Scheduling for embedded systems with multiple real-time constraints

Yves SOREL, Liliana CUCU, OSTRE team
INRIA Rocquencourt

Outline

- Context and objectives
- State of the art
- Model and problem to solve
- Schedulability conditions
- Optimal scheduling algorithm for systems with multiple constraints in the monoprocessor case
- Distribution and scheduling for systems with multiple constraints in the multiprocessor case
- Conclusion and work in progress

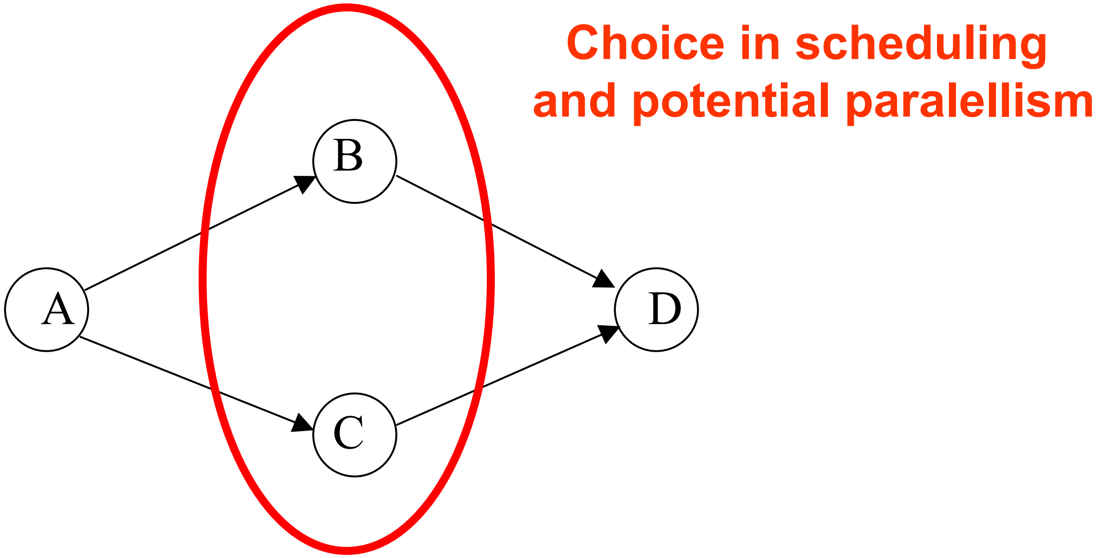
RTE systems characteristics

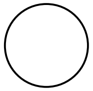
- **Functionalities**: Automatic Control, Signal & Image Processing algorithms
- **Reactive**: *Stimulus event - Operations – Reaction event*
- **Real-Time**: Constraints: Latency = bounded Reaction Time
  Cadence = bounded Input Rate
- **Distributed**: Power, Modularity, Wires minimization
  Heterogeneous Multicomponent Architecture
  - Network of Processors and Specific Integrated Circuits
  - Specific Integrated Circuits = ASIC, ASIP, FPGA, IP
- **Embedded**: Resources minimization

Algorithm-architecture adequation (AAA)

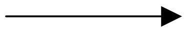
- **Global approach** based on the Synchronous Languages Semantics and the hardware RTL models
- **Unified Model:** Directed graphs
  - **A**lgorithm: Operation / Data-Conditioning Dependence
  - **A**rchitecture: FSM / Connection
  - Implementation: *distribution* and *scheduling* through graphs Transformations
- **Adequation:** Optimized Implementation (best matching)
- **Macro-Generation:**
  - Real-Time Executives for Multicomponent
  - Structural VHDL for Integrated Circuit Synthesis

Typical model: precedence constraints





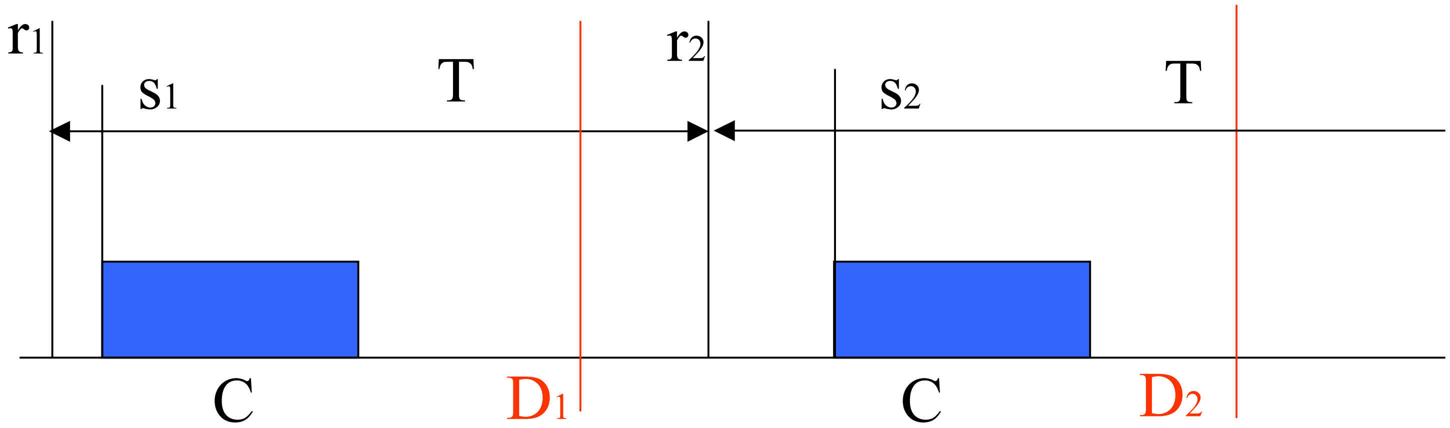
Task



Precedence

Directed Acyclic Graph (DAG)

Typical model: real-time constraints



- Period: T
- Deadline: D
- Computation time: C
- Release time: r
- Start time: s

State of art: tasks with periodicity constraints

Processors / characteristics / optimality criterium

1 / T = D / -	RMS : optimal for static assignment
1 / T ≤ D / -	DM : optimal for dynamic assigment
1 / T ≤ D, r / -	NP-hard (non-preemptive)
m / T = D / -	sufficient and necessary condition
m / T≤ D / -	NP-hard
m / strict T / -	NP-hard (non-preemptive)

State of art: tasks with precedence constraints

- 1 / prec, D / min L_{max} EDD – optimal
- 1 / prec / f_{max} Lawler – optimal
- 1 / prec, r, D / - NP-hard
- 1 / prec, r / f_{max} Baker – $O(n^2)$ (preemptive)
- 1 / prec, r / min L_{max} NP-hard
- 1 / $s_A - s_B < a_{AB}$ / min schedule

