CAPD Center for Advanced Process Decision-making

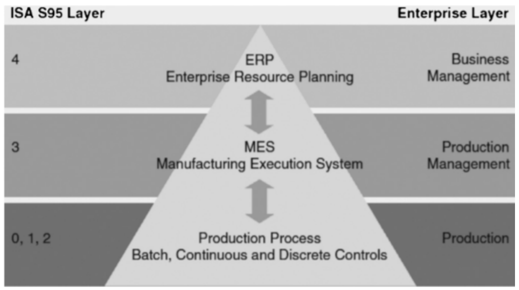
Integrating RTN scheduling models with ISA-95 standard

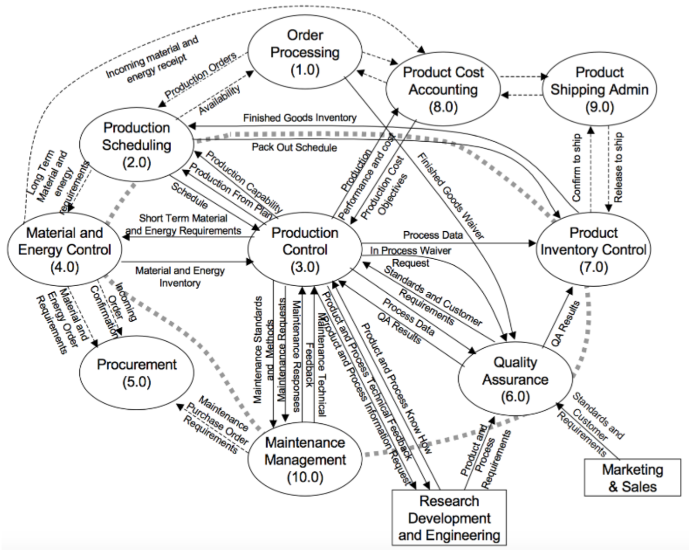
Pedro M. Castro
Ignacio E. Grossmann
Iiro Harjunkoski

IF INVESTIGADOR FCT
Ciências ULisboa
Carnegie Mellon
ABB

Motivation

- EWO aims to simultaneously account for KPI across multiple business units
  – Integration of supply chain management, production control, planning & scheduling
- Need to efficiently transfer data and information between different systems
  – Focus on production management system and scheduling solution
- ISA-95 standard can act as data-exchange platform
  – Goal: Integrate with generic scheduling framework to cover wide variety of problems

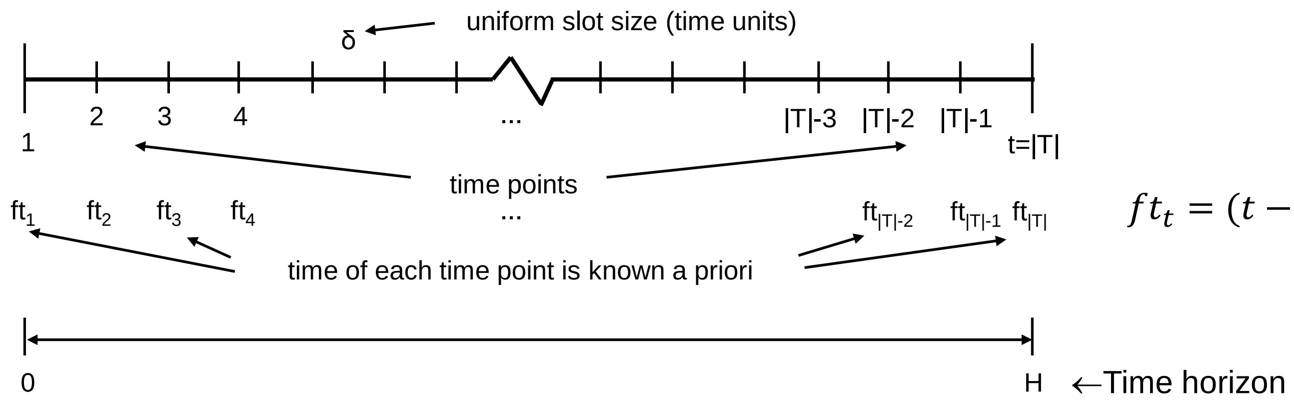




Generic scheduling framework

- Resource-Task Network formulation (Pantelides '94)
  - Key success drivers
    - Modeling paradigm easily understood by business stakeholders
    - Flexible approach, easily modified when new information becomes available
  - Discrete-time representation
    - Better at finding good solutions in short computational time

