Overview of Optimization Models for Planning and Scheduling

Ignacio E. Grossmann

Center for Advanced Process Decision-making

Department of Chemical Engineering

Carnegie Mellon University

Pittsburgh, PA

January 18, 2006

Enterprise-wide Optimization Seminar

Outline

I. Classification of batch scheduling problems
   Classification of optimization models for batch scheduling

II. Discrete and continuous time scheduling models

III. Numerical comparison of optimization models

IV. Alternative solution approaches

V. Commercial software for scheduling of batch plants

VI. Beyond current scheduling capabilities

References

Mendez, C.A., J. Cerda, I.E. Grossmann, I. Harjunkoski, M. Fahl, "State-of-the-art Review of Optimization Methods for Short-term Scheduling of Batch Processes," to appear in *Comp. & Chemical Engineering* (2006).

Floudas, C.A.; Lin, X. "Continuous-time versus discrete-time approaches for scheduling of chemical processes: a review." *Comp. and Chem. Eng.*, 28, 2109 – 2129 (2004).

Kallrath, J. "Planning and scheduling in the process industry," *OR Spectrum*, 24, 219-250 (2002).

Pinto, J.M. & Grossmann, I.E., "Assignments and sequencing models of the scheduling of process systems," *Annals of Operations Research*, 81, 433 – 466 (1998).

Shah, N., "Single- and multisite planning and scheduling: Current status and future challenges," Proceedings of FOCAPO-98 75 – 90 (1998).

Pekny, J. F. & Reklaitis, G. V., "Towards the convergence of theory and practice: A technology guide for scheduling/planning methodology. Proceedings of FOCAPO-98 91 – 111 (1998).

Mauderli. A. M.: Rippin. D. W. T. Production Planning and Scheduling for Mu1tip;;pose Batch Chemical Plants. *Comp. Chem.Eng.* 3, 199 (1979).

Reklaitis, G. V. Review of Scheduling of Process Operation. AIChE Symp. Ser. 78, 119-133 (1978).

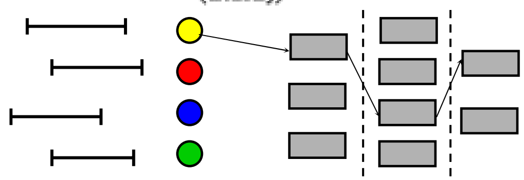
Major Academic Research Efforts

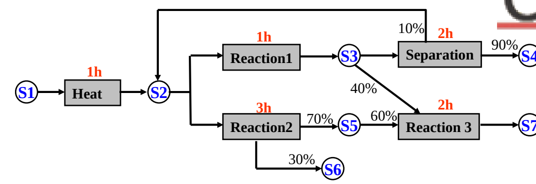
School	Researcher(s) Weblink
Åbo Akademi University	*T. Westerlund* http://www.abo.fi/~twesterl/
Carnegie Mellon University	*I.E. Grossmann* http://egon.cheme.cmu.edu
Imperial College	*C. Pantelides, N. Shah* http://www.ps.ic.ac.uk
Instituto Superior Lisbon	*A. Barbosa Povoa* http://alfa.ist.utl.pt/~d3662/
INTEC - CONICET	*J. Cerdá* http://intecwww.arcride.edu.ar/~jcerda/
National University of Singapore	*I.A. Karimi* http://www.chee.nus.edu.sg/staff/000731karimi.html
Polytechnic University	*J. Pinto* http://www.poly.edu/faculty/josempinto/
Princeton University	*C.A. Floudas* http://titan.princeton.edu/home.html
Purdue University	*J. Pekny and G.V. Reklaitis* http://engineering.purdue.edu/ChE/Research/Systems/index.html
Rutgers University	*M. Ierapetritou* http://sol.rutgers.edu/staff/marianth/
Technical University Graz	*R. E. Burkard* http://www.opt.math.tu-graz.ac.at/burkard/
Universitat Politécnica de Catalunya	*L. Puigjaner* http://deq.upc.es/wwwdeq/cat/infogral/curriculs/Lluis%20Puigjaner.htm
University College London	*L. Papageorgiou* http://www.chemeng.ucl.ac.uk/staff/papageorgiou.html
University of Dortmund	*S. Engell* http://www.bci.uni-dortmund.de/ast/en/content/mitarbeiter/elehrstuhlinhaber/engell.html
University of Sao Paulo	J. Pinto http://www.lscp.pqi.ep.usp.br/pro_zeca.html
University of Tessaloniki	*M. Georgiadis* http://www.cperi.certh.gr/en/compro.shtml#SECT2
University of Wisconsin	*C. Maravelias* http://www.engr.wisc.edu/che/faculty/maravelias_christos.html

Sequential vs Network

(1) Process topology

- Sequential
  - Single stage
    - Single unit
    - Parallel units
  - Multiple stages
    - Multiproduct (Flow-shop)
    - Multipurpose (Job-shop)
- Network (arbitrary)





(3) Equipment connectivity

- Partial (restricted)
- Full

Types of storage

(4) Inventory storage policies

- Unlimited Intermediate Storage (UIS)
- Non-Intermediate Storage (NIS)
- Finite Intermediate Storage (FIS)
  - Dedicated storage units
  - Shared storage units
- Zero Wait (ZW)

(5) Material transfer

- Instantaneous (neglected)
- Time-consuming
  - No-resources
  - Pipes
  - Vessels (Pipeless)

Fixed or variable batch size

(6) Batch size

- Fixed
- Variable (Mixing and Splitting)

(7) Batch processing time

- Fixed
  - Unit independent
  - Unit dependent
- Variable (unit/batch-size dependent)

(8) Demand patterns

- Due dates
  - Single product demand
  - multiple product demands
- Scheduling horizon
  - Fixed requirements
  - Minimum / maximum requirements

Changeovers?

(9) Changeovers

- None
- Unit dependent
- Sequence dependent
  - Product dependent
  - Product and unit dependent

Resource constraints

(10) Resource Constraints

- None (only equipment)
- Discrete
- Continuous

(11) Time Constraints

- None
- Non-working periods
- Maintenance
- Shifts

(12) Costs

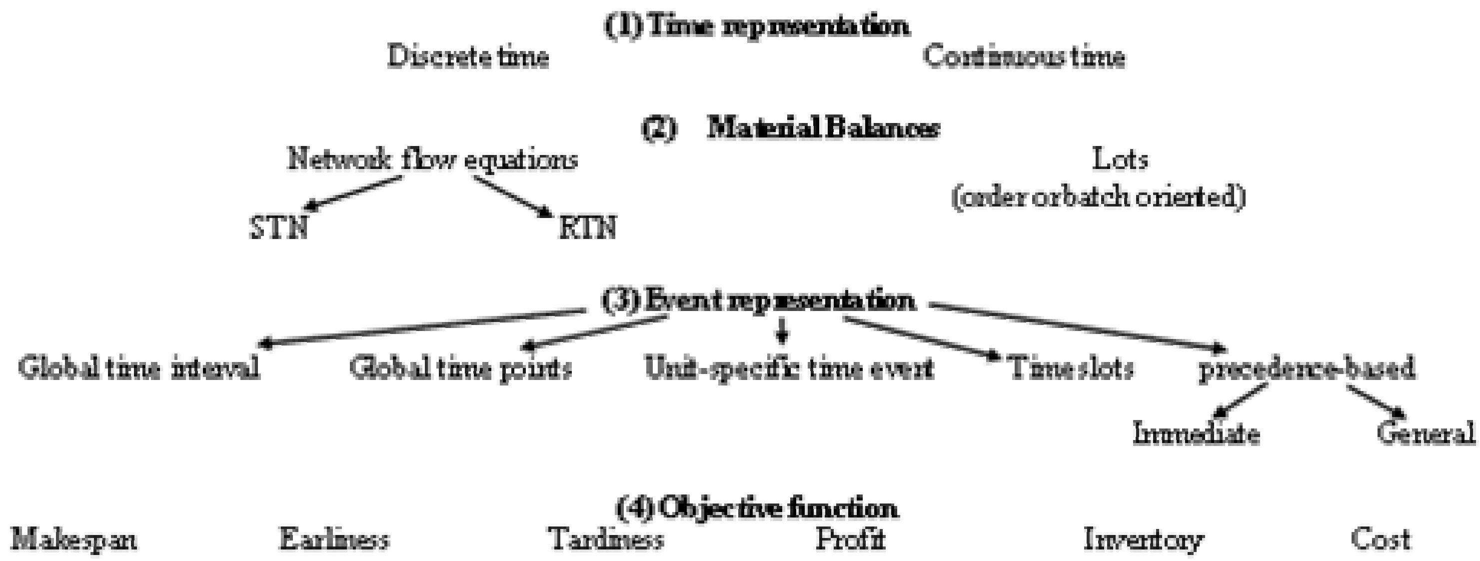
- Equipment
- Utilities
- Inventory
- Changeover

(13) Degree of certainty

- Deterministic
- Stochastic

Classification problems

Classification optimization models



Major differences in methods:

discrete vs continuous time

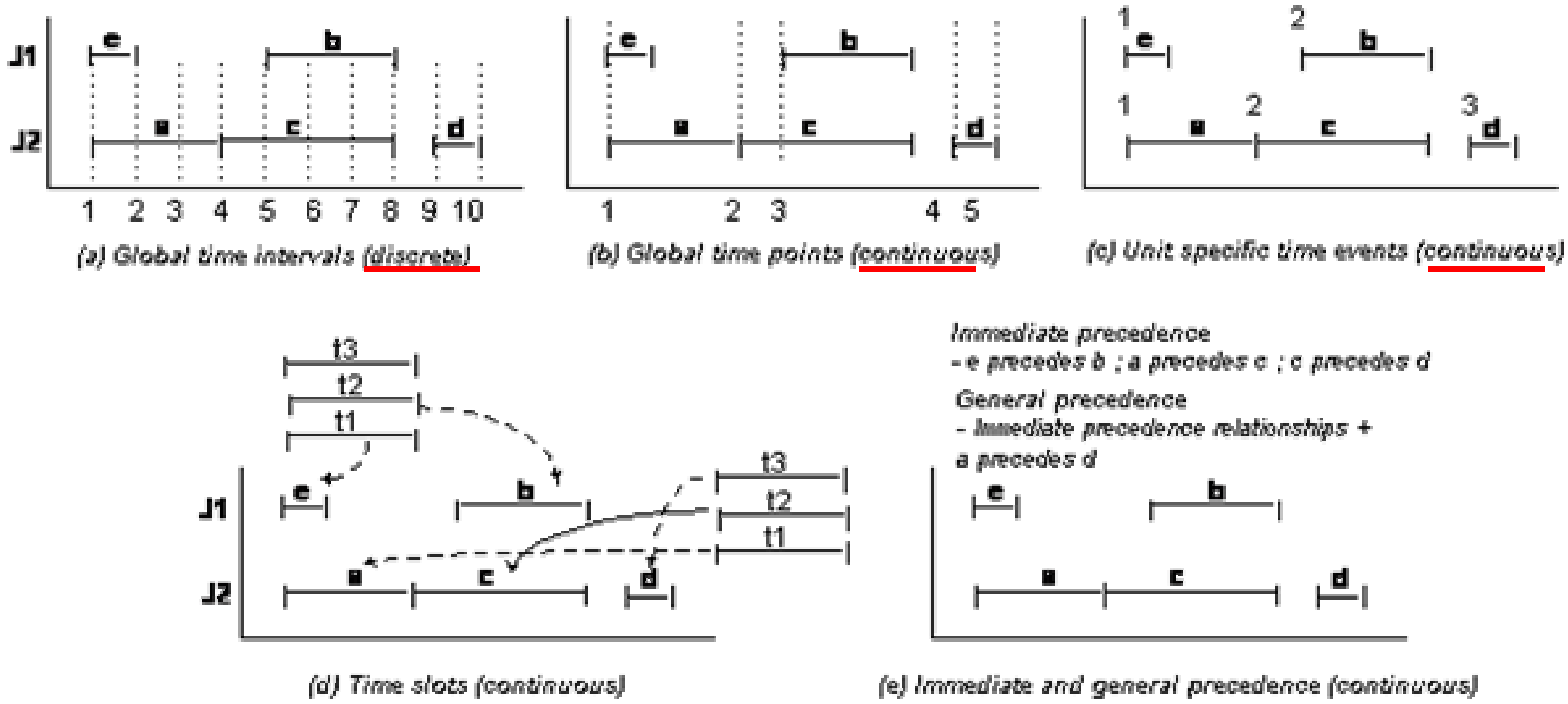
fixed variable batch sizes (splitting/mixing)

Performance models VERY sensitive to objective function

"Easiest": maximize profit

"Most difficult": minimize makespan (completion time)

Time representations



(a) Global time intervals (discrete)

(b) Global time points (continuous)

(c) Unit specific time events (continuous)

(d) Time slots (continuous)

(e) Immediate and general precedence (continuous)

Features of Discrete and Continuous Methods

Time representation	DISCRETE	CONTINUOUS					
Event representation	Global time intervals	Global time points	Unit-specific time events	Time slots*	Unit-specific immediate precedence*	Immediate precedence*	General precedence*
Main decisions	Lot-sizing, allocation, sequencing, timing	Lot-sizing, allocation, sequencing, timing		------------------- Allocation, sequencing, timing -------------------			
Key discrete variables	W_{ijt} defines if task *i* starts in unit *j* at the beginning of time interval *t*.	Ws_{in} / Wf_{in} define if task *i* starts/ends at time point *n*. $W_{inn'}$ defines if task *i* starts at time point *n* and ends at time point *n'*.	Ws_{in} / W_{in} / Wf_{in} define if task *i* starts/is active/ends at event point *n*.	W_{iljk} define if stage *l* of batch *i* is allocated to time slot *k* of unit *j*.	$X_{ii'j}$ defines if batch *i* is processed right before of batch *i'* in unit *j*. XF_{ij} defines if batch *i* starts the processing sequence of unit *j*	$X_{ii'}$ defines if batch *i* is processed right before of batch *i'*. XF_{ij} / W_{ij} defines if batch *i* starts/is assigned to unit *j*	$X'_{ii'}$ define if batch *i* is processed before or after of batch *i'*. W_{ij} defines if batch *i* is assigned to unit *j*
Type of process	General network	------- General network --------		------------------------------ Sequential --------------------------------			
Material balances	Network flow equations (STN or RTN)	Network flow equations (STN or RTN)	Network flow equations (STN)	---------------------------- Batch-oriented ----------------------------			
Critical modeling issues	Time interval duration, scheduling period (data dependent)	Number of time points (iteratively estimated)	Number of time events (iteratively estimated)	Number of time slots (estimated) and batch tasks (lot-sizing)	Number of batch tasks sharing units (lot-sizing) and units	Number of batch tasks sharing units (lot-sizing)	Number of batch tasks sharing resources (lot-sizing)
Critical problem features	Variable processing time, sequence-dependent changeovers	Intermediate due dates and raw-material supplies	Intermediate due dates and raw-material supplies	Inventory, resource limitations	Inventory, resource limitations	Inventory, resource limitations	Inventory

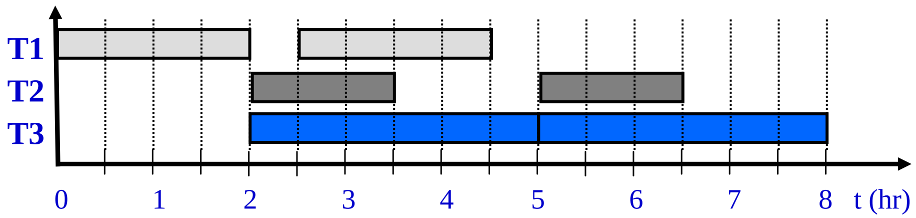
* Batch-oriented formulations assume that the overall problem is decomposed into the lot-sizing and the short-term scheduling issues. The lot-sizing or "batching" problem is solved first in order to determine the number and size of "batches" to be scheduled.