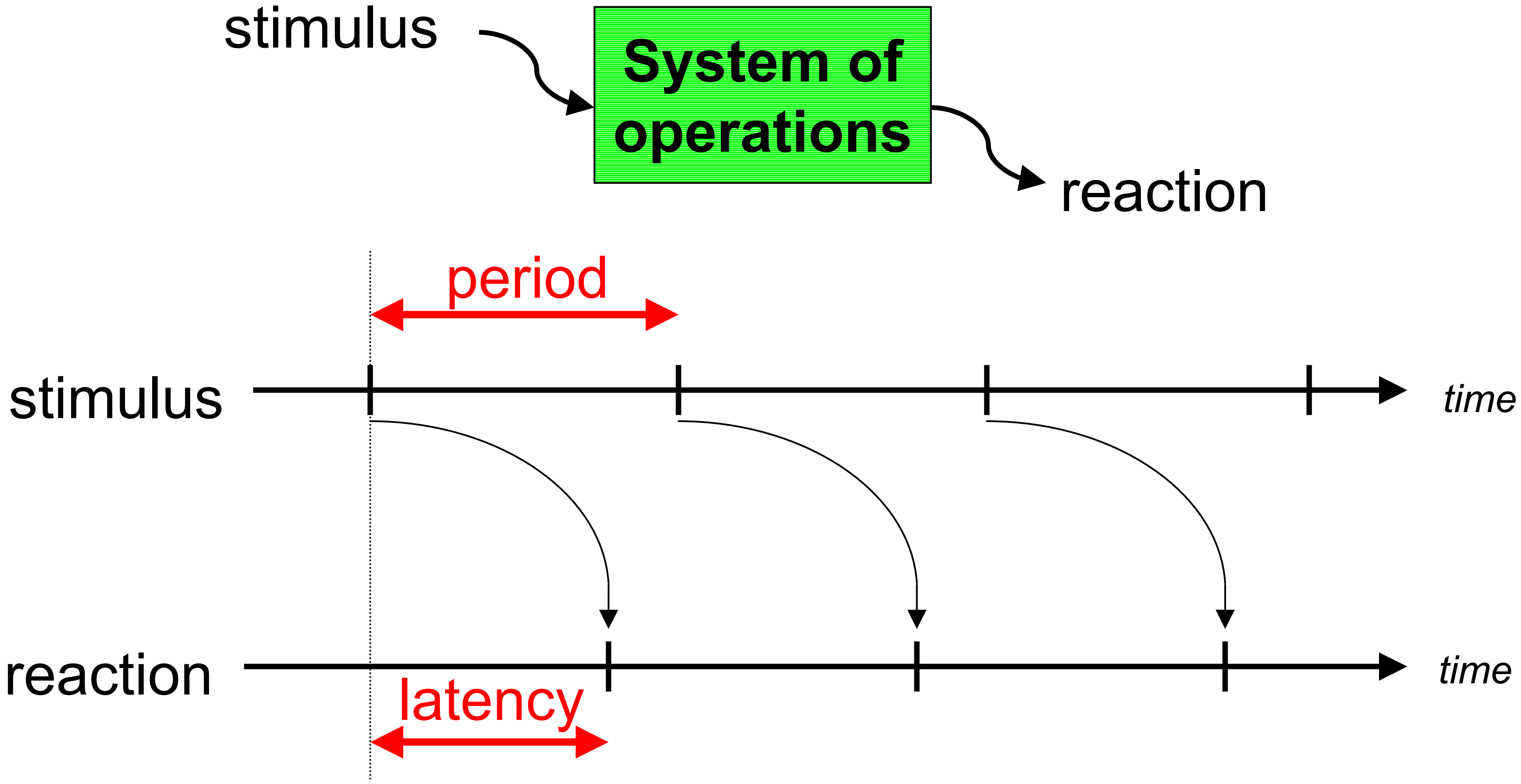
Tasks with precedence & periodicity constraints

- 1 / r, D const. partial order, T / - modified EDF
- 1 / prec-subtasks /- schedulability condition
- 1 / T, prec for sporadic tasks / - schedulability test
- m / T / minimize communications
- m / T, D- tasks ; T, D, prec-subtasks/ -

Model and problem to solve

- Reactive systems features
- Typical vs. new model
- Latency: new constraint
- Repetitive graph
- Latency and periodicity constraints
- Problem to solve