Processing times p_i (h) rounded up to multiples of δ
$\tau_i = \lceil p_i / \delta \rceil$ (# time slots)
↑ Parameter used by model



$$ft_t = (t-1)\cdot\delta$$

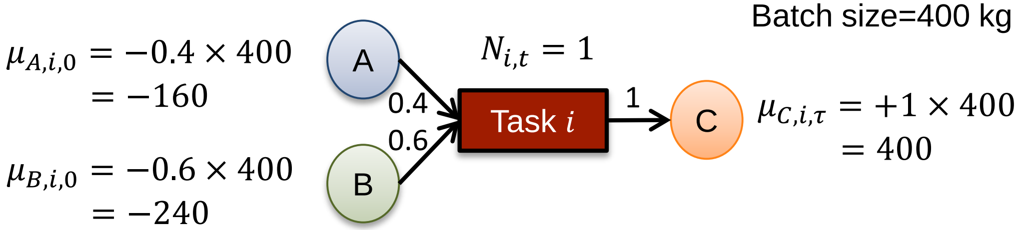
- Challenge
  - Develop RTN models that map ISA-95 information
    - Need for RTN Handbook

Key model entities: Tasks (i)

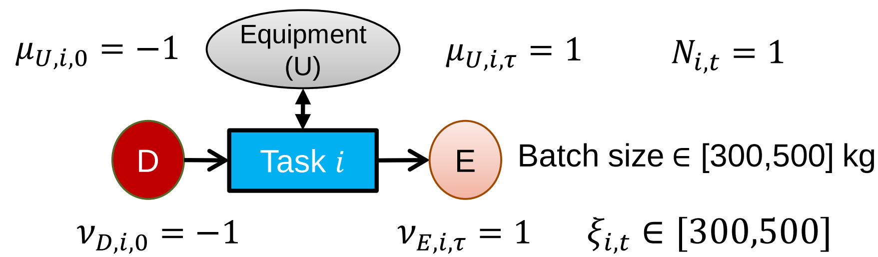
Carnegie Mellon ABB

- Characterized by:
  - Extent variables
    - Binary $N_{i,t}$
    - Continuous $\xi_{i,t}$
  - Structural parameters (from production recipe)
    - Binary interaction $\mu_{r,i,\theta}$
    - Continuous interaction $\nu_{r,i,\theta}$
    - Arrow indicates production and/or consumption
- Relative time index θ
  - Measure unit: # time slots
    - Events at start of task: $\theta = 0$
    - Events at end of task: $\theta = \tau$
    - Intermediate events: $\theta \in \{1, \dots, \tau - 1\}$

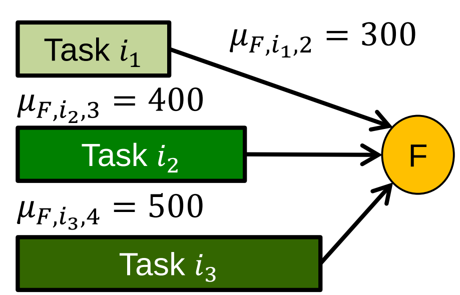
- Fixed amount



- Variable amount, fixed duration



- Variable amount & duration



Size (kg)	Duration
300	2
400	3
500	4

Key model entities: Resources (r)

Carnegie Mellon ABB

- Characterized by:
  - Excess resource variables
    - Continuous $R_{r,t}$
      - Unit assigned to task $\Leftrightarrow R_{r,t} = 0$
      - Idle unit $\Leftrightarrow R_{r,t} = 1$
  - Parameters
    - Initial availability R_r^0
      - Typically $R_r^0 = 1$ (equipment units)
      - $R_r^0 = n$ (if there are n equivalent units)
    - Time-dependent availability (dedicated storage capacity)

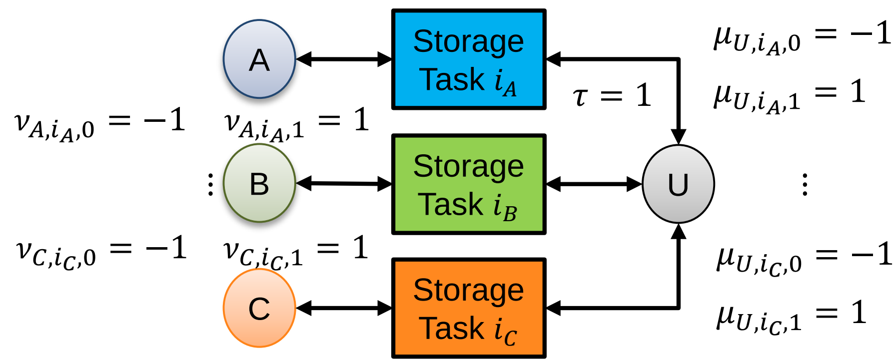
$$R_{r,t}^{min} \le R_{r,t} \le R_{r,t}^{max}$$

