# Contents

# List of Figures



# List of Tables

# 1 Introduction

Scheduling is a decision-making process that plays an important role in most manufacturing and service industries. Scheduling problems arise in almost any type of industrial production facilities where given tasks need to be processes using specified resources. In a chemical process, production must be planned such that equipment, material and utilities are available at the manufacturing facility when they are needed to realize the production tasks. Production scheduling comprises the activity of planning in detail the production of a product or products in a given production facility. It boils down to the following main decisions (Harjunkoski et al., 2014):

- What production tasks to execute?
- Where to process the production tasks?
- In which sequence to produce?
- When to execute the production tasks?

For batch processes, short-term scheduling deals with the allocation of a set of limited resources over time to manufacture one or more products following a batch recipe (Méndez et al., 2006). There has been significant development of optimization approaches to scheduling over the last two decades. The first mathematical programming approach the scheduling of multi-purpose, multi-product batch plants was proposed by Kondili et al. (1993). This approach introduces the state task network (STN) representation where the process is described as a bipartite graph consisting of states and tasks.

## 1.1 Parameter uncertainty in process scheduling

In realistic scenarios, many of the parameters associated with scheduling are not known exactly. Parameters such as processing time, yields, prices, etc. can vary with respect to time and are subject to unexpected deviations. Robust optimization is an approach that has been suggested to mitigate these uncertainties while designing a schedule.

Robust optimization seeks to generate a solution that is immune to uncertainty by ensuring that it remains feasible for all possible realizations of the uncertain parameters from within a set chosen *a priori* by the modeler.

This work supports two frameworks to handle uncertainty of parameters:

- **Static Robust Optimization:** The first application of robust optimization in process scheduling was by Lin et al. (2004). This work, which utilized box uncertainty sets, was later extended by Janak et al. (2007) to consider uncertainty sets derived from probabilistic information. Li and Ierapetritou (2008) considered box, ellipsoidal and budget uncertainty sets. All these single-stage approaches are collectively referred to as Static Robust Optimization (SRO).

- **Adjustable Robust Optimization:** SRO approaches are generally conservative, as they assume that all of the decisions have to be made "here-and-now", before the schedule begins to be implemented. In reality, many of the decisions can be "wait-and-see", meaning that they can be delayed until a later point when the a subset of the uncertain parameters have revealed their values. To handle such multi-stage decision making strategies, Adjustable Robust Optimization (ARO) is used, where an optimal policy is derived instead of a single, static solution. The optimal policy constitutes a family of solutions that are parameterized in the uncertain parameter realizations (Lappas and Gounaris, 2016).

# 2 Problem Statement?

# 3 Scheduling models used

This chapter briefly describes the scheduling models supported by the online optimization tool. All the models are mixed-integer linear models.

## 3.1 CBMN

This model is best. Best model ever.

$$\min_{\substack{W,T,B,\\U,S,G,z}} z \tag{3.1.1}$$

$$\text{s.t. } z \geq \sum_{s \in \mathcal{S}} P_s (S_{s0} - S_{sN}) \tag{3.1.2}$$

$$z \geq (T_N - T_1) \tag{3.1.3}$$

$$T_{n'} - T_n \geq \sum_{i \in \mathcal{I}_j} (\alpha_i W_{inn'} + \beta_i B_{inn'}) \qquad \forall j \in \mathcal{J}, \forall n \in \mathcal{N}, \forall n' \in \mathcal{N}_n^+ \tag{3.1.4}$$

$$W_{inn'} B_i^{\min} \leq B_{inn'} \leq W_{inn'} B_i^{max} \qquad \forall j \in \mathcal{J}, \forall n \in \mathcal{N}, \forall n' \in \mathcal{N}_n^+ \tag{3.1.5}$$

$$G_{jn} = G_{j(n-1)} + \sum_{i \in \mathcal{I}_j} \left[ \sum_{n' \in \mathcal{N}_n^+} W_{inn'} - \sum_{n' \in \mathcal{N}_n^-} W_{in'n} \right] \qquad \forall j \in \mathcal{J}, \forall n \in \mathcal{N} : \{n > 1\} \tag{3.1.6}$$

$$S_{sn} = S_{s(n-1)} + \sum_{i \in \mathcal{I}_s^p} \rho_{is} \sum_{n' \in \mathcal{N}_n^-} B_{in'n} - \sum_{i \in \mathcal{I}_s^c} \rho_{is} \sum_{n' \in \mathcal{N}_n^+} B_{inn'} \qquad \forall s \in \mathcal{S}, \forall n \in \mathcal{N} \tag{3.1.7}$$

