Sanjoy Baruah

Techniques for Multiprocessor Global Schedulability Analysis

Presented by Artem Burmyakov

Guideline

Introduction to cluster scheduling

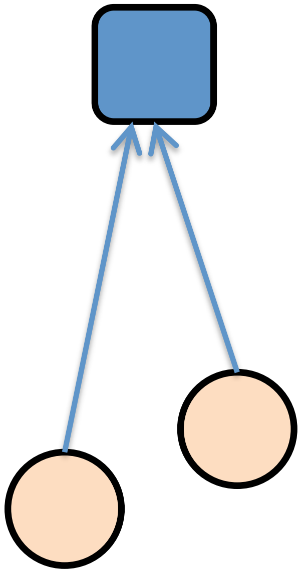
Problem Statement

Prior Results

An improved schedulability algorithm

Conclusions

Introduction: Scheduling

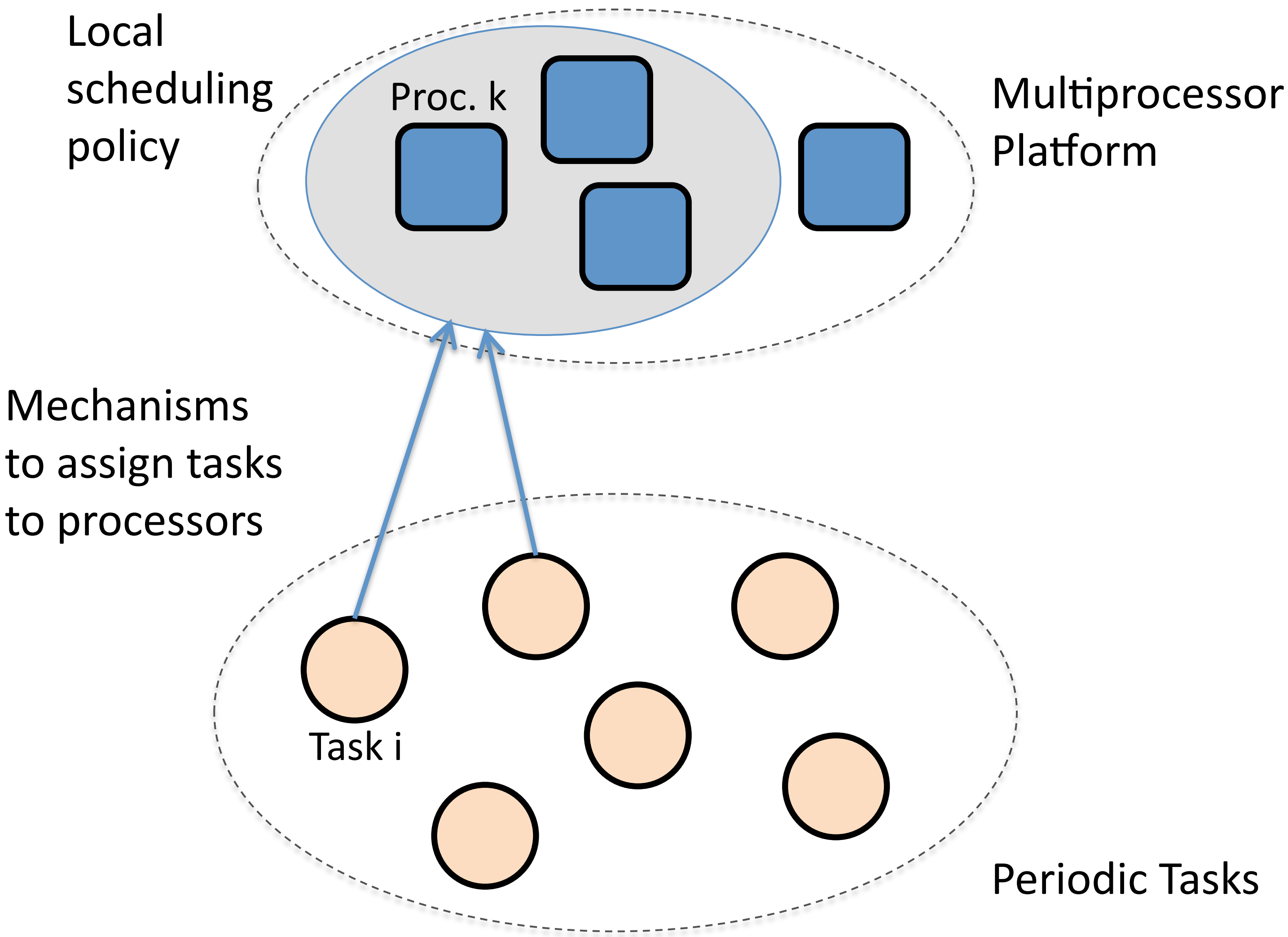