Discrete Time Formulations

Main Assumptions

- The scheduling horizon is divided into a finite number of time intervals with known duration
- Tasks can only start or finish at the boundaries of these time intervals

Discrete Time Representation



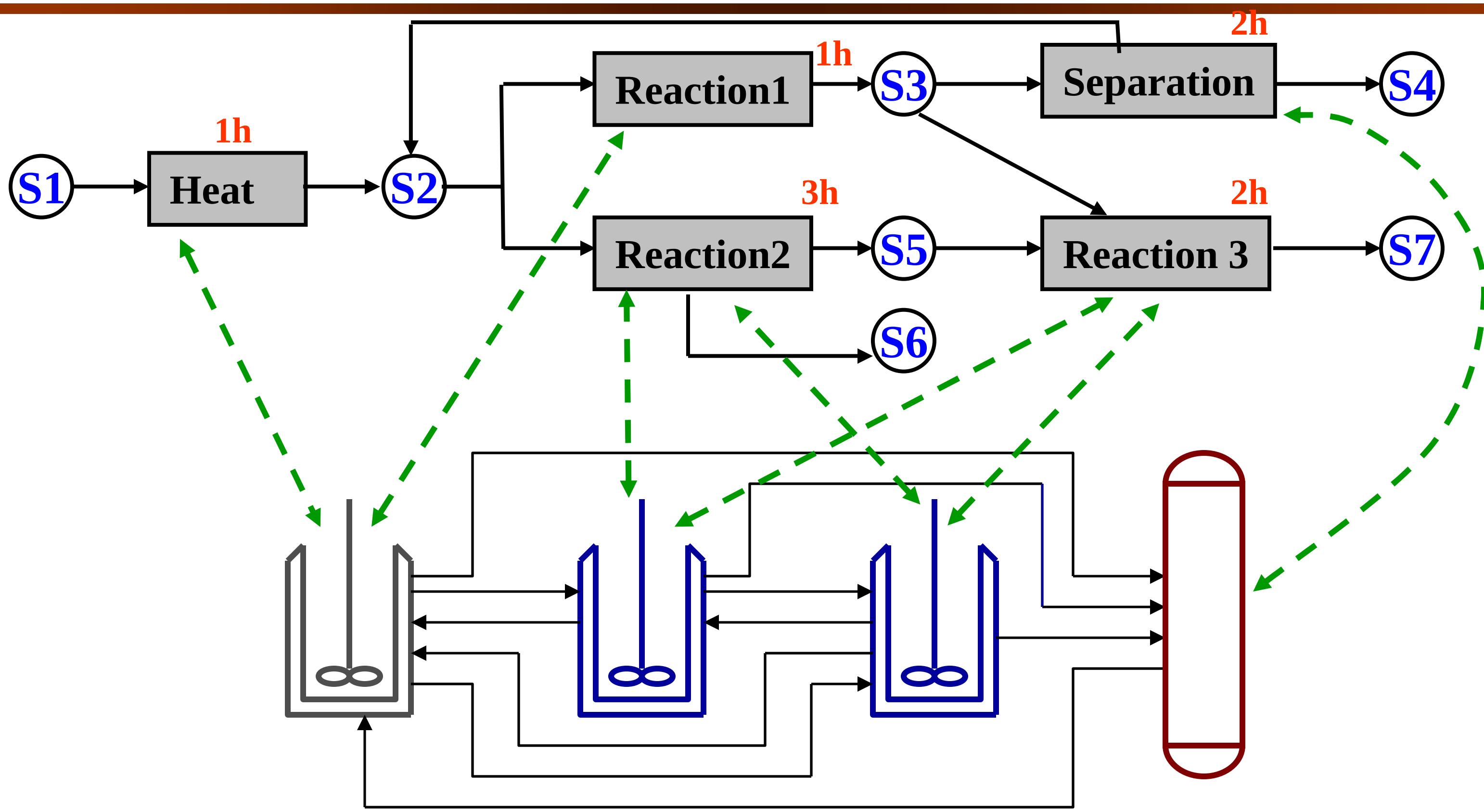
Advantages

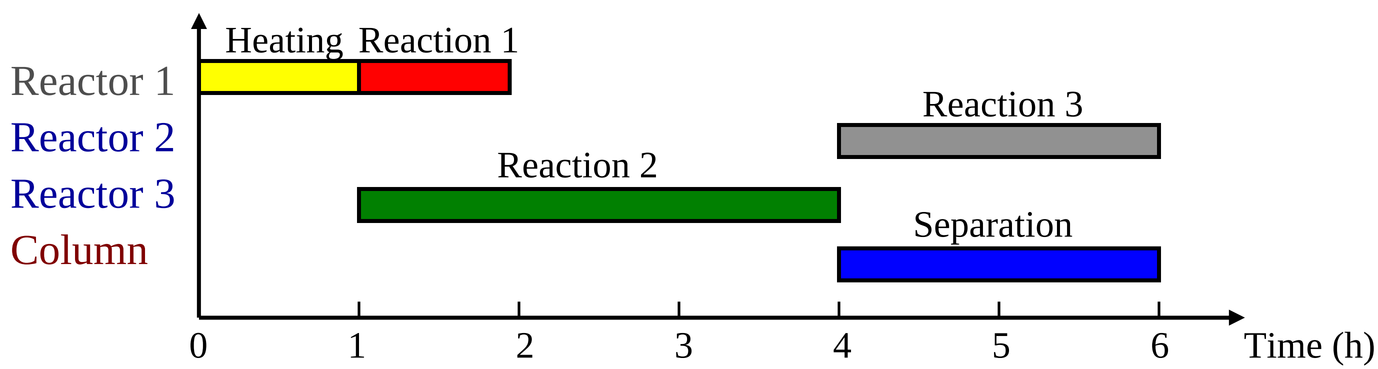
- Resource constraints are only monitored at predefined and fixed time points
- Simple models and easy representation of a wide variety of scheduling features

Disadvantages

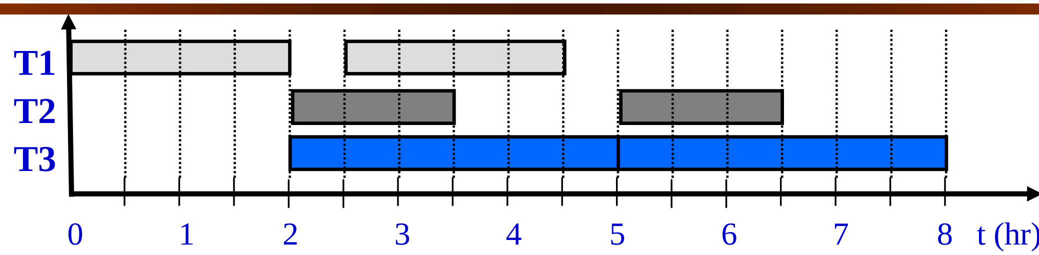
- Model size and complexity depend on the number of time intervals
- Constant processing times are required (rounding may be suboptimal)
- Changeovers are difficult to handle

State Task Network (STN) *(Kondili, Pantelides, Sargent, 1993)*





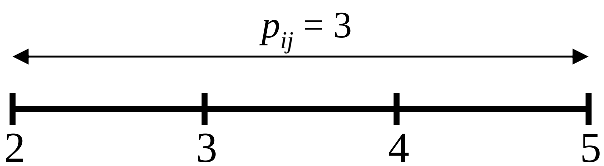
Discrete-time STN Model (1)



Variables:

W_{ijt} = 1 if unit *j* starts processing task *i* at the beginning of time period *t*; 0 otherwise.
B_{ijt} = Amount of material which starts undergoing task *i* in unit *j* at the beginning of period *t*.
S_{st} = Amount of material stored in state s, at the beginning of period t.
U_{ut} = Demand of utility *u* over time interval *t*.

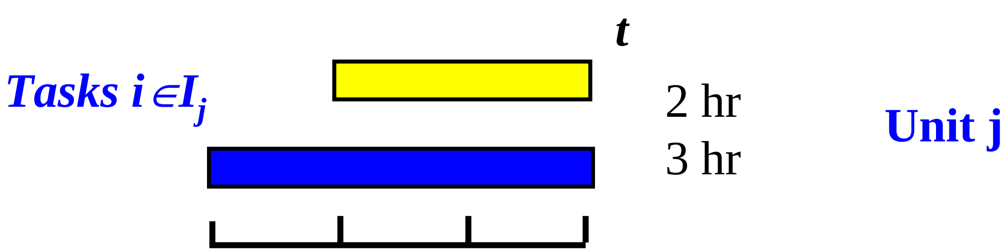
Task *i* starts at *t*=2 in unit *j*



$W_{ij2} = 1,\ B_{ij2} \neq 0$
$W_{ijt} = 0,\ B_{ijt} = 0,\ t \neq 2$

Allocation Constraints:

$$\sum_{i \in I_j} \sum_{\hat{t}=t}^{t-p_i+1} W_{ij\hat{t}} \leq 1 \quad \forall j,t$$



Discrete-time STN Model (2)

Capacity limitations:

$$W_{ijt}V_{ij}^{\min} \leq B_{ijt} \leq W_{ijt}V_{ij}^{\max} \quad \forall i,t,j \in K_i$$

Batch Units

$$0 \leq ST_{st} \leq C_s \quad \forall s,t$$

Storage capacity

Material balances:

Inventories **Produced** **Consumed** **Purchased/Sold**

$$ST_{st} = ST_{st-1} + \sum_{i \in \overline{T}_s} \overline{\rho}_{is} \sum_{j \in K_i} B_{ij,t-p_{is}} - \sum_{i \in T_s} \rho_{is} \sum_{j \in K_i} B_{ijt} + R_{st} - D_{st} \quad \forall s,t$$

Availability of utilities:

$$U_{ut} = \sum_{t} \sum_{j \in K_i} \sum_{\theta=0}^{p_i-1} (\alpha_{ui\theta} W_{ijt-\theta} + \beta_{ui\theta} B_{ijt-\theta}) \quad \forall u,t$$

Linear function batch size

$$0 \leq U_{ut} \leq U_{ut}^{\max} \quad \forall u,t$$

Objective Function:

$$\max \ Z = \sum_{s} \sum_{t=1}^{H} C_{st}^{D} D_{st} - \sum_{s} \sum_{t=1}^{H} C_{st}^{R} R_{st} + \sum_{s} C_{s,H+1} S_{s,H+1} - \sum_{u} \sum_{t=1}^{H} C_{ut} U_{ut}$$

Reformulation

Original MILP of Kondili was expensive to solve

Culprit: big-M allocation constraints

$$\sum_{i \in I_j} W_{ijt} \le 1 \qquad \forall j,t$$

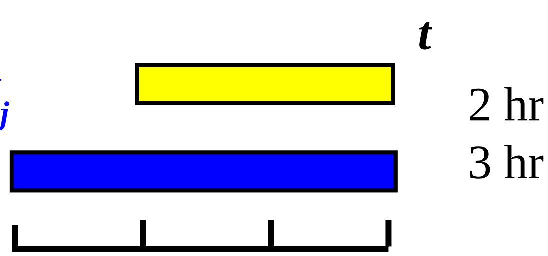
$$\sum_{i' \in I_j} \sum_{t'=t}^{t+p_{ij}-1} W_{i'jt'} - 1 \le M_{ij}\left(1 - W_{ijt}\right) \quad \forall i \in I_j, j, t$$

Solution: Replace by constraint below *(Shah, 1992)*

$$\sum_{i \in I_j} \sum_{\hat{t}=t}^{t-p_i+1} W_{ij\hat{t}} \le 1 \quad \forall j,t$$

Fewer and tighter !

Tasks $i \in I_j$

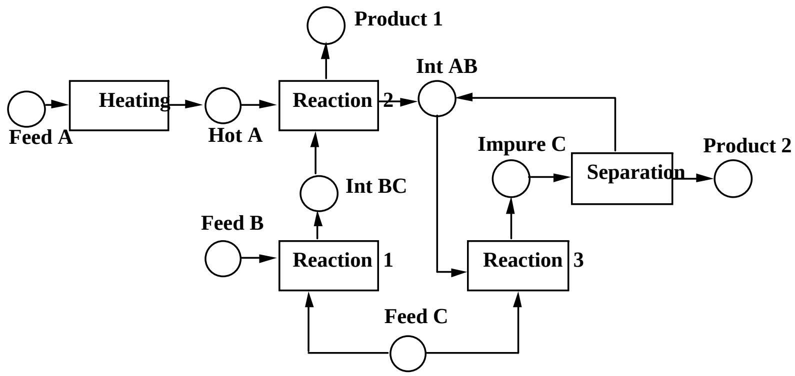


Unit j

Classical Kondili Example

Carnegie Mellon

STN



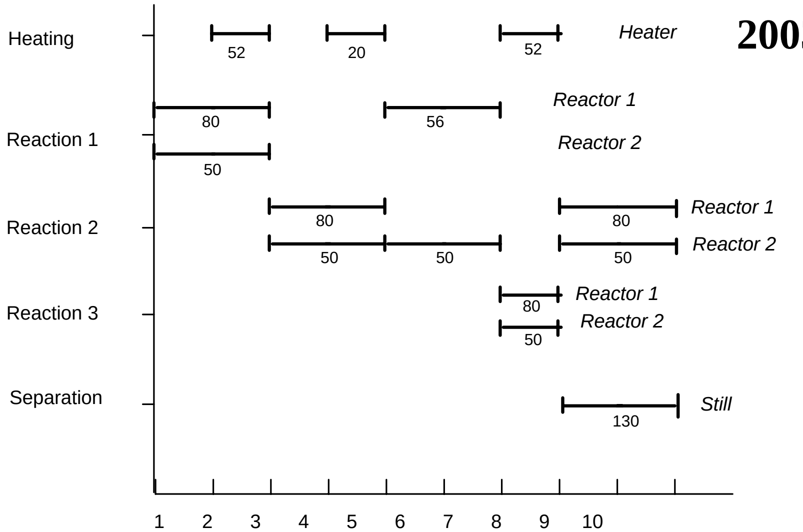
MILP

72 0-1 variables

179 continuous variables

250 constraints

Optimal Schedule



1987 Kondili's B&B: 908 sec, 1466 nodes, Vax-8600

1992 Shah's B&B: 119 sec, 419 nodes, SUN Sparc

2003 CPLEX 7.5: 0.45 sec, 22 nodes, IBM-T40

Disadvantages discrete-time STN:

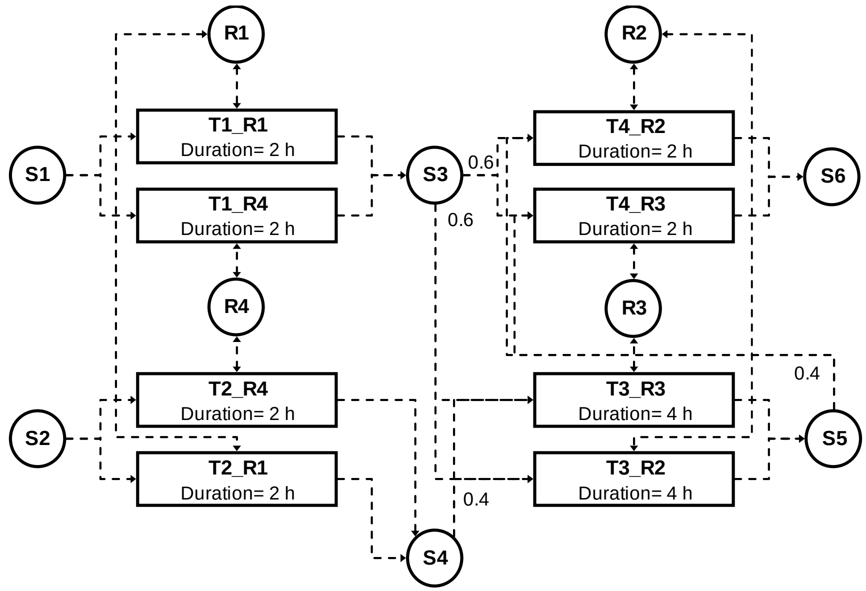
No. time intervals may be large

Changeovers not easy to handle: require definition cleanup tasks

RTN-based Discrete Time Formulation

Basic idea: material, equipment, labor treated are resources

(Pantelides, 1994).



