TECHNISCHE UNIVERSITÄT DRESDEN

Department of Computer Science Institute for System Architecture, Operating Systems Group

PROBABILISTIC ADMISSION CONTROL

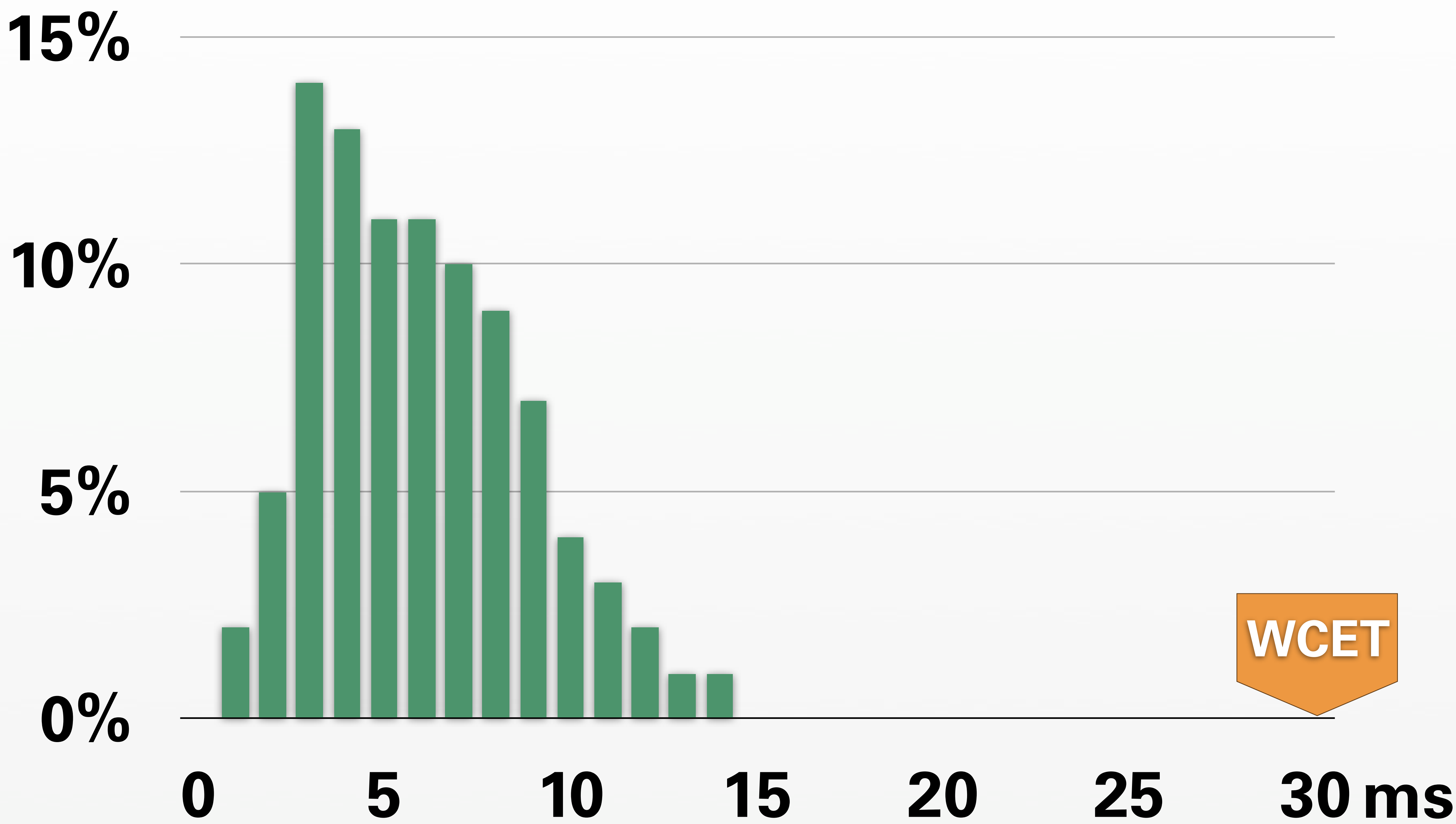
TO GOVERN REAL-TIME SYSTEMS UNDER OVERLOAD