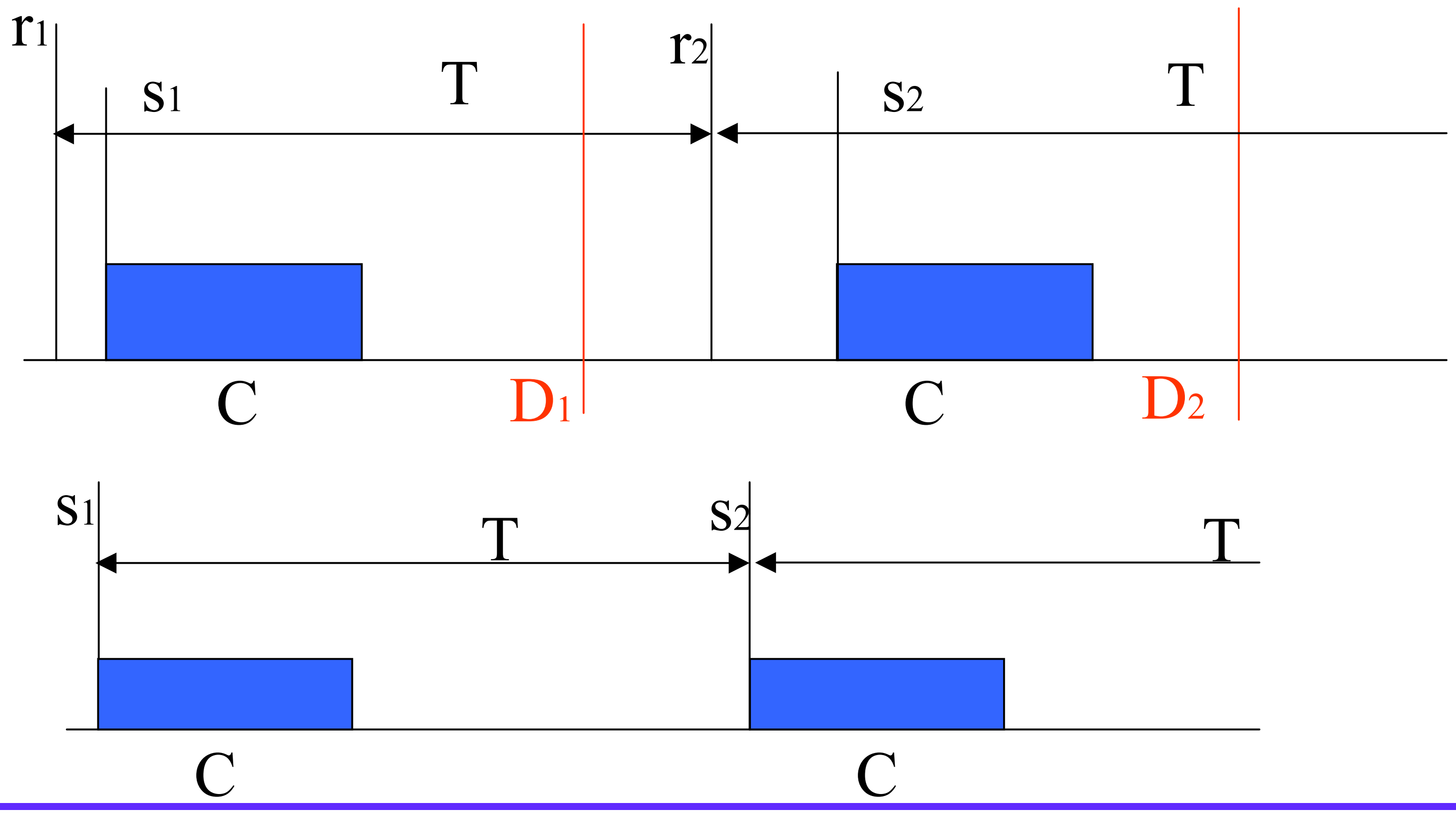
Reactive systems features



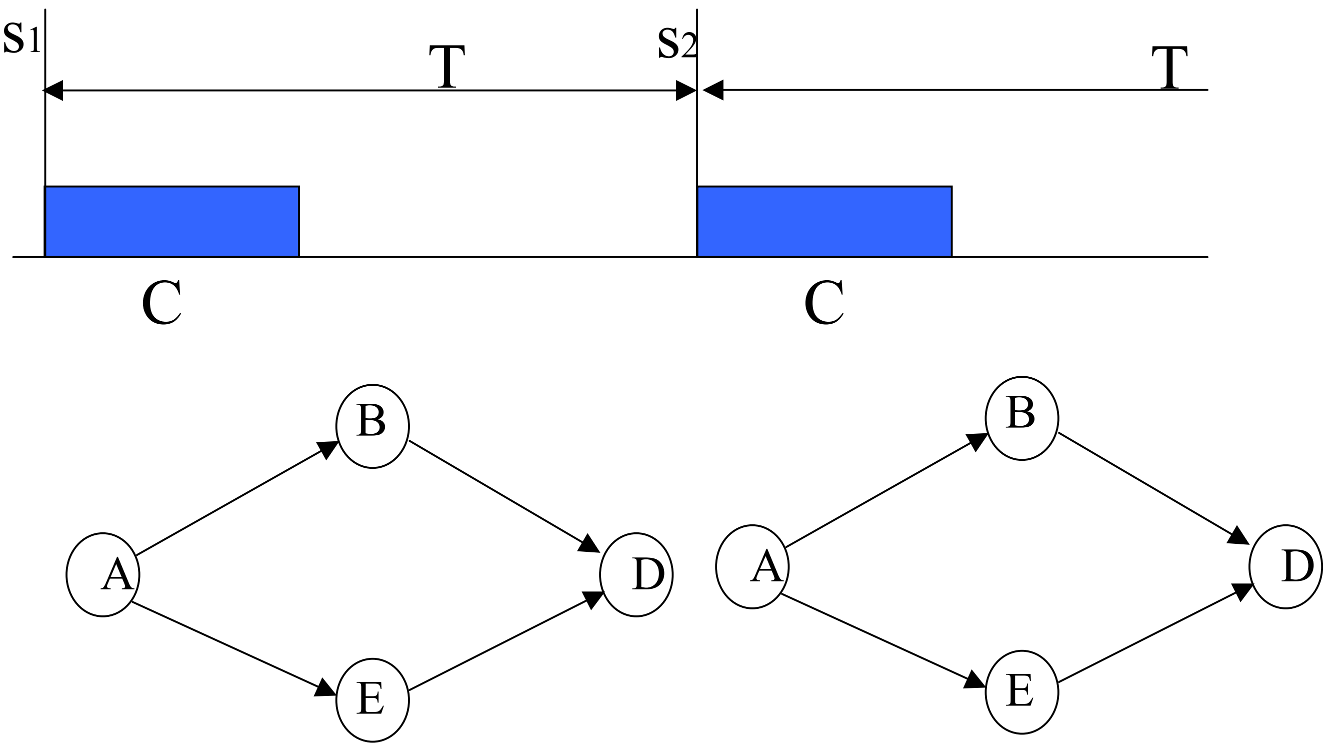
Extended to each operation and each pair of operations

Typical vs. new model

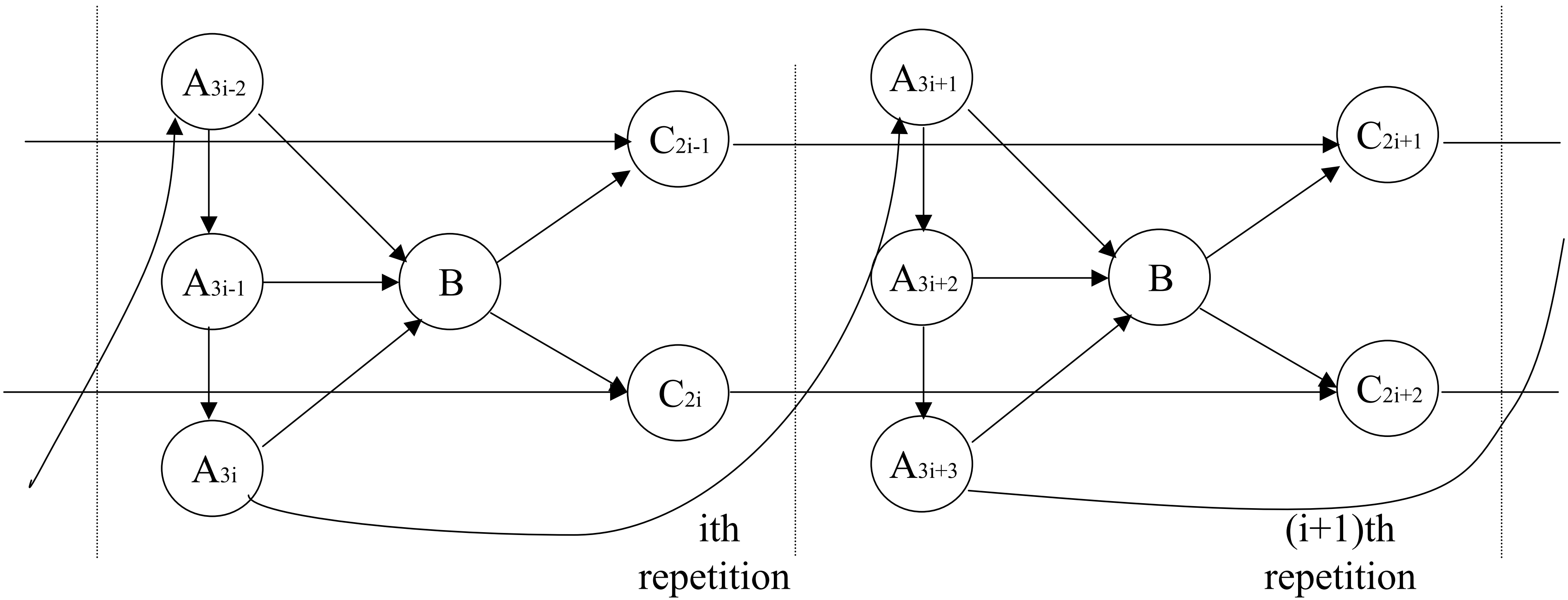
Operation instead of task or job to be independent of implementation aspects



