Memory-aware Response Time Analysis for P-FRP Tasks

Xingliang Zou
Albert M. K. Cheng

Department of Computer Science, University of Houston
Texas, USA
What is P-FRP?

- Functional Programming
  - Functional Reactive Programming
    - Priority-based FRP
P-FRP: AR

<table>
<thead>
<tr>
<th>Task</th>
<th>Execution Time</th>
<th>Period</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>τ₁</td>
<td>1</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>τ₂</td>
<td>2</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>τ₃</td>
<td>2</td>
<td>20</td>
<td>3</td>
</tr>
</tbody>
</table>
Memory-aware Scheduling Model Taking into consideration that the resumed job is able to use the code/data previously stored in the system, and hence has shorter execution time than that of cold started job.

Memory-aware P-FRP scheduling of fixed priority task set \((C_1^1 = 1, C_1^2 = 1; C_2^1 = 2, C_2^2 = 1; C_3^1 = 2, C_3^2 = 1; T_1 = 5, T_2 = 4, T_3 = 20)\)
Experiments and Results

Memory-aware scheduling leads to fewer unschedulable task sets than that of the original P-FRP scheduling:
(1) 22.9%, 21.3%, 15.6%, 14.4%, 13.4% and 9.6% when $f = 0.7$, or
(2) 46.9%, 39.9%, 31.4%, 27.3%, 22.9% and 18.0% when $f = 0.5$. 
We propose to consider more accurate, more practical task model in P-FRP task scheduling.
Thank you!

Welcome to watch our poster.