size 1, throughput 1
- Second packet: size 1, throughput 1/2
- Third packet: size 2, throughput 2
- Fourth packet: size 1, throughput 1
Scenario:
- First packet: size 1, throughput 1
- Second packet: size 1, throughput 1/2
- Third packet: size 2, throughput 2
- Fourth packet: size 1, throughput 1
Interval Bounding Pair (IBP)

- Real behaviours are unknown at design
- Performance studies based on contract
- Interval Bounding Pair: renaming of arrival curves/event stream
  \( \phi = (\underline{\phi}, \overline{\phi}) \) is an Interval Bounding Pair (IBP) of \( f \) iff

  \[
  \forall t, d \geq 0 : \underline{\phi}(d) \leq f(t + d) - f(t) \leq \overline{\phi}(d)
  \]

- Handle the contract tuple \( \langle \alpha, \pi, \eta \rangle \) where \( \alpha, \pi, \eta \) are respective IBPs of \( A, P, E \)
Outline

Context and goal

The unifying model

Model evaluation

Conclusion
Defining a new model is easy
Model evaluation is hard
Taking in hand the model:
  - basic properties of the model itself
  - modelling basic component
    1. packetizer
    2. aggregation
  - model accuracy (new)
**Mathematical operators**

- **Min/max-plus convolution:** associative, commutative, monotonous

\[
(f \ast g)(t) = \inf_{0 \leq s \leq t} f(t - s) + g(s) \quad (f \bar{\ast} g)(t) = \sup_{0 \leq s \leq t} f(t - s) + g(s)
\]

- **Composition:** associative, monotonous

\[
(f \circ g)(t) = f(g(t))
\]

- **Pseudo-inverses**

\[
f(x) \quad x \quad f^{-1}(y) \quad y \quad f^{-1}(y) \quad y
\]
Intrinsic properties

- IBP properties (from NC and CPA)
Intrinsic properties

- IBP properties (from NC and CPA)
  - min/max: if \((\phi, \overline{\phi})\) and \((\phi', \overline{\phi'})\) are IBP of \(f\), also is \((\max(\phi, \phi'), \min(\overline{\phi}, \overline{\phi'}))\)
Intrinsic properties

- IBP properties (from NC and CPA)
  - min/max: if \((\phi, \overline{\phi})\) and \((\phi', \overline{\phi'})\) are IBP of \(f\), also is \((\max(\phi, \phi'), \min(\overline{\phi}, \overline{\phi'}))\)
  - Kleene star closure: if \((\phi, \overline{\phi})\) is an IBP of \(f\), also is \((\overline{\phi}^*, \overline{\phi}^*)\)
    where \(\cdot^*\) are Kleene-star of min/max convolutions.
Intrinsic properties

- IBP properties (from NC and CPA)
  - min/max: if $(\phi, \overline{\phi})$ and $(\phi', \overline{\phi}')$ are IBP of $f$, also is $(\max(\phi, \phi'), \min(\overline{\phi}, \overline{\phi}'))$
  - Kleene star closure: if $(\phi, \overline{\phi})$ is an IBP of $f$, also is $(\phi^*, \overline{\phi}^*)$
    where $\cdot^*$, $\cdot^\star$ are Kleene-star of min/max convolutions.

- Between IBP (contribution): from two IBPs, build the missing one
Intrinsic properties

- **IBP properties (from NC and CPA)**
  - min/max: if $(\phi, \phi')$ and $(\phi', \phi'')$ are IBP of $f$, also is $(\max(\phi, \phi'), \min(\phi, \phi'))$
  - Kleene star closure: if $(\phi, \phi)$ is an IBP of $f$, also is $(\phi^*, \phi^*)$
    where $\iddots$, $\iddots^*$ are Kleene-star of min/max convolutions.
- **Between IBP (contribution):** from two IBPs, build the missing one

<table>
<thead>
<tr>
<th>$A$</th>
<th>$P$</th>
<th>$E$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(\alpha, \overline{\alpha})$</td>
<td>$(\pi, \overline{\pi})$</td>
<td></td>
</tr>
</tbody>
</table>
Intrinsic properties

- IBP properties (from NC and CPA)
  - min/max: if \((\phi, \phi')\) and \((\phi', \phi'')\) are IBP of \(f\), also is 
    \[(\max(\phi, \phi'), \min(\phi, \phi'))\]
  - Kleene star closure: if \((\phi, \phi')\) is an IBP of \(f\), also is \((\phi^*, \phi'^*)\)
    where \(\cdot^*, \cdot'^*\) are Kleene-star of min/max convolutions.

