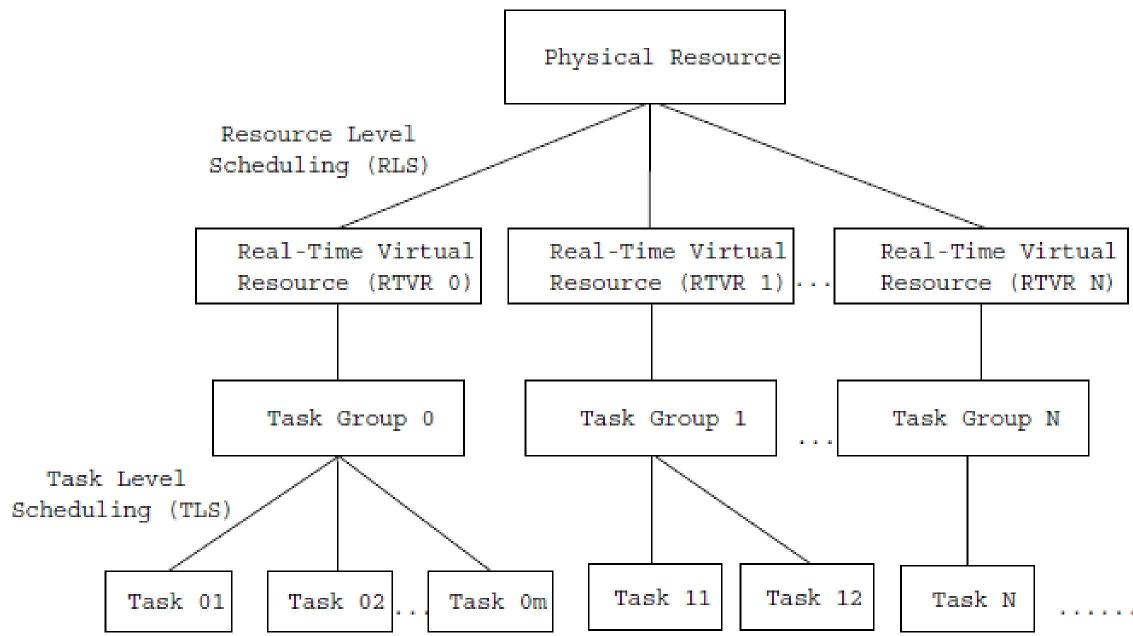
# Static Approximation Algorithms for Regularity-based Resource Partitioning

Yu Li, Albert M. K. Cheng
Computer Science Department
University of Houston

# Motivation

- Deploying real-time systems on powerful modern hardware causes **low resource utilization**
- Integrating multiple real-time systems onto one single platform via **hierarchical real-time scheduling (HRTS)** techniques



# Key Problem of HRTS

- How to define the specification of the Real-Time Virtual Resource (RTVR)?

1. The Manner of Resource Partitioning

2. The Real-time Guarantee of RTVRs

# Difficulty

**Higher Resource Utilization**

**V.S.**

**Better Real-time Guarantee**

# Other HRTS Technologies

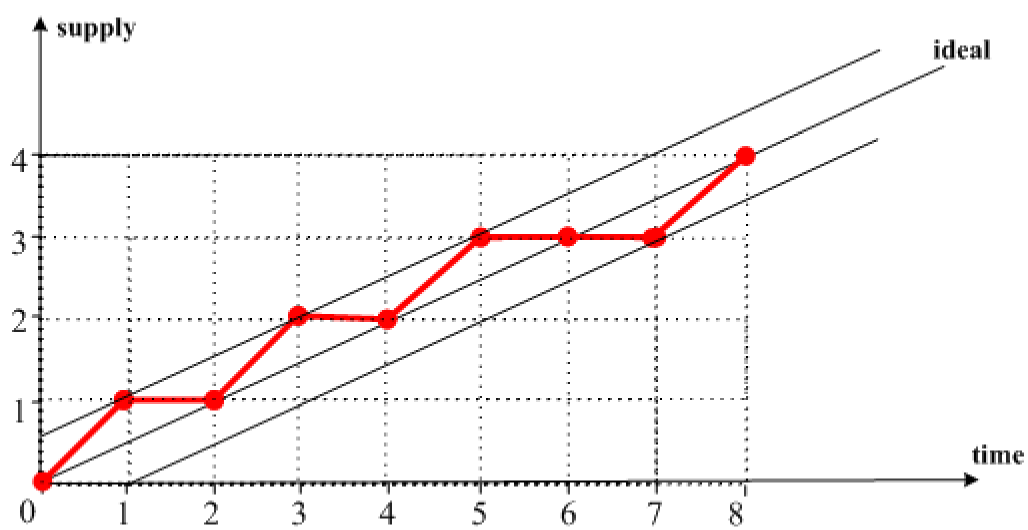
- **The Bounded-Delay Model** (Mok & Feng)
  - A clean separation
  - No effective scheduling algorithm
- **The Periodic Model** (Shin & Lee)
  - Easy for scheduling
  - Poor resource utilization
- **The EDP Model** (Easwaran, Anand & Lee)
  - Higher resource utilization
  - Only work on single-resource platforms

