Evaluation of existing schedulability tests for global EDF

Marko Bertogna
Scuola Superiore S.Anna,
Pisa, Italy

XRTS workshop – Vienna – September, 22
Outline

- System model
- Schedulability problem for global EDF
- Existing schedulability tests
- Evaluation metrics
- Combined the tests
- Simulations
- Conclusions and future extensions
System model

- Platform with m identical processors
- Task set \( \tau \) with n periodic or sporadic tasks \( \tau_i \)
  - Worst-case execution time \( C_i \)
  - Period or minimum inter-arrival time \( T_i \)
  - Deadline \( D_i \leq T_i \)
  - Utilization \( U_i = \frac{C_i}{T_i} \), density \( \lambda_i = \frac{C_i}{D_i} \)
Assumptions

- Independent tasks
- Job-level parallelism prohibited
  - the same job cannot be *contemporarily* executed on more than one processor
- Preemption and Migration support
  - a preempted task can resume its execution on a different processor
- Cost of preemption/migration integrated into task WCET
Global vs partitioned scheduling

- Single system-wide queue or multiple per-processor queues:

Global scheduler

- CPU1: \( \tau_1 \)
- CPU2: \( \tau_2 \)
- CPU3: \( \tau_3 \)

Partitioned scheduler

- CPU1: \( \tau_1 \)
- CPU2: \( \tau_2 \)
- CPU3: Empty
Multiprocessor anomaly

- Synchronous periodic arrival of jobs is not a critical instant for multiprocessors:

  \[ m = 2 \]
  \[ \tau_1 = (1,1,2) \]
  \[ \tau_2 = (1,1,3) \]
  \[ \tau_3 = (5,6,6) \]

from [Bar07]

Need to find pessimistic situations to derive sufficient schedulability tests
Problem window

\[ \tau_k \]

First missed deadline

\[ \varepsilon_i \]

Carry-in

\[ C_i \]

\[ D_i \]

\[ T_i \]

\[ L \]

\[ D_k \]

\[ C_k \]
Existing tests

- **GFB** (RTSJ’01)
- **BAK** (RTSS’03 → TPDS’05)
- **BAR** (RTSS’07)
- **LOAD** (ECRTS’07, ECRTS’08, RTSJ’08 → RTSJ’09)
- **BCL** (ECRTS’05 → TPDS’09)
- **RTA** (RTSS’07)
- **FF-DBF** (ECRTS’09)
Adopted techniques

- Consider the interference on the problem job
- Bound the interference with the workload
- Use an upper bound on the workload
- Existing schedulability tests differ in
  - Problem window selection: L
  - Carry-in bound $\varepsilon_i$ in the considered window
    - Amount of each contribution (BAK, LOAD, BCL, RTA)
    - Number of carry-in contributions (BAR, LOAD)
    - Total amount of all contributions (FF-DBF, GFB)
Possible metrics for evaluation

- Percentage of schedulable task set detected
  - Over a randomly generated load
  - Depends on the task generation method
- Processor speedup factor $s$
  - All feasible task sets pass the test on a platform in which all processors are $s$ times as fast
- Run-time complexity
- Sustainability and predictability properties
  - Tests still succeeds if $C_i \downarrow$, $T_i \uparrow$, $D_i \uparrow$
Processor speedup factor

- All feasible task sets pass the schedulability test on a platform in which all processors are $s$ times as fast.
- Phillips et al.’97: Each collection of jobs that is feasible on $m$ processors can be scheduled with EDF when processors are $(2 - \frac{1}{m})$ times as fast.
- A test is better if its speedup bound $\rightarrow (2 - \frac{1}{m})$.
Sustainability

- A *scheduling algorithm* is **sustainable** iff schedulability of a task set is preserved when
  1. decreasing execution requirements
  2. increasing periods of inter-arrival times
  3. increasing relative deadlines
- Baker and Baruah [ECRTS’09]: global EDF for sporadic task sets is sustainable w.r.t. points 1. and 2.
- Sustainable *schedulability test*
Theorem: (GFB). A task set $\tau$ is schedulable with global EDF if

$$\lambda_{\text{tot}} \leq m(1 - \lambda_{\text{max}}) + \lambda_{\text{max}}.$$ 

- Density-based test $\rightarrow$ linear complexity
- Sustainable w.r.t. all parameters
Theorem 1 (BAK): A task set $\tau$ is schedulable with global EDF if, for all $\tau_k \in \tau$, there is a $\lambda \in \{\lambda_k\} \cup \{U_\ell | U_\ell \geq \lambda_k, \ell < k\}$ such that

$$\sum_{\tau_i \in \tau} \min(1, \beta_{i,k}(\lambda)) \leq m(1 - \lambda) + \lambda,$$