Scheduling algorithm – mechanism to allocate processor time to jobs in order to meet all deadlines.

Schedulability test – verification if a given algorithm can schedule all permissible combinations of jobs, generated by tasks.

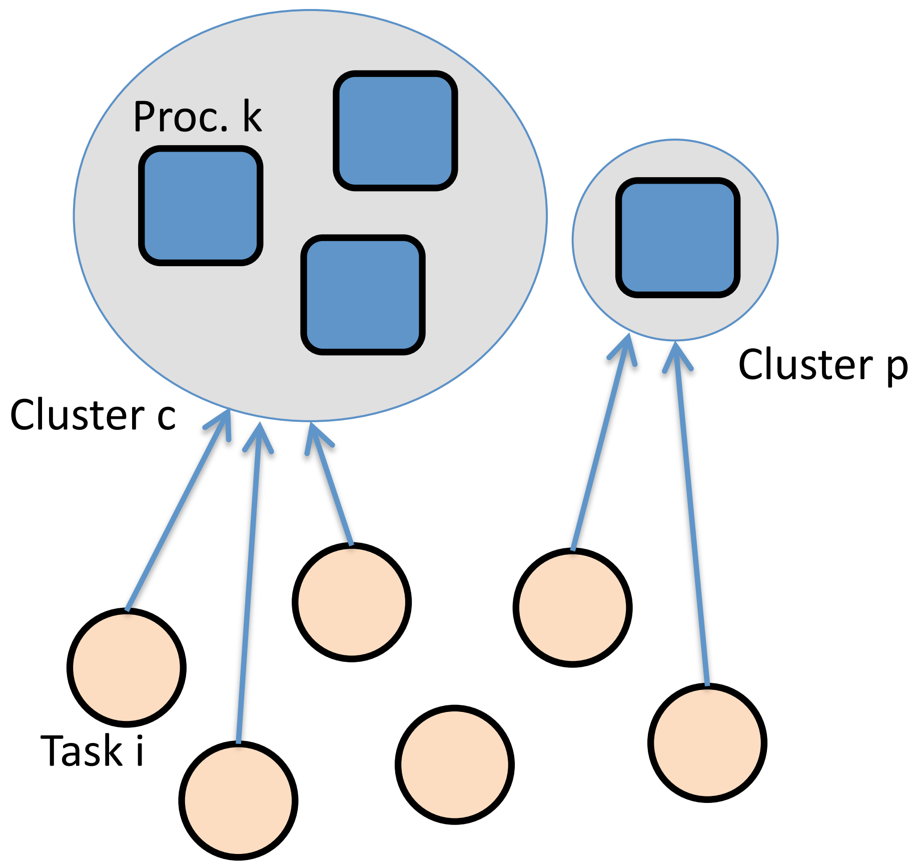
Principle of test: to identify worst-case job arrival sequences and check if they are schedulable.

