

Techniques and Analysis for Mixed-criticality Scheduling with Mode-dependent Server Execution Budgets

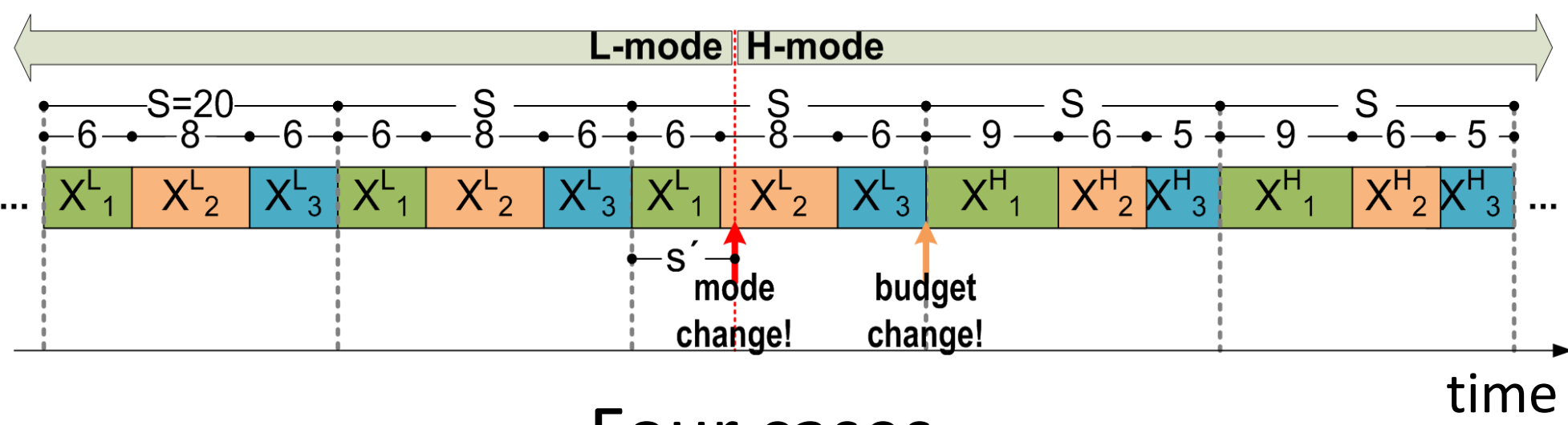
Problem Description

- Develop a mixed-criticality system that schedules applications of different criticality to share resources and reduce cost
- Ensure the temporal behavior of each application
- Simplify the certification process by providing sufficient temporal isolation
- Any misbehaving computational task should not affect the execution of any other task of the same or higher criticality
- Efficiently use the processing capacity to reduce costs