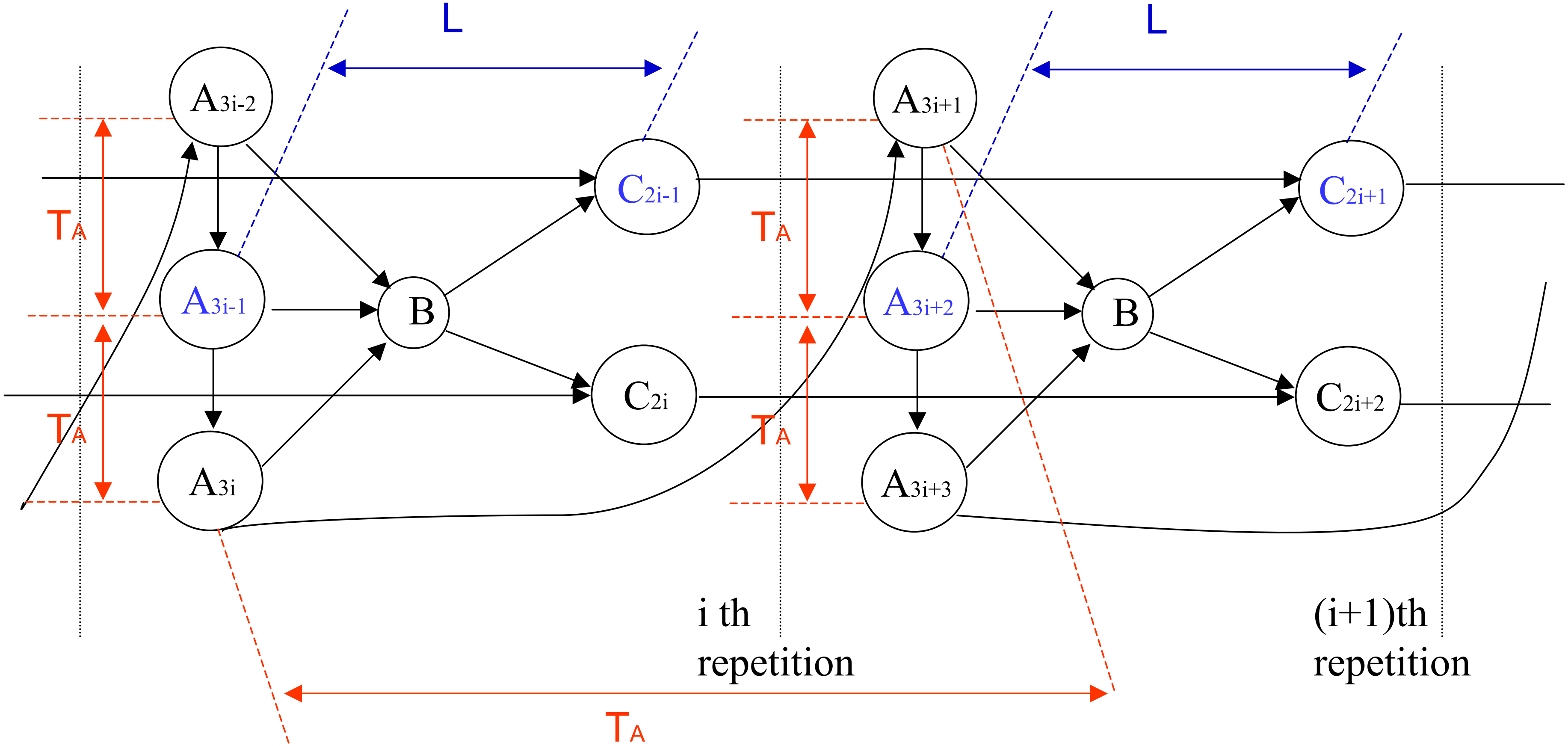
Periodic operations: Repetitive Graph



Repetitive Graph with repeated operations



Latency and periodicity constraints



$s_{C2i-1} - s_{A3i-1} + C_C \leq L$ $\qquad$ $s_{Ai+1} - s_{Ai} = T_A, \forall i \in \mathbb{N}^*$

Relation between periodicity and latency

Theorem: the periodicity constraint is a strict latency constraint

$$s_{Ai+1} - s_{Ai} = T_A \,,\ \forall\, i \in \mathbb{N}^* \quad \Rightarrow \quad s_{Ai+1} - s_{Ai} \leq L - C_A,\ \forall\, i \in \mathbb{N}^*$$

Problem to solve

- Several processors
- Precedence constraints
- Latency constraints
- Divisible periods and execution times

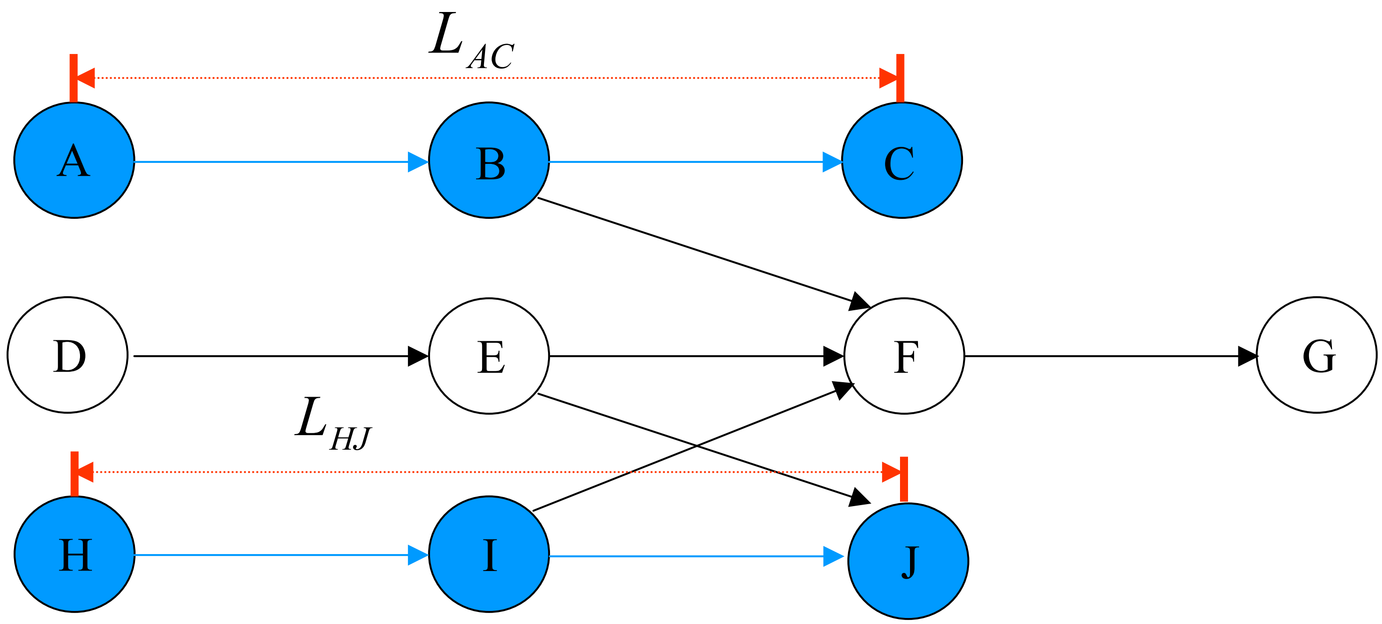
- Off-line scheduling
- Without preemption
- With idle time