# Approximation Methodology

- Partition resources by a ***quantum*** size
  - Bound context-switch overhead
  - Simplify real-time scheduling
- Round the weight of RTVRs by a static boundary sequence
  - Help find effective scheduling algorithms
- Bounded worst-case utilization loss
- High average performance

# Regularity-based Resource Model–Feng & Mok 2003

- Work in integral-number domain
- A resource partition is defined as $(\alpha, k)$, where $\alpha$ is the availability factor (weight), and integer $k$ is the supply regularity (deviation range of actual resource allocation)



# Regularity-based Resource Partitioning Problem

Given a set $\{(a_i, k_i), 1 \le i \le n\}$ as the availability factors and supply regularities of *n* resource partitions, can they be scheduled on a physical resource (single-resource or uniform multi-resource) ?

Definition: A <u>Regular Partition</u> is a partition with supply regularity of 1. A <u>*k*-Irregular Partition</u> is a partition with supply regularity of *k* where $k>1$.

# AAF-based Scheduling

- Convert irregular partitions to regular partitions
  - When $k$ regular partitions are combined together, they form a partition with supply regularity of $k$
- Round availability factors by the powers of 0.5
  - Boundary sequence is <1, 0.5, 0.25, 0.125, …>
- Function AAF($\alpha$, $k$): Adjusted Availability Factor
  - AAF(0.67, 2) = 0.5 + 0.25
  - AAF(0.67, 3) = 0.5 + 0.125 + 0.0675
  - AAF(0.75, 2) = AAF(0.75, 3) = 0.5 + 0.25

# AAF-Single (Feng & Mok)

- Only work in single-resource scenario
- Given a set $\{(a_i, k_i), 1 \le i \le n\}$ as the availability factors and supply regularities of n regular or irregular partitions, they are schedulable on a single-resource if

$$\sum_{i=1}^{n} AAF(\alpha_i, k_i) \le 1$$

- Example: {(0.3, 2), (0.24, 1), (0.42, 3)} is the set of availability factors and supply regularities of P1, P2, P3

P1, AAF(0.3, 2) = 0.25 + 0.0675

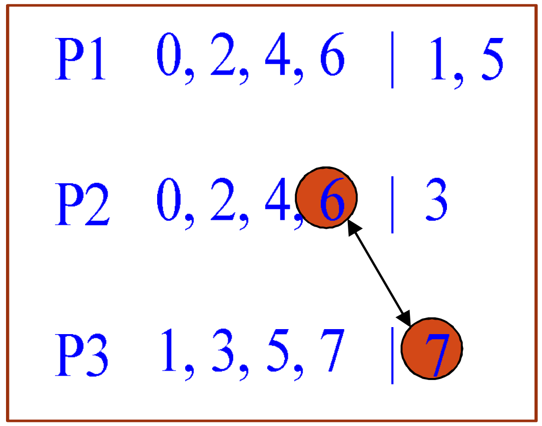
P2, AAF(0.24, 1) = 0.25

P3, AAF(0.42, 3) =0.25 + 0.125 + 0.0675

| | | | |
|---|---|---|---|
| P1 | 0, 4, 8, 12 | | 7 |
| P2 | 1, 5, 9, 13 | | |
| P3 | 2, 6, 10, 14 | 3, 11 | 15 |

# AAF-Multi (Li, Cheng & Mok)

- AAF-Single does not work in multi-resource scenario




- AAF-Multi:irregular partitions are converted to **<u>sub-regular partitions</u>** instead of regular partitions

