Uniprocessor Scheduling under Precedence Constraints

PARADES G.E.I.E.
Massimo Baletti, Alberto Ferrari, Leonardo Mangeruca, Alberto Sangiovanni-Vincentelli

Outline

• The meaning of precedence constraints
• Motivating example
• Conditions for scheduling under precedence constraints
  – General necessary and sufficient conditions
  – Sufficient conditions for relevant scheduling techniques
• Optimization problems
• Conclusions
Model Based Design

- Emerging solution to embedded software design issues
  - from manual coding and informal specifications ...
  - to capturing functional and non-functional requirements by mathematical models

Motivation

Abstract: “zero computation time”

Implementation: non-zero computation time
New Precedence Constraints

- $\tau_j$ precedes $\tau_i$: if the $T_j$ event arrives before the $T_i$ event, the corresponding job of $\tau_j$ completes before the corresponding job of $\tau_i$ starts.
- Tasks subjected to precedence constraints may have different periods.
- Deterministic communication in both data-flow and event driven systems.

**Def.:** $\tau_j(k)$ precedes $\tau_i(h)$, denoted $\tau_j(k) \leq \tau_i(h)$, if $f_j(k) \leq s_i(h)$

$$a_j(k) \leq a_i(h) \implies \tau_j(k) \leq \tau_i(h)$$

Motivating example

Data is still in the buffer when Consumer executes

$$a_i(h) \leq a_j(k+1) \implies \tau_i(h) \leq \tau_j(k+B_{len})$$
General Precedence Constraints

\[ a_i(h) \leq a_j(k) \Rightarrow \tau_i(h-d_{i,j}) \leq \tau_j(k+e_{i,j}) \]
\[ a_j(k) \leq a_i(h) \Rightarrow \tau_j(k-d_{j,i}) \leq \tau_i(h+e_{j,i}) \]

Deriving the Precedence Conditions

\[ a_i(h) \leq a_j(k) \Rightarrow \tau_i(h) \leq \tau_j(k+1) \]
\[ a_j(k) \leq a_i(h) \Rightarrow \tau_j(k) \leq \tau_i(h) \]
Notation: offset, idle, release
inter-release time

Precedence Condition: An Example

\[ \tau_i(h) \leq \tau_j(h+2) \text{ if } 2^kT_j + O_{ij} \geq R_i \]

Sufficient condition
Precedence Relation

Def: \( \tau_i(h-1) \leq \tau_j(k+1) \Leftrightarrow f_i(h-1) \leq s_j(k+1) \)

Precedence Relation

\[ T_i(h-1) - R_i(h-1) + O_{ij}(h,k) + T_j(k,k+1) + I_j(k+1) \geq 0 \]
The priority relation

- **Priority function**: $P_i(h,t)$
  - priority of job $\tau_i(h)$ at time $t$
- **Priority relation**: $p_{ij}(h,k)$
  - $p_{ij}(h,k)=1$ if $P_i(h,t) > P_j(k,t)$ for all $t$
  - $p_{ij}(h,k)=0$ otherwise

$p_{ij}(h,k)=1$ implies that $\tau_j(k)$ cannot preempt $\tau_i(h)$

Formally: $p_{ij}(h,k)=1$ and $r_i(h) \leq s_j(k) \Rightarrow \tau_i(h) \leq \tau_j(k)$

Precedence Constraint with Priorities

$p_{ij}(h-1,k+1)=1$ and $r_i(h-1) \leq s_j(k+1) \Rightarrow \tau_i(h-1) \leq \tau_j(k+1)$

$t_i(h-1) \leq s_j(k+1)$

$T_i(h-1,h) + O_{ij}(h,k) + T_j(k,k+1) + I_j(k+1) \geq 0$
Precedence Constraints: Necessary and Sufficient Condition

without priorities:

\[ T_i(h-d_{ij},h) - R_i(h-d_{ij}) + O_{ij}(h,k) + T_j(k,k+e_{ij}) + I_j(k+e_{ij}) \geq 0 \]

with priorities:

\[ p_{ij}(h-d_{ij},k+e_{ij}) = 1 \ and \ T_i(h-d_{ij},h) + O_{ij}(h,k) + T_j(k,k+e_{ij}) + I_j(k+e_{ij}) \geq 0 \]

joining the two conditions together:

\[ T_i(h-d_{ij},h) - (1-p_{ij}(h-d_{ij},k+e_{ij})) * R_i(h-d_{ij}) + O_{ij}(h,k) + T_j(k,k+e_{ij}) + I_j(k+e_{ij}) \geq 0 \]

Application to Relevant Scheduling Techniques
Fixed Priority Scheduling: Sufficient Condition

General necessary and sufficient condition:

\[ T_i(h-d_{i,j}) - [(1-p_{i,j}(h-d_{i,j},k+e_{i,j}))*R_i(h-d_{i,j}) + O_{i,j}(h,k) + T_j(k,k+e_{i,j}) + I_{j(k+e_{i,j})}] \geq 0 \]

