Optimization of Crude-Oil Blending Operations

Sylvain Mouret   Ignacio E. Grossmann   Pierre Pestiaux

Introduction

Goal

Optimize the schedule of operations for the crude-oil problem using a MINLP scheduling model

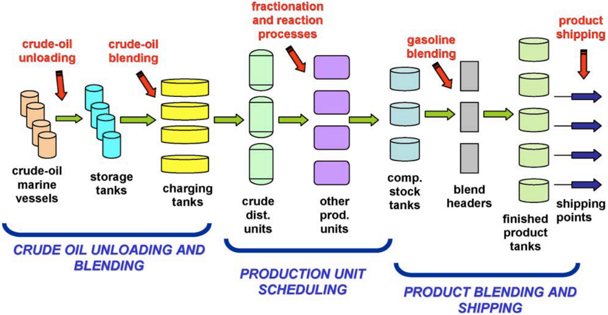
Tools

- MINLP: Mixed Integer NonLinear Programming
- MILP: Mixed Integer Linear Programming
- NLP: NonLinear Programming

1. Problem statement
   - Oil refinery
   - Crude-oil blending scheduling
   - Scheduling formulations
2. Proposed approach
   - Basic idea
   - MINLP model
   - Search procedure
3. Results and comparisons
   - Computational results
4. Conclusion

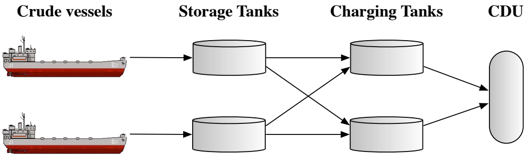
A typical oil refinery

- Refining crude-oil into useful petroleum products:
  LPG, gasoline, diesel fuel, kerosene, heating oil, asphalt base
- 3 phases:
  - Crude-oil unloading and blending
  - Fractionation and reaction processes
  - Product blending and shipping



Crude-oil operations scheduling problem

- Scheduling horizon $[0, H]$
- 4 types of resources:
  - Crude-oil marine vessels
  - Storage tanks
  - Charging tanks
  - Crude Distillation Units (CDUs)
- 3 types of operations:
  - Unloading: Vessel unloading to storage tanks
  - Transfer: Transfer from storage tanks to charging tanks
  - Distillation: Distillation of charging tanks



Problem definition

Given

- Refinery configuration
- Logistics constraints
- Initial tank inventory and composition
- Vessel arrival time, inventory level and composition
- Distillation specifications and demands (planning decisions)

Determine

- Required operations
- Timing decisions
- Transfer volumes

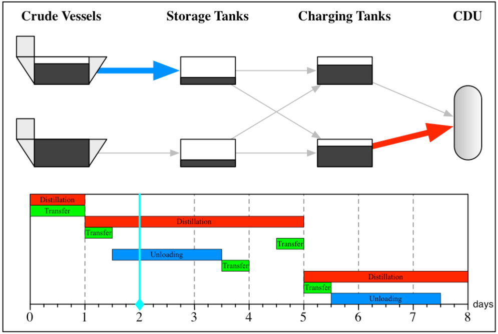
Minimize

- Costs of distilled crude-oil mixtures

Example of crude-oil operations schedule

Common logistics constraints

- Only one docking station available for vessel unloadings
- No simultaneous inlet and outlet operations on tanks
- Continuous distillation



Refinery operations

Gantt chart

Scheduling formulations

- Fixed Time Grid
  - Kondili et al. (1993), Shah et al. (1993), Pantelides (1994)
  - Crude-oil scheduling: Shah (1996), Lee et al. (1996)
- Variable Time Grid
  - Zhang and Sargent (1996), Schilling and Pantelides (1996), Mockus and Reklaitis (1997)
  - Crude-oil scheduling: Moro and Pinto (2004)
- Single-Operation Time-Slots (event-based formulation)
  - Ierapetritou and Floudas (1998a, 1998b)
  - Crude-oil scheduling: Jia et al. (2003)
- *Multi-Operation Time-Slots*

Basic Idea

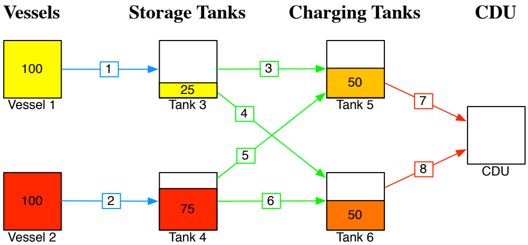
Basic steps

1. **Postulate** the number of time-slots that are needed
2. **Define** an ordered set of time-slots
3. **Define** the set of all transfer operations
4. **Assign** exactly one operation to each time-slot and **determine** the timing and volume decisions

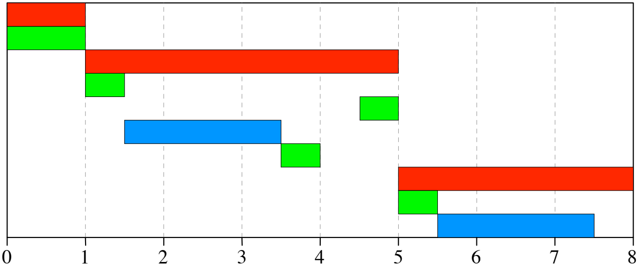
MINLP model

- Binary variables: assignment variables
- Continuous variables: time, volume and level variables