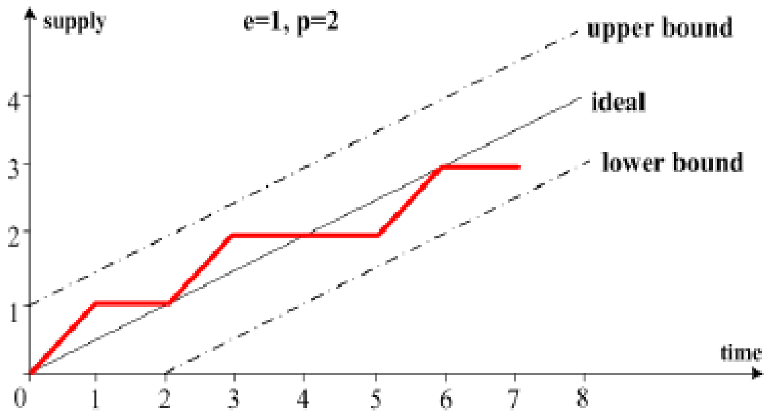
$$\sum_{i=1}^{n} AAF(\alpha_i, k_i) \leq m$$

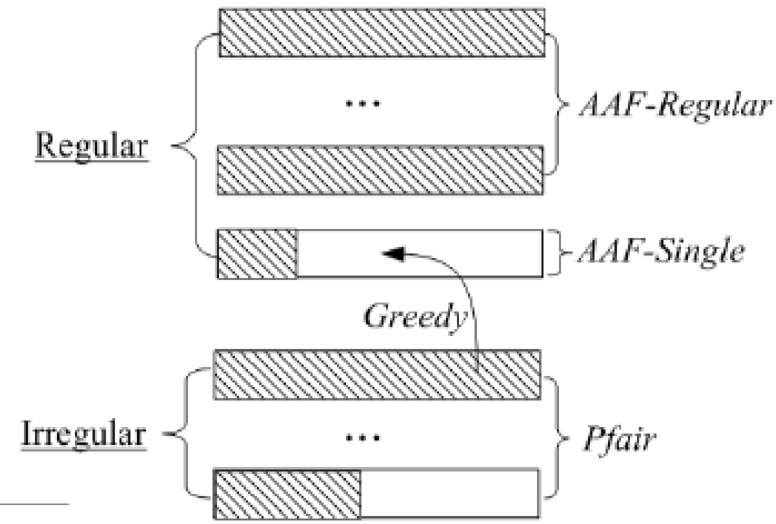
# Cons of AAF-based algorithms

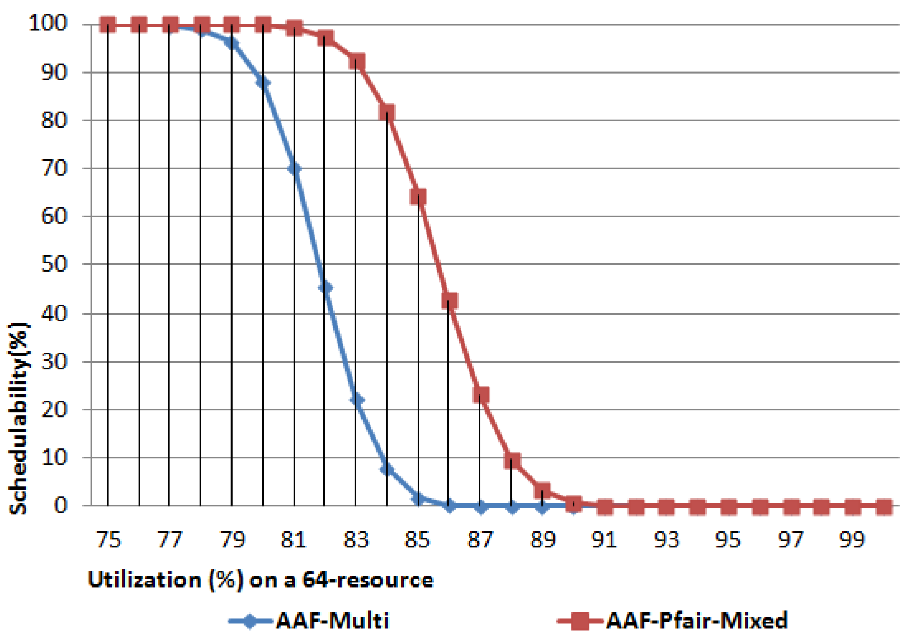
- Convert one irregular partition to multiple regular/sub-regular partitions
  - Regular/sub-regular partitions require more restricted timing constraint than irregular partitions
  - Overlap problem in multi-resource scenario

