CAPD Center for Advanced Process Decision-making

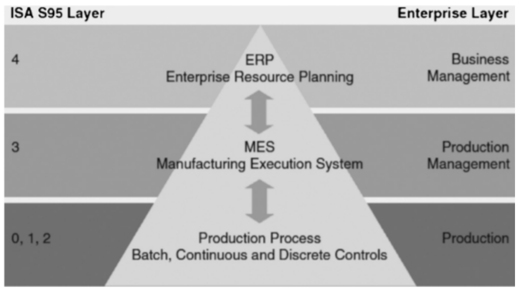
ISA-95 Friendly RTN Models for Industrial Production Scheduling

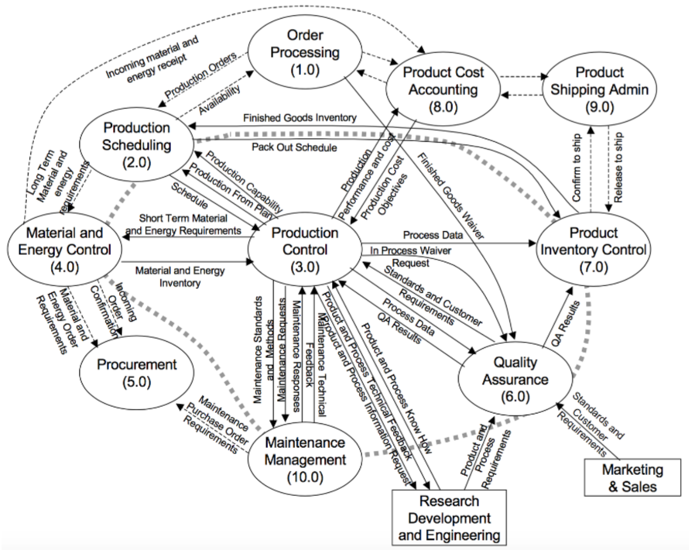
Pedro M. Castro
Ignacio E. Grossmann
Iiro Harjunkoski

IF INVESTIGADOR FCT
Ciências ULisboa
Carnegie Mellon
ABB

Motivation

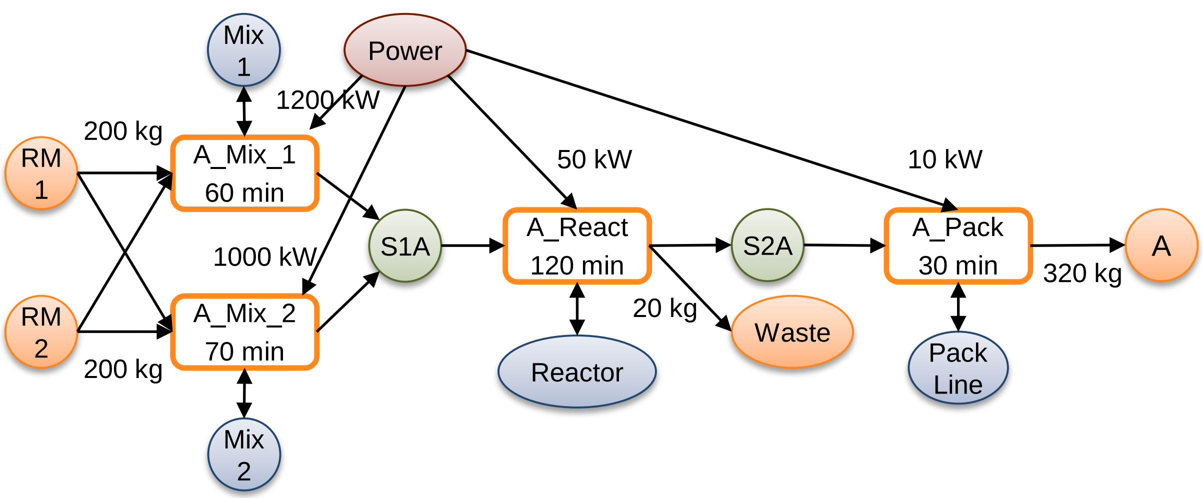
- EWO aims to simultaneously account for KPI across multiple business units
  - Integration of supply chain management, production control, planning & scheduling
- Need to efficiently transfer data and information between different systems
  - Focus on production management system and scheduling solution
- ISA-95 standard can act as data-exchange platform
  - Goal: Adopt standard in a way that fulfills requirements of most common scheduling problems





