Linear Time Admission Control for Elastic Scheduling

Work Already Published

Marion Sudvarg, Chris Gill, Sanjoy Baruah

Real-Time Systems **57,** 485–490 (2021)

Presented by Marion Sudvarg

RTSS 2021

This work was funded by the United States National Science Foundation.

Washington University in St. Louis

James McKelvey School of Engineering

Overview

- Elastic scheduling is a model for compressing task utilizations in an overloaded system
- The original algorithm* performs compression in time quadratic in the number of tasks
- In this work, we present an implementation that, with O(n log n) initialization, performs compression for online task admission in linear time

*** Giorgio C. Buttazzo, Giuseppe Lipari, and Luca Abeni. "Elastic task model for adaptive rate control." RTSS 1998.**

Review of the Original Algorithm

Each task τ_i characterized by:

- U_i^{max}: Initial, uncompressed utilization
- $\mathbf{E_i}$: Elasticity, flexibility to vary utilization
- U_i^{min}: Constraint on minimum utilization

System global variables:

- $\mathbf{E_{SUM}}$: Total elasticity (sum of E_i)
- $\mathbf{U_{SUM}}$: Initial total utilization (sum of U_i^{max})
- $\mathbf{U_d}$: Desired total utilization

During overload, total required compression is $U_{SUM} - U_d$

Algorithm

1. Compress each task's utilization proportionally to its elasticity:

   $U_i = U_i^{max} - (U_{SUM} - U_d)\frac{E_i}{E_{SUM}}$

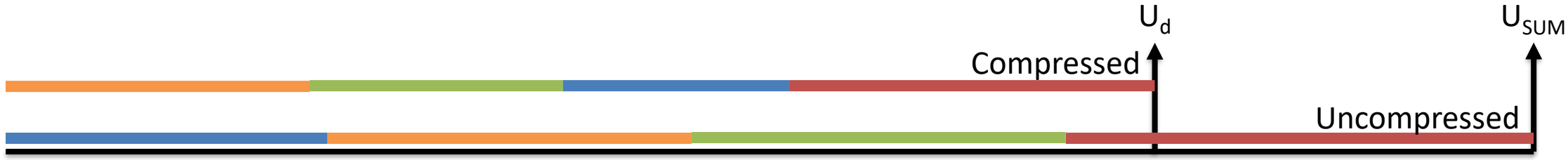
2. While any task τ_i has $U_i < U_i^{min}$:
   a. Set $U_i = U_i^{min}$
   b. Repeat from 1

Improved Compression Algorithm

Initialization: Sort tasks in a list according to $\emptyset_i = \frac{U_i^{max} - U_i^{min}}{E_i}$

Task Admission:

1. Insert new task τ_j according to $\emptyset_j$
2. For each task τ_i :
   a. Compress according to $U_i = U_i^{max} - (U_{SUM} - U_d)\frac{E_1}{E_{SUM}}$
   b. Is $U_i < U_i^{min}$?
   c. If so, set $U_i = U_i^{min}$
   d. If not, no remaining task will have $U_i < U_i^{min}$, and so compress normally



Conclusions

- We have presented a new algorithm for elastic compression, which improves on the prior $O(n^2)$ algorithm:
- It requires one-time O(n log n) initialization to sort tasks
- Enables new task admission and compression in O(n) time
- If a task leaves, it is removed from the (still sorted) list, allowing decompression in O(n) time
- Questions? msudvarg@wustl.edu
- Sudvarg, M., Gill, C. & Baruah, S. Linear-time admission control for elastic scheduling. *Real-Time Systems* **57,** 485–490 (2021)