Introduction: Multiprocessor Platform



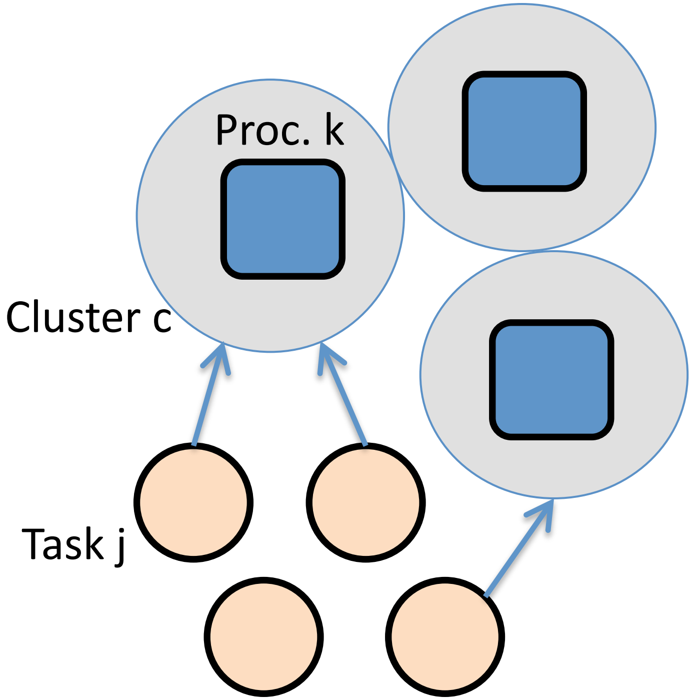
Introduction: Clustering

Cluster – a group of processors.

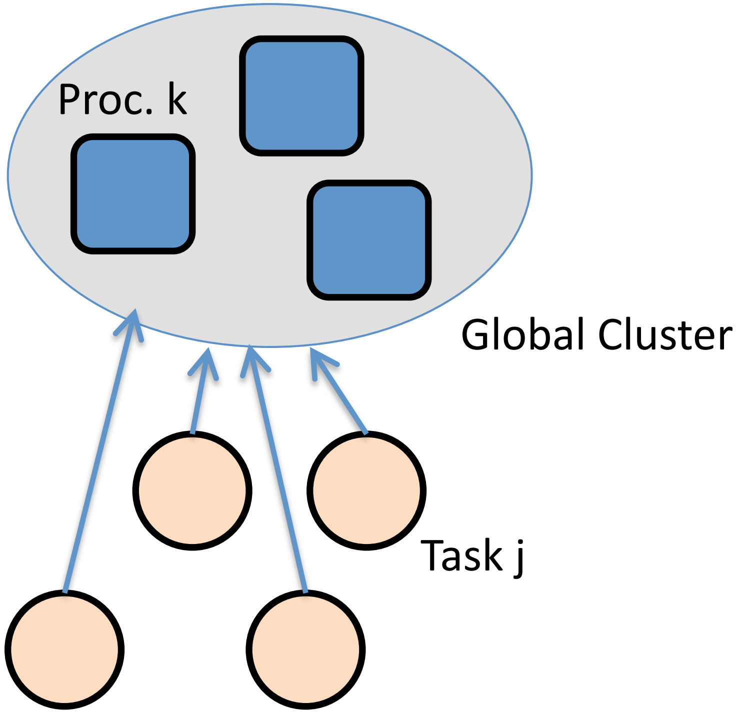


Problem Statement

Partitioned Scheduling



Global Scheduling

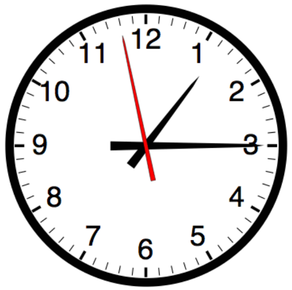


Much better understood;
Sufficient schedulability tests;
Simulations proving tests' effectiveness