$$R_{rt} = R_{r(t-1)} + \sum_{i \in I_r} \sum_{t'=0}^{pt_i} \left(\mu_{irt'} W_{i(t-t')} + \nu_{irt'} B_{i(t-t')} \right) + \prod_{rt} \qquad \forall r, t$$

$$0 \le R_{rt} \le R_{rt}^{\max} \qquad \forall r, t$$

RESOURCE BALANCE

$$V_{ir}^{\min} W_{it} \le B_{it} \le V_{ir}^{\max} W_{it} \qquad \forall i, r \in R_i^J, t$$

BATCH SIZE

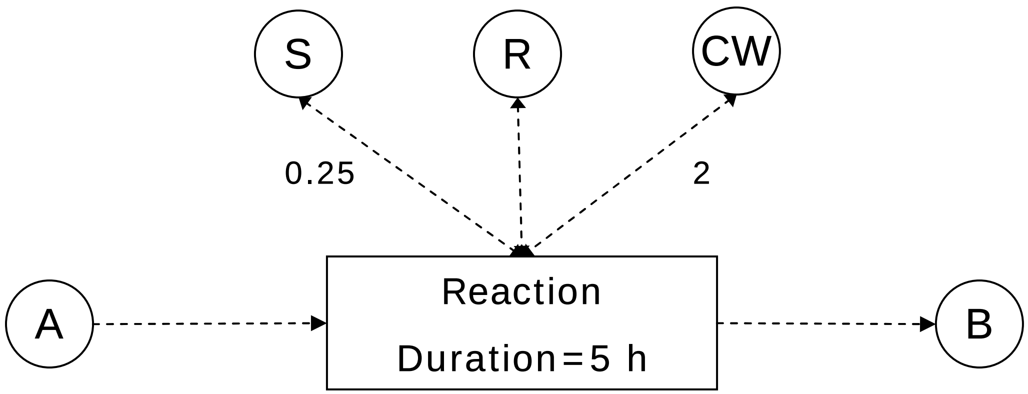
FEATURES

- **Very compact representation**
- **Not as intuitive as STN**
- **Computationally similar to STN**

RTN Modeling Example

Consider a <u>reaction task i, that lasts 5 hours</u>. It converts material A to B. It is carried out in a reactor. It uses <u>0.25 kg/s of steam per t of material being processed during the first hour</u>. Then it used <u>2 kg/s of cooling water per t of material being processed until the end of the operation</u>

Assume time intervals of one hour: δ=1 h



Reactor: $\mu_{i,R,0}=-1$; $\mu_{i,R,5}=1$

Materials: $\nu_{i,A,0}=-1$; $\nu_{i,B,5}=1$

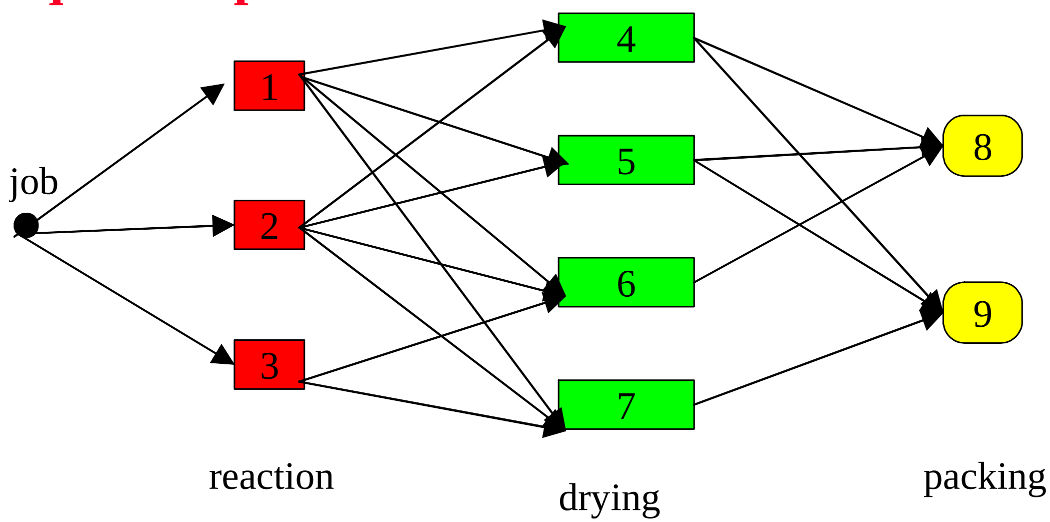
Utilities: $\nu_{i,S,0}=-0.25$; $\nu_{i,S,1}=0.25$

$\nu_{i,CW,1}=-2$; $\nu_{i,CW,5}$

Extension to continuous time STN/RTN has proved VERY difficult

- Zhang & Sargent (1995); Schilling & Pantelides (1996): RTN – Continuous
- Mockus & Reklaitis (1999): STN – Continuous
- Maravelias & Grossmann (2003): STN - Continuous
- Ierapetritou & Floudas (1998): Continuous Event-Based Formulation
- Cerda and Mendez (2000); Rodriguez et al. (2001); Lee et al. (2001); Castro et al. (2001)

Alternative: Sequential processes



Advantages: continuous time, can handle changeovers

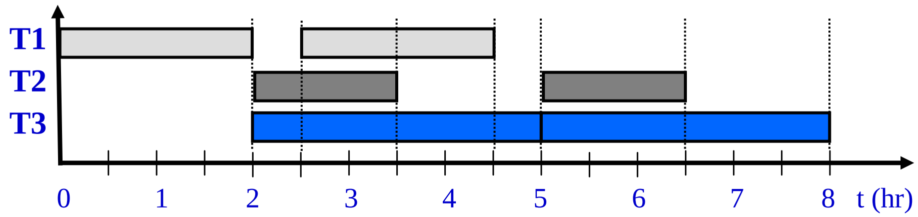
Disadvantages: cannot easily handle variable batch sizes, resource constraints

STN-based Continuous Time Formulation
(Global Time Points)

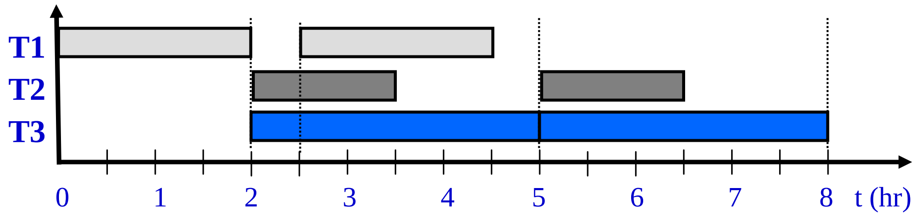
(Pantelides, 1996; Zhang and Sargent, 1996; Mockus and Reklaitis,1999; Mockus and Reklaitis, 1999; Lee et al., 2001, Giannelos and Georgiadis, 2002; Maravelias and Grossmann, 2003)

- Define a common time grid for all shared resources
- The maximum number of time points is predefined
- The time at which each time point takes place is a model decision (continuous domain)
- Tasks allocated to a certain time point *n* must start at the same time
- Only zero wait tasks must finish at a time point, others may finish before

Continuous Time Representation I



Continuous Time Representation II



ADVANTAGES

- Significant reduction in model size when the minimum number of time points is predefined
- Variable processing times
- Resource constraints are monitored at each time point

DISADVANTAGES

- Definition of the minimum number of time points
- Model size, complexity and optimality depend on the number of time points predefined

STN-based Continuous Formulation
(Global Time Points)

Carnegie Mellon

(Maravelias and Grossmann, 2003)

ALLOCATION CONSTRAINTS

$$\sum_{i \in I_j} Ws_{in} \le 1 \qquad \forall j,n$$

$$\sum_{i \in I_j} Wf_{in} \le 1 \qquad \forall j,n$$

$$\sum_{i \in I_j} \sum_{n' \le n} (Ws_{in'} - Wf_{in'}) \le 1 \qquad \forall j,n$$

$$\sum_n Ws_{in} = \sum_n Wf_{in} \qquad \forall i$$

BATCH SIZE CONSTRAINTS

$$V_i^{\min} Ws_{in} \le Bs_{in} \le V_i^{\max} Ws_{in} \qquad \forall i,n$$

$$V_i^{\min} Wf_{in} \le Bf_{in} \le V_i^{\max} Wf_{in} \qquad \forall i,n$$

$$V_i^{\min}\left(\sum_{n'<n} Ws_{in'} - \sum_{n' \le n} Wf_{in'}\right) \le Bp_{in} \le V_i^{\max}\left(\sum_{n'<n} Ws_{in'} - \sum_{n' \le n} Wf_{in'}\right) \qquad \forall i,n$$

$$Bs_{in-1} + Bp_{i(n-1)} = Bp_{in} + Bf_{in} \qquad \forall i, n > 1$$

MATERIAL AND RESOURCE BALANCES

$$S_{sn} = S_{s(n-1)} - \sum_{i \in I_s^c} \rho_{is}^c Bs_{in} + \sum_{i \in I_s^p} \rho_{is}^p Bf_{in} \qquad \forall s, n > 1$$

$$S_{sn} \le C_s^{\max} \qquad \forall s,n$$

$$R_{rn} = R_{r(n-1)} - \sum_i \mu_{ir}^c Ws_{in} + \nu_{ir}^c Bs_{in} + \sum_i \mu_{ir}^p Wf_{in} + \nu_{ir}^p Bf_{in} \qquad \forall r,n$$

TIMING AND SEQUENCING CONSTRAINTS

$$T_{n+1} \ge T_n \qquad \forall n$$

$$Tf_{in} \le T_n + \alpha_i Ws_{in} + \beta_i Bs_{in} + H(1 - Ws_{in}) \qquad \forall i,n$$

$$Tf_{in} \ge T_n + \alpha_i Ws_{in} + \beta_i Bs_{in} - H(1 - Ws_{in}) \qquad \forall i,n$$

$$Tf_{i(n-1)} \le T_n + H(1 - Wf_{in}) \qquad \forall i, n > 1$$

$$Tf_{i(n-1)} \ge T_n - H(1 - Wf_{in}) \qquad \forall i \in I^{ZW}, n > 1$$

$$Ts_{i'n} \ge Tf_{i(n-1)} + cl_{ii'} \qquad \forall j, i \in I_j, i' \in I_j, n$$

SHARED STORAGE TASKS

$$\sum_{s \in S_j} V_{jsn} \le 1 \qquad \forall j \in J^T, n$$

$$S_{sjn} \le C_j V_{jsn} \qquad \forall j \in J^T, s \in S_j, n$$

$$S_{sn} = \sum_{j \in J_s^T} S_{sjn} \qquad \forall s \in S^T, n$$

RTN-Based Continuous Formulation
(Global Time Points)

Carnegie Mellon

(Castro et al., 2004)

$$T_{n'} - T_n \geq \sum_{i \in I_r} \left(\alpha_i W_{inn'} + \beta_i B_{inn'} \right) \qquad \forall r \in R^J, n, n', (n < n')$$

$$T_{n'} - T_n \leq H \left(1 - \sum_{i \in I_r^{ZW}} W_{inn'} \right) + \sum_{i \in I_r^{ZW}} \left(\alpha_i W_{inn'} + \beta_i B_{inn'} \right) \qquad \forall r \in R^J, n, n', (n < n')$$

TIMING CONSTRAINTS

$$V_i^{min} W_{inn'} \leq B_{inn'} \leq V_i^{max} W_{inn'} \qquad \forall i, n, n', (n < n')$$

BATCH SIZE

$$R_{rn} = R_{r(n-1)} + \sum_{i \in I_r} \left[\sum_{n' < n} \left(\mu_{ir}^p W_{in'n} + \nu_{ir}^p B_{in'n} \right) - \sum_{n' > n} \left(\mu_{ir}^c W_{inn'} + \nu_{ir}^c B_{inn'} \right) \right] +$$
$$\sum_{i \in I^S} \left(\mu_{ir}^p W_{i(n-1)n} - \mu_{ir}^c W_{in(n+1)} \right) \qquad \forall r, n > 1$$

$$R_r^{min} \leq R_{rn} \leq R_r^{max} \qquad \forall r, n$$

RESOURCE BALANCE

$$V_i^{min} W_{in(n+1)} \leq \sum_{r \in R_i^S} R_{rt} \leq V_i^{max} W_{in(n+1)} \qquad \forall i \in I^s, n, (n \neq |N|)$$

$$V_i^{min} W_{i(n-1)n} \leq \sum_{r \in R_i^S} R_{rt} \leq V_i^{max} W_{i(n-1)n} \qquad \forall i \in I^s, n, (n \neq 1)$$

STORAGE CONSTRAINTS

STN-based Continuous Formulation
(Unit-specific Time Event)

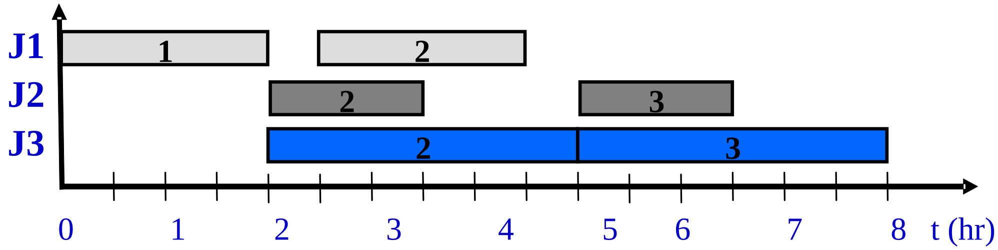
Carnegie Mellon

(Ierapetritou and Floudas, 1998; Vin and Ierapetritou, 2000; Lin et al., 2002; Janak et al., 2004).

