Real-Time Calculus

Prof. Dr. Jian-Jia Chen

LS 12, TU Dortmund

09, Dec., 2014
Arbitrary Deadlines

The worst-case response time of \( \tau_i \) by only considering the first job of \( \tau_i \) at the critical instant is too optimistic when the relative deadline of \( \tau_i \) is larger than the period.

Consider two tasks:

- \( \tau_1 \) has period 70 and execution time 26 and \( \tau_2 \) is with period 100 and execution time 62.
- \( \tau_2 \)'s seven jobs have the following response times, respectively: 114, 102, 116, 104, 118, 106, 94.
- Note that the first job’s response time is not the longest.
Busy Intervals

**Definition**

A $\tau_i$-level busy interval $(t_0, t]$ of task $\tau_i$ begins at an instant $t_0$ when

1. all jobs in $\tau_i$ released before $t$ have completed, and
2. a job of $\tau_i$ releases.

The interval ends at the first instant $t$ after $t_0$ when all jobs in $\tau_i$ released since $t_0$ are complete.
Abstract Models for Real-Time Calculus

Concrete Instance

Abstract Representation

Input Stream

Tasks

Processor

Service Model

Processing Model

Load Model
Abstract Models for Module Performance Analysis
Overview

System View
- Module Performance Analysis (MPA)

Math. View
- Real-Time Calculus (RTC)
- Min-Plus Calculus, Max-Plus Calculus
Backgrounds

- Real-Time Calculus can be regarded as a worst-case/best-case variant of classical queuing theory. It is a formal method for the analysis of distributed real-time embedded systems.

- Related Work:
Plus-Times and Min-Plus Algebras

- **Algebraic structure**
  - a set of (finite or infinite) elements $S$
  - one or more operators defined on the elements of this set
- **Plus-Times Algebra**: Two operators $+$ and $\times$, denoted by $(S, +, \times)$
- **Min-Plus Algebra**
  - Two operators $\oplus$ (min) and $\otimes$ (plus), denoted by $(S \cup \{+\infty\}, \inf, +)$
  - **Infimum**: The infimum of a subset of some set is the greatest element, not necessarily in the subset, that is less than or equal to all other elements of the subset.
    - For example, $\inf\{[a, b]\} = a$, $\inf\{(a, b)\} = a$, where $\min\{[a, b]\} = a$, $\min\{(a, b)\} = \text{undefined}$.
  - **Supremum**: The supremum of a subset of some set is the smallest element, not necessarily in the subset, that is more than or equal to all other elements of the subset.
    - For example, $\sup\{[a, b]\} = b$, $\sup\{[a, b]\} = b$, where $\max\{[a, b]\} = b$, $\max\{[a, b]\} = \text{undefined}$.
Definition of Arrival Curves and Service Curves

- For a specific trace:
  - Data streams: \( R(t) = \) number of events in \([0, t)\)
  - Resource stream: \( C(t) = \) available resource in \([0, t)\)

- For the worst cases and the best cases in any interval with length \( \Delta \):
  - Arrival Curve \([\alpha^l, \alpha^u]\):
    \[
    \alpha^l(\Delta) = \inf_{\lambda \geq 0, \forall R} \{ R(\Delta + \lambda) - R(\lambda) \}
    \]
    \[
    \alpha^u(\Delta) = \sup_{\lambda \geq 0, \forall R} \{ R(\Delta + \lambda) - R(\lambda) \}
    \]

  - Service Curve \([\beta^l, \beta^u]\):
    \[
    \beta^l(\Delta) = \inf_{\lambda \geq 0, \forall C} \{ C(\Delta + \lambda) - C(\lambda) \}
    \]
    \[
    \beta^u(\Delta) = \sup_{\lambda \geq 0, \forall C} \{ C(\Delta + \lambda) - C(\lambda) \}
    \]
Abstract Models for Real-Time Calculus