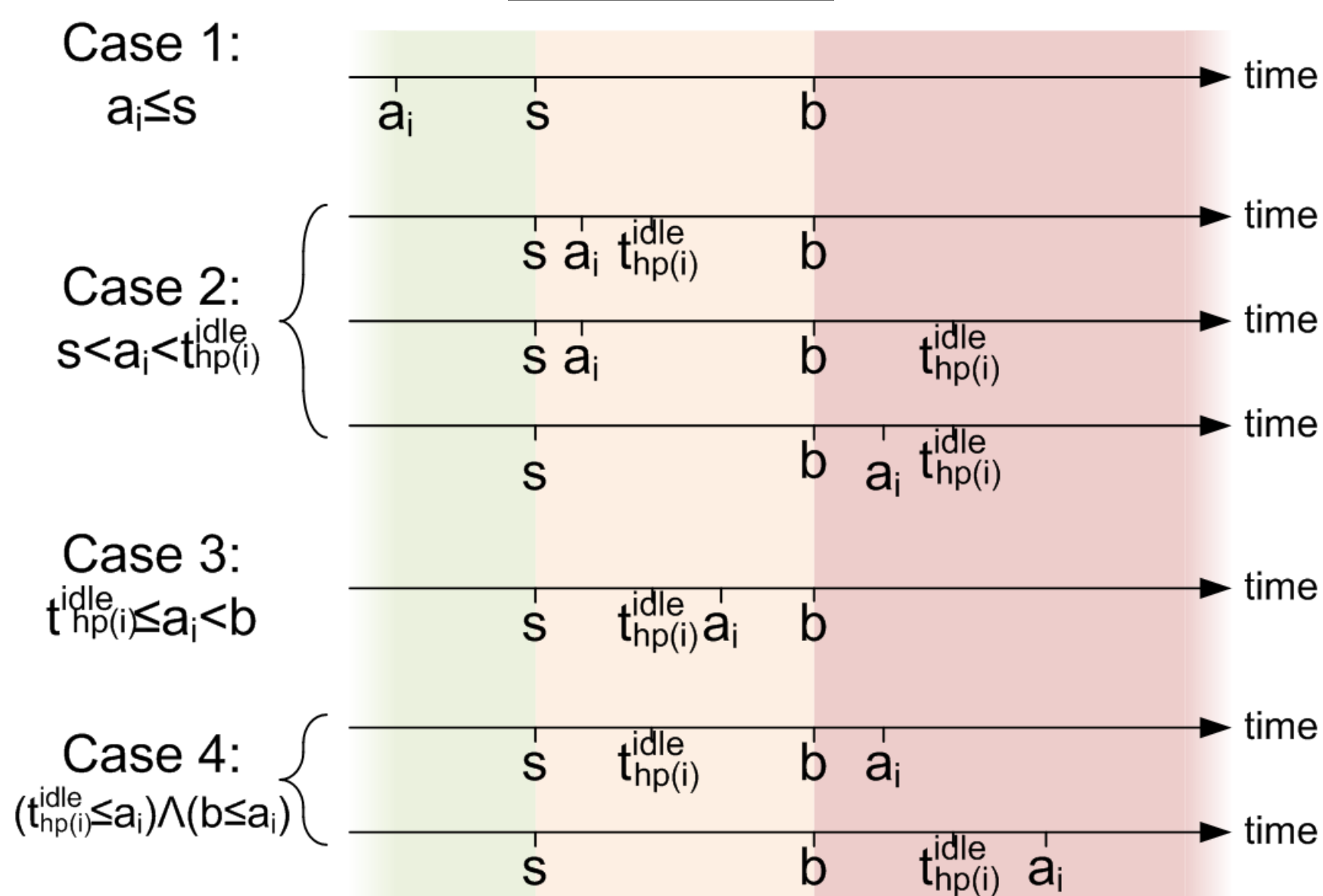
Main Idea

- An adaptive mode-based scheduling arrangement is used to schedule mixed-criticality applications
- Isolation among applications is achieved through servers
- Servers are scheduled in a cyclic executive manner to reduce certification cost
- Vary the server budgets dynamically upon a mode-switch to improve processing resource utilization
- Assign per-mode execution budgets to servers via heuristics

