Simulation and Implementation of the PTIDES Programming Model

Jia Zou
Slobodan Matic
Edward A. Lee
Thomas Huining Feng
Patricia Derler

PTIDES
Programming Temporally Integrated Distributed Embedded Systems
Based on Discrete-Event simulation
Relate model time to physical time at specific points in the system
Leverages time synchronization across distributed platforms
(IEEE 1588 protocol)

Flexotask Implementation
The Flexotask system enables implementation of both real-time applications and real-time schedulers in a Java Virtual Machine.

Program Feasibility Analysis

\[
\max_{i \in G(i)} \{d(i) - \delta(i, i')\} \leq \min_{E \subseteq G} \{\delta(i, i')\}
\]

PTIDES Simulation
A domain in Ptolemy II Environment

Implementation Strategy
Event Safe-To-Process Analysis

1. Physical time has exceeded
   \[\tau + \max_{p \in \cal{C}, \tau < \tau_i} \{\delta(p) - \delta(p, r')\},\]
   and
   2. for each source-actor input port \( p \in \cal{C} \), an event has been received with time stamp greater than
   \[\tau + \max_{i \in G(i)} \{\delta(p, r')\},\]
   and
   3. for each port \( p' \in \cal{I} \) such that there exists \( p' \in \cal{C} \) with \( \delta(p, p') \leq \infty \), each event in input queue of \( p' \) has time stamp
   \[\tau + \max_{i \in G(i)} \{\delta(p', r')\} \leq \tau \]
   for \( p' \in \cal{I} \).

Dependency Cut

\[
\max(d_{i1}, d_{i2} - \delta(i, i'_1), d_{i3} - \delta(i, i'_2)) \leq \delta(i, i'_1)
\]

Three PTIDES Schedulers
Each with a different set of assumptions and execution strategy
1. Event queue ordered by timestamp, check smallest event for processing
2. Event queue ordered by timestamp, check all events for processing.
3. Event queue ordered by deadline, check all events for processing => fusion between Earliest-Deadline-First (EDF) and PTIDES.

JAviator Control Application

Preliminary Results