Main Assumptions

- The number of event points is predefined
- Event points can take place at different times in different units

Event-Based Representation



Advantages

- More flexible timing decisions
- Fewer number of event points

Disadvantages

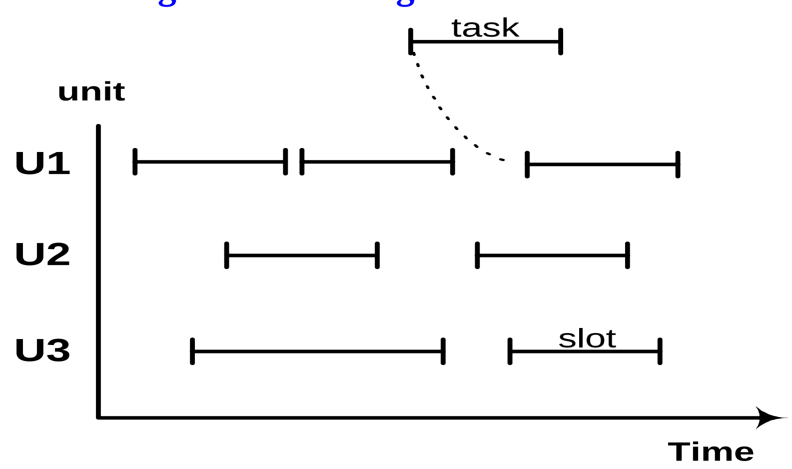
- Definition of event points (especially resource constraints, inventories)
- More complex models
- Additional tasks for storage and utilities

Continuous Time Formulations Sequential: Slot-based

(Pinto and Grossmann (1995, 1996); Chen et. al. ,2002; Lim and Karimi, 2003)

Main Assumptions

- **A number of time slots with unknown duration are postulated to be allocated to batches**
- **Batches to be scheduled are defined a priori**
- **No mixing and splitting operations**
- **Batches can start and finish at any time during the scheduling horizon**



Advantages

- **Significant reduction in model size when a minimum number of time slots is predefined**
- **Simple model and easy representation for sequencing and allocation scheduling problems**

Disadvantages

- **Resource and inventory constraints are difficult to model**
- **Model size, complexity and optimality depend on the number of time slots predefined**

Time-slot Continuous Time Formulation

(Pinto and Grossmann (1995)

$$\sum_{j}\sum_{k \in K_j} W_{ijkl} = 1 \qquad \forall i, l \in L_i$$ **BATCH ALLOCATION**

$$\sum_{i}\sum_{l \in L_i} W_{ijkl} \le 1 \qquad \forall j, k \in K_j$$ **SLOT ALLOCATION**

$$Tf_{jk} = Ts_{jk} + \sum_{i}\sum_{l \in L_i} W_{ijkl}\left(p_{ij} + su_{ij}\right) \qquad \forall j, k \in K_j$$ **SLOT TIMING**

$$Tf_{il} = Ts_{il} + \sum_{j}\sum_{k \in K_j} W_{ijkl}\left(p_{ij} + su_{ij}\right) \qquad \forall i, l \in L_i$$ **BATCH TIMING**

$$Tf_{jk} \le Ts_{j(k+1)} \qquad \forall j, k \in K_j$$ **SLOT SEQUENCING**

$$Tf_{il} \le Ts_{i(l+1)} \qquad \forall j, k \in K_j$$ **STAGE SEQUENCING**

$$-M\left(1 - W_{ijkl}\right) \le Ts_{il} - Ts_{jk} \qquad \forall i, j, k \in K_j, l \in L_i$$

$$M\left(1 - W_{ijkl}\right) \ge Ts_{il} - Ts_{jk} \qquad \forall i, j, k \in K_j, l \in L_i$$

SLOT-BATCH MATCHING

Global General Precedence Sequential Plants

Carnegie Mellon

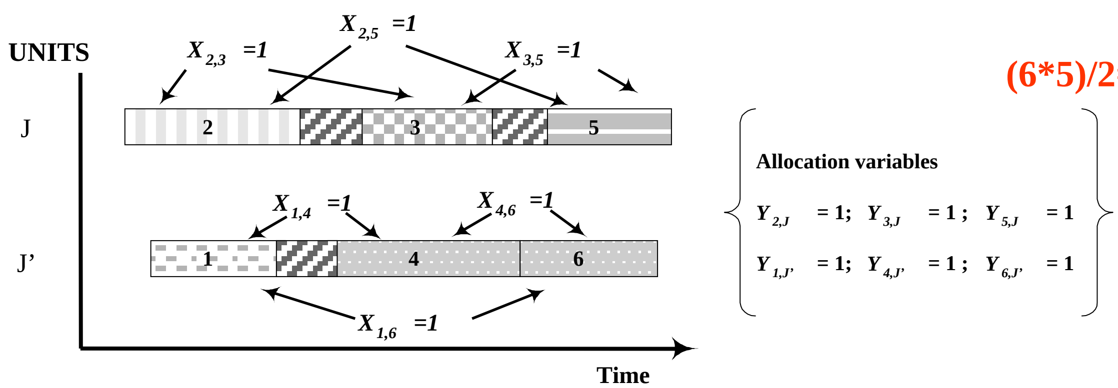
(Méndez et al., 2001; Méndez and Cerdá (2003,2004))

Main Assumptions

- Batches to be scheduled are defined a priori
- No mixing and splitting operations
- Batches can start and finish at any time during the scheduling horizon

6 BATCHES, 2 UNITS

(6*5)/2= 15 SEQUENCING VARIABLES



Advantages

- General sequencing is explicitly considered in model variables
- Changeover times and costs are easy to implement
- Lower number of sequencing decisions

Disadvantages

- Resource constraints are difficult to model
- Material balances cannot be handled

Global General Precedence Sequential Plants

Carnegie Mellon

(Méndez and Cerdá, 2003)

$$\sum_{j \in J_{il}} W_{ilj} = 1 \quad \forall i, l \in L_i$$

ALLOCATION CONSTRAINT

$$Tf_{il} = Ts_{il} + \sum_{j \in J_{il}} tp_{ilj} W_{ilj} \quad \forall i, l \in L_i$$

PROCESSING TIME

$$Ts_{i'l'} \geq Tf_{il} + cl_{il,i'l'} + su_{i'l'} - M\left(1 - X_{il,i'l'}\right) - M\left(2 - W_{ilj} - W_{i'l'j}\right) \quad \forall i, i', l \in L_i, l' \in L_{i'}, j \in J_{il,i'l'}$$

SEQUENCING CONSTRAINTS

$$Ts_{il} \geq Tf_{i'l'} + cl_{i'l',il} + su_{il} - M\, X_{il,i'l'} - M\left(2 - W_{ilj} - W_{i'l'j}\right) \quad \forall i, i', l \in L_i, l' \in L_{i'}, j \in J_{il,i'l'}$$

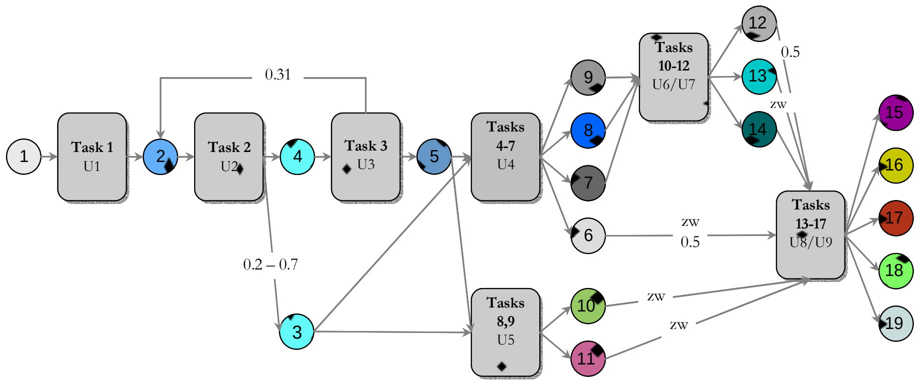
$$Ts_{il} \geq Tf_{i(l-1)} \quad \forall i, l \in L_i, l > 1$$

STAGE PRECEDENCE

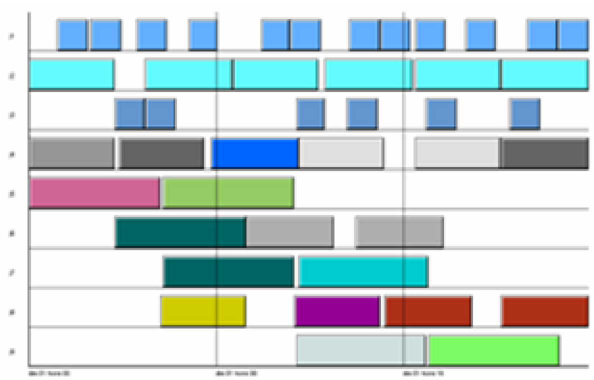
Comparison of models

Case study I

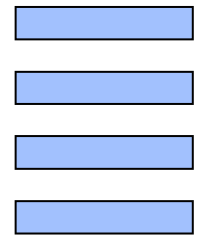
17 tasks, 19 states, 9 units



Westenberger, Kallrath (1995)

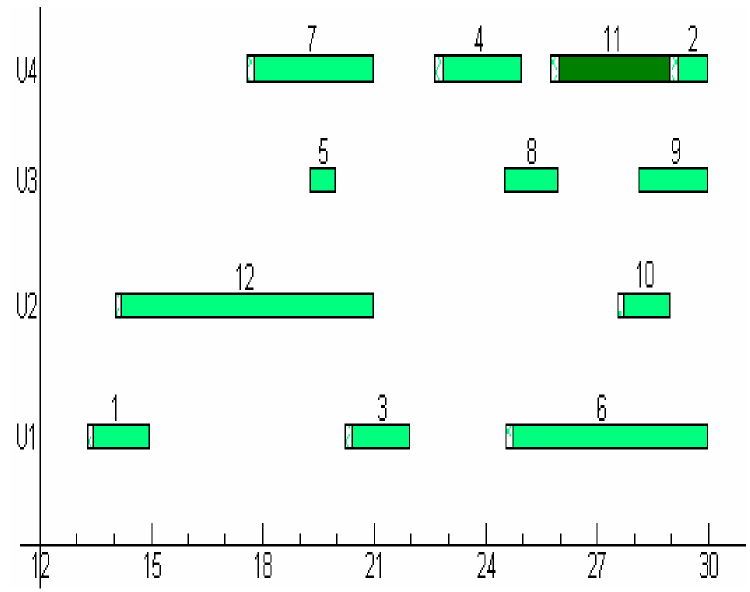


Case study II



4 reactors
12 orders
Manpower constraints

Pinto, Grossmann (1997)



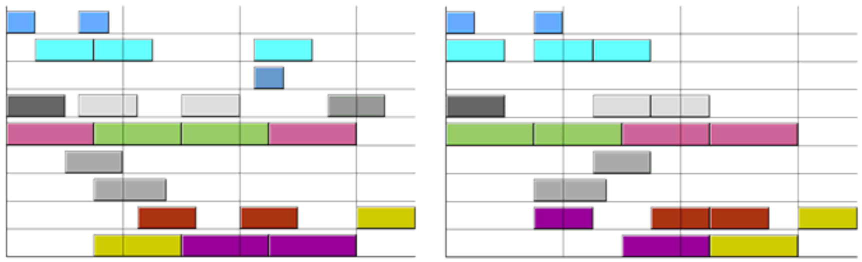
Case Study 1

Computational results for discrete and continuous STN models

Case Study	Event representation (time intervals or points)	Binary vars, cont. vars, constraints	LP relaxation	Objective function	CPU time[a]	Relative gap
1.a	Global time intervals (30)	720, 3542, 6713	9.9	28	1.34	0.0
	Global time points (8)	384, 2258, 4962	24.2	28	108.39	0.0
1.b	Global time intervals (240)	5760, 28322, 47851	1769.9	1425.8	7202	0.122
	Global time points (14)	672, 3950, 8476	1647	1407.4	258.54	0.042

[a] Seconds on Pentium IV PC with CPLEX 8.1 in GAMS 21.

Makespan Minimization

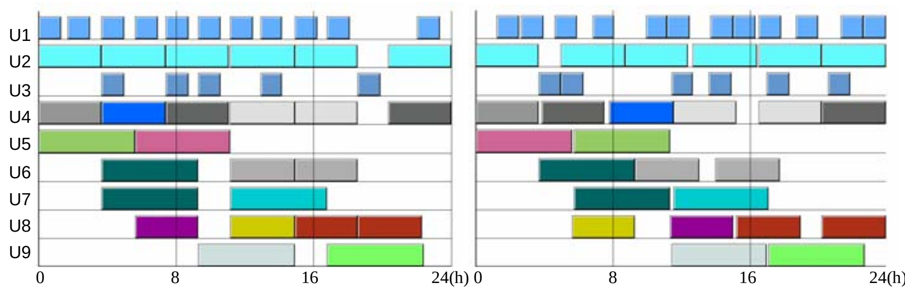


Discrete STN model   Continuous STN model

Gantt charts for case 1.a (Makespan minimization)

Discrete Time Faster

Profit Maximization



Discrete STN model   Continuous STN model

Gantt charts for case 1.b (Profit maximization)

Continuous time faster

Case Study 2

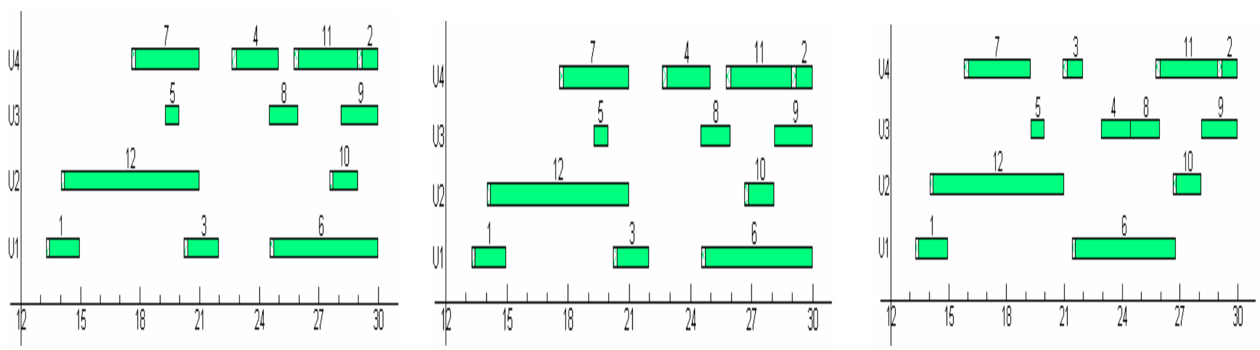
Carnegie Mellon

Comparison of model sizes and computational requirements

Case Study	Event representation	Binary vars, cont. vars, constraints	Objective function	CPU time	Nodes
2.a	Time slots & preordering	100, 220, 478	1.581	67.74[a] (*113.35*)*	456
	General precedence	82, 12, 202	1.026	0.11[b]	64
	Unit-based time events (4)	150, 513, 1389	1.026	0.07[c]	7
2.b	Time slots & preordering	289, 329, 1156	2.424	2224[a] (*210.7*)*	1941
	General precedence	127, 12, 610	1.895	7.91[b]	3071
	Unit-based time events (12)	458, 2137, 10382	1.895	6.53[c]	1374
2.c	Time slots & preordering	289, 329, 1156	8.323	76390[a] (*927.16*)*	99148
	General precedence	115, 12, 478	7.334	35.87[b]	19853
	Unit-based time events (12)	446, 2137, 10381	7.909	178.85[c]	42193

Seconds on [a] IBM 6000-530 with GAMS/OSL / [b] Pentium III PC with ILOG/CPLEX / [c] 3.0 GHz Linux workstation with GAMS 2.5/CPLEX 8.1.
*Seconds for disjunctive branch and bound

General precedence fastest



(a) without manpower limitation (b) 3 operator crews (c) 2 operators crews

Alternative Solution Approaches

Carnegie Mellon

(1) Exact methods
MILP
MINLP

(2) Constraint programming (CP)
Constraint satisfaction methods

(3) Meta-heuristics
Simulated annealing (SA)
Tabu search (TS)
Genetic algorithms (GA)

(4) Heuristics
Dispatching rules

(5) Artificial Intelligence (AI)
Rule-based methods
Agent-based methods
Expert systems

(6) Hybrid-methods
Exact methods + CP
Exact methods + Heuristics
Meta-heuristics + Heuristics

Exact methods provide rigorous and general basis

Solution of real-world problems requires:

Hybrid methods
Aggregation
Decomposition

MILP Formulation Continuous Time STN

(Maravelias, Grossmann, 2004)

$$\max Z = \sum_s SS_{sn}\zeta_s$$

$$\sum_{i\in I(j)}\sum_{n'\le n}(Ws_{in'} - Wf_{in'}) \le 1 \quad \forall j, \forall n$$

$$\sum_n Ws_{in} = \sum_n Wf_{in} \quad \forall i$$

Novel Assignment Constraints

$$D_{in} = \alpha_i Ws_{in} + \beta_i Bs_{in} \quad \forall i, \forall n$$

$$Tf_{in} \le Ts_{in} + D_{in} + H(1 - Ws_{in}) \quad \forall i, \forall n$$

$$Tf_{in} \ge Ts_{in} + D_{in} - H(1 - Ws_{in}) \quad \forall i, \forall n$$

$$Ts_{in} = T_n \quad \forall i, \forall n$$

$$Tf_{in-1} \le T_n + H(1 - Wf_{in}) \quad \forall i, \forall n$$

$$Tf_{in-1} \ge T_n - H(1 - Wf_{in}) \quad \forall i \in ZW(i), \forall n$$

Finish time of task *i*

Big-M

$$B_i^{MIN} Ws_{in} \le Bs_{in} \le B_i^{MAX} Ws_{in} \quad \forall i, \forall n$$

$$B_i^{MIN} Wf_{in} \le Bf_{in} \le B_i^{MAX} Wf_{in} \quad \forall i, \forall n$$

$$Bs_{in-1} + Bp_{in-1} = Bp_{in} + Bf_{in} \quad \forall i, \forall n$$

$$B_{isn}^I = \rho_{is} Bs_{in} \quad \forall i, \forall n, \forall s \in SI(i)$$

$$B_{isn}^O = \rho_{is} Bf_{in} \quad \forall i, \forall n, \forall s \in SO(i)$$

$$S_{sn} = S_{s,n-1} + \sum_{i\in O(s)} B_{isn}^O - \sum_{i\in I(s)} B_{isn}^I + SP_{sn} - SS_{sn} \quad \forall s, \forall n > 1$$

Mass balances

$$R_{irn}^I = \gamma_{ir} Ws_{in} + \delta_{irs} Bs_{in} \quad \forall i, \forall r, \forall n$$

$$R_{irn}^O = \gamma_{ir} Wf_{in} + \delta_{irs} Bf_{in} \quad \forall i, \forall r, \forall n$$

$$R_{rn} = R_{rn-1} - \sum_i R_{irn-1}^O + \sum_i R_{irn}^I \quad \forall r, \forall n$$

Utility Constraints

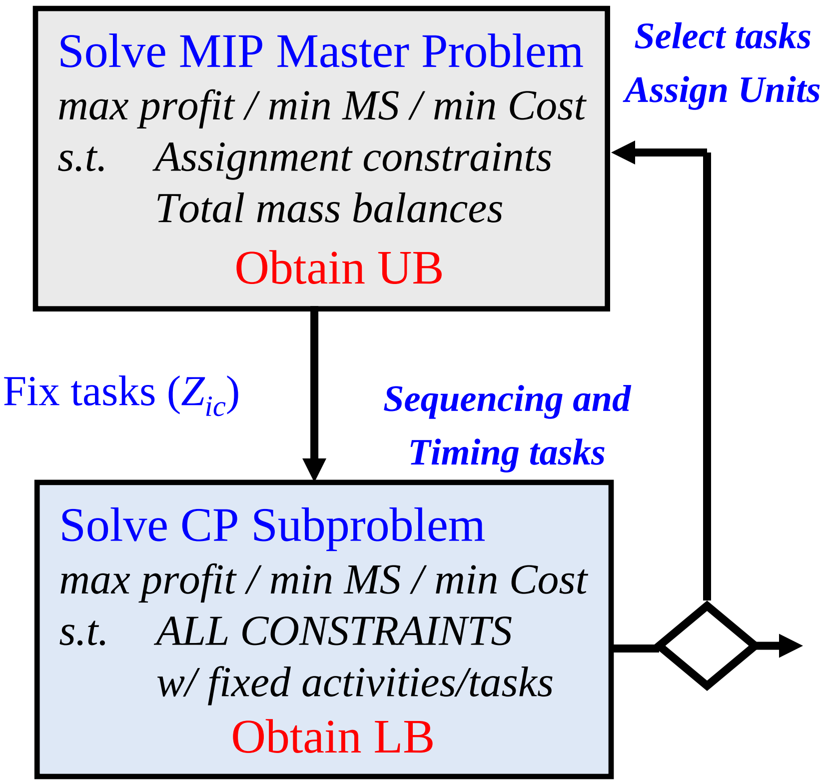
$$\sum_{i\in I(j)}\sum_{n'\ge n} D_{in'} \le H - T_n \quad \forall j, \forall n$$

$$\sum_{i\in I(j)}\sum_{n'\le n} (Wf_{in'} fd_i + Bf_{in'} vd_i) \le T_n \quad \forall j, \forall n$$

Novel Tightening Constraints

Hybrid MILP/Constraint Programming Method

$Z_{ic} = 1$ if copy c of task i is carried out



$$\sum_{i \in I(j)} \sum_{c} D_{ic} Z_{ic} \le H \quad \forall j$$

$$B_i^{MIN} Z_{ic} \le B_{ic} \le B_i^{MAX} Z_{ic} \quad \forall i, \forall c$$

$$S_s = S_0 + \sum_i \sum_c \rho_{is}^O B_{ic} - \sum_i \sum_c \rho_{is}^I B_{ic} \quad \forall s$$

$$S_s \ge d_s \quad \forall s \in FP$$

$$S_s \le C_s \quad \forall s \in INT$$

$$Z_{ic+1} \le Z_{ic} \quad \forall i, \forall c < |C|$$

Integer Cuts

$$B_i^{MIN} \le B_{ic} \le B_i^{MAX} \quad \forall i, \forall c$$

$$B_{ics}^I = \rho_{is}^I B_{ic} \quad \forall i, \forall c, \forall s$$

$$B_{ics}^O = \rho_{is}^O B_{ic} \quad \forall i, \forall c, \forall s$$

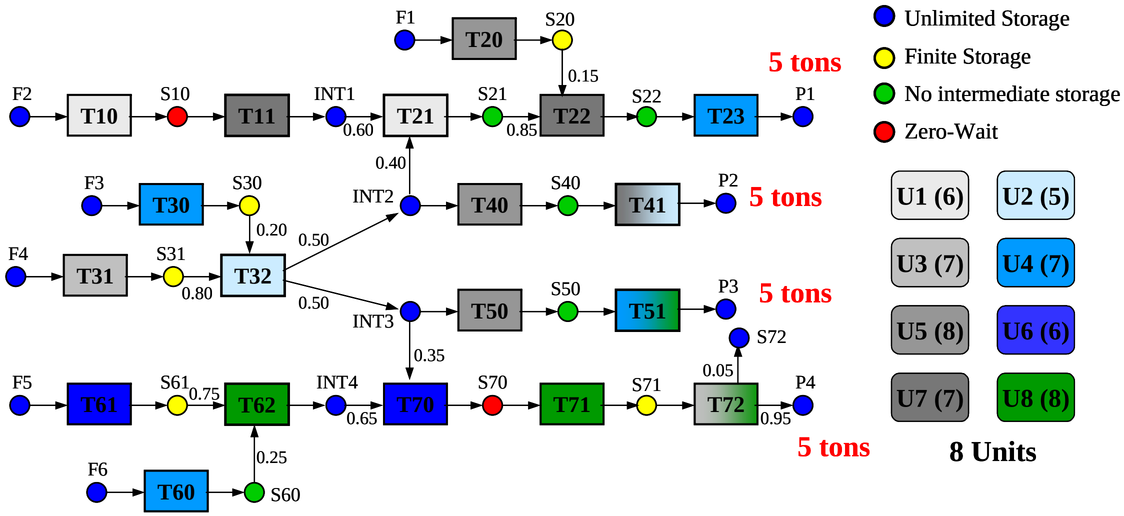
$$R_{ic} = \alpha_i + \beta_i B_{ic} \quad \forall i, \forall c$$

$$\sum_i \sum_c B_{ics}^O \ge d_s \quad \forall s \in FP$$

Task[i,c] requires Unit[j]	$\forall j, \forall i \in I(j), \forall c$
Task[i,c] requires R_{ic} *Utility[r]*	$\forall i, \forall c$
Task[i,c] consumes B^I_{ics} *State[s]*	$\forall i, \forall c, \forall s$
Task[i,c] produces B^O_{ics} *State[s]*	$\forall i, \forall c, \forall s$
Task[i,c].end $\le MS$	$\forall i, \forall c$
Task[i,c] precedes Task[i,c+1]	$\forall i, \forall c < \|C\|$

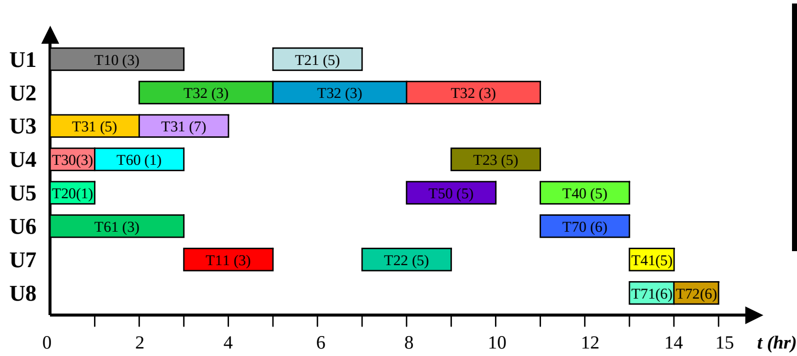
- Tasks ⇒ Activities
- Units ⇒ Unary Resources
- Utilities ⇒ Discrete Resources
- States ⇒ Reservoirs

Example Makespan Minimization



Continuous-time MILP: Unsolvable

Optimal Schedule



Minimum completion time = 15 hours

Proposed Method

Optimum schedule found in 5 seconds !!

CPLEX 7.5/ILOG Solver 5.2

Commercial Software

Aspen Plant Scheduler
(Aspentech)

Model Enterprise Optimal Single Site Scheduler *(OSS Scheduler)*
(PSEnterprise)

VirtECS Schedule
(Advanced Process Combinatorics)

SAP Advanced Planner and Optimizer *(SAP APO)*
(SAP)

Beyond current Scheduling Capabilities

1. Reactive Scheduling

Changes while executing a schedule:

- New orders

- Equipment breakdown

Approaches rely mostly on making small changes (e.g. Mendez et al.)

2. Integration of Planning and Scheduling

Key: Aggregated models, decomposition

Example: Erdirik & Grossmann (2005)

3. Integration of Process Models

Generally leads to MINLP problems

Examples: Mendez et. al (2005), Flores & Grossmann (2005)

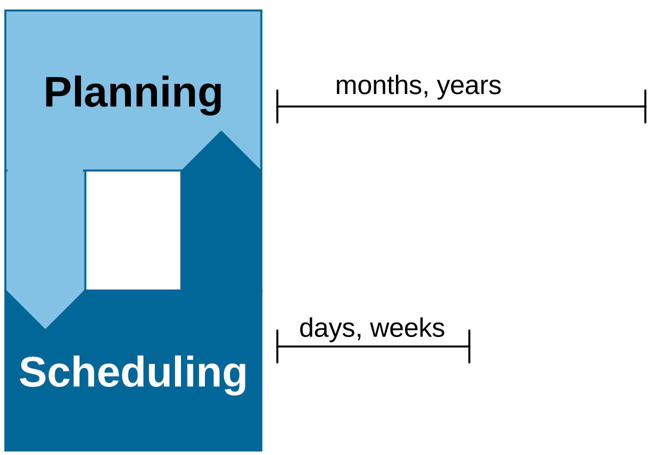
Carnegie Mellon

Approaches to Planning and Scheduling

Erdirik, Grossmann, 2005)

Decomposition

Sequential Hierarchical Approach

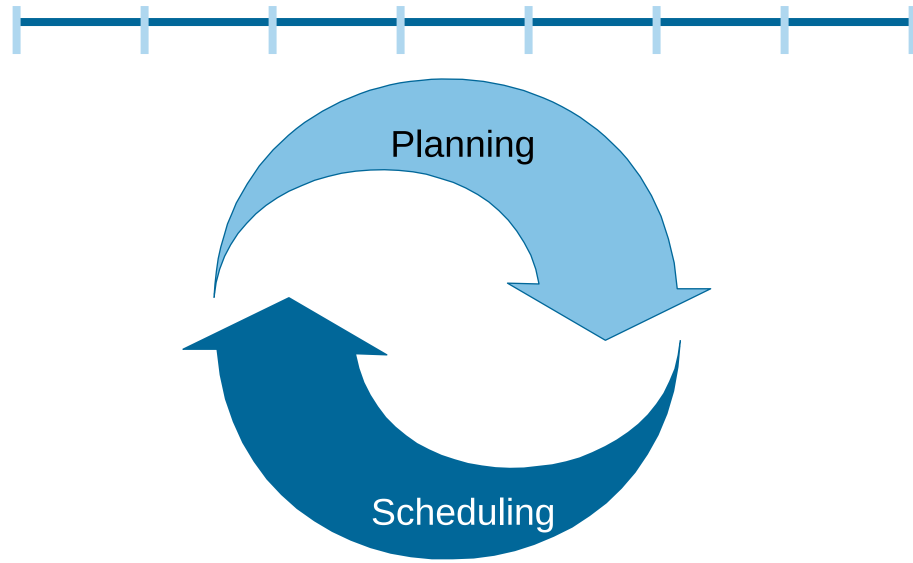


Challenges:

- Different models / different time scales
- Mismatches between the levels

Simultaneous Planning and Scheduling

Detailed scheduling over the entire horizon



Challenges:

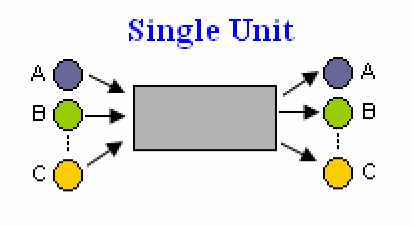
- Very Large Scale Problem
- Solution times quickly intractable

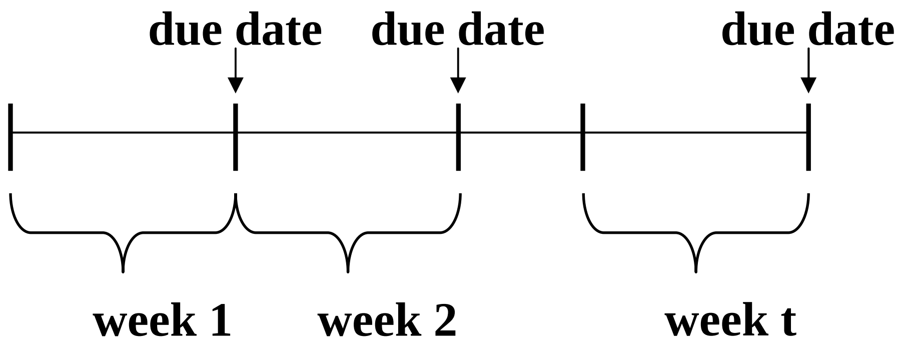
GOAL:

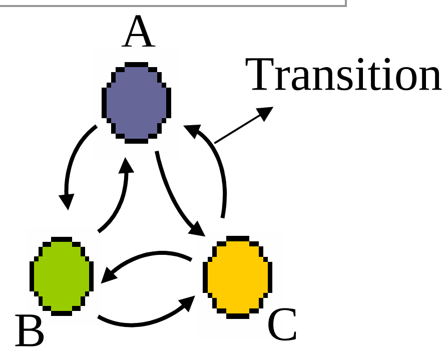
- Propose a novel decomposition algorithm to integrate planning and scheduling for multiproduct continuous plants.
- Ensure optimality and consistency between the two levels.

Planning and Scheduling of Continuous Plants

- **Multiproducts** to be processed in a single **continuous** unit/production line
- Time horizon subdivided into **weeks** at the end of which demands are specified.
- Transition times are **sequence dependent**
- **Continuous** time representation is used.