- Boundary sequence is <1, 0.5,0.25,0.125, …>
  - Not dense enough?

# Involve Pfair for Irregular Partitions



# Pros of Pfair-Mixed

- Improve the resource utilization
  - Pfair has no utilization loss
  - Greedy mitigates the resource wasting problem due to the partitioned strategy
- Eliminate the overlap problem on irregular partitions
  - Do not divide them any more

Given $<\alpha_1, \alpha_2, \ldots, \alpha_n>$ as the availability factors of $n$ regular partitions, can they be scheduled on a multi-resource platform?

# Approximating Boundary Sequence (ABS)

- Definition: a rational number sequence $<b_1, b_2, b_3, \ldots>$, where $\forall\, i > 0$, $1 > b_i > b_{i+1} > 0$, $\lim b_n = 0$ if $n \to \infty$

- A <u>Static Approximation Algorithm</u> (SAA) adjusts each availability factor to the closest greater or equal value in a specific ABS

The <u>Approximating Function</u> of an ABS $\boldsymbol{B} = <b_1, b_2, b_3, \ldots>$ for $\alpha \in (0,1)$ is

$$\boldsymbol{R}(\boldsymbol{B}, \alpha) = b_{i-1} \text{ when } b_i < \alpha \le b_{i-1},$$

where $b_0 = 1$; $b_{|\boldsymbol{B}|+1} = 0$ if $\boldsymbol{B}$ is finite.

# Feasibility of ABS

- An ABS $\boldsymbol{B}$ is **feasible** if and only if $\forall \langle \alpha_1, \alpha_2, \ldots, \alpha_n \rangle$ as the availability factors of $n$ time regular partitions, they are schedulable on $m = \left\lceil \sum_{i=1}^{n} R(B, \alpha_i) \right\rceil$ time resources after being approximated by $\boldsymbol{B}$.
  - For convenience, we derive directly that $\langle \alpha_1, \alpha_2, \ldots, \alpha_n \rangle$ is (un)schedulable under $\boldsymbol{B}$.

**Goal: Find optimal feasible ABS**

# ABS: Schedulability Bound - Y(*B*)

- If a feasible ABS $\boldsymbol{B}$ is finite, $\Upsilon(\boldsymbol{B}) = 0$.