$$\begin{aligned} U_{un} &= U_{u(n-1)} \\ &+ \sum_{u \in \mathcal{I}_u} \left[ \sum_{n' \in \mathcal{N}_n^+} (\gamma_{iu} W_{inn'} + \delta_{iu} B_{inn'}) - \sum_{n' \in \mathcal{N}_n^-} (\gamma_{iu} W_{in'n} + \delta_{iu} B_{in'n}) \right] \\ &\qquad \forall u \in \mathcal{U}, \forall n \in \mathcal{N} : \{n > 1\} \end{aligned} \tag{3.1.8}$$

$$T_{n'} - T_n \leq \bar{M} \left[ 1 - \sum_{i \ in \mathcal{I}_j \cap \mathcal{I}^{\text{zw}}} W_{inn'} \right] + \sum_{i \in \mathcal{I}_j} (\alpha_i W_{inn'} + \beta_{inn'}) \quad \forall j \in \mathcal{J}, \forall n \in \mathcal{N}, \forall n' \in \mathcal{N}_n^+ \tag{3.1.9}$$

$$\sum_{i \in \mathcal{I}_j} \sum_{\substack{n' \in \mathcal{N}: \\ \{n' \geq n\}}} \sum_{n'' \in \mathcal{N}_{n'}^+} (\alpha_i W_{in'n''} + \beta_i B_{in'n''}) \leq T_N - T_n \quad \forall j \in \mathcal{J}, \forall n \in \mathcal{N} \tag{3.1.10}$$

$$T_N = H \tag{3.1.11}$$

$$S_{sN} \geq D_s \qquad \forall s \in \mathcal{S} \tag{3.1.12}$$

$$0 \leq G_{jn} \leq 1 \qquad \forall j \in \mathcal{J}, \forall n \in \mathcal{N} \tag{3.1.13}$$

$$0 \leq S_{sn} \leq S_s^{\max} \qquad \forall s \in \mathcal{S}, n \in \mathcal{N} \tag{3.1.14}$$

$$0 \leq U_{un} \leq U_u^{\max} \qquad \forall u \in \mathcal{U}, n \in \mathcal{N} \tag{3.1.15}$$

$$W_{inn'} = 0 \qquad \forall i \in \mathcal{I}, \forall n \in \mathcal{N}, \forall n' \in \mathcal{N} \backslash \mathcal{N}_n^+ \tag{3.1.16}$$

$$B_{inn'} = 0 \qquad \forall i \in \mathcal{I}, \forall n \in \mathcal{N}, \forall n' \in \mathcal{N} \backslash \mathcal{N}_n^+ \tag{3.1.17}$$

$$T_1 = 0 \tag{3.1.18}$$

$$G_{jN} = 0 \qquad \forall j \in \mathcal{J} \tag{3.1.19}$$

$$W_{inn'} \in 0, 1 \qquad \forall i \in \mathcal{I}, \forall n \in \mathcal{N}, \forall n' \in \mathcal{N} \tag{3.1.20}$$

## 3.2 MG

Second best model but big as fuck.

$$\min \ z \tag{3.2.1}$$

$$\text{s.t. } z \geq \sum_{s \in \mathcal{S}} P_s (S_{s0} - S_{sN}) \tag{3.2.2}$$

$$z \geq (T_N - T_1) \tag{3.2.3}$$

$$T_1 = 0 \tag{3.2.4}$$

$$T_N = H \tag{3.2.5}$$

$$T_{n+1} \geq T_n \qquad \forall n \tag{3.2.6}$$

$$\sum_{i \in \mathcal{I}_j} Ws_{\text{in}} \leq 1 \qquad \forall j, \forall n \qquad (3.2.7)$$

$$\sum_{i \in \mathcal{I}_j} Wf_{\text{in}} \leq 1 \qquad \forall j, \forall n \qquad (3.2.8)$$

$$\sum_{n} Ws_{\text{in}} = \sum_{n} Wf_{\text{in}} \qquad \forall i \qquad (3.2.9)$$

$$S_{sn} = S_{sn-1} + \sum_{i \in O(s)} B^{O}_{isn} - \sum_{i \in I(s)} B^{I}_{isn} \qquad \forall s, \forall n > 1 \qquad (3.2.10)$$

$$S_{sn} \leq S^{\max}_{s} \qquad \forall s, \forall n \qquad (3.2.11)$$

$$U_{un} = U_{un-1} - \sum_{i} U^{O}_{iun-1} + \sum_{i} U^{I}_{iun} \qquad \forall u, \forall n \qquad (3.2.12)$$

$$U_{un} \leq U^{\max}_{u} \qquad \forall u, \forall n \qquad (3.2.13)$$

$$\sum_{i \in \mathcal{I}_j} \sum_{n' \leq n} (Ws_{in'} - Wf_{in'}) \leq 1 \qquad \forall j, \forall n \qquad (3.2.14)$$