    - Shared storage (multiple materials r simultaneously in unit u)

$$v_u^{min} \le \sum_r R_{r,t} \le v_u^{max}$$

    - External interactions $\pi_{r,t}$
      - Supplied to (+) & removed from (-) system

- Shared storage (one material at a time)
  - Define one storage task per material
    - All consume unit u ($R_u^0 = 1$)
  - Hide material resources r temporarily (one time slot)
    - $R_{A,t} = R_{B,t} = R_{B,t} = 0 \ \forall t$

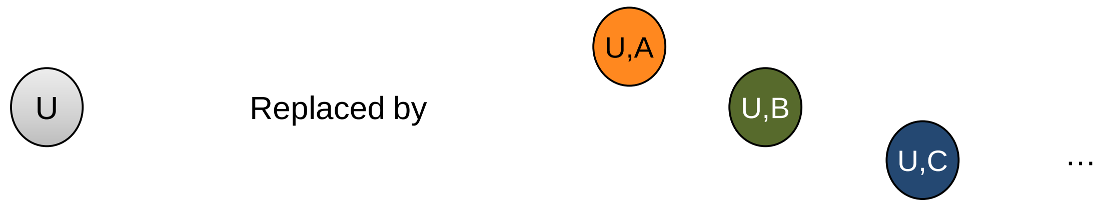


- Task/resource interactions
  - Excess resource balances

$$R_{r,t} = R_r^0|_{t=1} + R_{r,t-1} + \sum_i \sum_{\theta=0}^{\tau_i} (\mu_{r,i,\theta} N_{i,t-\theta} + \nu_{r,i,\theta} \xi_{i,t-\theta}) + \pi_{r,t} \ \forall r, t$$

Sequence dependent changeovers

- Need to disaggregate equipment resource
  - One for each operation mode



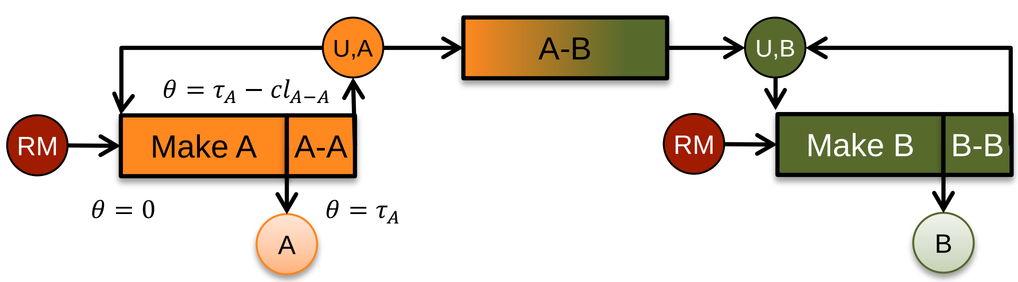
- Include easy clean (e.g. A to A) in processing task
  - Avoids the need for clean and dirty states
    - Same mode can immediately follow
  - Deduct easy clean from real values
  - Takes advantage of intermediate events

Real values

Time	A	B	C
A	15	60	120
B	120	15	60
C	60	60	15

Model values

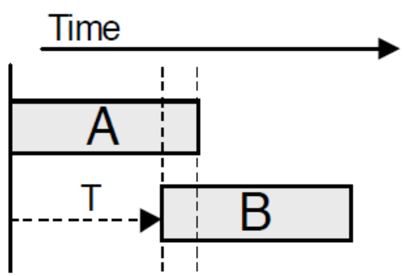
Time	A	B	C
A	-	45	105
B	105	-	45
C	45	45	-