Concrete Instance

Abstract Representation

Input Stream

Processor

Tasks

C(t)

R(t) → R'(t)

Load Model

Service Model

Processing Model

α(Δ)

β(Δ)

Prof. Dr. Jian-Jia Chen
Arrival Curve: An Example

Use a sliding window to get the upper bound of the number of events in a specified interval length.
Arrival Curve: An Example

Use a sliding window to get the upper bound of the number of events in a specified interval length.
Arrival Curve: An Example

Use a sliding window to get the upper bound of the number of events in a specified interval length.
Arrival Curve: An Example

Use a sliding window to get the upper bound of the number of events in a specified interval length.
Arrival Curve: An Example

Use a sliding window to get the upper bound of the number of events in a specified interval length.
Arrival Curve: An Example

Use a sliding window to get the upper bound of the number of events in a specified interval length.
Arrival Curve: An Example

Use a sliding window to get the upper bound of the number of events in a specified interval length.
Arrival Curve: An Example

Use a sliding window to get the upper bound of the number of events in a specified interval length.
Arrival Curve: An Example

Use a sliding window to get the upper bound of the number of events in a specified interval length.
Service Curve: An Example

Resource Availability

Service Curves
\( \beta = [\beta^l, \beta^u] \)
Example 1: Periodic with Jitter

A common event pattern that is used in literature can be specified by the parameter triple \((p, j, d)\), where \(p\) denotes the period, \(j\) the jitter, and \(d\) the minimum inter-arrival distance of events in the modeled stream.
Example 1: Periodic with Jitter

Periodic

\[
\alpha^u(\Delta) = \left\lceil \frac{\Delta}{p} \right\rceil
\]

\[
\alpha^l(\Delta) = \left\lfloor \frac{\Delta}{p} \right\rfloor
\]

Periodic with Jitter

\[
\alpha^u(\Delta) = \left\lceil \frac{\Delta + j}{p} \right\rceil
\]

\[
\alpha^l(\Delta) = \left\lfloor \frac{\Delta - j}{p} \right\rfloor
\]
Example 1: Periodic with Jitter

\[ \alpha^u(\Delta) = \min \left\{ \left\lceil \frac{\Delta + j}{p} \right\rceil, \left\lfloor \frac{\Delta}{d} \right\rfloor \right\} \]

\[ \alpha^l(\Delta) = \left\lfloor \frac{\Delta - j}{p} \right\rfloor \]
More Examples on Arrival Curves

(a) \[ \Delta \]

(b) \[ \Delta \]

(c) \[ \Delta \]

(d) \[ \Delta \]
Example 2: TDMA Resource

- Consider a real-time system consisting of $n$ applications that are executed on a resource with bandwidth $B$ that controls resource access using a TDMA (Time Division Multiple Access) policy.

- Analogously, we could consider a distributed system with $n$ communicating nodes, that communicate via a shared bus with bandwidth $B$, with a bus arbitrator that implements a TDMA policy.

- TDMA policy: In every TDMA cycle of length $\bar{c}$, one single resource slot of length $s_i$ is assigned to application $i$. 

![Diagram of TDMA Resource]
Example 2: TDMA Resource

\[ \beta^u(\Delta) = B \min \left\{ \left\lfloor \frac{\Delta}{\bar{c}} \right\rfloor s_i, \Delta - \left\lceil \frac{\Delta}{\bar{c}} \right\rceil (\bar{c} - s_i) \right\} \]

\[ \beta^l(\Delta) = B \max \left\{ \left\lfloor \frac{\Delta}{\bar{c}} \right\rfloor s_i, \Delta - \left\lceil \frac{\Delta}{\bar{c}} \right\rceil (\bar{c} - s_i) \right\} \]
More Examples on Service Curves

- **Full Resource**
  - Graph showing the number of cycles versus $\Delta$ for $\beta^u$ and $\beta^l$.