$$Wf_{i0} = 0 \qquad \forall i \qquad (3.2.15)$$

$$Ws_{i0} = 0 \qquad \forall i, n = |N| \qquad (3.2.16)$$

$$D_{in} = \alpha_i Ws_{in} + \beta Bs_{in} \qquad \forall i, \forall n \qquad (3.2.17)$$

$$Tf_{in} \leq Ts_{in} + D_{in} + H(1 - Ws_{in}) \qquad \forall i, \forall n \qquad (3.2.18)$$

$$Tf_{in} \geq Ts_{in} + D_{in} + H(1 - Ws_{in}) \qquad \forall i, \forall n \qquad (3.2.19)$$

$$Tf_{in} - Tf_{in-1} \leq HWs_{in} \qquad \forall i, \forall n \qquad (3.2.20)$$

$$Tf_{in} - Tf_{in-1} \geq D_{in} \qquad \forall i, \forall n \qquad (3.2.21)$$

$$Ts_{in} = T_n \qquad \forall i, \forall n \qquad (3.2.22)$$

$$Tf_{in} \leq T_n + H(1 - Wf_{in}) \qquad \forall i, \forall n \qquad (3.2.23)$$

$$Tf_{in} \geq T_n - H(1 - Wf_{in}) \qquad \forall i \in I^{\text{zw}}, \forall n \qquad (3.2.24)$$

$$B^{min}_{i} Ws_{in} \leq Bs_{in} \leq B^{max}_{i} Ws_{in} \qquad \forall i, \forall n \qquad (3.2.25)$$

$$B^{min}_{i} Wf_{in} \leq Bf_{in} \leq B^{max}_{i} Wf_{in} \qquad \forall i, \forall n \qquad (3.2.26)$$

$$B^{min}_{i}\Big(\sum_{n' < n} Ws_{in'} - \sum_{n' \leq n} Wf_{in'}\Big) \leq Bp_{in} \leq B^{max}_{i}\Big(\sum_{n' < n} Ws_{in'} - \sum_{n' \leq n} Wf_{in'}\Big) \qquad \forall i, \forall n \qquad (3.2.27)$$

$$Bs_{in-1} + Bp_{in-1} = Bp_{in} + Bf_{in} \qquad \forall i, \forall n \qquad (3.2.28)$$

$$B^I_{isn} = \rho_{is} Bs_{in} \qquad \forall i \in \mathcal{I}^c_s, \forall n, \forall s \quad (3.2.29)$$

$$B^I_{isn} \leq B^{max}_i \rho_{is} Ws_{in} \qquad \forall i \in \mathcal{I}^c_s, \forall n, \forall s \quad (3.2.30)$$

$$B^O_{isn} = \rho_{is} Bf_{in} \qquad \forall i \in \mathcal{I}^p_s, \forall n, \forall s \quad (3.2.31)$$

$$B^O_{isn} \leq B^{max}_i \rho_{is} Wf_{in} \qquad \forall i \in \mathcal{I}^p_s, \forall n, \forall s \quad (3.2.32)$$

$$U^I_{iun} = \gamma_{iu} Ws_{in} + \delta_{iu} Bs_{in} \qquad \forall i, \forall u, \forall n \quad (3.2.33)$$

$$U^O_{iun} = \gamma_{iu} Wf_{in} + \delta_{iu} Bf_{in} \qquad \forall i, \forall u, \forall n \quad (3.2.34)$$

## 3.3 GHM

## 3.4 IF

Ierapetroiyadocjhoueotiou

$$\min \ z \quad (3.4.1)$$

$$\text{s.t. } z \geq \sum_{s \in \mathcal{S}} P_s (S_{s0} - S_{sN} - \sum_{i \in \mathcal{I}^p_s} \sum_{j \in \mathcal{J}_i} B_{iN}) \quad (3.4.2)$$

$$z \geq T_{ijn} + \alpha_i W_{iN} + \beta_i B_{iN} - T^s_{ij0} \qquad \forall i \in \mathcal{I}, \forall j \in \mathcal{J}_i \quad (3.4.3)$$

$$\sum_{i \in \mathcal{I}_j} W_{in} = G_{jn} \qquad \forall j \in \mathcal{J}, \forall n \in \mathcal{N} \quad (3.4.4)$$

$$B^{\min}_i \leq B_{in} \leq B^{\max}_i W_{in} \qquad \forall i \in \mathcal{I}, \forall j \in \mathcal{J}_i, \forall n \in \mathcal{N} \quad (3.4.5)$$

$$S_{sn} \leq S^{\max}_s \qquad \forall s \in \mathcal{S}, \forall n \in \mathcal{N} \quad (3.4.6)$$