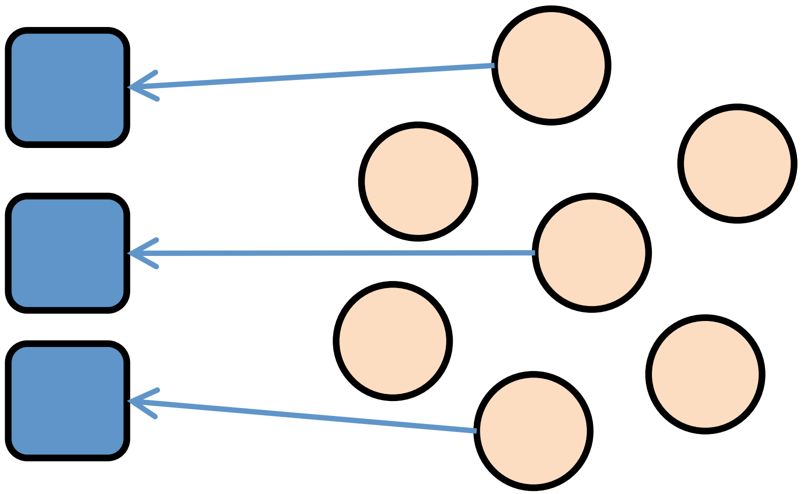
Only trivial theoretical bounds on performance

Global-EDF

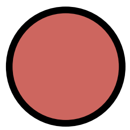
EDF = Earliest Deadline First

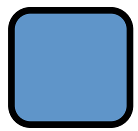


At every
time instant



m processors n tasks

Select **m** tasks  with jobs having earliest deadlines

Assign selected tasks to processors 

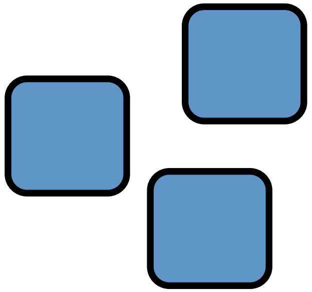
Purpose of this Paper

Study of the Global Scheduling:

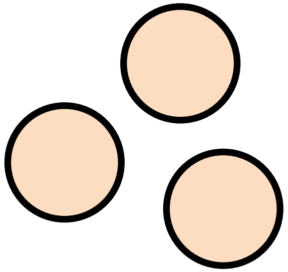
- to identify problems of schedulability analysis;
- to propose a new **global-EDF** schedulability test.

Achieved techniques to be applied to other scheduling algorithms.

Model Definition



Processors are identical;
Preemptive execution.



Task system:

$$\tau = \{\tau_1, \tau_2, .., \tau_n\}$$

Sporadic tasks:

$$\tau_i = (C_i, D_i, T_i),$$

worst-case execution time → C_i; relative deadline → D_i; minimum inter-arrival separation → T_i