CLAUDE-J. HAMANN, MICHAEL ROITZSCH,
LARS REUTHER, JEAN WOLTER, HERMANN HÄRTIG

MOTIVATION

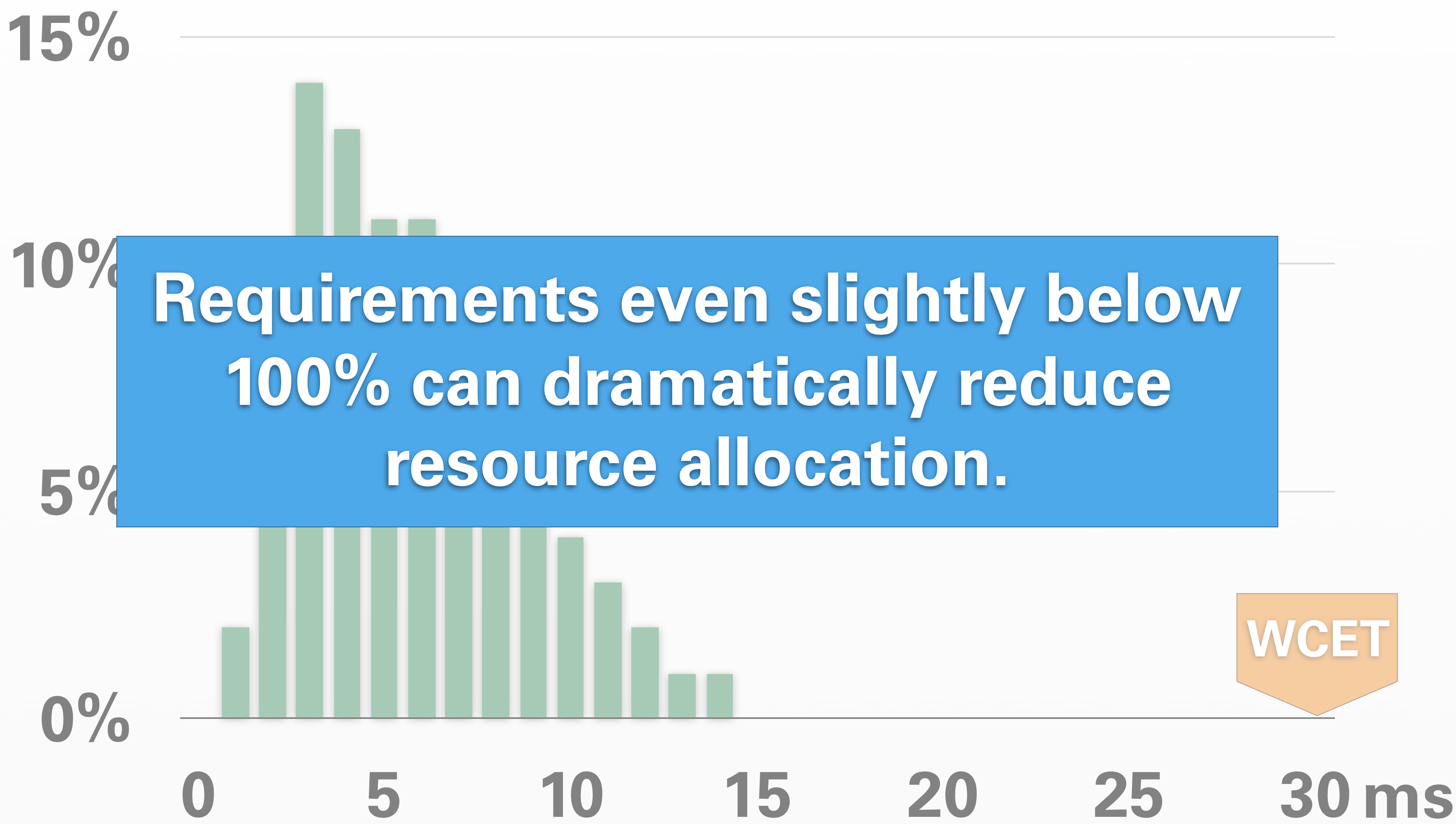
- desktop real-time
- there are no hard real-time applications on desktops
- there is a lot of firm and soft real-time
  - low-latency audio processing
  - smooth video playback
  - desktop effects
  - user interface responsiveness