- Between IBP (contribution): from two IBPs, build the missing one

<table>
<thead>
<tr>
<th>(A)</th>
<th>(P)</th>
<th>(E)</th>
</tr>
</thead>
<tbody>
<tr>
<td>((\alpha, \bar{\alpha}))</td>
<td>((\bar{\pi}, \bar{\pi}))</td>
<td>((\bar{\pi} \circ \alpha, \bar{\pi} \circ \bar{\alpha}))</td>
</tr>
</tbody>
</table>
Intrinsic properties

- IBP properties (from NC and CPA)
  - min/max: if $(\phi, \phi')$ and $(\phi', \phi'')$ are IBP of $f$, also is $(\max(\phi, \phi'), \min(\phi, \phi'))$
  - Kleene star closure: if $(\phi, \phi)$ is an IBP of $f$, also is $(\phi^*, \phi^*)$
    where $\cdot^*$, $\cdot^*$ are Kleene-star of min/max convolutions.
- Between IBP (contribution): from two IBPs, build the missing one

<table>
<thead>
<tr>
<th>$A$</th>
<th>$P$</th>
<th>$E$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(\alpha, \overline{\alpha})$</td>
<td>$(\pi, \overline{\pi})$</td>
<td>$(\pi \circ \alpha, \overline{\pi} \circ \overline{\alpha})$</td>
</tr>
<tr>
<td>$(\pi, \overline{\pi})$</td>
<td>$(\pi, \overline{\pi})$</td>
<td>$(\eta, \overline{\eta})$</td>
</tr>
</tbody>
</table>
Intrinsic properties

- IBP properties (from NC and CPA)
  - min/max: if \((\phi, \phi')\) and \((\phi', \phi')\) are IBP of \(f\), also is 
    \((\max(\phi, \phi'), \min(\phi, \phi'))\)
  - Kleene star closure: if \((\phi, \phi)\) is an IBP of \(f\), also is \((\phi^*, \phi^*)\)
    where \(\cdot^*, \cdot^*\) are Kleene-star of min/max convolutions.
- Between IBP (contribution): from two IBPs, build the missing one

<table>
<thead>
<tr>
<th>(A)</th>
<th>(P)</th>
<th>(E)</th>
</tr>
</thead>
<tbody>
<tr>
<td>((\alpha, \bar{\alpha}))</td>
<td>((\pi, \bar{\pi}))</td>
<td>((\pi \circ \alpha, \bar{\pi} \circ \bar{\alpha}))</td>
</tr>
<tr>
<td>((\pi^{-1} \circ \eta, \pi^{-1} \circ \bar{\eta}))</td>
<td>((\pi, \bar{\pi}))</td>
<td>((\eta, \bar{\eta}))</td>
</tr>
</tbody>
</table>
Intrinsic properties

- **IBP properties (from NC and CPA)**
  - min/max: if \((\phi, \phi')\) and \((\phi', \phi'')\) are IBP of \(f\), also is 
    \((\max(\phi, \phi'), \min(\phi, \phi''))\)
  - Kleene star closure: if \((\phi, \phi)\) is an IBP of \(f\), also is \((\phi^*, \phi^*)\) 
    where \(-\cdot^*, \cdot^*\) are Kleene-star of min/max convolutions.
- Between IBP (contribution): from two IBPs, build the missing one

<table>
<thead>
<tr>
<th>(A)</th>
<th>(P)</th>
<th>(E)</th>
</tr>
</thead>
<tbody>
<tr>
<td>((\alpha, \overline{\alpha}))</td>
<td>((\pi, \overline{\pi}))</td>
<td>((\pi \circ \alpha, \overline{\pi} \circ \overline{\alpha}))</td>
</tr>
<tr>
<td>((\pi^{-1} \circ \eta, \pi^{-1} \circ \eta))</td>
<td>((\pi, \overline{\pi}))</td>
<td>((\eta, \overline{\eta}))</td>
</tr>
<tr>
<td>((\alpha, \overline{\alpha}))</td>
<td></td>
<td>((\eta, \overline{\eta}))</td>
</tr>
</tbody>
</table>
Intrinsic properties

- **IBP properties (from NC and CPA)**
  - **min/max:** if \((\phi, \overline{\phi})\) and \((\phi', \overline{\phi'})\) are IBP of \(f\), also is \((\max(\phi, \phi'), \min(\overline{\phi}, \overline{\phi'}))\)
  - **Kleene star closure:** if \((\phi, \overline{\phi})\) is an IBP of \(f\), also is \((\phi^*, \overline{\phi}^*)\)
    where \(\cdot^*, \cdot^\ast\) are Kleene-star of min/max convolutions.
  - **Between IBP (contribution):** from two IBPs, build the missing one