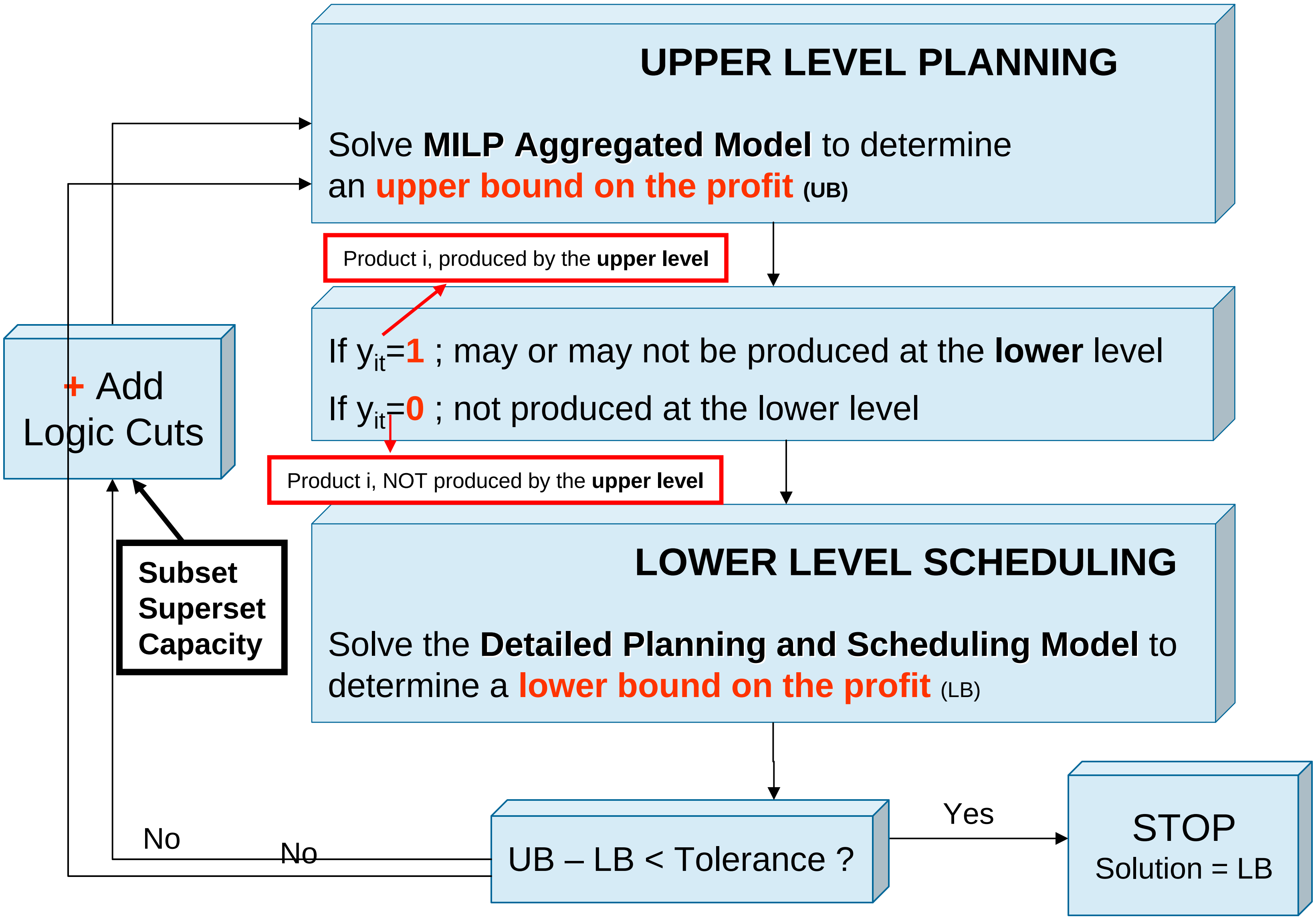
Decisions

- Amounts to be produced
- Length of processing times
- Product inventories
- Sequencing of products

Objective

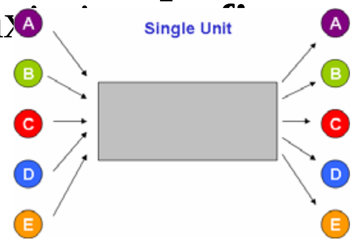
- Max Profit = Sales –Operating Costs – Inventory Costs – Transition Costs

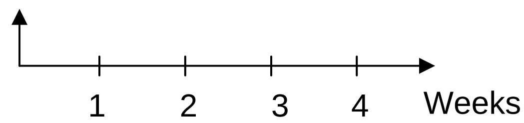
Proposed Decomposition Algorithm



Example

- Determine **plan and schedule** for **5** products for a planning horizon of **4 weeks** to max





Demand input data for Example 1

	Time Period			
	1	**2**	**3**	**4**
	Demand (kg)			
A	10,000	20,000	30,000	10,000
B	25,000	20,000	15,000	25,000
C	30,000	40,000	50,000	30,000
D	30,000	20,000	13,000	30,000
E	30,000	20,000	12,000	30,000

Production rated data for Example 1

Product	Production Rates (kg/hr)
A	800
B	900
C	1,000
D	1,000
E	1,200

Transition Data for Example 1

	Product				
Product	**A**	**B**	**C**	**D**	**E**
	Transition times (hrs)				
A	0.00	2.00	1.50	1.00	0.75
B	1.00	0.00	2.00	0.75	0.50
C	1.00	1.25	0.00	1.50	2.00
D	0.50	1.00	2.00	0.00	1.75
E	0.70	1.75	2.00	1.50	0.00
	Transition costs ($)				
A	0	760	760	750	760
B	745	0	750	770	740
C	770	760	0	765	765
D	740	740	745	0	750
E	740	740	750	750	0

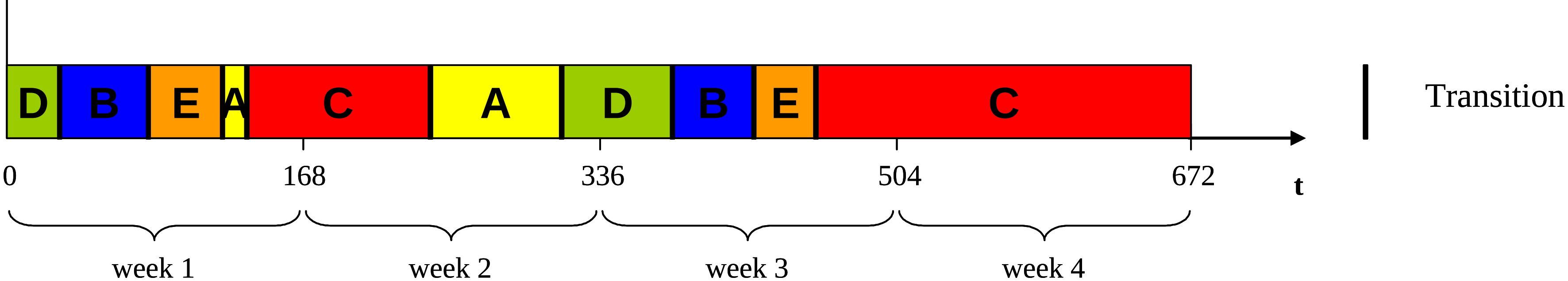
Cost data for Example 1

	Operating Costs ($/kg)	Selling Price ($/kg
A	0.19	0.25
B	0.32	0.40
C	0.55	0.65
D	0.49	0.55
E	0.38	0.45

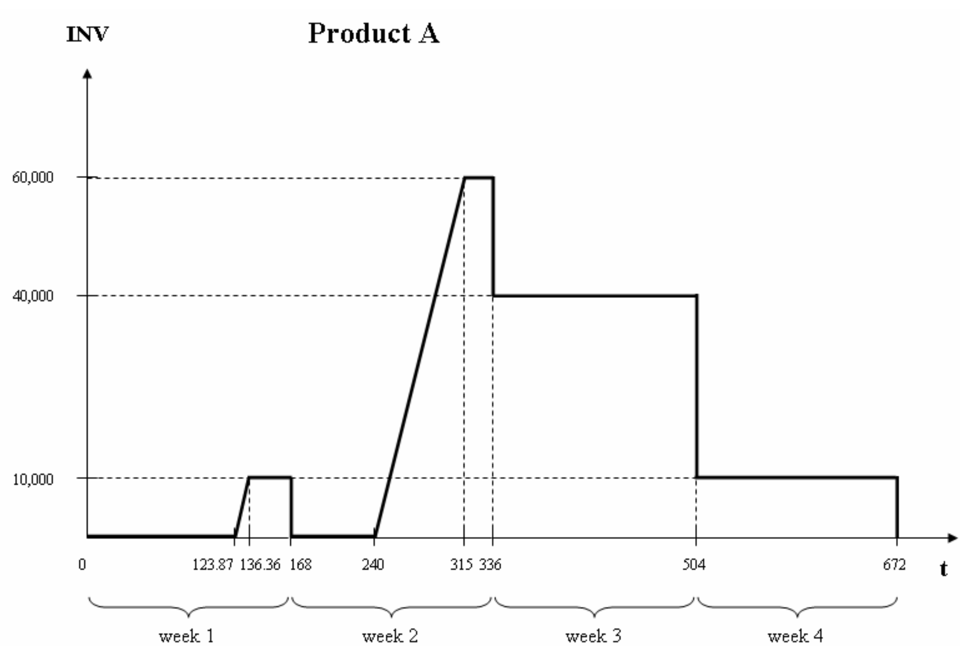
Inventory Cost ($/kg.h)
0.0000306

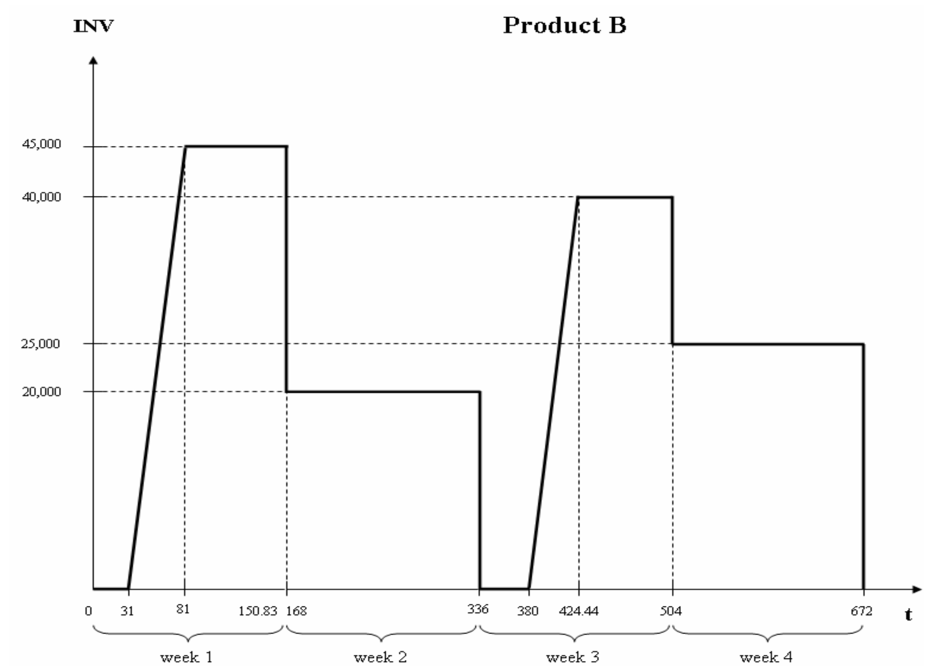
Results of Example

Gantt Chart for the planning horizon



Inventory levels for Product A and B



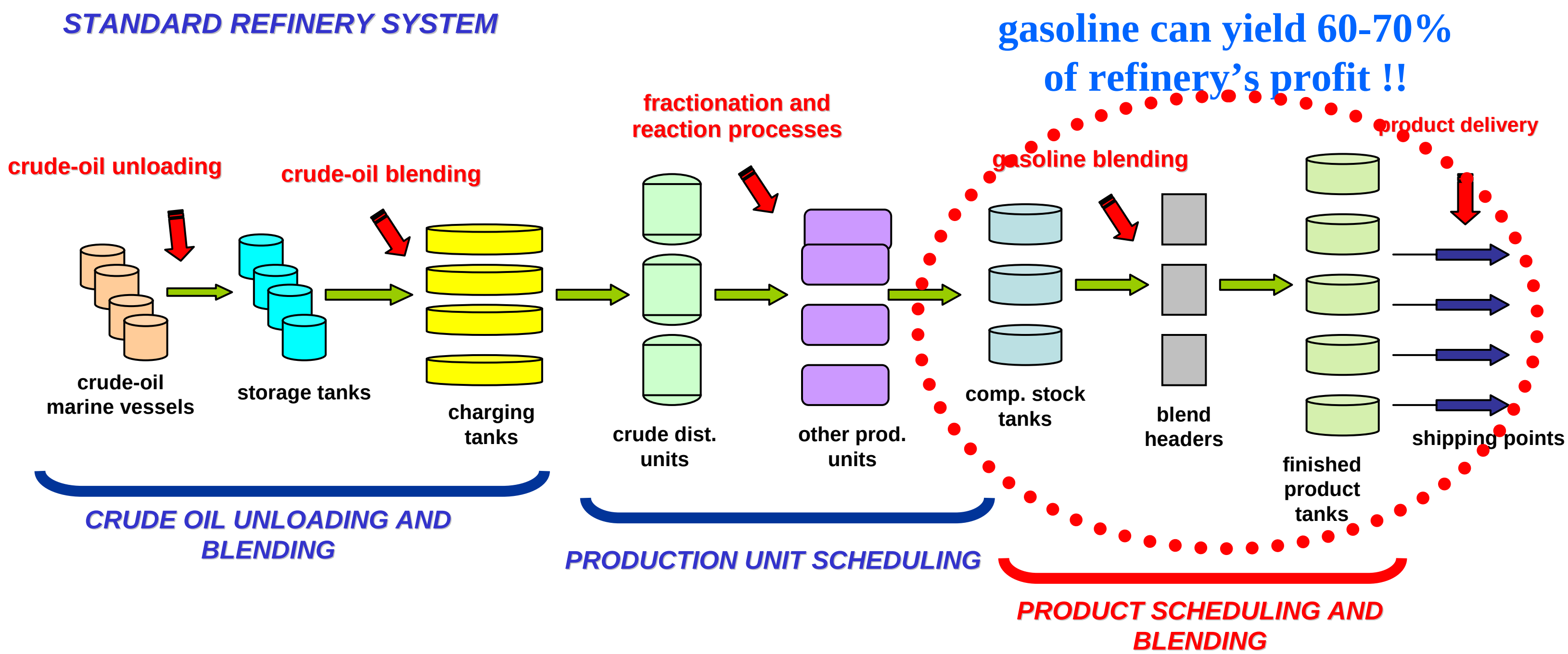


Computational Results

Method	Number of binary variables	Number of continuous variables	Number of Equations	Number of Major Iterations	Time (CPUs)	Solution ($)
Full Space	120	987	906		**6000***	**43,120.8**
Proposed algorithm (0%)				**15**	**207.9**	**43,120.8**
Problem UB	20	151	564		2.0	43,013.0
Problem LB	120	996	949		205.9	43,120.8
Proposed algorithm (1%)				**4**	**28.9**	**43,120.8**
Problem UB	20	151	251		0.2	43,540.5
Problem LB	120	996	938		28.7	43,120.8