An example of time-slots assignment



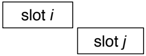
Task	Operation	Volume
1	Distillation 7	5
2	Transfer 6	50
3	Distillation 8	100
4	Transfer 3	25
5	Transfer 5	19.5
6	Unloading 1	100
7	Transfer 3	5.5
8	Distillation 7	95
9	Transfer 6	5.5
10	Unloading 2	100

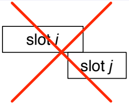


Main sets and variables

- Ordered set of unspecified time-slots $i \in \{1, .., n\}$
  - Start time: $TS^i \in [0, H]$
  - End time: $TE^i \in [0, H]$
- Set of operations $v \in \{1, .., N_v\}$
- Assignment variables $Z_v^i \in \{0, 1\}$
  - Operation v is assigned to time-slot i iff $Z_v^i = 1$
  - Exactly one operation for each time-slot: $\sum_v Z_v^i = 1$

Non-overlapping constraints





- For each ordered pair of time-slots $i \prec j$ and for each pair of non-overlapping operations v and w:

$$\begin{cases} Z_v^i = Z_w^j = 1 & \Rightarrow NO_{ij} = 1 \\ NO_{ij} = 1 & \Rightarrow TE^i \leq TS^j \end{cases} \Rightarrow \begin{cases} NO_{ij} & \geq Z_v^i + Z_w^j - 1 \\ TE^i & \leq TS^j + M \cdot (1 - NO_{ij}) \end{cases}$$

- For example,
  - Vessel unoadings 1 and 2:

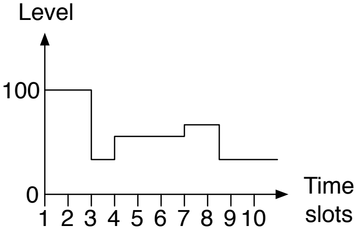
$$NO_{ij} \geq Z_1^i + Z_2^j - 1$$

  - Distillation transfers 7 and 8 (use same CDU):

$$NO_{ij} \geq Z_7^i + Z_8^j - 1$$

Tank inventory and composition constraints

- Tanks $t \in \{1, .., N_t\}$
- Crude-oil types $c \in \{1, .., N_c\}$
- Volume variables $V_v^i, V_{vc}^i \geq 0$
- Level variables $L_t^i, L_{tc}^i \geq 0$
- Tank inventory constraints:
  - $L_t^{i+1} = L_t^i + \sum_{v \in IN(t)} V_v^i - \sum_{v \in OUT(t)} V_v^i$
  - $\underline{L_t} \leq L_t^i \leq \overline{L_t}$
- Tank composition constraints:
  - $Z_v^i = 1 \Rightarrow \frac{V_{vc}^i}{V_v^i} = \frac{L_{tc}^i}{L_t^i}$ (nonlinear)
  - $\left.\begin{array}{l} L_t^i = \sum_c L_{tc}^i \\ V_v^i = \sum_c V_{vc}^i \end{array}\right\}$ linear relaxation



Other constraints

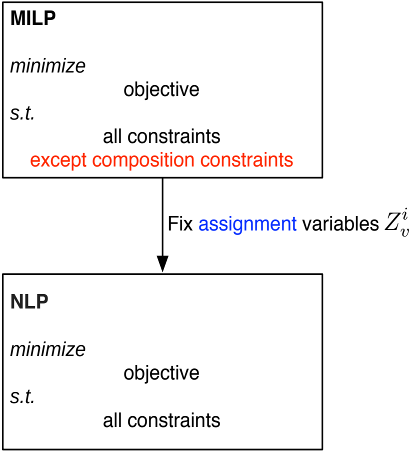
- Continuous distillation
- Flowrate limitations: $\underline{FR}_v \cdot (TE^i - TS^i) \leq V_v^i \leq \overline{FR}_v \cdot (TE^i - TS^i)$
- Scheduling constraints:
  - Vessels availability time window
  - Precedence constraints
- Crude-oil mixture specification and demand
- Objective function:

$$OBJ = \sum_{t \in CDUs} \sum_{c} COST_c \cdot L_{tc}^n$$

MILP-NLP decomposition

Decomposition steps

1. **Master problem**: find optimal solution for the MILP relaxation
2. Solution may not satisfy the nonlinear composition constraints
3. Fix assignment variables
4. **Slave problem**: find optimal solution for the resulting NLP (with nonlinear composition constraints)



Sensitivity to the number of time-slots

- MILP-NLP decomposition tested on case-study with 5 to 13 slots
- Size of the MINLP with 13 time-slots:
  1575 binary variables, 1419 continuous variables, 4429 constraints
- Feasible schedule obtained with 9 time-slots
- Optimal schedule obtained with 10 time-slots