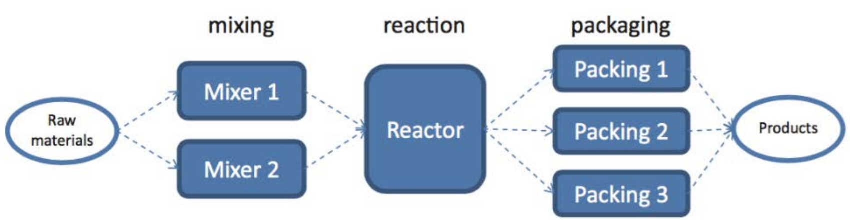
Generic scheduling approach

- Resource-Task Network
  - Process representation
    - Production recipe & network topology
  - Mathematical formulations
    - Discrete- & continuous-time
- Key success drivers
  - Modeling paradigm easily understood by business stakeholders
  - Flexible approach, easily modified when new information becomes available
- Challenge
  - Develop RTN models complying to the ISA-95 standard



ISA-95 data implementation

- Variety of complex XML files
  - Process segment information
  - Equipment
  - Material
  - Personnel
  - Operations capability
  - Operations definition
  - Operations schedule



Example ProcessSegmentInformation.xml UNREGISTERED

Example EquipmentInformation.xml UNREGISTERED

```xml
<Description>Small test example to illustrate ISA-95</
Description>
<PublishedDate>2014-06-16T00:00:00</PublishedDate>
<Equipment>
  <ID>Mixer1</ID>
  <Description>Example mixer</Description>
  <EquipmentProperty>
    <ID>SetupTime</ID>
    <Value>
      <ValueString>15</ValueString>
      <DataType>double</DataType>
      <UnitOfMeasure>min</UnitOfMeasure>
      <Key />
    </Value>
  </EquipmentProperty>
  <EquipmentClassID>Mixer</EquipmentClassID>
</Equipment>
<Equipment>
  <ID>Mixer2</ID>
  <Description>Example mixer</Description>
  <EquipmentProperty> ...
  </EquipmentProperty>
  <EquipmentClassID>Mixer</EquipmentClassID>
</Equipment>
<Equipment>
  <ID>Reactor1</ID>
  <Description>Process reactor</Description>
  <EquipmentProperty> ...
  </EquipmentProperty>
  <EquipmentClassID>Reactor</EquipmentClassID>
</Equipment>
<Equipment>
  <ID>Packing1</ID>
  <Description>Packaging line 1</Description>
  <EquipmentProperty> ...
  </EquipmentProperty>
  <EquipmentClassID>Pack</EquipmentClassID>
</Equipment>
```

Mapping data with RTN process model

- Sets definition

Set	Description	Elements	B2MML file (XML implementation ISA-95)
MT	Materials	Power, RM1, RM2, Waste, A, B, C	MaterialInformation
EQ	Equipment	Mixer1, Mixer2, Reactor1, Packing1, Packing2, Packing3	EquipmentInformation
ID	Class ID	Mixer, Reactor, Pack	EquipmentInformation
OP	Operations	ProdA, ProdB, ProdC	OperationsDefinitionInformation
SG	Segments	Mixing, Reaction, Packaging	ProcessSegmentInformation
P1(SG,SG)	Immediate precedence	Mixing.Reaction, Reaction.Packaging	ProcessSegmentInformation
P2(SG,ID)	Segment belongs to class	Mixing.Mixer, Reaction.Reactor, Packaging.Pack	ProcessSegmentInformation
P3(ID,EQ)	ID belonging to equipment	Mixer.Mixer1, Mixer.Mixer2, Reactor.Reactor1, Pack.Packing1, Pack.Packing2	EquipmentInformation