Worst-case for Global Scheduling

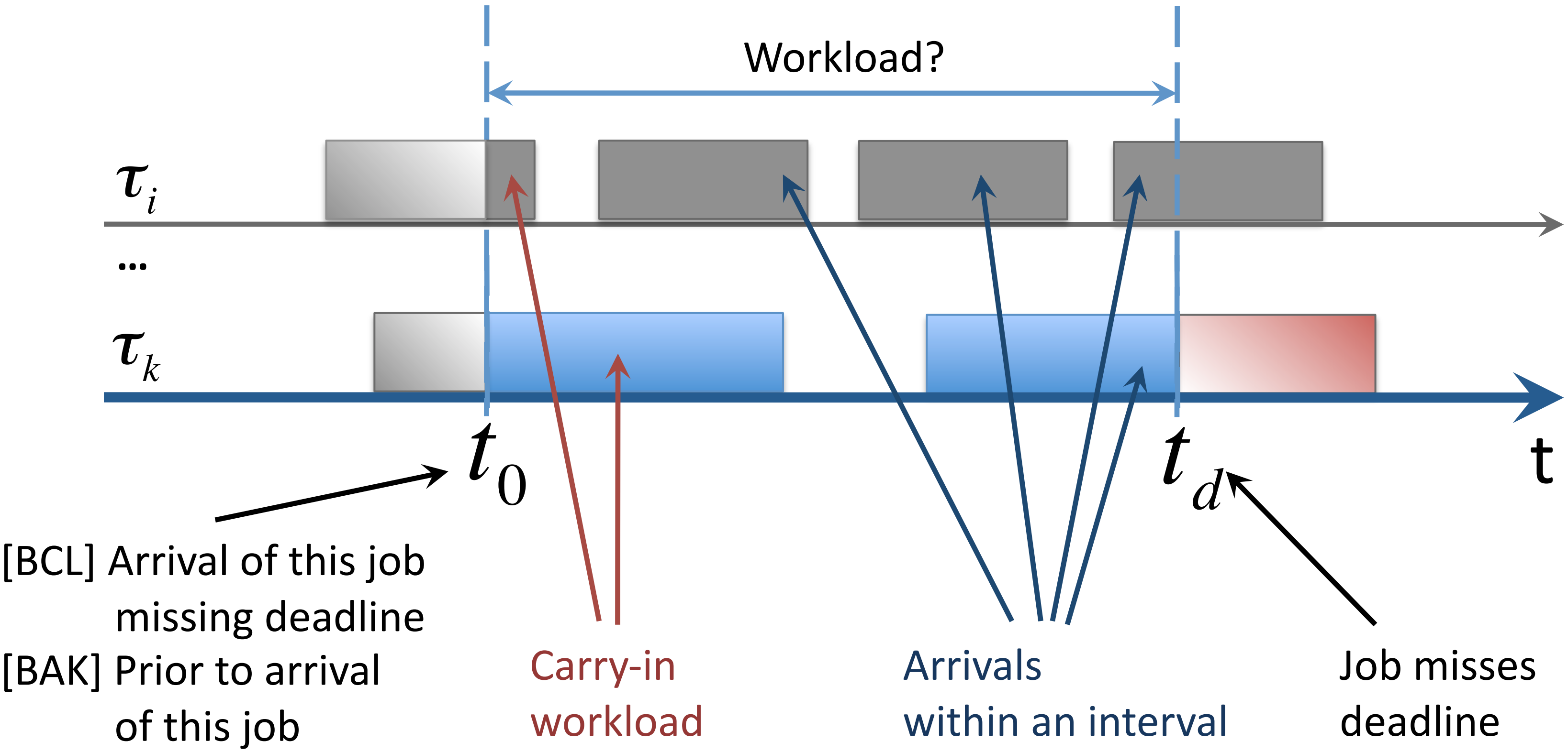
Partitioned Scheduling (using EDF, RM):
synchronous arrival sequences

Global Scheduling (e.g. using EDF):
synchronous arrival sequences
are not the worst case

[BAK] and [BCL] tests

The concept is similar:

to build unschedulability condition



[BAK]: less pessimistic, more sophisticated;
[BCL]: trivial techniques.

Main Shortcoming of [BAK] and [BCL]

[BAK] and [BCL] tests are very pessimistic

Tests give overestimated carry-in load (carry-in of all n tasks is counted)

Tests perform poorly:
they flag systems as "unschedulable",
while they are schedulable