Study for monoprocessor case
then results extention for multiprocessor case

Schedulability condition for latencies

- Relations between pairs of operations
  - II: schedulability condition for imposed latencies on pairs of operations which are in relation II
  - Z: schedulability condition for imposed latencies on pairs of operations which are in relation Z
  - X: schedulability condition for imposed latencies on pairs of operations which are in relation X
- Schedulability condition

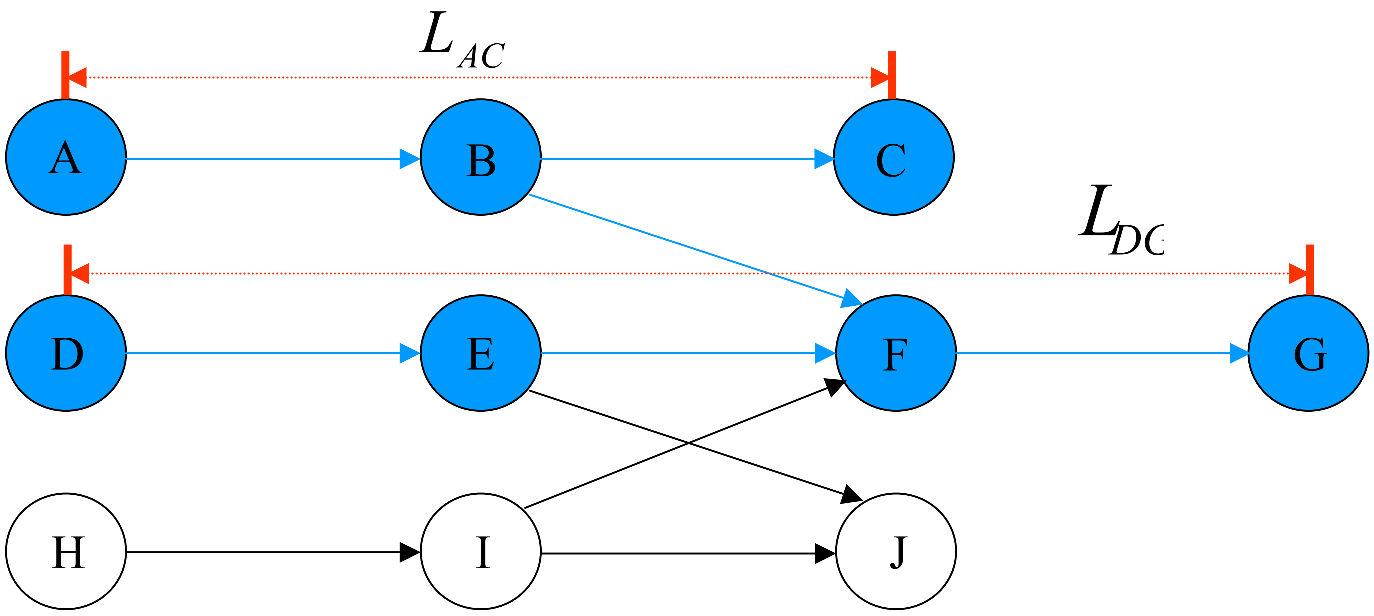
Relations between pairs of operations: II



(A,C) II *(H,J)*

Theorem: the system is schedulable if and only if $L_{AC} \geq \sum_{H \in I(A,C)} C_H$ and $L_{HJ} \geq \sum_{H \in I(H,J)} C_H$

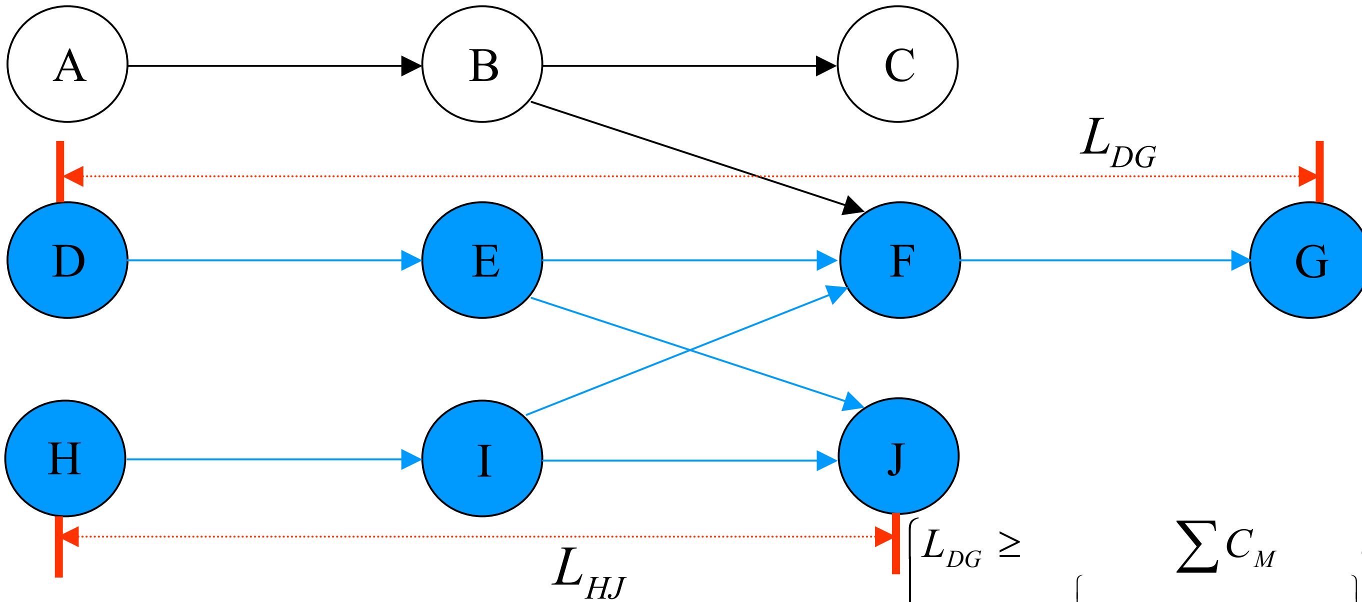
Relations between pairs of operations: Z