- Parameters from production recipe

MatReq1(OP,SG,MT)	RM1	RM2	Waste	A	B	C
ProdA.Mixing	-200	-200				
ProdA.Reaction			20			
ProdA.Packaging				320		
ProdB.Mixing	-500					
ProdB.Reaction			50			
ProdB.Packaging					400	
ProdC.Mixing	-150	-300				
ProdC.Reaction			15			
ProdC.Packaging						380

MatReq2 (OP,EQ,MT)	Mixer1. Power	Mixer2. Power	Reactor1 .Power	Packing1 .Power	Packing2 .Power	Packing3 .Power
ProdA	-1200	-1000	-50	-20	-10	
ProdB	-900		-120	-12	-12	-10
ProdC	-800	-750	-50	-10	-10	-10

OperationsDefinitionInformation.xml

PTime (OP,EQ)	Mixer1	Mixer2	Reactor1	Packing1	Packing2	Packing3
ProdA	60	70	120	30	30	
ProdB	110		240	45	45	60
ProdC	80	80	150	40	40	40

RTN discrete-time model (GAMS)

- Few lines of code generate:
  - Resources
    - Time availability parameters
    - Subsets for domain constraints
  - Tasks
    - Structural parameters
      - Processing times
      - Timing of resource consumption/production
- Actual model quite simple

```
#Resources generation
#Equipment resources
loop(EQ,
          loop(R$(ord(R) EQ CURR),
          MATCHEQ(R,EQ)=yes;
          REQ(R)=yes;
          ER0(R)=1;
          );
CURR=CURR+1;
);
```

```
#Tasks and structural parameters generation
loop(OP,
loop(EQ$(PTIME(OP,EQ) GT 0),
loop(I$(ord(I) EQ CURI),
MATCHI(I,OP,EQ)=yes;
DUR(I)=PTIME(OP,EQ);
TAU(I)=ceil(DUR(I)/DELTA);
U(R,I,THETA)$(MATCHEQ(R,EQ) and ord(THETA) EQ 1)=-1;
U(R,I,THETA)$(MATCHEQ(R,EQ) and ord(THETA) EQ TAU(I)+1)=1;
          loop(MT,
          U(R,I,THETA)$(MATCHMT(R,MT) and REL(R) and ord(THETA) EQ 1)=MATREQ2(OP,EQ,MT);
          U(R,I,THETA)$(MATCHMT(R,MT) and RRM(R) and ord(THETA) EQ 1)=SUM(SG$(OPSGEQ(OP,SG,EQ)),MATREQ1(OP,SG,MT));
          U(R,I,THETA)$(MATCHMT(R,MT) and (RWS(R) or RFP(R)) and ord(THETA) EQ TAU(I)+1)=SUM(SG$(OPSGEQ(OP,SG,EQ)),MATREQ1(OP,SG,MT));
          );
          loop(SG$(OPSGEQ(OP,SG,EQ)),
```

```
OBJECT..  TAMOUNT=e=SUM((R,T)$(RFP(R) and FIM(T)),ER(R,T));
BALREC(R,T)$(ACTR(R) and ACTT(T))..     ER(R,T)=e=ER0(R)$(INICIO(T))+ER0V(R)$((REL(R) or RRM(R)) and INICIO(T))+ER(R,T-1)+SUM((I,THETA)$(ACTI(I) and ord(THETA) LE TAU(I)+1),U(R,I,THETA)*N(I,T-(ord(THETA)-1)));
EXCBAT(OP,SG)..                         SUM((I,EQ,T)$(MATCHI(I,OP,EQ) and OPSGEQ(OP,SG,EQ) and ACTT(T)),N(I,T))=e=NBATCHS(OP);
CNSBAT(R)$(RRM(R))..                    ER0V(R)=e=-SUM((OP,MT)$(MATCHMT(R,MT)),MATREQ1(OP,'Mixing',MT)*NBATCHS(OP));
BALGLO(R,T)$(REL(R) and FIM(T))..       ER(R,T)=e=0;
PRDBAT(R,T)$(RFP(R) and FIM(T))..       ER(R,T)=e=SUM((OP,MT)$(MATCHMT(R,MT)),MATREQ1(OP,'Packaging',MT)*NBATCHS(OP));
```