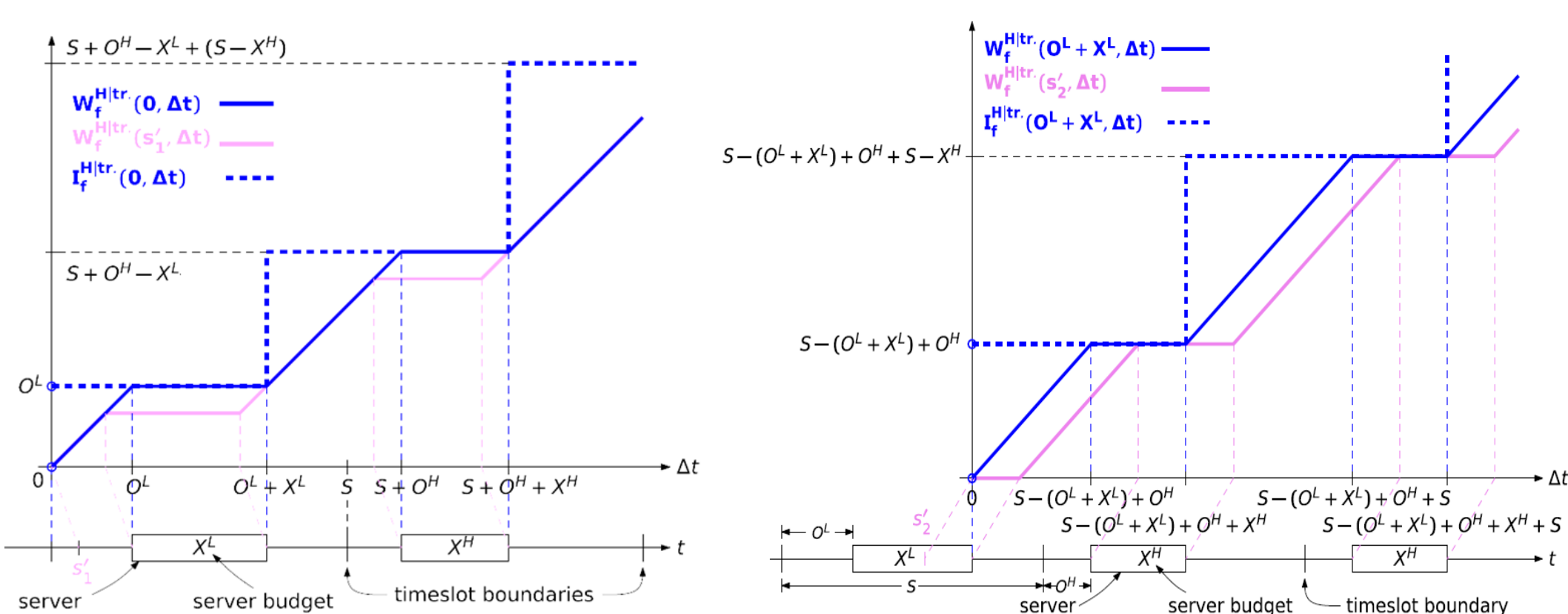
Server-level Schedulability



Four cases



Fake task interference two worst-cases

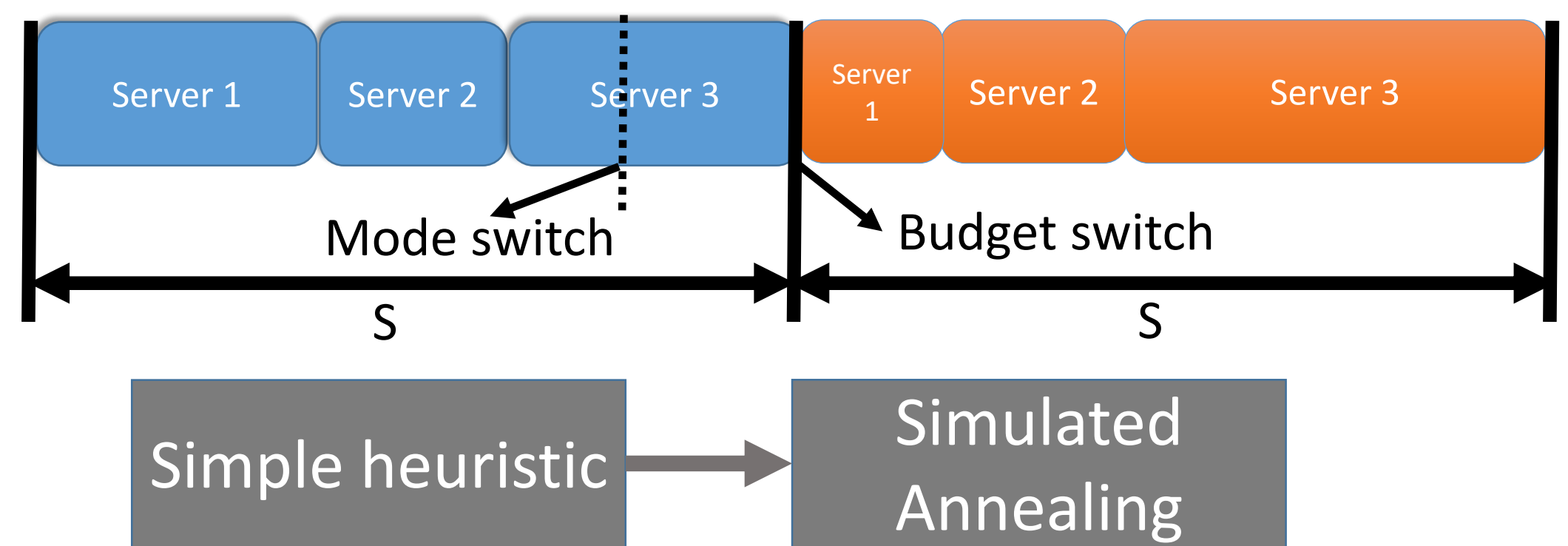


System-level Schedulability

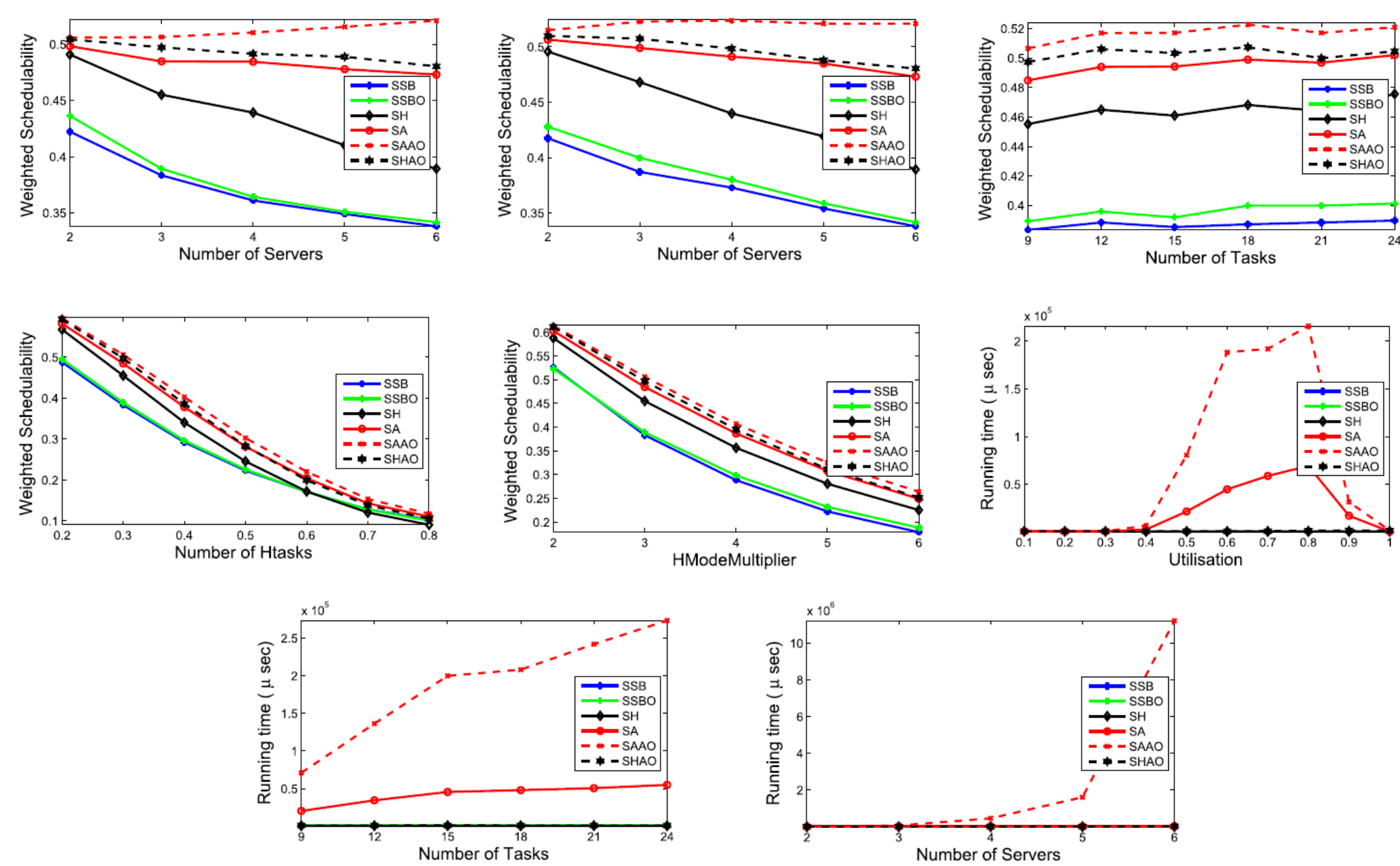
- Assign server budgets in both modes (X^L and X^H)
- Assign offsets in both modes (O^L and O^H)
- Ensure the processing capacity is not exceeded in any mode, (i.e., $\sum_i X_i^L \leq S$ and $\sum_i X_i^H \leq S$)
- Computation of X_i^L and X_i^H depends on O_i^L and O_i^H , which in turn depend on X_j^L and X_j^H of preceding servers ($j < i$)
- Non-decreasing order of $X^L - X^H$ leads to benign jitter
- A more generous X_i^L may permit using a smaller X_i^H

Budget Assignment Heuristics

Static ($X^L=X^H$) vs Dynamic ($X^L \neq X^H$) server budgets



Evaluation



Concluding Remarks

- New schedulability analysis for mixed-criticality systems
- Periodic servers scheduled in a cyclic executive manner
- AMC scheduling policy within each server
- Varying server budgets in different modes
- Strict temporal isolation
- Mode dependent budgets can improve the schedulability ratio by up to 52.8% vs. static budgets
- Even simple heuristics can yield up to 27% of improvement
- The order of servers can influence the schedulability ratio and the proposed ordering heuristics perform well