- Polynomial complexity: $O(n^3)$
- Pessimistic version $O(n^2)$
- Not so good performances
When $U_{\text{tot}} < m \rightarrow \text{pseudo-polynomial complexity}$
LOAD

\[ \text{LOAD} = \max \sum_{\tau \in \tau} \frac{DBF_i(t)}{t} \]

- Computation is exponential in the worst-case
- Polynomial and pseudo-polynomial approximations
Theorem (LOAD) A task set $\tau$ is schedulable with global EDF if

$$\text{LOAD} \leq \max\{\mu - \lambda_{\text{max}}^\mu, ([\mu]-1) - \lambda_{\text{max}}^\mu - 1\},$$

where $\mu = m - (m-1)\lambda_{\text{max}}$, and $\lambda_{\text{max}}^x$ is the sum of the $(\left\lfloor x \right\rfloor - 1)$ largest densities among all tasks.

- Sustainable
- Proc. Speedup bound of

$$\frac{2(m-1)}{(3m-1) - \sqrt{5m^2 - 2m + 1}},$$

$\approx 2.62$ as $m \to \infty$. 
Problem window = problem job

Compute slack lower bound of task \( \tau_k \) as:

\[
D_k - C_k - \left[ \frac{1}{m} \sum_{i \neq k} \min (\mathcal{J}_k^i, D_k - C_k + 1) \right]
\]

Use slack lower bounds to compute an iteratively refined bound on the workload

Polynomial complexity: \( O(n^2N) \)
An upper bound on the WCRT of task $k$ is given by the fixed point of $R_k$ in the iteration:

$$R_k^{ub} \leftarrow C_k + \left[ \frac{1}{m} \sum_{i \neq k} \min \left( \mathcal{M}_i(R_k), \tau_i, R_k^{ub} - C_k + 1 \right) \right]$$

- Iteratively refine the response time bounds using already computed values
- Pseudo-polynomial complexity
- Sustainable w.r.t. task periods
FF-DBF

\[ FF_{DBF}(t, \sigma) = \left\lfloor \frac{t - D_i}{T_i} \right\rfloor C_i + \sigma \left( (t - D_i) \mod T_i + \frac{C_i}{\sigma} \right) \]

**Theorem (FF-DBF)**

A task set \( \tau \) is schedulable with global EDF if \( \exists \sigma \mid \lambda_{\text{max}} \leq \sigma < \frac{m - U_{\text{tot}}}{m - 1} - \epsilon \) (with an arbitrarily small \( \epsilon \)), such that \( \forall t \geq 0 \),

\[
\sum_{\tau_i \in \tau} FF_{DBF}(t, \sigma) \leq (m - (m - 1)\sigma)t
\]

- Pseudo-polynomial complexity
- Best processor speedup bound: \( (2 - \frac{1}{m}) \)
Dominance relations

- FF-DBF dominates GFB
- RTA dominates BCL
- All other tests are *incomparable*!

- A more efficient test (COMP) can be obtained composing existing techniques
Combining the tests (COMP)

- Apply RTA, storing the computed slacks
- If RTA fails, apply BAR refining the interference with the computed slacks
- If still no success, apply FF-DBF
- If the test fails, the task set is not feasible on a platform $1/ (2 - \frac{1}{m}) = \frac{m}{2m-1}$ times as fast

- Other tests are either dominated by COMP or would lead to insignificant improvements
Simulations

- No exact test is known for multiprocessor feasibility nor for EDF-schedulability
- $10^6$ task sets generated for each experiment
- All task sets satisfy the tightest known feasibility condition for sporadic task sets (Baker’06)
Simulations

m = 2
σ = 0.25
Simulations

- Generated task sets

Graph showing the number of detected task sets with different task set utilizations. The graph plots the number of task sets against task set utilization.

- TOT
- COMP
- RTA
- BCL
- FF-DBF
- GFB
- LOAD
- BAR
- BAK

- Parameters:
  - $m = 4$
  - $\sigma = 0.25$
Simulations

\[ m = 8 \]
\[ \sigma = 0.25 \]
Simulations

\[ m = 2 \]
\[ \sigma = 0.5 \]
Conclusions

- Different techniques led to many incomparable EDF schedulability tests
- Test selection depends on adopted metric
- Best performances given by COMP, although with pseudo-polynomial complexity
- For faster tests
  - BCL \(\rightarrow\) good results with polynomial complexity
  - GFB \(\rightarrow\) linear complexity and sustainability
- Future work: consider blocking due to shared resources
Marko Bertogna
marko@sssup.it
http://retis.sssup.it/~marko