(A,C) Z *(D,G)*

Theorem: the system is schedulable if and only if $L_{AC} \geq \sum_{H \in I(A,C)} C_H$ and $L_{DG} \geq \sum_{H \in I(D,G)} C_H$

Relations between pairs of operations: X



(D,G) X *(H,J)*

Theorem: the system is schedulable if and only if one of the following relations is satisfied

$$
\begin{cases}
L_{DG} \geq \sum\limits_{M \in \left\{ I(D,G) \cup \bigcup\limits_{N \in \Gamma^{-}_{DG}(H,J)} I(H,E) \right\}} C_M \quad and L_{HJ} \geq \sum\limits_{M \in \left\{ I(H,J) \cup \bigcup\limits_{N \in \Gamma^{+}_{HJ}(D,G)} I(N,G) \right\}} C_M \\
L_{DG} \geq \sum\limits_{M \in \left\{ I(D,G) \cup \bigcup\limits_{N \in \Gamma^{+}_{DG}(H,J} I(N,J) \right\}} C_M \quad and L_{HJ} \geq \sum\limits_{M \in \left\{ I(H,J) \cup \bigcup\limits_{N \in \Gamma^{-}_{HJ}(D,G)} I(D,N) \right\}} C_M \\
L_{DG} = \sum\limits_{M \in I(D,G)} C_M and L_{HJ} \geq \sum\limits_{M \in I(D,G) \cup I(H,J)} C_M \\
L_{DG} \geq \sum\limits_{M \in I(D,G) \cup I(H,J)} C_M \quad and L_{HJ} = \sum\limits_{M \in I(H,J)} C_M
\end{cases}
$$

Schedulability condition for latencies

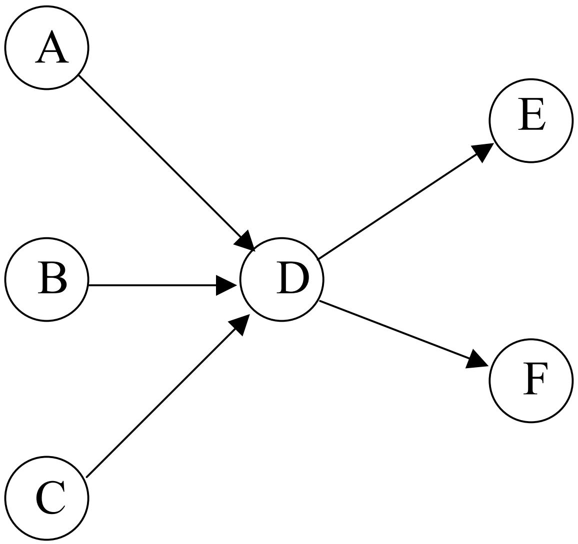
Theorem: the system is schedulable if and only if:

- for all pairs *(A,C)* II *(H,J)*, $L_{AC} \geq \sum_{H \in I(A,C)} C_H$ and $L_{DG} \geq \sum_{H \in I(D,G)} C_H$
- for all pairs *(A,C)* Z *(D,G)*, $L_{AC} \geq \sum_{H \in I(A,C)} C_H$ and $L_{HJ} \geq \sum_{H \in I(H,J)} C_H$
- for all pairs *(D,G)* X *(H_i,J_i)*, one of following relations is satisfied:

$$
\left\{
\begin{array}{l}
L_{DG} = \sum_{M \in I(D,G)} C_M \; and L_{H_i J_i} \geq \sum_{M \in I(D,G) \cup I(H_i,J_i)} C_M \\
L_{DG} \geq \sum_{M \in I(D,G) \cup \bigcup_{i \in \{1,j\}; N \in \Gamma_{\overline{DG}}(H_i,J_i)} I(H_i,N) \cup \bigcup_{i \in \{j,k\}, N \in \Gamma^{+}_{DG}(H_i,J_i)} I(N,J_i) \cup \bigcup_{i \in [k,n]} I(Hi,Ji)} C_M \\
\left\{
\begin{array}{l}
L_{Hi,Ji} \geq \sum_{M \in \left\{ I(Hi,Ji) \cup \bigcup_{N \in \Gamma^{+}_{HiJi}(D,G)} I(N,G) \right\}} C_M, \forall i \in \{1,\dots, j\}; \\
L_{Hi,Ji} \geq \sum_{M \in \left\{ I(Hi,Ji) \cup \bigcup_{N \in \Gamma_{\overline{HiJi}}(D,G)} I(Hi,N) \right\}} C_M, \forall i \in \{j,\dots, k\}; \\
L_{Hi,Ji} = \sum_{M \in I(hi,Ji)} C_M, \forall i \in \{k,\dots, n\}.
\end{array}
\right.
\end{array}
\right.
$$

Schedulability condition for periodicities

Theorems:



$T_D = \max\{T_A, T_B, T_C\}$

$T_D = \min\{T_E, T_F\}$

$T_D = \min\{T_E, T_F\}$

Schedulability condition for periodicities

- Theorem: for a system with periodicity and precedence constraints
  - the existence of a hyperperiod from s_{max} to $s_{max}+T$, where T is the least common multiple of all periodicity constraints
  - if the system is schedulable then $\sum_{A \in V} \frac{C_A}{T_A} \leq 1$

General schedulability condition

- Theorem: if the system is schedulable then

$\sum_{A \in V} \frac{C_A}{T_A} \le 1$ and

$$
\begin{cases}
L_{AB} \ge \sum_{M \in W_1} C_H + \sum_{H \in V, T_H < T_A} C_H \frac{L_{AB}}{T_H} \\
W_1 = M(A,B) \bigcup \bigcup_{i \in \{1,j\}; E \in lt_{AB}(C_i D_i)} M(C_i, E) \bigcup \bigcup_{i \in \{j+1,k\}, E \in ft_{AB}(C_i D_i)} M(E, D_i) \bigcup \bigcup_{i \in \{k+1,m\}} M(Ci, Di) \bigcup \bigcup_{i \in \{m+1,n\}} M(Ci, Di) \\
\begin{cases}
L_{CiDi} \ge \sum_{H \in \left\{ M(CiDi) \bigcup \bigcup_{E \in ft_{CiDi}(AB)} M(E,B) \right\}} C_H, \forall i \in \{1,\dots, j\}; \\
L_{CiDi} \ge \sum_{H \in \left\{ M(Ci,Di) \bigcup \bigcup_{N \in lt_{CiDi}(AB)} M(A,E) \right\}} C_H, \forall i \in \{j+1,\dots, k\}; \\
L_{CiDi} \ge \sum_{H \in M(CiDi)} C_H, \forall i \in \{k+1,\dots, m\} \\
L_{CiDi} \ge \sum_{H \in \left\{ M(Ci,Di) \bigcup \bigcup M(A,B) \right\}} C_H, \forall i \in \{m+1,\dots, n\}
\end{cases}
\end{cases}
$$