H.264 DECODING



TECHNISCHE UNIVERSITÄT DRESDEN

H.264 DECODING



SRMS

- **Statistical Rate Monotonic Scheduling**
- local admission ensures percentage of successful jobs
- execution time of each job must be known in advance

PROBLEMS

- WCET largely exceeds average case
- poor utilization efficiency
- restricted to specific task types
- tough runtime requirements
- missed deadlines can at best be predicted

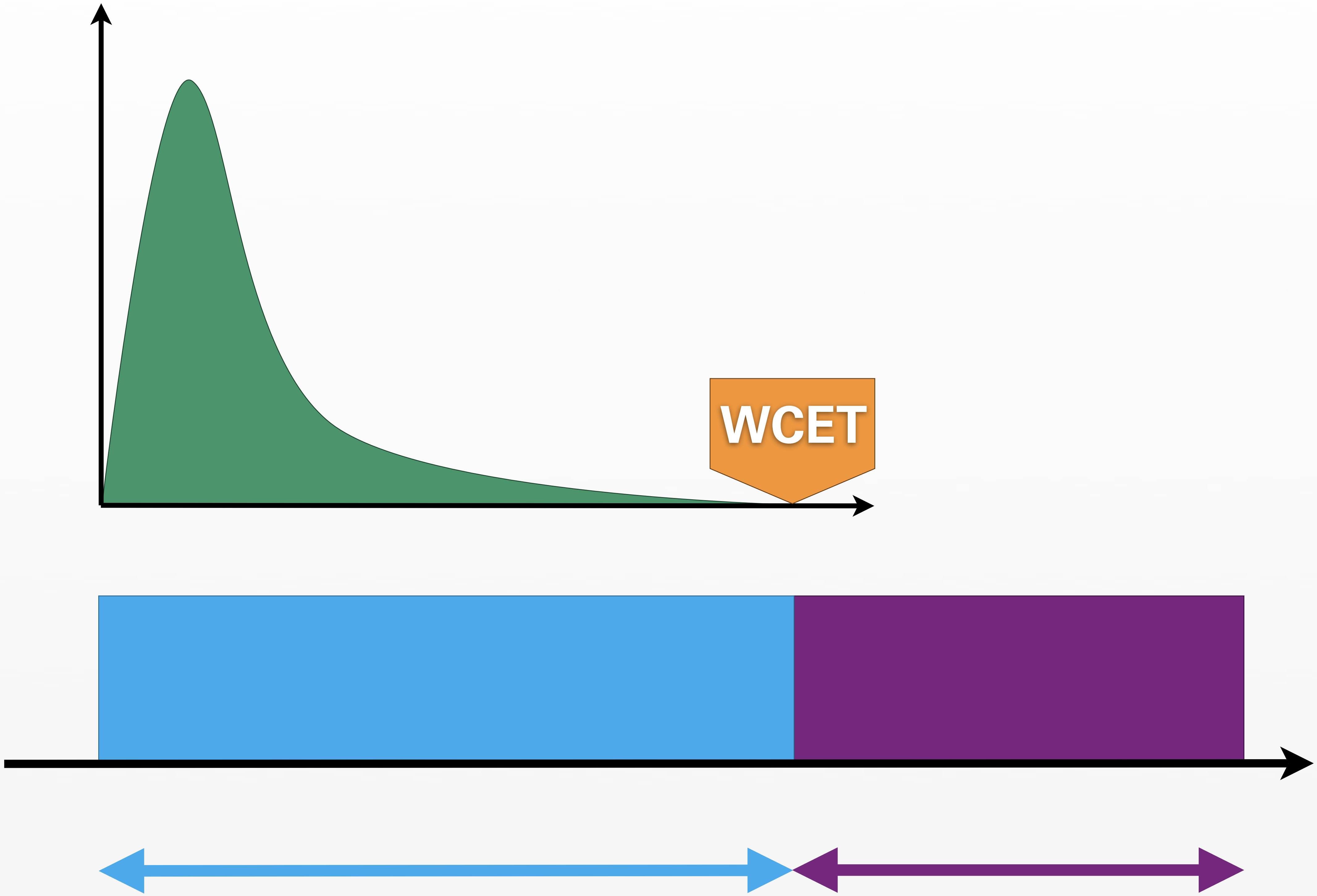
DESIGN GOALS

- use distribution instead of WCET
- relax guarantees, improve utilization
- hard, firm, preemptible, non-preemptible
- minimal runtime dispatcher requirements
- controllable fraction of missed deadlines

KEY IDEA

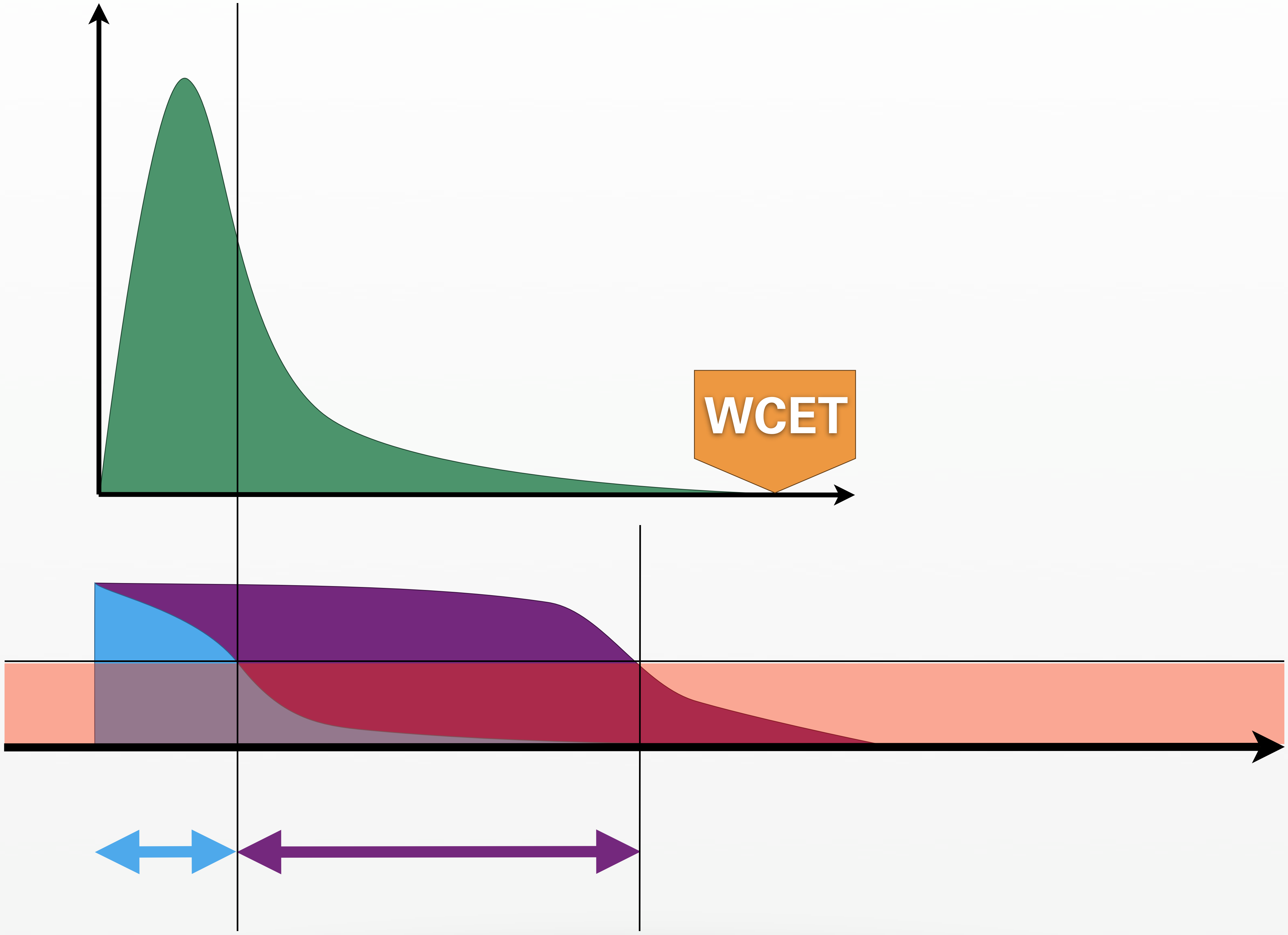
Use probabilistic admission control to model the actual run-time dispatching.

