CHESS
CENTER FOR HYBRID & EMBEDDED SOFTWARE SYSTEMS

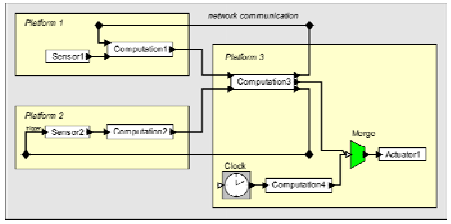
COLLEGE OF ENGINEERING, UC BERKELEY

Simulation and Implementation of the PTIDES Programming Model

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

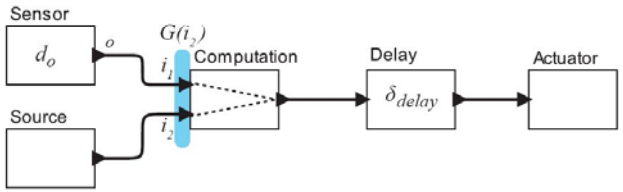
http://chess.eecs.berkeley.edu/

Distributed Embedded System Implementation



PTIDES

Programming **T**emporally **I**ntegrated **D**istributed **E**mbedded **S**ystems
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)



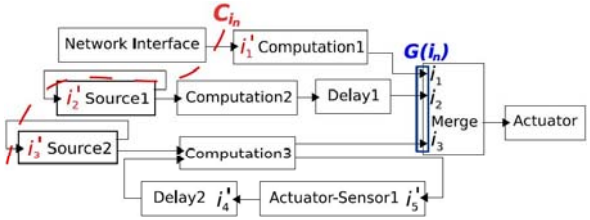
Assumption on sensors (and network interface outputs)

$$\tau + d_o \geq t$$

Requirement on actuators (and network interface inputs)

$$t \leq \tau$$

Dependency Cut



Implementation Strategy

Event Safe-To-Process Analysis

An event at input port $i \in I$ with time stamp τ is safe to process when:

1) a) *physical time has exceeded*

$$\tau + \max_{p \in \mathbb{C}_i, i' \in G(i)} \{d(p) - \delta(p, i')\},$$

and

b) *at each source actor input port $p \in \mathbb{C}_i$ an event has been received with time stamp greater than*

$$\tau + \max_{i' \in G(i)} (-\delta(p, i')),$$

and

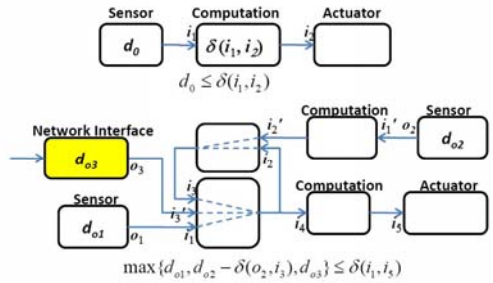
2) *for each port $p' \in I$ such that there exists port $p \in \mathbb{C}_i$ with $\delta(p, p') < \infty$, each event in input queue of p' has time stamp*

a) *greater than or equal to τ for $p' \in G(i)$,*

b) *greater than*
$\tau + \max_{i' \in G(i)}(-\delta(p', i'))$ *for* $p' \notin G(i)$.

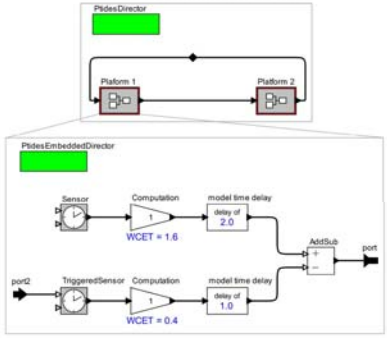
Program Feasibility Analysis

$$\max_{\substack{i' \in G(i) \\ o \in O}} \{d(o) - \delta(o, i')\} \leq \min_{i \in I} \{\delta(i, i)\}$$



PTIDES Simulation

A domain in Ptolemy II Environment



Flexotask Implementation

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

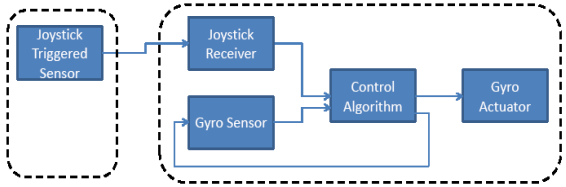
Flexotask uses a **restricted memory schemes** to ensure the real-time properties of the system. This is achieved with separated memory spaces and restricted garbage collection behaviors.

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