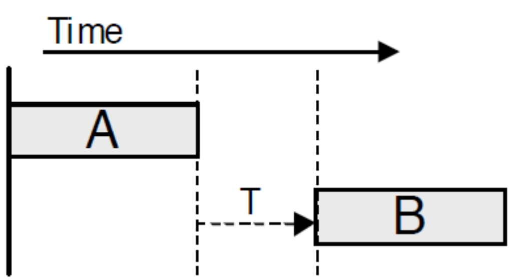
Time-based dependencies

- Define auxiliary timing variables
  - Starting time $Ts_i = \sum_t ft_t N_{i,t}$
  - Ending time $Te_i = \sum_t (ft_t + \tau_i \cdot \delta) \cdot N_{i,t}$
    - Assumes execution of a single instance of task i during time horizon

- ANSI/ISA–95.00.02–2001, Figure B-6

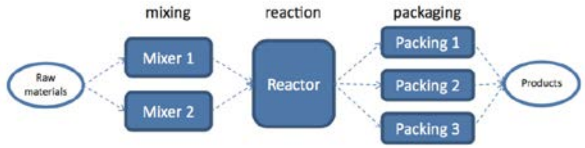


B may run in parallel to A	No constraint needed
Start B at A start	$Ts_B = Ts_A$
Start B after A start	$Ts_B \geq Ts_A$
Start B no later than T time after A start	$Ts_B \leq Ts_A + T$
Start B no earlier than T time after A start	$Ts_B \geq Ts_A + T$



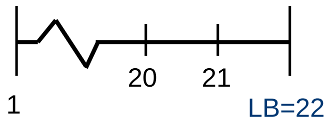
B may not run in parallel to A	$Ts_B \geq Te_A \vee Ts_A \geq Te_B$
Start B at A end	$Ts_B = Te_A$
Start B after A end	$Ts_B \geq Te_A$
Start B no later then T time after A end	$Ts_B \leq Te_A + T$
Start B no earlier than T time after A end	$Ts_B \geq Te_A + T$

Minimizing makespan

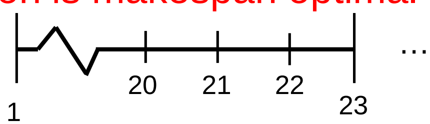


CarnegieMellon ABB

- Option 1 (Maravelias & Grossmann ‘03)
  - Iterative procedure (more efficient)
  - Start with short horizon (infeasible)

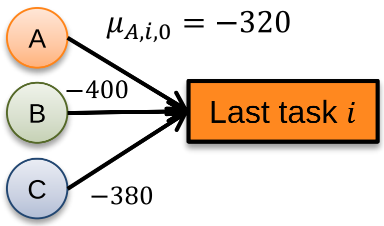


  - Optimize auxiliary objective
    - E.g. Maximize production
  - Increase # slots until feasibility
    - 1st solution is makespan optimal



δ	#Batches	Iterations	Makespan	CPUs
5	(1,1,1)	9	365	4.06
5	(2,2,2)	26	535	31.5
5	(3,3,3)	49	730	2066
10	(3,3,3)	26	750	78
15	(3,3,3)	17	750	18.5
10	(5,5,5)	42	1080	1433
15	(5,5,5)	26	1050	40.0

- Option 2 (Castro ‘01)
  - Define sufficiently long horizon
  - Minimize starting time of last task
    - Consumes all product demand



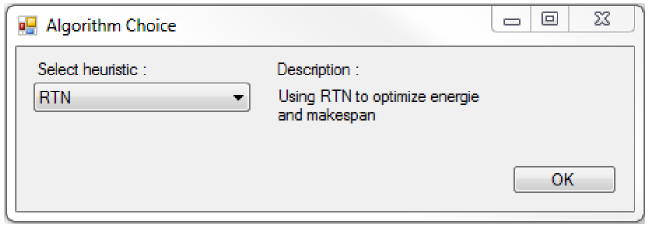
  - LP relaxation ⇒ Lower bound (LB)
  - Returns solutions before optimum

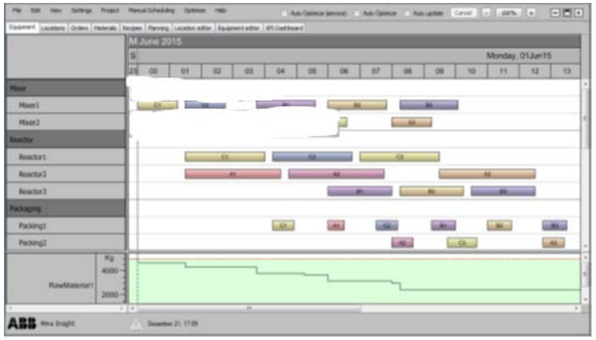
δ	#Batches	Makespan	CPUs
5	(1,1,1)	365	4.83
5	(2,2,2)	535	60.1
5	(3,3,3)	730	3600
10	(3,3,3)	750	2343
15	(3,3,3)	750	106
10	(5,5,5)	1080	3600
15	(5,5,5)	1050	60.2