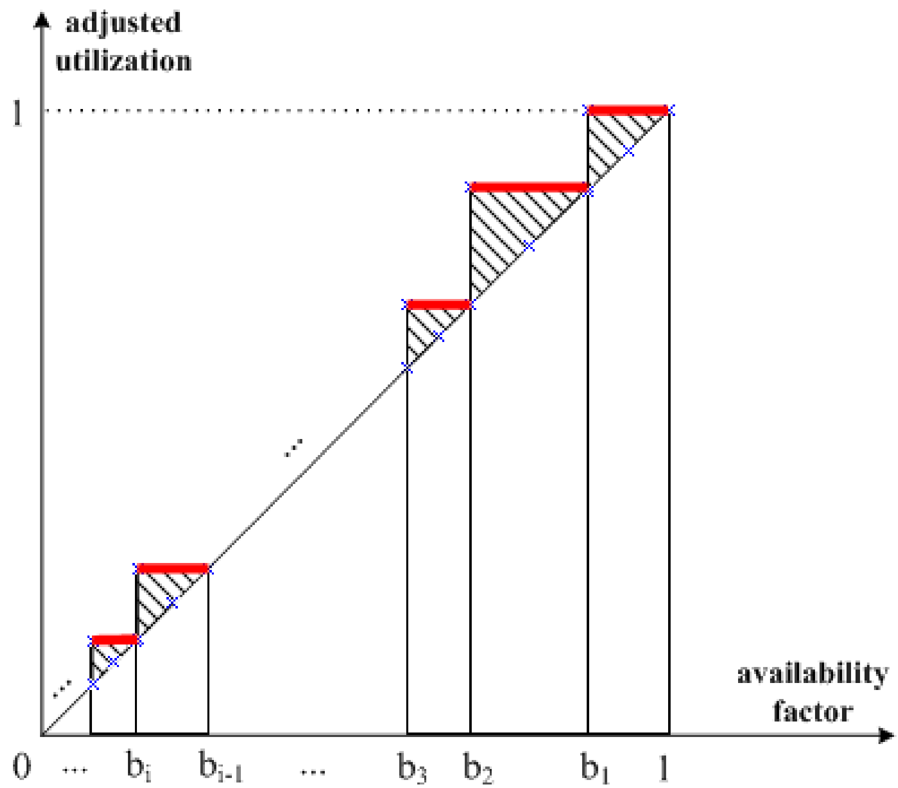
- If a feasible ABS $\boldsymbol{B} = <b_1, b_2, b_3, \ldots>$ is infinite,

$$\Upsilon(\boldsymbol{B}) = \min\{b_{i+1}/b_i : i \geq 0\},$$

where $b_0 = 1$.

**Any finite ABS is not optimal**

# ABS: Average Resource Utilization



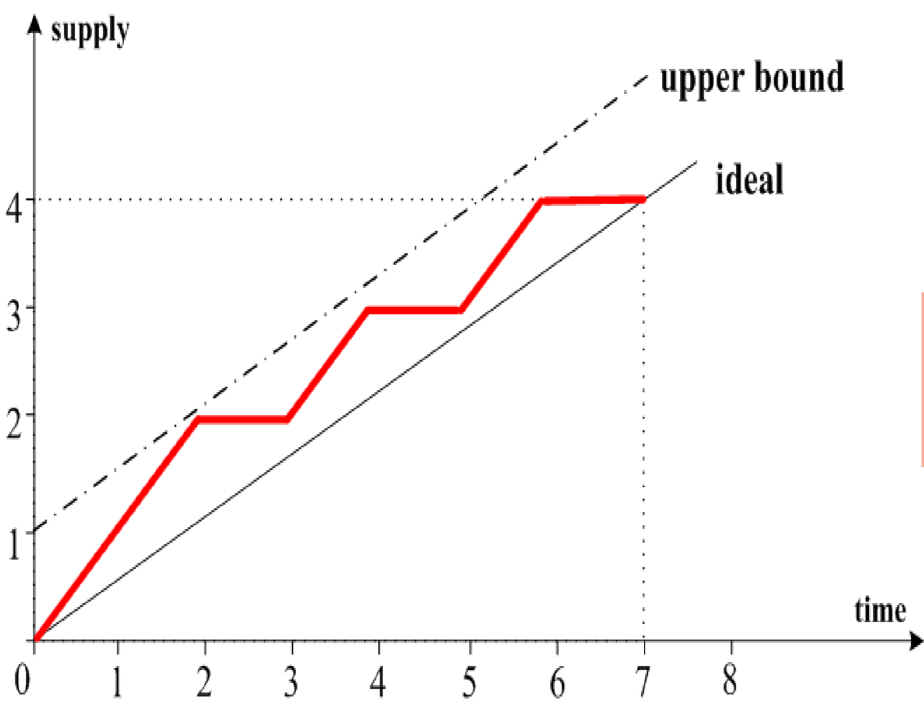

The Utilization Overhead of ABS $\boldsymbol{B}$ = <$b_1$, $b_2$, $b_3$, …> is

$$\Omega(B) = \frac{1}{2}\sum_{i\geq 0}(b_i - b_{i+1})^2,$$

where $b_0 = 1$; $b_{|\boldsymbol{B}|+1} = 0$ if $\boldsymbol{B}$ is finite.