KEY IDEA

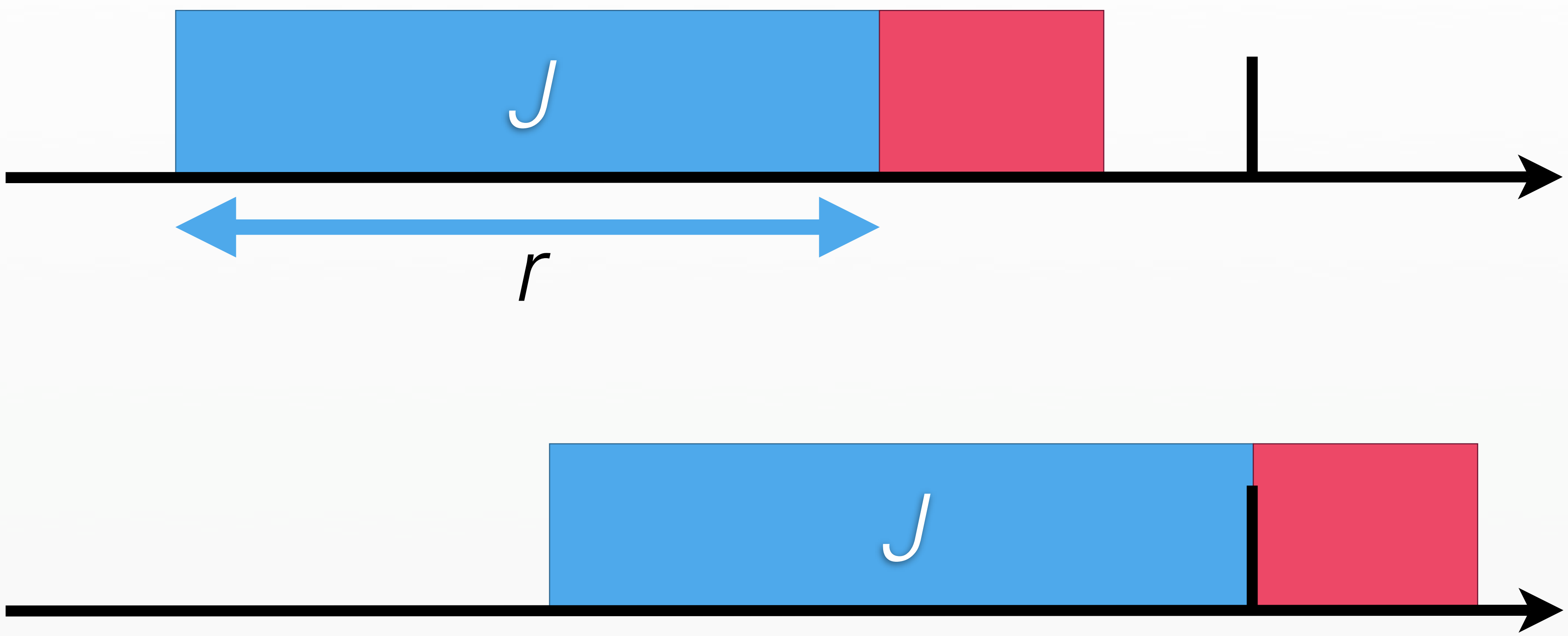


TECHNISCHE UNIVERSITÄT DRESDEN

KEY IDEA



RESERVATION



$$\mathbf{P}(J \text{ does not run longer than } r \wedge \\ J \text{ is completed until its relative deadline}) \geq q$$