Nb of slots	LP	MILP	CPU	Nb of Nodes	NLP	Gap
5	Infeas					
6	117.5	Infeas	0s	1		
7	83.7	Infeas	1s	29		
8	82.5	Infeas	2s	115		
9	82.5	120.28	3s	147	121.25	0.8%
10	82.5	120.25	4s	116	120.25	0%
11	82.5	120.25	7s	97	120.25	0%
12	82.5	120.25	6s	75	120.25	0%
13	82.5	120.25	2s	17	120.25	0%

Comparison with other algorithms

- Number of time-slots: 13
- Algorithms used:
  - MILP-NLP decomposition: Xpress (MILP), CONOPT (NLP)
  - MINLP solvers: DICOPT, SBB, AlphaECP, BARON (global optimum)

Algorithm	Solution	CPU time
MINLP-NLP	120.25	2s
DICOPT	120.25	18s
SBB	120.25	81s
AlphaECP	120.25	468s
BARON	120.25	219s

$\Rightarrow$ Order of magnitude reduction for CPU time.

Larger instances

- Approach tested on the 4 problems from Lee et al. (1996)
- Problems solved with the maximum number of time-slots needed
- Problem 3 shows a gap of 4.9% between the MILP and NLP solutions

Example	Vessels/Storage/Charging/CDUs	Slots	MILP	NLP	Gap	CPU
1	2 / 2 / 2 / 1	13	120.25	120.25	0%	2s
2	3 / 3 / 3 / 2	21	198.83	198.83	0%	104s
3	3 / 3 / 3 / 2	21	59.60	62.50	4.9%	73s
4	3 / 6 / 4 / 3	26	107.47	107.47	0%	506s

Conclusion and future work

Conclusion

- New MINLP formulation for the crude-oil operations problem
- Handles logistics constraints and minimization crude-oil costs
- MILP-NLP decomposition algorithm compares well to MINLP solvers

Future work

- Hybrid optimization: Contraint Programming as a symmetry-breaking branching tool
- Enhance the MILP-NLP decomposition
- Take into account stochastic parameters (vessels arrival time)
- Practical case-study

Discrete formulation

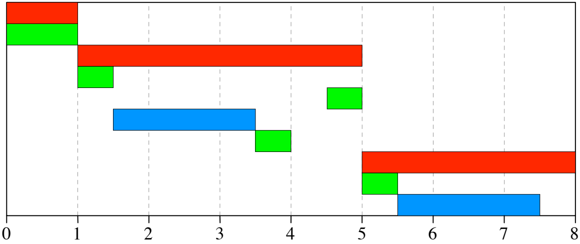
- Discretization of the time horizon into n fixed-length adjacent time-slots
- Identical MILP-NLP decomposition
- Example 4 from Lee et al. (1996), optimal solution: 107.47

Nb of slots	LP	MILP	CPU	Nb of nodes	NLP	Gap
5	107.87	Infeas	0s	1		
10	107.49	107.80	385s	74627	107.80	0%
15	107.47	107.77	29s	1853	Local Infeas	
20	107.45	107.68	*3600s	*199800	107.68	0%
25	107.45	107.63	*3600s	*41300	107.63	0%

Symmetry breaking

- Multiple operation assignment may lead to the same schedule
- For instance, exchanging operations assigned to slots 1 and 2 in the following gantt chart leads to the same solution

Task	Operation	Volume
1	Distillation 7	5
2	Transfer 6	50
3	Distillation 8	100
4	Transfer 3	25
5	Transfer 5	19.5
6	Unloading 1	100
7	Transfer 3	5.5
8	Distillation 7	95
9	Transfer 6	5.5
10	Unloading 2	100

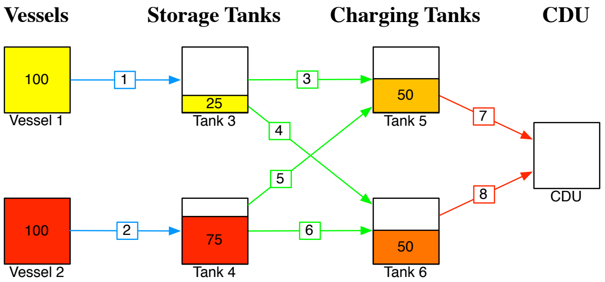


Regular language derivation

- The possible sequences of operations are represented by a regular language (Regular constraint by Côté et al., 2007)
- Example 1 has 2 refinery states: distillation 7 or 8
- During distillation state 7:

$$L_7 = 7(\epsilon + 4)(\epsilon + 6)(\epsilon + 1 + 14)(\epsilon + 2 + 26)$$

- Overall: $L = (\epsilon + L_7)(L_8 L_7)^*(\epsilon + L_8)$



Automaton representation

- The regular language $L_7 = 7(\epsilon + 4)(\epsilon + 6)(\epsilon + 1 + 14)(\epsilon + 2 + 26)$ can be recognized by the following Deterministic Finite Automaton (DFA)
- Each node is equivalent to a state
- Each arc corresponds to an operation assigned to a task