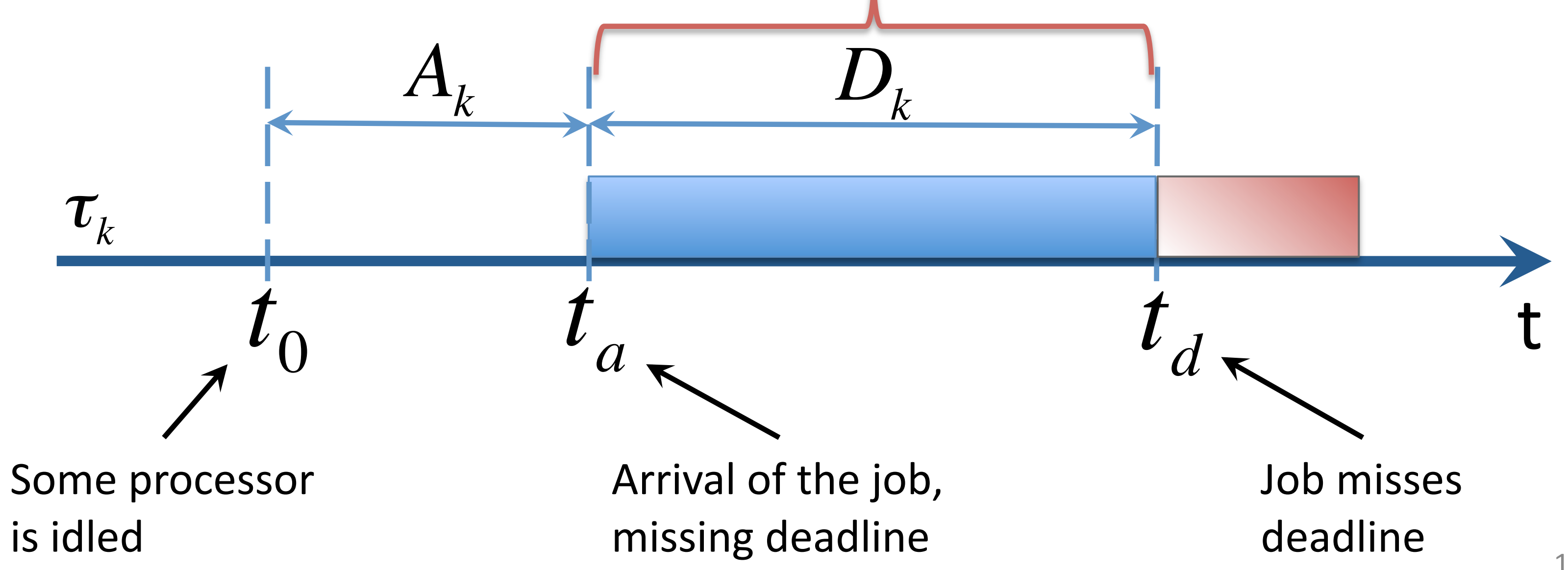
An Improved Schedulability test (1)

Idea: to define schedulability condition for each task

Condition to miss deadline for τ_k :

τ_k executes for less than C_k time units

⇒ All processors are busy with HP tasks for more than $(D_k - C_k)$ each (set of intervals Γ_k)