TASK MODEL

- tasks T_i are sequences of periodic jobs
- period length = relative deadline d_i
- jobs are partitioned into one mandatory part and m_i optional parts
- mandatory part's execution time X_i with WCET w_i
- optional part's execution time Y_i
- quality q_i: fraction of completed optional parts

ADMISSION GOAL

priorities and reservation times for all jobs to generate a feasible schedule

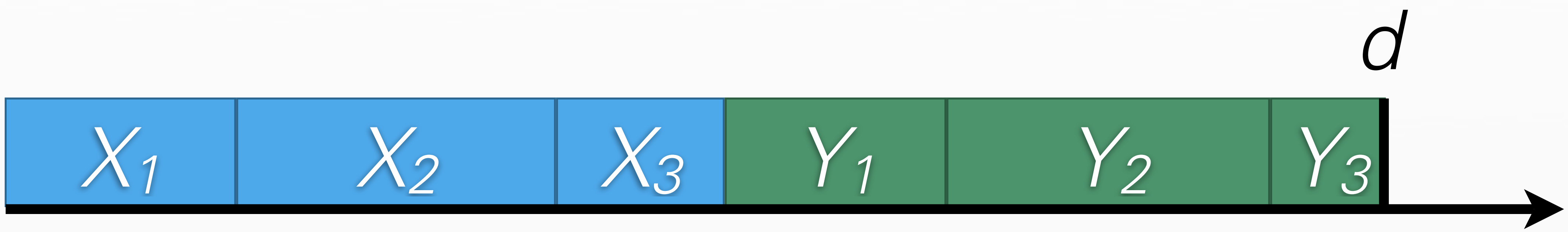
- all mandatory parts meet their deadlines
- all optional parts meet their requested qualities

- **Quality-Assuring Scheduling** (RTSS'01)
- priority assignment:
  - all mandatory parts first
  - higher quality → higher priority
- reservation times:

$$p_i(r) = \mathbf{P}(Y_i \leq r \wedge \sum_{i=1}^{n} X_i + \sum_{j=1}^{i-1} \min(Y_j, r_j) + Y_i \leq d)$$

EXAMPLE

3 Tasks: 1 mandatory, 1 optional part each



$$p_i(r) = \mathbf{P}(Y_i \leq r \wedge \sum_{i=1}^{n} X_i + \sum_{j=1}^{i-1} \min(Y_j, r_j) + Y_i \leq d)$$

DOWNSIDE

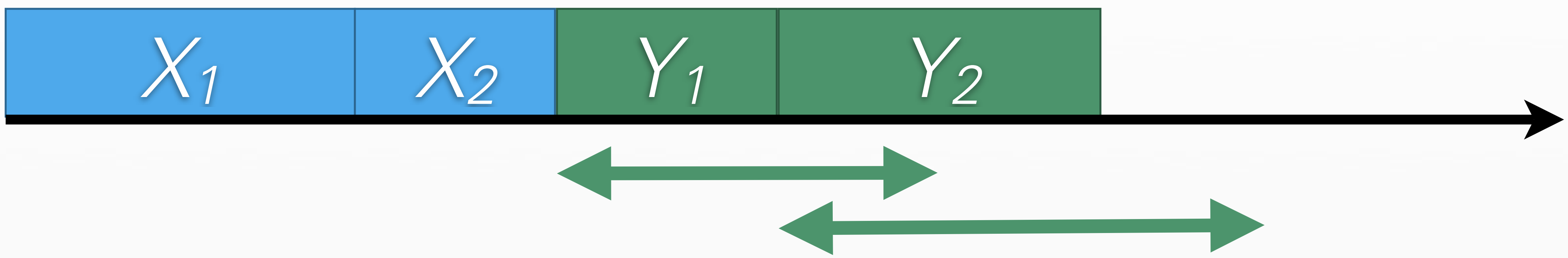
- expensive computation for arbitrary periods
- hyperperiod explodes for task sets with close-by period lengths (LCM of 503 and 510 anyone?)
- new algorithm differs in three ways
  - priority assignment
  - notion of reservation time
  - very low-cost admission