- **Bounded Delay**
  - Graph showing the number of cycles versus $\Delta$ for $\beta^u$ and $\beta^l$.

- **TDMA Resource**
  - Graph showing the number of cycles versus $\Delta$ for $\beta^u$ and $\beta^l$.

- **Periodic Resource**
  - Graph showing the number of cycles versus $\Delta$ for $\beta^u$ and $\beta^l$.
On-Site Exercise

Consider a stream of events. There are at most three events and at least 2 events arriving within a time interval \([k \cdot p, (k + 1) \cdot p]\) for \(k = 0, 1, 2, \ldots\). These 2 or 3 events within a period \(p\) are defined as follows:

- At time \(k \cdot p\), there is one event.
- At time \(k \cdot p + 0.3p\), there is another event.
- Between time \(k \cdot p + 0.4p\) and \(k \cdot p + 0.5p\), there may be one event.

How do the upper and lower arrival curves look like?
Convolutions

- Plus-times system theory: signals $f$, impulse response $g$, convolution in time domain:

$$h(t) = (f \times g)(t) = \int_0^t f(t - s)g(s)ds,$$

where $f, g$ can be thought as signals and impulse response, respectively.

- Min-Plus system theory: streams $R$, variability curves $g$, convolution in time-interval domain:

$$R'(t) \geq (R \otimes g)(t) = \inf_{0 \leq \lambda \leq t} \{ R(t - \lambda) + g(\lambda) \}.$$
Abstraction

time domain cumulative functions

time-interval domain variability curves
Convolution and De-convolution

- $f \otimes g$ is called \textit{min-plus convolution}
  \[
  (f \otimes g)(t) = \inf_{0 \leq \lambda \leq t} \{ f(t - \lambda) + g(\lambda) \}
  \]

- $f \oslash g$ is called \textit{min-plus de-convolution}
  \[
  (f \oslash g)(t) = \sup_{0 \leq \lambda} \{ f(t + \lambda) - g(\lambda) \}
  \]

- $f \bar{\otimes} g$ is called \textit{max-plus convolution}
  \[
  (f \bar{\otimes} g)(t) = \sup_{0 \leq \lambda \leq t} \{ f(t - \lambda) + g(\lambda) \}
  \]

- $f \bar{\oslash} g$ is called \textit{max-plus de-convolution}
  \[
  (f \bar{\oslash} g)(t) = \inf_{0 \leq \lambda} \{ f(t + \lambda) - g(\lambda) \}
  \]
Arrival and Service Curves Revisit

\[ \alpha^l(t - s) \leq R(t) - R(s) \leq \alpha^u(t - s) \forall s \leq t. \]
\[ \beta^l(t - s) \leq C(t) - C(s) \leq \beta^u(t - s) \forall s \leq t. \]

Therefore, by using the convolution and de-convolution, we know that

\[ \alpha^u = R \circ R; \quad \alpha^l = R \circ R; \quad \beta^u = C \circ C; \quad \beta^l = C \circ C; \]

The proof for \( \alpha^u \):

\[ \alpha^u(\Delta) = \sup_{\lambda \geq 0} \{ R(\Delta + \lambda) - R(\lambda) \} \geq R(\Delta + \lambda) - R(\lambda), \quad \forall \lambda \geq 0. \]
Tight Curves

A curve $f$ is sub-additive, if

$$f(a) + f(b) \geq f(a + b) \quad \forall a, b \geq 0.$$ 

The sub-additive closure $\overline{f}$ of a curve $f$ is the largest sub-additive curve with $\overline{f} \leq f$ and is computed as

$$\overline{f} = \min\{f, (f \otimes f), (f \otimes f \otimes f), \ldots\}.$$ 

If $f$ is interpreted as an arrival curve, then any trace $R$ that is upper bounded by $f$ is also upper bounded by the sub-additive closure $\overline{f}$.
Tight Curves

A curve $f$ is sub-additive, if

$$ f(a) + f(b) \geq f(a + b) \quad \forall a, b \geq 0. $$

The sub-additive closure $\overline{f}$ of a curve $f$ is the largest sub-additive curve with $\overline{f} \leq f$ and is computed as

$$ \overline{f} = \min\{f, (f \otimes f), (f \otimes f \otimes f), \ldots\}. $$

If $f$ is interpreted as an arrival curve, then any trace $R$ that is upper bounded by $f$ is also upper bounded by the sub-additive closure $\overline{f}$.