# ABS: Standard Regular Partition

- A regular partition is uniquely determined by its availability factor except for the offset.
  - Standard Regular Partition: $T_0(p, q)$
  - General Regular Partition: $T(p,q,\delta)$




$T_0(7,4) = <0,1,3,5>;$
$T(7,4,2) = <0,2,3,5>$

# ABS: Split Regular Partitions

- If $T(p, r, \delta) \subset T_0(p, q)$, where $q+r< p$ and $r<q<2r$, then $\exists\, n > 0$,

$$\frac{q}{p} = \frac{2n+1}{4n+3}; \frac{r}{p} = \frac{n+1}{4n+3}.$$

- If $p \geq 3q$ and $r<q<2r$, $T(p, r, \delta) \subset T_0(p, q)$ is impossible.

$T(20,7,\delta)$ is not a subsequence of $T_0(20,10)$ for any $\delta \in [0,20)$.

$\forall\, p \geq 6$ but $p \neq 7$, $T(p,2,\delta)$ is not a subsequence of $T_0(p,3)$ for any $\delta \in [0,p)$.

# ABS: Schedulability Bound Theorem

**The Schedulability Bound of any feasible ABS is not greater than one-half.**

**Proof:** $\boldsymbol{B} = \langle b_1, b_2, b_3, \ldots \rangle$, $b_1 = q/p$ ($p$ and $q$ are co-prime).

$\exists m, n > 0, m * p = n * q + 1.$

If $Y(\boldsymbol{B}) > 0.5$, then $\forall i \geq 0, \frac{b_{i+1}}{b_i} > 0.5$, where $b_0 = 1$.

Hence, $p \geq 3$, and $\exists j, b_j \in (\frac{1}{2p}, \frac{1}{p})$.

$\langle b_1, b_1, \ldots, b_1, b_j \rangle$ with $n$ times $b_1$ is unschedulable under $\boldsymbol{B}$. $(n * b_1 = m - 1/p)$ ■

# ABS: Special Cases

- Geometric ABS: $\boldsymbol{G}(m) = \left\langle \frac{1}{m}, \frac{1}{m^2}, \frac{1}{m^3}, \ldots \right\rangle$
- Arithmetic ABS: $\boldsymbol{A}(m) = \left\langle \frac{n-1}{n}, \frac{n-2}{n}, \frac{n-3}{n}, \ldots, \frac{1}{n} \right\rangle$
- Hybrid ABS: $\boldsymbol{H}(n, m) = \left\langle \frac{n-1}{n}, \frac{n-2}{n}, \frac{n-3}{n}, \ldots, \frac{1}{n}, \frac{1}{n \cdot m}, \frac{1}{n \cdot m^2}, \frac{1}{n \cdot m^3}, \ldots \right\rangle$

| $\mathcal{B}$ | $\Upsilon(\mathcal{B})$ | $\mathcal{O}(\mathcal{B})$ | Feasibility |
|---|---|---|---|
| $\mathcal{G}(m)$ | $\frac{1}{m}$ | $\frac{1}{2} - \frac{1}{m+1}$ | feasible |
| $\mathcal{A}(n)$ | 0 | $\frac{1}{2n}$ | feasible iff $n$ is an RMN |
| $\mathcal{H}(n, m)$ | $\frac{1}{m}$ | $\frac{1}{2n} - \frac{1}{n^2(m+1)}$ | feasible iff $n$ is an RMN |