Scheduling algorithm for monoprocessor

- Algorithm of latency marking
- Scheduling algorithm
- Optimality

Scheduling algorithm

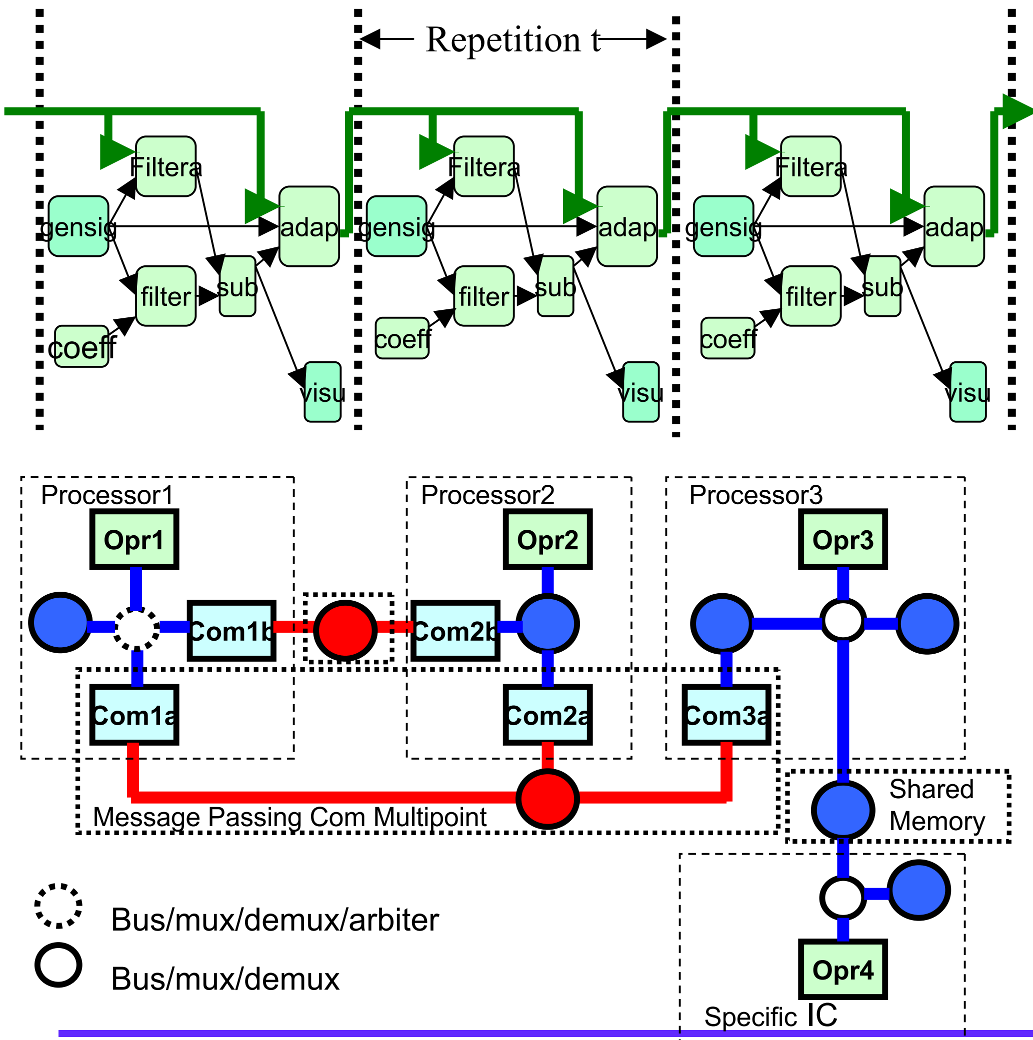
- Algorithm of latency marking
  - the mark of an operation is the smallest value of all latency constraints for which there is a path from this operation to the second operation of the latency constraint
- Infinite scheduling algorithm
  - the steps of initialization schedule the operations in this order: first, operations without constraints, then operations with mark $\neq 0$, and finally periodic operations
  - once a periodic operation is scheduled, the order of the scheduling is the opposite order of the initialization order

Optimality

Scheduling algorithm applied, only, from 0 to s_{max} +T

- Theorem: the scheduling algorithm is optimal (if there is a schedule, the algorithm will find it)
  - The system has only precedence and latency constraints (By contradiction)
  - The system has only periodicity and precedence constraints (Theorem)
  - The system has periodicity, latency and precedence constraints (Combination of previous cases)

Distribution and scheduling for multiprocessor



- Scheduling of operations is not sufficient
- Distribution of operations onto processors
- Distribution and scheduling based on algorithm graph and architecture graph transformations

Distribution and scheduling model (1/2)

The set of all possible implementations is described as the composition of three binary relations:

$$(Gal, Gar) \xrightarrow{\textit{rout o distrib o sched}} (Gal', Gar')$$

- **Routing**: creation of all the inter-operator routes
- **Distribution**: *spatial allocation*
  - Partitioning and allocation: operations onto operator
  - Partitioning of inter-partition edges according to routes
  - Creation and allocation:
    - Communication vertices **onto** communicators of the route
    - Allocation vertices **onto** memories
    - Identity vertices **onto** bus/mux/demux/ with or without arbiter

Distribution and scheduling model (2/2)