- \( R_i \geq R_i(h-d_{i,j}) \): upper bound on the response time
- \( T_i \leq T_i(h-1,h) \): lower bound on the inter-release time
- \( I_j \leq I_j(k) \): lower bound on idle time
- \( O_{i,j} \leq O_{i,j}(h,k) \): lower bound on offset
- \( p_{i,j} = p_{i,j}(h,k) \): fixed priority relation

Sufficient condition:

\[ d_{i,j} \cdot T_i - (1-p_{i,j}) \cdot R_i + O_{i,j} + e_{i,j} \cdot T_j + I_j \geq 0 \]

EDF Scheduling: Sufficient Condition

\[ p_{i,j}(h-d_{i,j},k+e_{i,j}) = 1 \text{ and } T_i(h-d_{i,j},h) + O_{i,j}(h,k) + T_j(k,k+e_{i,j}) + I_{j(k+e_{i,j})} \geq 0 \]

- \( D_i \): relative deadline
- \( d_i(h) = r_i(h) + D_i \): absolute deadline
- \( T_i \): lower bound on the inter-arrival time
- \( I_j \): lower bound on the idle time
- \( O_{i,j} \): lower bound on the offset

\[ p_{i,j}(h-d_{i,j},k+e_{i,j}) = 1 \Leftrightarrow d_i(h-d_{i,j}) < d_j(k+e_{i,j}) \]

\[ d_i(h-d_{i,j}) < d_j(k+e_{i,j}) \quad \text{and} \quad D_i < d_i + d_i \cdot T_i + O_{i,j} + e_{i,j} \cdot T_j \]

\[ d_i \cdot T_i + O_{i,j} + e_{i,j} \cdot T_j + I_j \geq 0 \]
Optimization Problems

Fixed Priority Scheduling

GIVEN: \( T_i, O_{ij}, I_j, D_i \)

FIND: \( p_{ij}, d_{ij}, \) and \( e_{ij} \)

SUCH THAT:

\[
\begin{align*}
& d_{ij} \cdot T_i - (1-p_{ij}) \cdot O_{ij} + e_{ij} \cdot T_j + I_j \geq 0 \\
& R_i \leq D_i \\
& d_{ij} \leq D_{ij} \text{ and } e_{ij} \leq E_{ij}
\end{align*}
\]

MINIMIZE COST: \( \sum_{ij}(A_{ij} \cdot d_{ij} + B_{ij} \cdot e_{ij}) \)

NOTE: \( R_i \) is an implicit function of \( p_{ij} \)
EDF Scheduling

**GIVEN:** \( T_i, D_j, O_{ij}, I_j, D_j \)

**LET:** \( b \) be a “small” number

**FIND:** \( a_i, a_j, d_{ij} \) and \( e_{ij} \)

**SUCH THAT:**
\[
D_i - b^*a_i < D_j - b^*a_i + d_{ij}^*T_i + O_{ij} + e_{ij}^*T_j
\]
\[
d_{ij}^*T_i + O_{ij} + e_{ij}^*T_j + I_j \geq 0
\]
\[
d_{ij} \leq D_{ij} \text{ and } e_{ij} \leq E_{ij}
\]

**MINIMIZE COST:**
\[
\sum_{i,j}(A_{ij} * d_{ij} + B_{ij} * e_{ij})
\]

---

**Conclusions**

- Generalization of precedence constraints
  - Arising from embedded systems design problems
- General necessary and sufficient conditions
  - Scheduling algorithm accounted for by priority relation function
- Sufficient conditions for relevant scheduling techniques
  - Fixed priority, EDF as special cases of general conditions
- Optimization problems for scheduling embedded systems
  - under precedence and schedulability constraints
  - Minimization of buffers for deterministic communications
- Future works
  - Efficient algorithms for optimization problems
  - Extension to multiprocessor and distributed computation
Thanks for Thanks for
Your
Attention

Notation
Precedence Relation

**Def.:** \( \tau_j(k) \) precedes \( \tau_i(h) \), denoted \( \tau_j(k) \leq \tau_i(h) \), if \( f_j(k) \leq s_i(h) \)

\[
a_j(k) \leq a_i(h) \Rightarrow \tau_j(k) \leq \tau_i(h)
\]

Motivating example

Data is still in the buffer when Consumer executes

\[
a_i(h) \leq a_j(k) \Rightarrow \tau_i(h) \leq \tau_j(k+B_{len}-1)
\]
Bounding the Inter-release Time

- $T_i \leq T_i(h-1,h)$: lower bound on the inter-release time
- $T_i \leq T_i(h-1,h) \Rightarrow d_{ij} \cdot T_i \leq T_i(h-d_{ij},h)$

![Diagram showing the inter-release time bounds](image_url)