RTN in C# Gantt scheduling program

Carnegie Mellon ABB

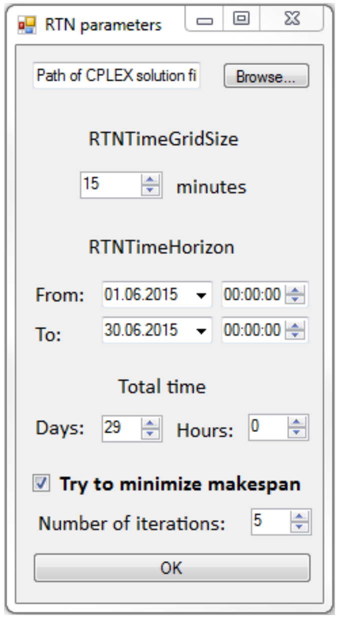
- Import files from XML files with ISA-95 info
  - RTN added to list of heuristics/algorithms

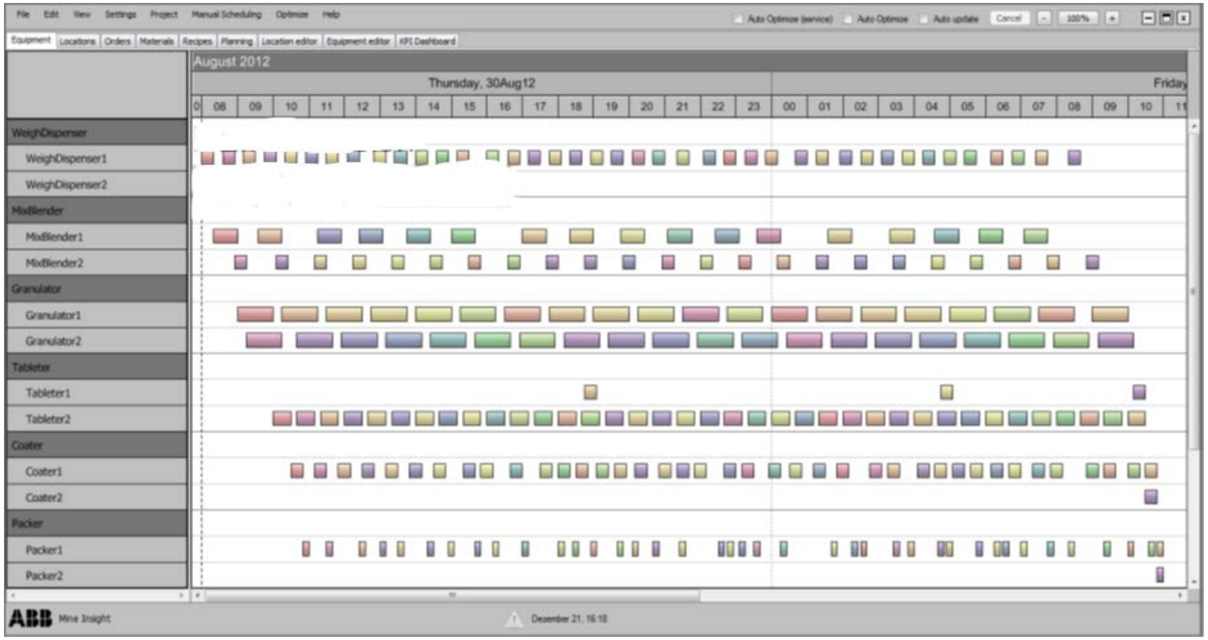




- Select tuning parameter δ
  - Makespan strategy

Pharma problem with 40 orders;
δ=5 min; Makespan=27 h; 95 CPUs





Conclusions

- Guidelines for modeling scheduling problems with RTN discrete-time formulations
  - Makespan minimization can be challenging
  - Efficient handling of sequence dependent changeovers
- RTN model integrated with ISA-95 standard
  - New feature in ABB's Gantt scheduling program
    - Rule-based heuristics (e.g., ASAP)
- Important features still missing
  - Time-dependent interactions
    - Intermediate events of a task (beyond start and finish)
    - Interaction with system boundaries (e.g. electricity prices)
  - Pre-emptive tasks
    - Key for completing after the weekend (e.g., maintenance)