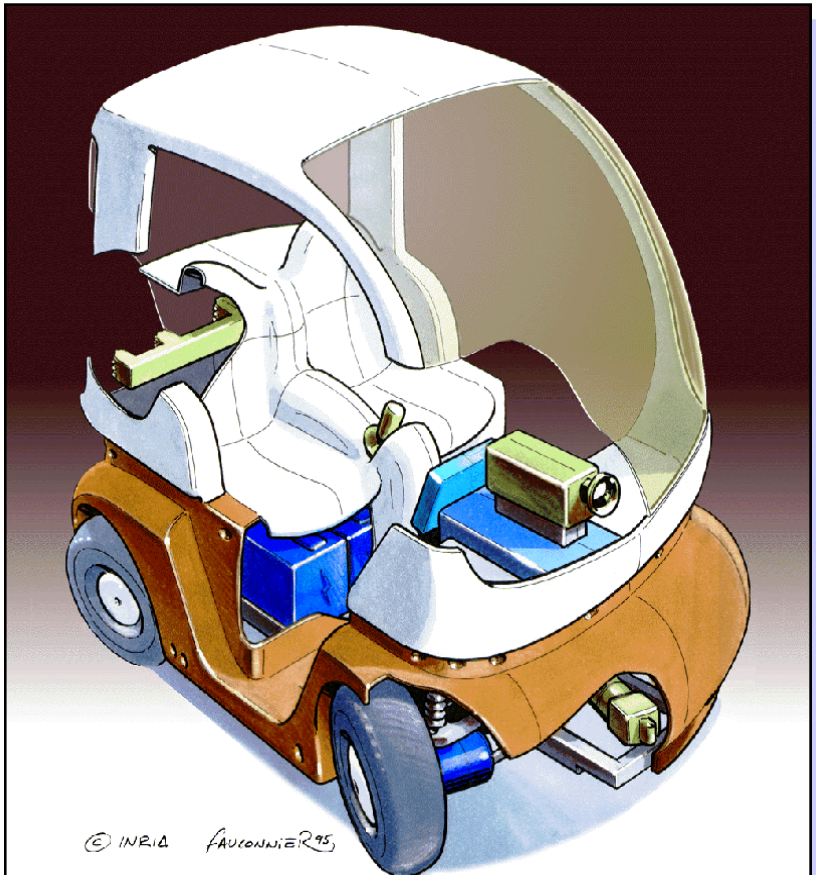
- **Scheduling**: *temporal allocation*
  - Partial Order $\rightarrow$ Total Order for:
    - Each partition of operations allocated onto an operator
    - Each partition of communication operations allocated onto a communicator

Routing, Distribution and Scheduling lead to a Partial Order consistent with the initial Partial Order of the Algorithm Graph

Distribution and scheduling optimization

- Distribution and scheduling optimizations lead to NP-hard problems
- Heuristics based on scheduling results for monoprocessor such that communication cost is taken into account
  - Fast: Greedy: list-scheduling for Rapid Prototyping
  - Slow: Neighboring list-scheduling with back-tracking

CyCab application

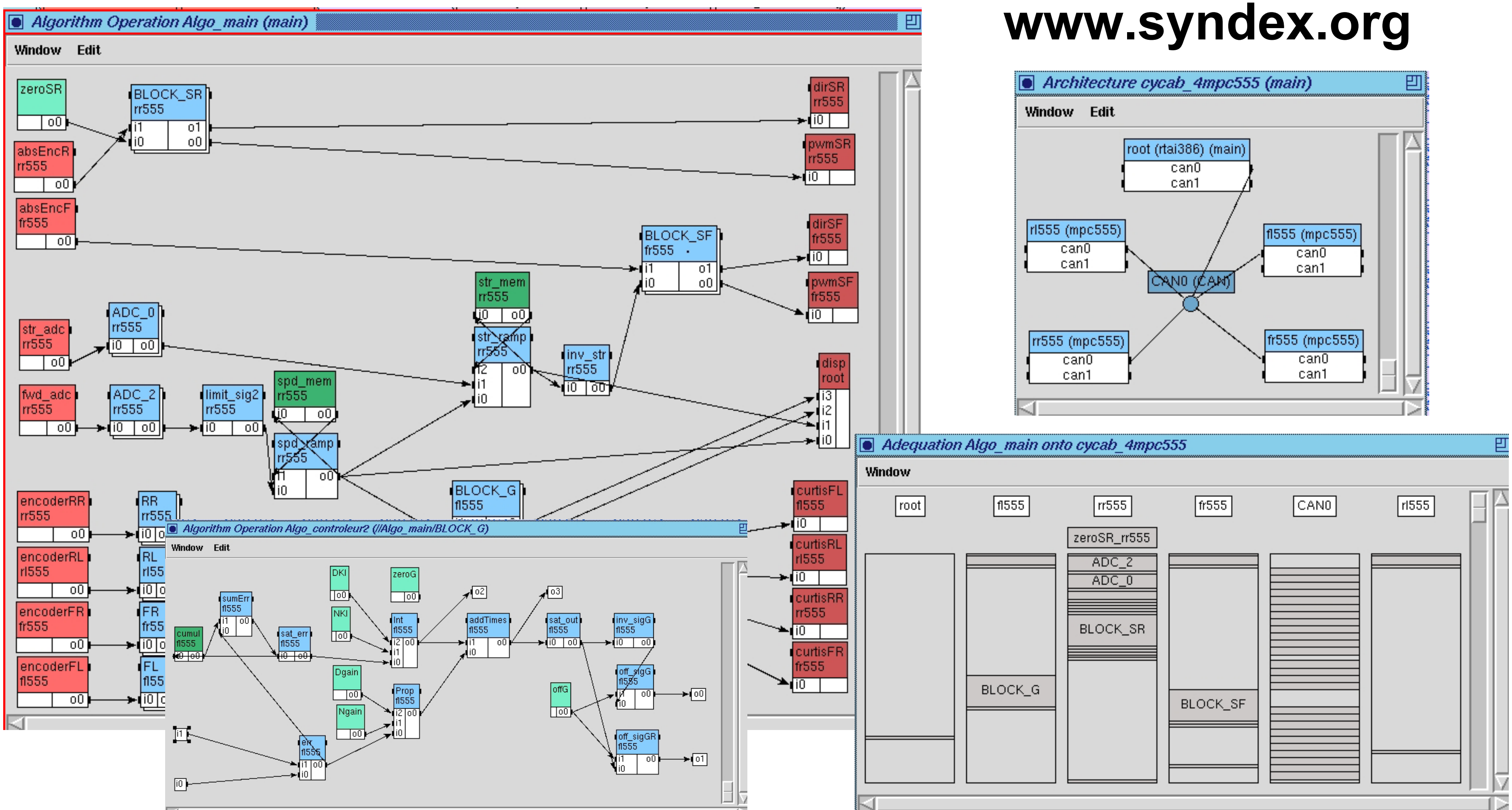


- Vitesse 30km/h
- Moteurs électriques
- 4 roues motrices
- 2 directions AV, AR
- Multi-processeur MPC555 + un Pentium
- Bus Can

Industrialisé par Robosoft
www.robosoft.fr

System level CAD software: SynDEx

www.syndex.org



Conclusion

- New model for real-time systems
- Relations between:
  - Latency and periodicity constraints
  - Latency constraint and deadline
- Monoprocessor
  - Optimal scheduling algorithm
  - Schedulability condition for latencies
  - Schedulability condition for periodicities
  - General schedulability condition
- Multiprocessor
  - Distribution and scheduling for one latency = period
  - Heuristics taking into account communication cost

Work in progress

- Extension to multiprocessor by using heuristics based on previous results
- Preemptive scheduling algorithm
- Periodicity with jitter