<table>
<thead>
<tr>
<th>(A)</th>
<th>(P)</th>
<th>(E)</th>
</tr>
</thead>
<tbody>
<tr>
<td>((\alpha, \overline{\alpha}))</td>
<td>((\pi, \overline{\pi}))</td>
<td>((\pi \circ \alpha, \overline{\pi} \circ \overline{\alpha}))</td>
</tr>
<tr>
<td>((\pi^{-1} \circ \eta, \pi^{-1} \circ \overline{\eta}))</td>
<td>((\pi, \overline{\pi}))</td>
<td>((\eta, \overline{\eta}))</td>
</tr>
<tr>
<td>((\alpha, \overline{\alpha}))</td>
<td>((\eta \circ \overline{\alpha}^{-1}, \overline{\eta} \circ \alpha^{-1}))</td>
<td>((\eta, \overline{\eta}))</td>
</tr>
</tbody>
</table>
Packetizer:
- store bits, up to end-of-packet
- instantaneous packet output
- model: $E, P$ unchanged

$$A' := P^{-1} \circ P \circ A \quad E' := E \quad P' := P$$
Packetizer:

- store bits, up to end-of-packet
- instantaneous packet output
- model: $E, P$ unchanged

$$A' := P^{-1} \circ P \circ A$$
$$E' := E$$
$$P' := P$$

$$\alpha' := \pi^{-1} \circ \eta$$
$$\overline{\alpha}' := \pi^{-1} \circ \overline{\eta}$$
Aggregation:

- mix of flows
- “sum” of flows
  - is a flow
- no delay

\[
\begin{align*}
A_1, E_1, P_1 & \rightarrow S \rightarrow A, E, P \\
A_2, E_2, P_2 & \rightarrow S
\end{align*}
\]

\[
A := A_1 + A_2 \\
E := E_1 + E_2 \\
P(A_1 + A_2) := P(A_1) + P(A_2)
\]

\[
\begin{align*}
\alpha := \alpha_1 + \alpha_2 \\
\eta := \eta_1 + \eta_2 \\
\pi := \lfloor \pi_1 \star \pi_2 \rfloor
\end{align*}
\]

\[
\begin{align*}
\overline{\alpha} := \overline{\alpha}_1 + \overline{\alpha}_2 \\
\overline{\eta} := \overline{\eta}_1 + \overline{\eta}_2 \\
\overline{\pi} := \lceil \overline{\pi}_1 \star \overline{\pi}_2 \rceil
\end{align*}
\]
Case study

Two data flows, \( F_1, F_2 \), from \( S \) to \( C \)

Using a link of throughput 1

<table>
<thead>
<tr>
<th>Flow</th>
<th>Packet size</th>
<th>Burst</th>
<th>Throughput</th>
<th>( \alpha_i )</th>
<th>( \pi_i )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( F_1 )</td>
<td>1/2</td>
<td>1</td>
<td>1/4</td>
<td>( x/4 + 1 )</td>
<td>( \lceil 2x \rceil )</td>
</tr>
<tr>
<td>( F_2 )</td>
<td>1</td>
<td>1</td>
<td>1/4</td>
<td>( x/4 + 1 )</td>
<td>( \lfloor x \rfloor )</td>
</tr>
</tbody>
</table>

Goal: evaluation of the packet throughput

- \( F = F_1 + F_2 \)
- what is \( \bar{\eta} \) ?
- challenge: modelling the link shaping
Packet throughput: no shaping

No shaping:

\[ \overline{\eta}_1 = \overline{\pi}_1 \circ \overline{\alpha}_1 = \left\lfloor \frac{x}{2} \right\rfloor + 2 \]

\[ \overline{\eta}_2 = \overline{\pi}_2 \circ \overline{\alpha}_2 = \left\lfloor \frac{x}{4} \right\rfloor + 1 \]

\[ \overline{\eta} \leq \overline{\eta}_1 + \overline{\eta}_2 \]
Packet throughput: with shaping

Link throughput: $\lambda(t) = t$

- Shaping reduces data throughput
  - for each flow, $\bar{\alpha}_i^s = \lambda \land \bar{\alpha}_i$
  - for the aggregate flow: $\bar{\alpha}_{1+2} = \lambda \land (\bar{\alpha}_1 + \bar{\alpha}_2)$

- Impact on packet throughput
  - per flow: $\bar{\eta}_i^s = \bar{\pi}_i \circ \bar{\alpha}_i^s$
  - aggregate flow:
    $\bar{\eta}_{1+2}^s = [\bar{\pi}_1 \ast \bar{\pi}_2] \circ \bar{\alpha}_{1+2}^s$
  - both $\bar{\eta}_1^s + \bar{\eta}_2^s$ and $\bar{\eta}_{1+2}^s$ are packet throughput bounds
- the shaping only affects start of curve
- the simple method has better long term throughput
- the new method is locally better
Outline

Context and goal

The unifying model

Model evaluation

Conclusion
Conclusion

- A new model, unifying NC and Event-Stream/CPA
- Taking the model in hand
  - algebraic results
  - some accuracy gains
- Next steps
  - composition implementation
  - aggregation improvement
  - realistic case study
Toward unifying model

Figure: http://xkcd.com/927/