**8% gap*

Refinery Scheduling and Blending



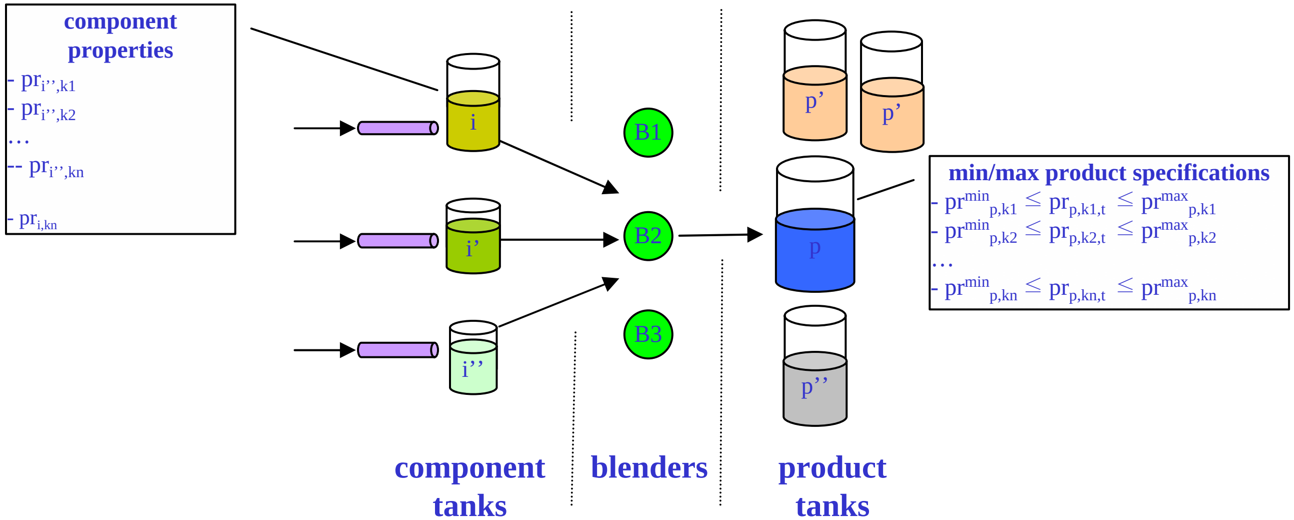
- **Production logistics (scheduling)**

  Multiple product demands
  Resource allocation
  Timing of operations
  Inventory and pumping constraints
  Logistic and operating rules

- **Production quality (blending)**

  Variable product recipes
  Product specifications
  Complex correlations for product properties

Problem statement



GIVEN:

- Scheduling horizon
- Components, Products
- Storage tanks, Blend headers
- Component properties, stocks and supplies
- Product specifications, stocks and demands
- Min/Max flowrates and concentrations
- Correlations for predicting product properties
- Operating rules

THE GOAL IS TO DETERMINE:

- Allocation of resources
- Inventory levels in tanks
- Component concentrations in each product
- Volume of each product
- Pumping rates
- Production and storage tasks timing

MAXIMIZE PRODUCTION PROFIT

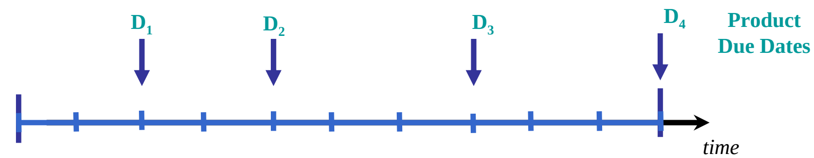
REQUIRES SIMULTANEOUS SCHEDULING AND BLENDING

Proposed optimization approach

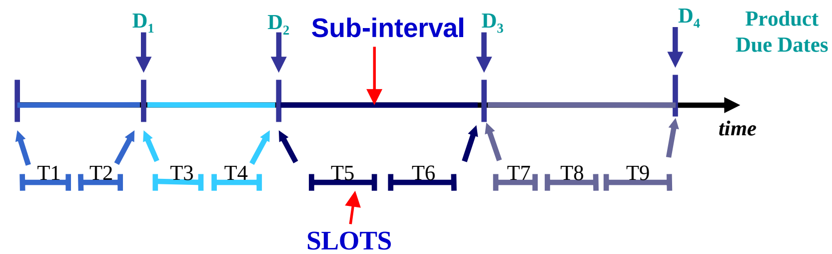
MILP multiperiod optimization method

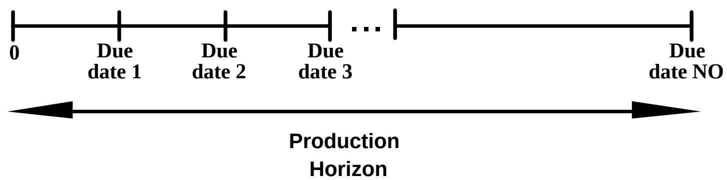
- Discrete or continuous time domain representation
- Linear approximations for product properties
- Iterative procedure for improving predictions
- Integrated production logistics and quality specifications
- Variable recipes and min/max component concentrations
- Discrete decisions for resource allocation and operating rules

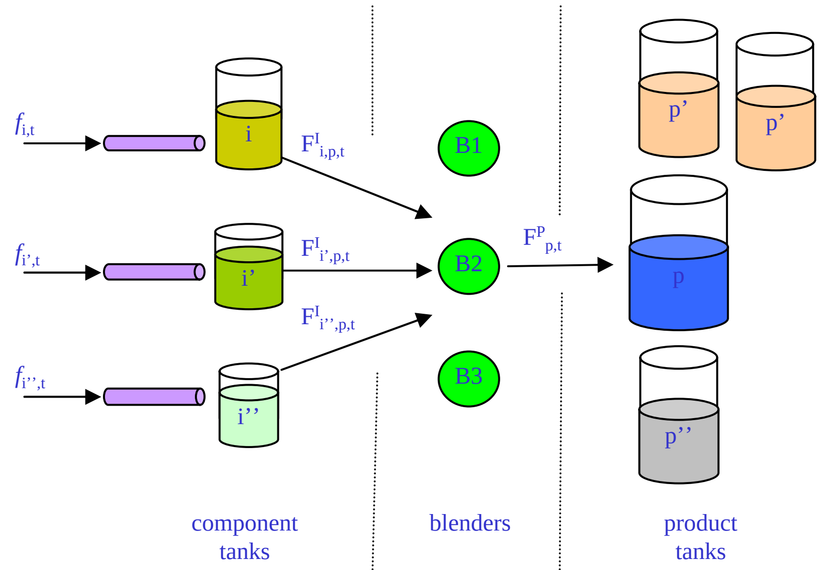
DISCRETE TIME FORMULATION



CONTINUOUS TIME FORMULATION







Product property prediction

•Volumetric average

$$pr_{p,k,t} = \sum_i pr_{i,k} v^I_{i,p,t} \qquad \forall p,k,t$$

•Non-linear flowrate-composition matching

$$v^I_{i,p,t} F^P_{p,t} = F^I_{i,p,t} \qquad \forall i,p,t$$

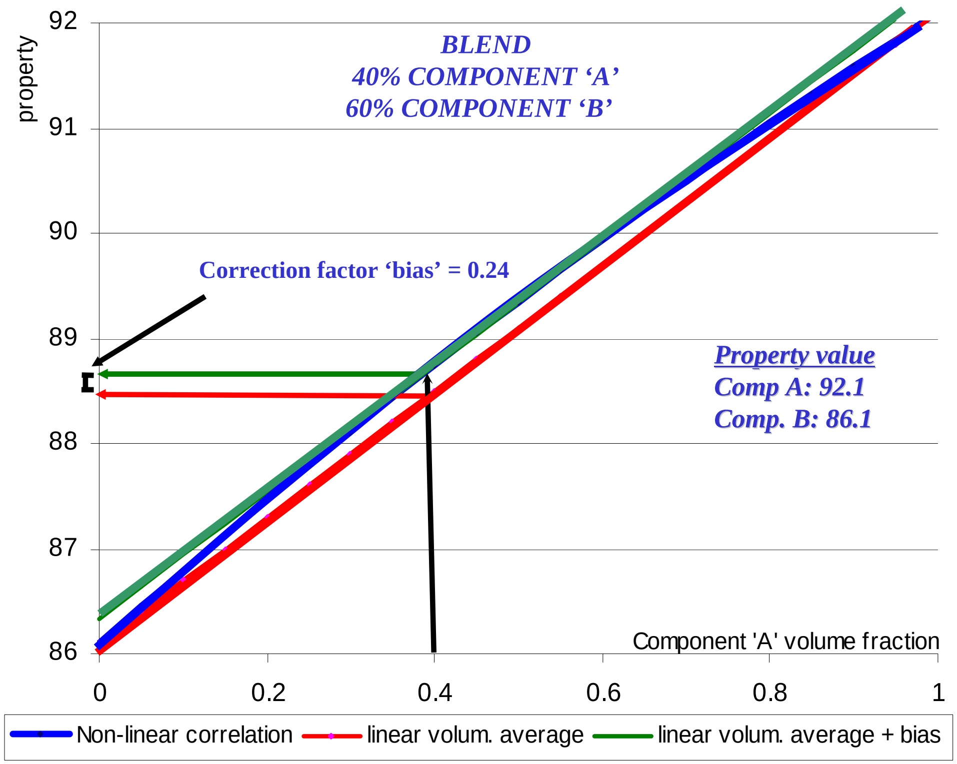
Volumetric product property correlation *(Linear)*

$$pr^{\min}_{p,k} F^P_{p,t} \le \sum_i pr_{i,k} F^I_{i,p,t} \le pr^{\max}_{p,k} F^P_{p,t} \qquad \forall p,k,t$$

Linear approximation for non-linear product properties

$$pr^{\min}_{p,k} F^P_{p,t} + bias_{p,k,t} F^P_{p,t} \le \sum_i pr_{i,k} F^I_{i,p,t} \le$$

$$pr^{\max}_{p,k} F^P_{p,t} + bias_{p,k,t} F^P_{p,t} \qquad \forall p,k,t$$



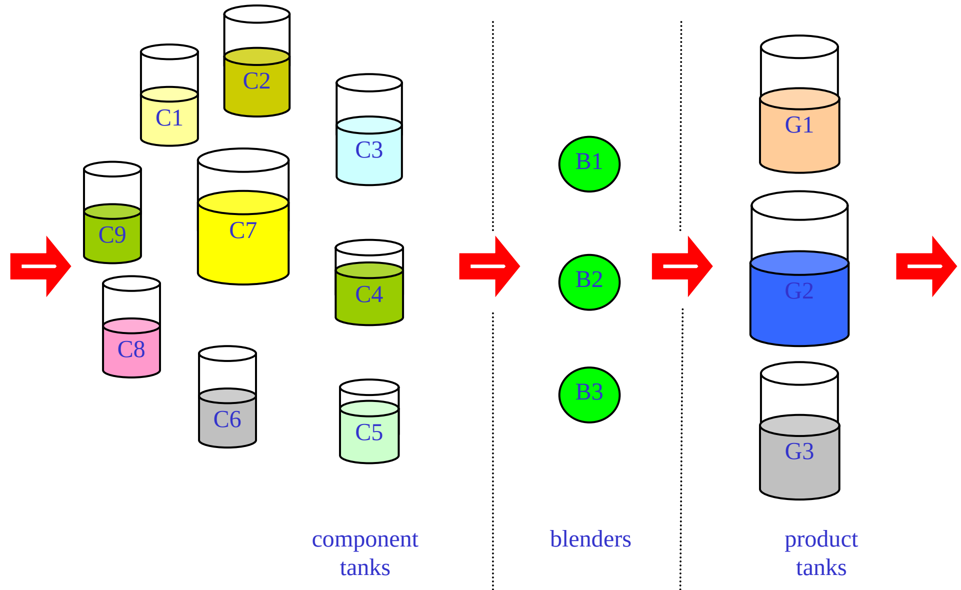
Example

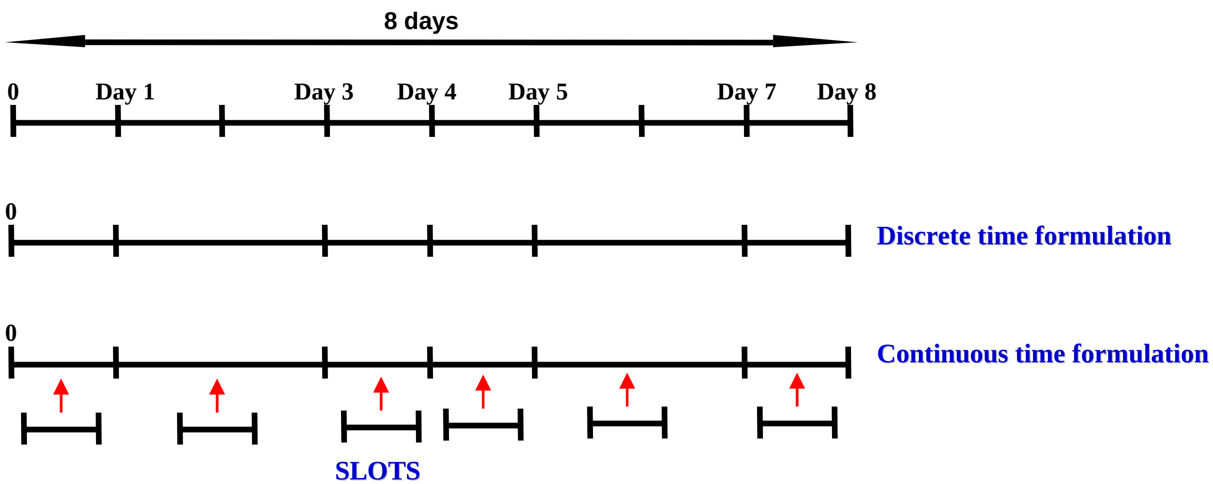
8-DAY TIME HORIZON

12 STORAGE TANKS

3 BLEND HEADERS

6 DUE DATES





- Motivation
- Modeling issues
- Problem statement
- Proposed optimization approach
- Product property prediction
- Integrated model
  - Discrete MILP formulation
  - Continuous MILP formulation
- Treatment of infeasible solutions
- **Numerical results**

Example: Blending and scheduling

Objective: Make scheduling and blending decisions that maximize profit

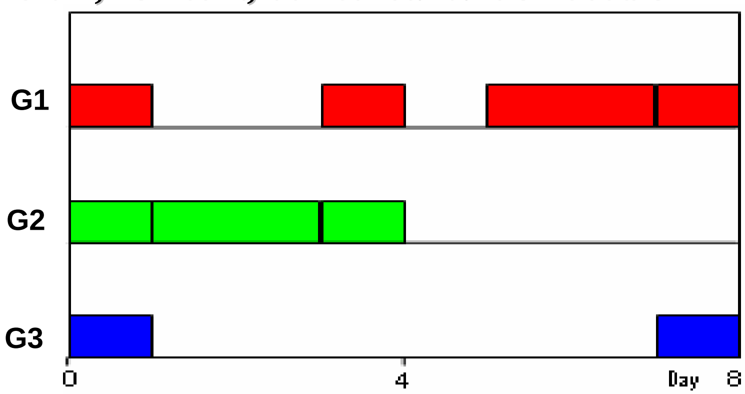
- Min/max production for each time interval
- Demands at specific due dates
- Product specifications
- Operating conditions