Computational studies

- How far can we go in terms of problem size?
  - Vary # batches per product (1:1:1 A,B,C)
  - Short-term scheduling models, minimizing makespan
    - Discrete-time (DT), 5 min slots
      - More batches $\Rightarrow$ multiple executions of the same task
    - Continuous-time with multiple time grids (CT)
      - More batches $\Rightarrow$ more variables and constraints (batch index)

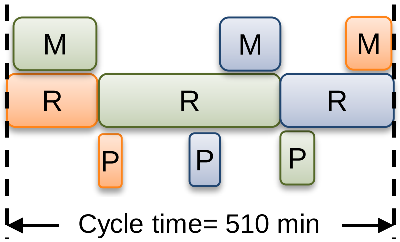
# batches	Makespan (min)	DT CPUs	CT CPUs
1	610	3.36	0.41
2	1110	1.52	2.31
3	1620	2.34	65.1
5	2640	12.8	>1600 (gap=71.3%)
10	5190	33.2	-
20	10290	397	-

- Discrete-time much better approach
  - But we want 100+ batches, sometimes 500!

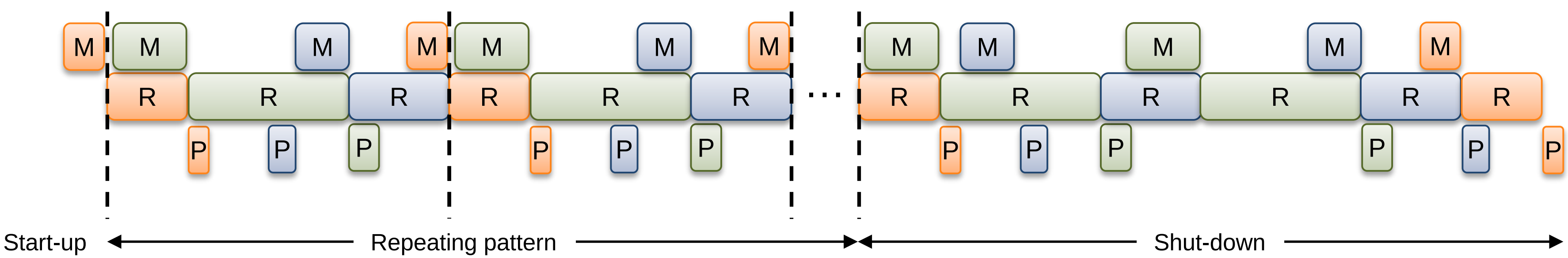
How about relying on periodic scheduling?

Carnegie Mellon ABB

- Step 1: Find optimal cyclic pattern
  - Given batch proportion between different recipes (ex. 1:1:1)



- Step 2: Solve short-term scheduling for a given number of repetitions
  - Find optimal start-up and shutdown phases
    - Almost all binary variables are fixed
    - Timing of events still free to vary



Results for periodic-based scheduling

Carnegie Mellon ABB

- Advantages
  - Exact same solution as short-term discrete-time model
    - Continuous-time is more accurate
  - Better computational performance
- Disadvantages
  - Periodic scheduling is for stable production of a few products
  - Underlying model works just for a specific type of plant topology

# batches	Makespan (min)	Total CPUs
5	2640	2.2
10	5190	2.8
20	10290	3.6
30	15390	4.3
50	25590	7.9
100	51090	36.2

- Why not periodic-based scheduling with discrete-time?
  - Need narrow time window for task execution
    - Difficult to generate because start-up phase determined by optimization

# batches	Binary variables	Total variables	Equations
5	53	751	1107
10	53	2851	4017
20	53	11101	15237
30	53	24751	33657
50	53	68251	92097
100	53	271501	364197

Conclusions

- Guidelines for generating RTN-based scheduling model from ISA-95 data
- Existing generic models simply not good enough to allow tackling problems of industrial relevance
  – Need for optimization based decomposition strategy that is competitive with simple heuristics
- Concept tested by repeating pattern from periodic scheduling
  – Continuous-time approach allows for more freedom
    - Using just a few binary variables
- Future work
  – Develop a generic and efficient matheuristic
    - Integration of (meta)heuristics and mathematical programming
  – Test in a set of real-life problems