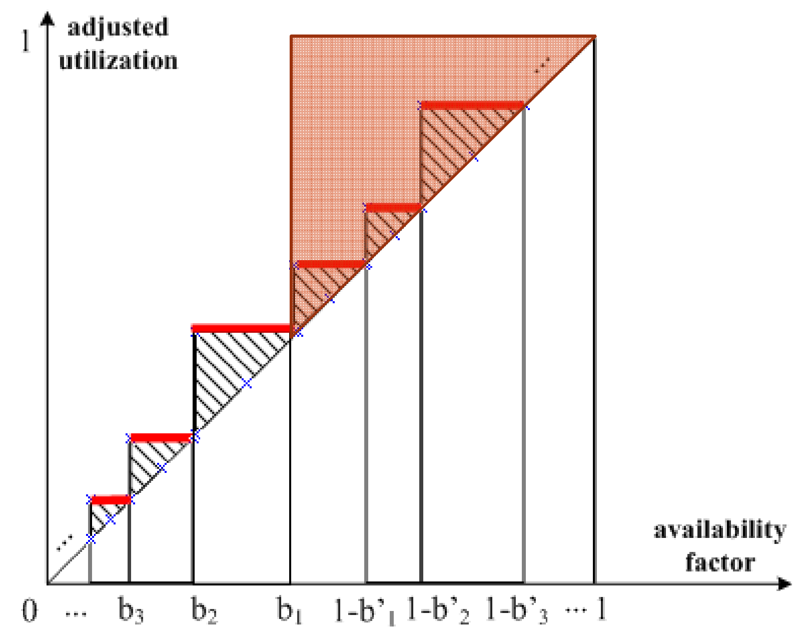
The Set of all RMNs = {2,3,4,5,7}

$\Omega(\boldsymbol{H}(7,2)) = 0.065$

$\Omega(\boldsymbol{G}(2)) = 0.167$

# Extended ABS (E-ABS)

- Definition: a tuple $\boldsymbol{E}=(\boldsymbol{B}, \boldsymbol{B'})$, where $\boldsymbol{B} = <b_1, b_2, b_3, \ldots>$, $\boldsymbol{B'}=<b'_1, b'_2, b'_3, \ldots>$ are two ABSes; $b_1+b'_1<1$ if neither $\boldsymbol{B}$ nor $\boldsymbol{B'}$ is empty
  - The boundary sequence of $\boldsymbol{E}$ is $<\ldots, 1\text{-}b'_3, 1\text{-}b'_2, 1\text{-}b'_1, b_1, b_2, b_3, \ldots>$
  - $\Omega(\boldsymbol{E}) < \Omega(\boldsymbol{B})$ if $\boldsymbol{B'}$ is not empty



# Optimal E-ABS Theorem

If a feasible E-ABS $\boldsymbol{E}$ has a different boundary sequence with $\boldsymbol{Z}(7, 2)$ and $\Upsilon(\boldsymbol{E})=0.5$, then

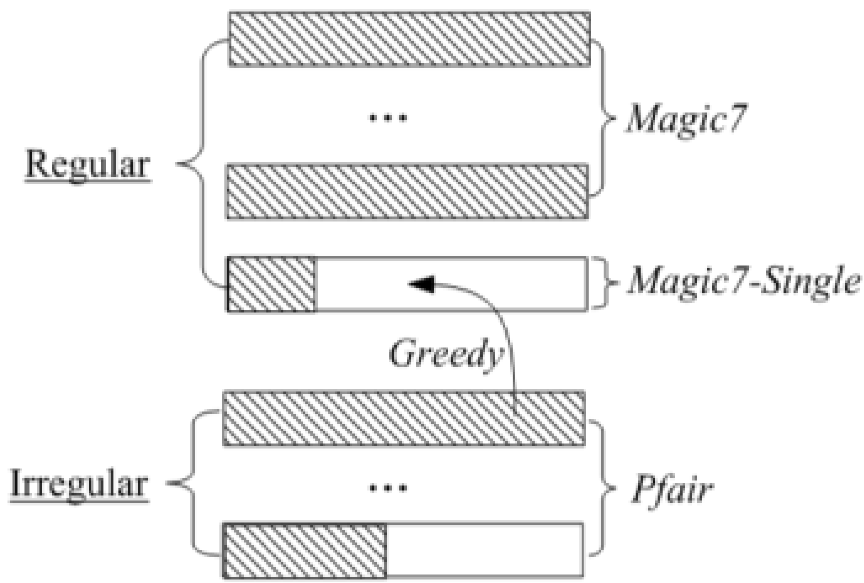
$$\Omega(\boldsymbol{E}) > \Omega(7,2),$$

$$\textbf{\textit{where}}\ \boldsymbol{Z}(\mathrm{n}, \mathrm{m}) = (\boldsymbol{H}(\mathrm{n}, \mathrm{m}), \left\langle \frac{1}{n \cdot m}, \frac{1}{n \cdot m^{2}}, \frac{1}{n \cdot m^{3}}, \ldots \right\rangle).$$