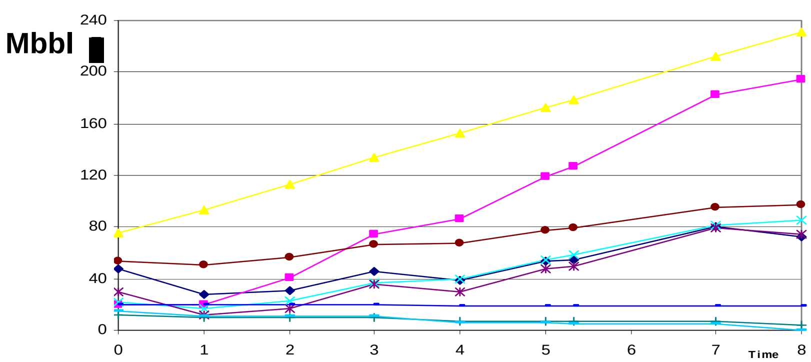
Profit: M$1,611.21

Operates blenders at full capacity for 2.67 days less than discrete time

Discrete time formulation

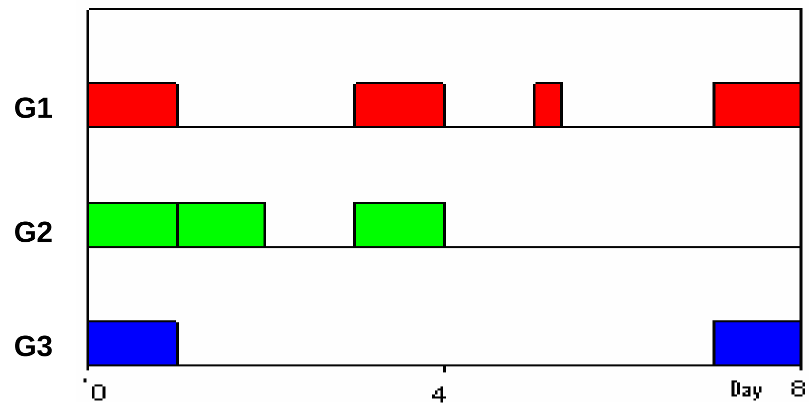
9 0-1, 757 cont, 667 const. 0.26CPUsec/CPLEX

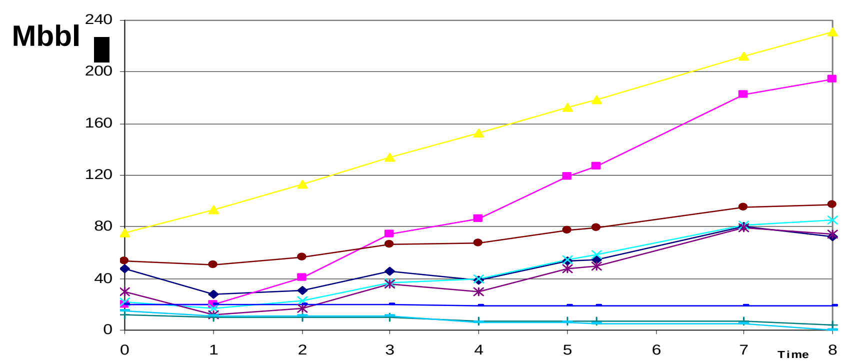




Continuous time formulation

9 0-1, 841 cont, 832 constr. 0.26 CPUsec/CPLEX





EVOLUTION OF COMPONENT STOCKS

Simultaneous Cyclic Scheduling and Control of a Multiproduct CSTR Reactor

Flores, Grossmann (2005)

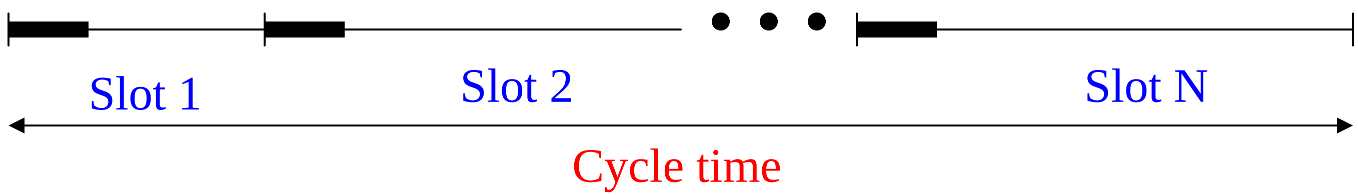
Given is a CSTR reactor
- N products
- Lower bounds demand rates
- Dynamic model for reactions

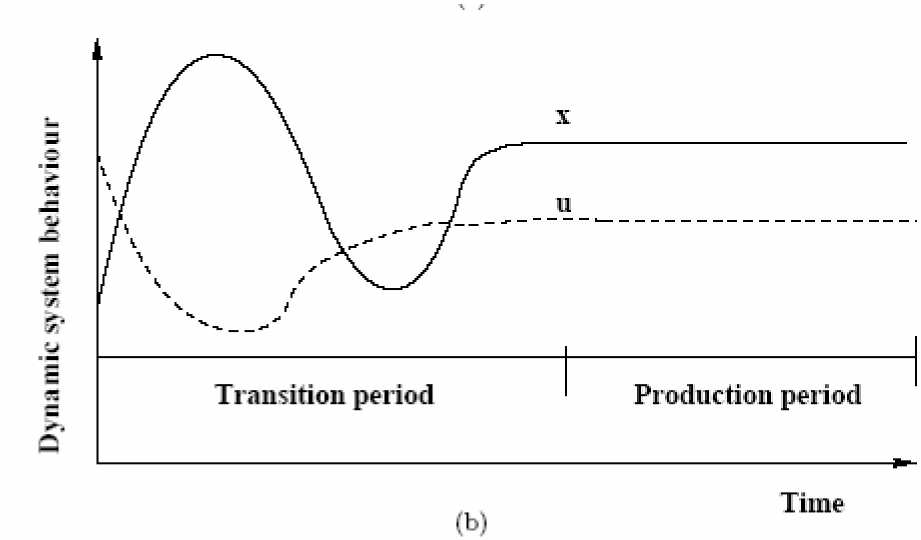
Determine cyclic schedule
- Cycle time
- Sequence
- Amounts to produce
- Lenghts transitions and their dynamic profile

Objective: Maximize total profit

Basic ideas MIDO model

$$y_{il} = \begin{cases} 1 & product\ i\ assigned\ \ slot\ \ l \\ 0 & otherwise \end{cases}$$





Requires guessing transition times

Use orthogonal collocation for converting dynamic eqtns into algebraic eqtns.

Discretized DEA solved as MINLP (DICOPT)

MIDO Optimization model

$$\max \quad \Bigg\{ \sum_{i=1}^{N_p} \frac{C_i^p W_i}{T_c} - \sum_{i=1}^{N_p} \frac{C_i^s (G_i - W_i)}{2\Theta_i T_c} - \sum_{k=1}^{N_s} \sum_{f=1}^{N_{fe}} h_{fk} \sum_{c=1}^{N_{cp}} \frac{C^r t_{fck} \Omega_{c,N_{cp}}}{T_c} \big((x_{fck}^1 - \bar{x}_k^1)^2 + \ldots + (x_{fck}^n - \bar{x}_k^n)^2 + (u_{fck}^1 - \bar{u}_k^1)^2 + \ldots + (u_{fck}^m - \bar{u}_k^m)^2 \big) \Bigg\} \qquad (1)$$

s.t. Scheduling constraints:

- Product assignments
- Amounts manufactured
- Processing times
- Transition constraints
- Timing relations

Dynamic and control optimization

- Dynamic mathematical model discretization
- Continuity constraint between finite elements
- Model behavior at each collocation point

MIDO:

$$\max \left\{ \sum_{i=1}^{N_p} \frac{C_i^p W_i}{T_c} - \sum_{i=1}^{N_p} \frac{C_i^s (G_i - W_i)}{2\Theta_i T_c} - \sum_{k=1}^{N_s} \sum_{f=1}^{N_{fe}} h_{fk} \sum_{c=1}^{N_{cp}} \frac{C^r t_{fck} \Omega_{c,N_{cp}}}{T_c} \left((x_{fck}^1 - \bar{x}_k^1)^2 + \ldots + (x_{fck}^n - \bar{x}_k^n)^2 + (u_{fck}^1 - \bar{u}_k^1)^2 + \ldots + (u_{fck}^m - \bar{u}_k^m)^2 \right) \right\}$$

Scheduling:

$$\sum_{k=1}^{N_s} y_{ik} = 1, \ \forall i$$
$$\sum_{i=1}^{N_p} y_{ik} = 1, \ \forall k$$
$$y'_{ik} = y_{i,k-1}, \ \forall i, k \neq 1$$
$$y'_{i,1} = y_{i,N_s}, \ \forall i$$

$$W_i \geqslant D_i T_c, \ \forall i$$
$$W_i = G_i \Theta_i, \ \forall i$$
$$G_i = F^o(1 - X_i), \ \forall i$$
$$\theta_{ik} \leqslant \theta^{max} y_{ik}, \ \forall i, k$$
$$\Theta_i = \sum_{k=1}^{N_s} \theta_{ik}, \ \forall i$$
$$p_k = \sum_{i=1}^{N_p} \theta_{ik}, \ \forall k$$

$$z_{ipk} \geqslant y'_{pk} + y_{ik} - 1, \ \forall i, p, k$$

$$\theta_k^t = \sum_{i=1}^{N_p} \sum_{p=1}^{N_p} t_{pi}^t z_{ipk}, \ \forall k$$
$$t_1^s = 0$$
$$t_k^e = t_k^s + p_k + \sum_{i=1}^{N_p} \sum_{p=1}^{N_p} t_{pi}^t z_{ipk}, \ \forall k$$
$$t_k^s = t_{k-1}^e, \ \forall k \neq 1$$
$$t_k^e \leqslant T_c, \ \forall k$$
$$t_{fck} = (f-1)\frac{\theta_k^t}{N_{fe}} + \frac{\theta_k^t}{N_{fe}} \gamma_c, \ \forall f, c, k$$

Dynamics:

$$x_{fck}^n = x_{o,fk}^n + \theta_k^t h_{fk} \sum_{l=1}^{N_{cp}} \Omega_{lc} \dot{x}_{flk}^n, \ \forall n, f, c, k$$
$$x_{o,fk}^n = x_{o,f-1,k}^n + \theta_k^t h_{f-1,k} \sum_{l=1}^{N_{cp}} \Omega_{l,N_{cp}} \dot{x}_{f-1,l,k}^n, \ \forall n, f \geqslant 2, k$$

$$\dot{x}_{fck}^n = f^n(x_{fck}^1, \ldots, x_{fck}^n, u_{fck}^1, \ldots u_{fck}^m), \ \forall n, f, c, k$$

$$x_{in,k}^n = \sum_{i=1}^{N_p} x_{ss,i}^n y_{i,k}, \ \forall n, k$$
$$\bar{x}_k^n = \sum_{i=1}^{N_p} x_{ss,i}^n y_{i,k+1}, \ \forall n, k \neq N_s$$
$$\bar{x}_k^n = \sum_{i=1}^{N_p} x_{ss,i}^n y_{i,1}, \ \forall n, k = N_s$$
$$u_{in,k}^m = \sum_{i=1}^{N_p} u_{ss,i}^m y_{i,k}, \ \forall m, k$$
$$\bar{u}_k^m = \sum_{i=1}^{N_p} u_{ss,i}^m y_{i,k+1}, \ \forall m, k \neq N_s - 1$$
$$\bar{u}_k^m = \sum_{i=1}^{N_p} u_{ss,i}^m y_{i,1}, \ \forall m, k = N_s$$
$$x_{N_{fe},N_{cp},k}^n = \bar{x}_k^n, \ \forall n, k$$
$$u_{1,1,k}^m = u_{in,k}^m, \ \forall m, k$$
$$u_{N_{fe},N_{cp},k}^m = \bar{u}_{in,k}^m, \ \forall m, k$$
$$x_{o,1,k}^n = x_{in,k}^n, \ \forall n, k$$

$$x_{min}^n \leqslant x_{fck}^n \leqslant x_{max}^n, \ \forall n, f, c, k$$
$$u_{min}^m \leqslant u_{fck}^m \leqslant u_{max}^m, \ \forall m, f, c, k$$

Example: 5 products

Third order kinetics: $R \xrightarrow{k} P, \qquad -R_R = kc_R^3$

$$\frac{dc_R}{dt} = \frac{Q}{V}(c_o - c_R) + R_R$$

Data

Product	Q [lt/hr]	c_o [mol/lt]	Demand rate [Kg/h]	Product cost [$/kg]	Inventory cost [$]
A	10	0.0967	3	200	1
B	100	0.2	8	150	1.5
C	400	0.3032	10	130	1.8
D	1000	0.393	10	125	2
E	2500	0.5	10	120	1.7

Results

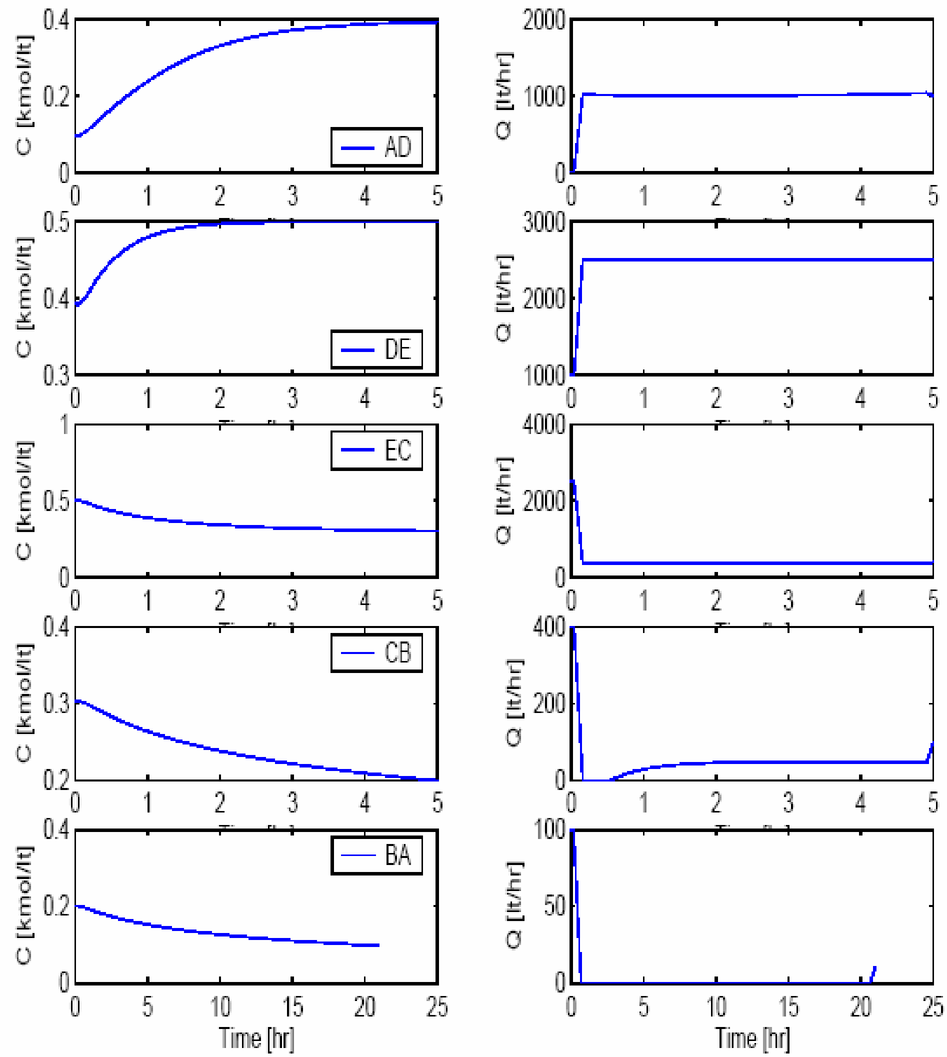
Slot	Product	Process time [h]	Production rate [Kg/h]	w [Kg]	Transition Time [h]	T start [h]	T end [h]
1	A	41.5	9.033	374.31	5	0	46.4
2	D	2.06	607	1249.4	5	46.4	53.6
3	E	23.4	1250	29270.4	5	53.6	82
4	C	4.48	278.72	1249.4	5	82	91.5
5	B	12.48	80	999.5	21	91.5	125

Optimal sequence

A→E→C→D→B→

Cycle time = 124.8 h

Profit = $7889/h



Conclusions

1. **Optimization-based scheduling very active area of research**

   Major modeling tool: mixed-integer optimization

2. **Great diversity in applications makes single solutions still elusive**

   Tailored solutions are still most effective,
   but MOVING target !

3. **Real–world industrial problems require combination of approaches and aggregation/decomposition**