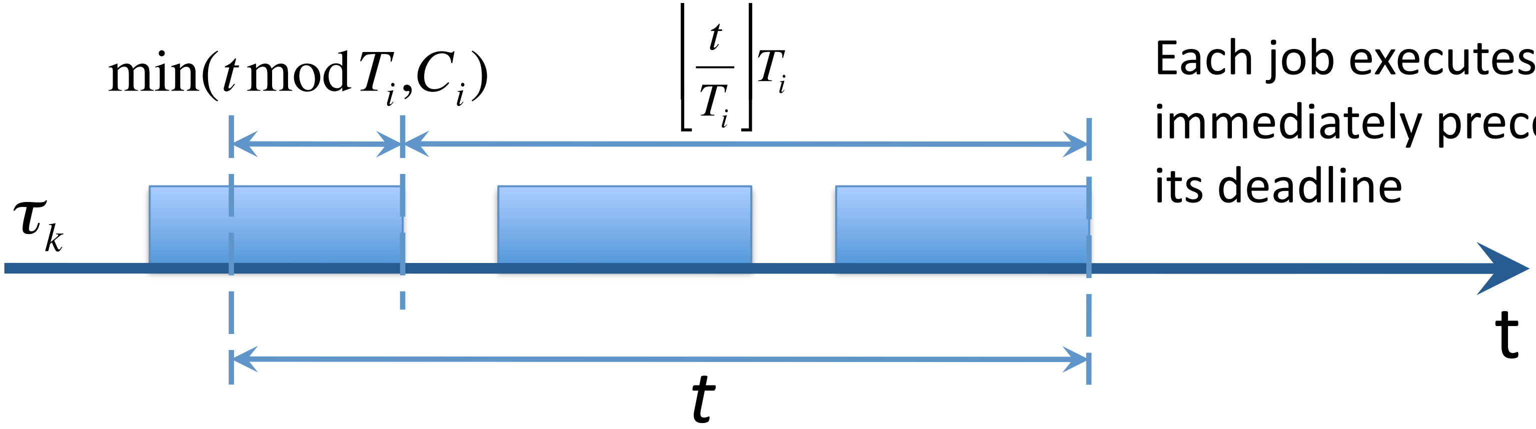


An Improved Schedulability test (2)

Contribution computation of each task
over $[t_0, t_a) \bigcup \Gamma_k$ (with max. length of $A_k + (D_k - C_k)$):

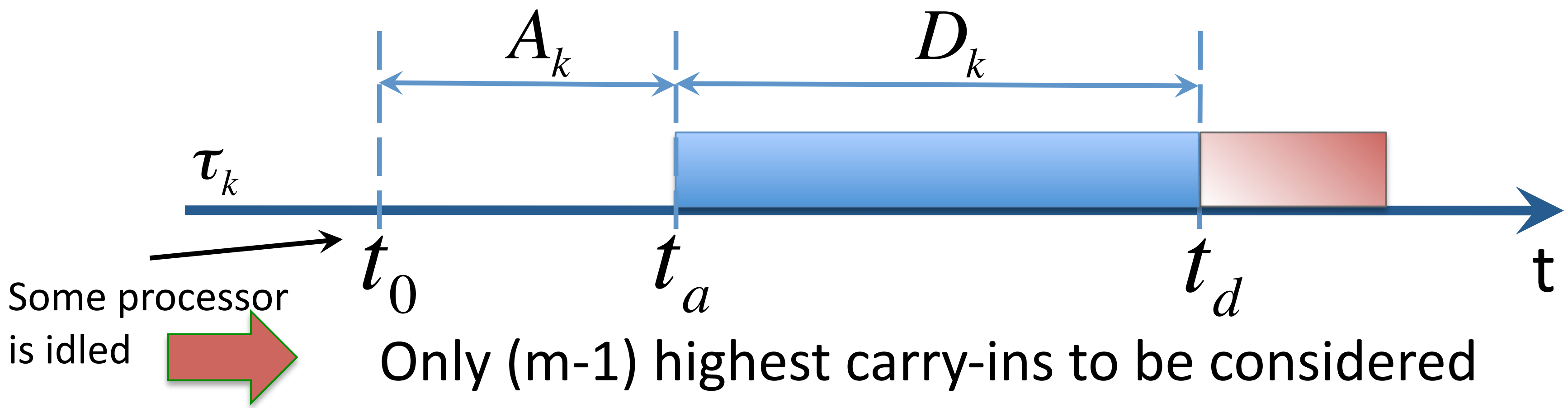
- **not** considering carry-in load
  min(DBF, maximum intervals length)

- considering carry-in load
  introduced DBF':



Each job executes immediately preceding its deadline

An Improved Schedulability Test (3)



Only (m-1) highest carry-ins to be considered

WORKLOAD OVER $[t_0, t_a) \bigcup \Gamma_k$ = **CONTRIBUTIONS WITHOUT CARRY-IN** + **(m-1) HIGHEST CARRY-INS**

SCHEDULABILTY CONDITION FOR τ_k : **WORKLOAD OVER $[t_0, t_a) \bigcup \Gamma_k$** $\leq m(A_k + D_k - C_k)$

Conclusions

Identified problems of the analysis of the Global Scheduling

Proposed a new schedulability test for global-EDF algorithm (less pessimistic)

Developed techniques, which are applicable to other scheduling algorithms, not only EDF