# Magic7 Algorithm

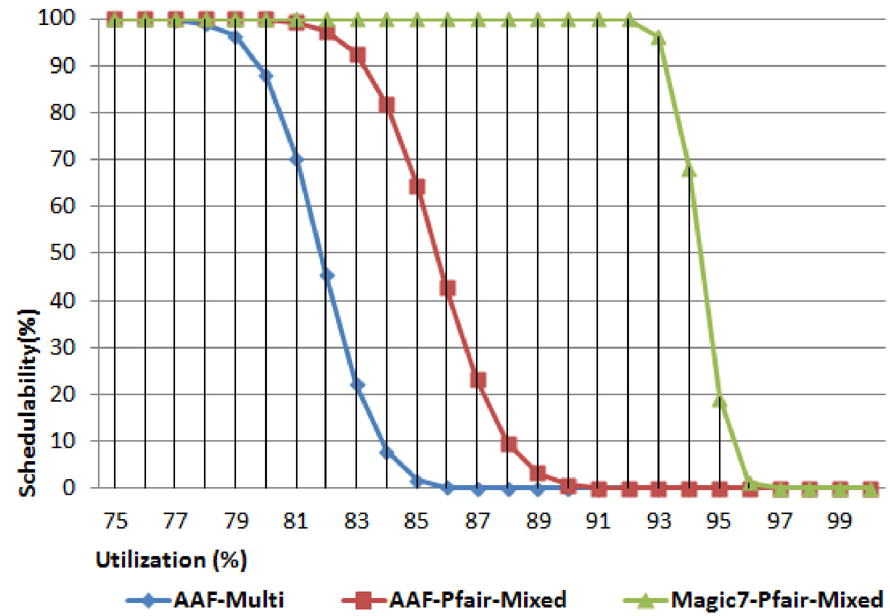
Magic7 is named for the SAA with $\boldsymbol{Z}(7,2)$, whose boundary sequence is

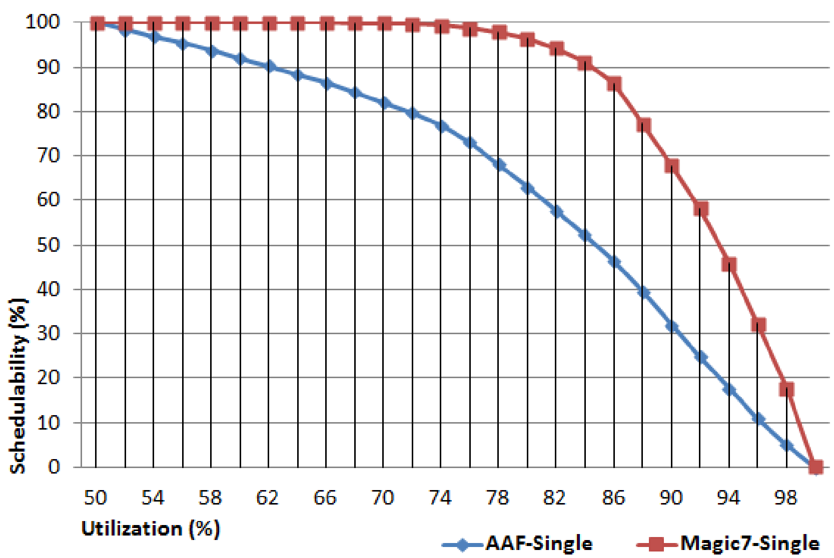
$$\left\langle \ldots, \frac{55}{56}, \frac{27}{28}, \frac{13}{14}, \frac{6}{7}, \frac{5}{7}, \frac{4}{7}, \frac{3}{7}, \frac{2}{7}, \frac{1}{7}, \frac{1}{14}, \frac{1}{28}, \frac{1}{56}, \ldots \right\rangle$$



# Experimental Result

64-resource ➔




← Single-resource

# Conclusion

- Our algorithms can improve the resource utilization of Regularity-base Resource Partitioning with a semi-partitioned scheduling strategy.
- The partition-migration overhead could be taken more into account in the future.

# Q&A