QRMS

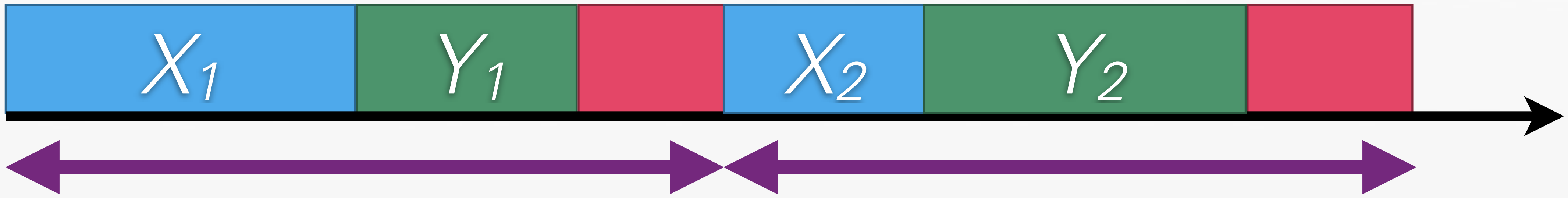
- **Quality-Rate-Monotonic Scheduling**
- cut down the exact modeling of dispatcher behavior in favor of a simpler algorithm:
  - priorities are assigned to tasks as in RMS
  - combined reservation for all parts of a job
  - reservation time regarded constant execution time in the admission
- tasks are independent for admission

EXAMPLE

QAS:



QRMS:



TECHNISCHE UNIVERSITÄT DRESDEN

RESERVATION

$$r'_i = \min(r \in \mathbb{R} \mid \frac{1}{m_i} \sum_{k=1}^{m_i} \mathbf{P}(X_i + k \cdot Y_i \leq r) \geq q_i)$$

$$r_i = \max(r'_i, w_i) \quad i = 1, \ldots, n$$

- Where is the deadline?
- consider reservation as constant execution time of a rate monotonic task
- use any RMS admission criterion
- aborting by deadline does not happen

COST

- Admission
  - computational cost dominated by convolutions
  - O(*number optional parts* × (*number of bins in distribution*)^2)
  - 5ms per part for hundreds of bins
- Runtime
  - static priorities

ACCURACY

Period	Mandatory Part	Optional Part	Requested Quality	Achieved Quality
20	N(5,1), w=6.5	N(3,1)	70%	70.23%
30	E(0.33), w=4	N(2,3)	90%	89.72%
50	E(0.25), w=2	N(5,19.5)	80%	78.44%

QRMS VS. SRMS

Period	Mandatory Part	Optional Part	Requested Quality	QRMS Quality	SRMS Quality
10	N(2,0.5), w=3	N(1.5,0.5)	70%	70.06%	85.9%
20	E(0.33), w=6	N(2,1)	50%	99.95%	77.5%
60	N(6,3), w=10	E(10)	75%	74.76%	79.3%

$$r_i = \max(r'_i, w_i) \quad i = 1, \ldots, n$$

QRMS VS. QAS

- performed simulations: random qualities, random distributions

	QAS	QRMS
uniform, optional only	++	
uniform	+	
harmonic		++

- yet to come: quantitative analysis, utilization discussion, application studies

CONCLUSION

- handles arbitrary, empiric distributions
- high utilization by probabilistic guarantees
- mandatory and optional parts, subjobs
- static priority dispatching
- intuitive quality parameter