A tight upper arrival curve should satisfy the sub-additive property.
Greedy Processing Component (GPC)

- Component is triggered by incoming events.
- A fully preemptable task is instantiated at every event arrival to process the incoming event.
- Active tasks are processed in a greedy fashion in FIFO order.
- Processing is restricted by the availability of resources.
Some Relations (only for your reference)

- The output upper arrival curve of a component satisfies

\[ \alpha^{u'} \leq (\alpha^{u} \ominus \beta^{l}) \]

with a simple and pessimistic calculation.

- The remaining lower service curve of a component satisfies

\[ \beta^{l''}(\Delta) = \sup_{0 \leq \lambda \leq \Delta} (\beta^{l}(\lambda) - \alpha^{u}(\lambda)) \]
More Relations (only for your reference)

\[
\alpha^u' = \left[ (\alpha^u \otimes \beta^u) \otimes \beta^l \right] \land \beta^u \\
\alpha^l'' = \left[ (\alpha^u \otimes \beta^l) \otimes \beta^l \right] \land \beta^l \\
\beta^u' = (\beta^u - \alpha^l) \ominus 0 \\
\beta^l'' = (\beta^l - \alpha^u) \ominus 0
\]

Without formal proofs....
Graphical Interpretation

\[ B = \sup_{t \geq 0} \{ R(t) - R'(t) \} \leq \sup_{\lambda \geq 0} \{ \alpha^u(\lambda) - \beta^l(\lambda) \} \]

\[ D = \sup_{t \geq 0} \{ \inf_{\tau \geq 0} : R(t) \leq R'(t + \tau) \} \]

\[ = \sup_{\Delta \geq 0} \{ \inf_{\tau \geq 0} : \alpha^u(\Delta) \leq \beta^l(\Delta + \tau) \} \]
System Composition

Concrete Instance

How to Interconnect service?
Scheduling

GPC

CPU

RM

TDMA

BUS

DSP

α

α′

β_{CPU}

β_{BUS}

β_{DSP}
Scheduling and Arbitration
Mixed Hierarchical Scheduling

...and many other combinations:

- RR + EDF
- FP/RM + RR
- FP/RM + GPS
- GPS + EDF

Prof. Dr. Jian-Jia Chen (LS 12, TU Dortmund)
Complete System Composition

Input Stream

Concrete Instance

Abstract Representation

Prof. Dr. Jian-Jia Chen (LS 12, TU Dortmund)
RTC Toolbox (http://www.mpa.ethz.ch/Rtctoolbox)

Overview

The Real-Time Calculus (RTC) Toolbox is a free Matlab toolbox for system-level performance analysis of distributed real-time and embedded systems.

The RTC Toolbox is based on an efficient representation of Variability Characterization Curves (VCC's) and implements most min-plus and max-plus algebra operators for these curves. On top of the min-plus and max-plus algebra operators, the RTC Toolbox provides a library of functions for Modular Performance Analysis with Real-Time Calculus.
Advantages and Disadvantages of RTC and MPA

- **Advantages**
  - More powerful abstraction than “classical” real-time analysis
  - Resources are first-class citizens of the method
  - Allows composition in terms of (a) tasks, (b) streams, (c) resources, (d) sharing strategies.

- **Disadvantages**
  - Needs some effort to understand and implement
  - Extension to new arbitration schemes not always simple
  - *Not applicable for schedulers that change the scheduling policies dynamically.*