$$S_{sn} = S_{s(n-1)} + \sum_{i \in \mathcal{I}^p_s} \rho_{is} \sum_{j \in \mathcal{J}_i} B_{i(n-1)} - \sum_{i \in \mathcal{I}^c_s} \rho_{is} \sum_{j \in \mathcal{J}_i} B_{in} \qquad \forall s \in \mathcal{S}, \forall n \in \mathcal{N} \quad (3.4.7)$$

$$S_{sN} \geq D_s \qquad \forall s \in \mathcal{S} \quad (3.4.8)$$

$$T^f_{ijn} = T^s_{ijn} + \alpha_i W_{in} + \beta_i B_{in} \qquad \forall i \in \mathcal{I}, \forall j \in \mathcal{J}_i, \forall n \in \mathcal{N} \quad (3.4.9)$$

$$T^s_{ij(n+1)} \geq T^f_{ijn} - H(2 - W_{in} - G_{jn}) \qquad \forall i \in \mathcal{I}, \forall j \in \mathcal{J}_i, \forall n \in \mathcal{N}, n \neq N \quad (3.4.10)$$

$$T^s_{ij(n+1)} \geq T^s_{ijn} \qquad \forall i \in \mathcal{I}, \forall j \in \mathcal{J}_i, \forall n \in \mathcal{N}, n \neq N \tag{3.4.11}$$

$$T^f_{ij(n+1)} \geq T^f_{ijn} \qquad \forall i \in \mathcal{I}, \forall j \in \mathcal{J}_i, \forall n \in \mathcal{N}, n \neq N \tag{3.4.12}$$

$$T^s_{ij(n+1)} \geq T^f_{i'jn} - H(2 - W_{i'n} - G_{jn}) \qquad \forall j \in \mathcal{J}, \forall i \in \mathcal{I}_j, \tag{3.4.13}$$

$$\forall i' \in \mathcal{I}_j, i \neq i', \forall n \in \mathcal{N}, n \neq N$$

$$T^s_{ij(n+1)} \geq \sum_{n' \in \mathcal{N}, n' \leq \mathcal{N}} \sum_{i' \in \mathcal{I}_j} (T^f_{i'jn'} - T^s_{i'jn'}) \qquad \forall i \in \mathcal{I}, \forall j \in \mathcal{J}_i, \tag{3.4.14}$$

$$\forall n \in \mathcal{N}, n \neq \mathcal{N}$$

$$T^f_{ijn} \leq H \qquad i \in \mathcal{I}, \forall j \in \mathcal{J}_i, n \in \mathcal{N} \tag{3.4.15}$$

$$T^s_{ijn} \leq H \qquad i \in \mathcal{I}, \forall j \in \mathcal{J}_i, n \in \mathcal{N} \tag{3.4.16}$$

## 3.5 SK

Shit model

$$\min \; z \tag{3.5.1}$$

$$\text{s.t. } z \geq \sum_{s \in \mathcal{S}} P_s (S_{s0} - S_{sN}) \tag{3.5.2}$$

$$z \geq \sum_{n \in \mathcal{N}} SL_n \tag{3.5.3}$$

$$\sum_{n=1}^{N} SL_n \leq H \tag{3.5.4}$$

$$G_{jn} = \sum_{i \in \mathcal{I}_j} Ws_{in} \qquad 0 \leq n < N \tag{3.5.5}$$

$$B_i^{min} Ws_{in} \leq Bp_{in} \leq B_i^{max} Ws_{in} \qquad i > 0 \tag{3.5.6}$$

$$Wp_{in} = Wp_{i(n-1)} + Ws_{i(n-1)} - Wf_{in} \qquad 0 < n < N \tag{3.5.7}$$

$$G_{jn} = \sum_{i \in \mathcal{I}_j} Wf_{in} \qquad 0 < n < N \tag{3.5.8}$$

$$\sum_{i \in \mathcal{I}_j} Wp_{in} + G_{jn} = 1 \qquad 0 < n < N \tag{3.5.9}$$

$$Wp_{in} + Ws_{in} \leq 1 \qquad 0 < n < N \tag{3.5.10}$$

$$Wp_{in} + Wf_{in} \leq 1 \qquad 0 < n < N \qquad (3.5.11)$$

$$t_{j(n+1)} \geq t_{jn} + \sum_{i \ in \mathcal{I}_j} (\alpha_i Ws_{in} + \beta_i Bp_{in}) - SL_{n+1} \qquad n < N \qquad (3.5.12)$$

$$Bs_{in} = Bs_{i(n-1)} + Bp_{i(n-1)} - Bf_{in} \qquad i > 0, n > 0 \qquad (3.5.13)$$

$$B_i^{min} Wp_{in} \leq Bs_{in} \leq B_i^{max} Wp_{in} \qquad i > 0, 0 < n < N \qquad (3.5.14)$$

$$B_i^{min} Wf_{in} \leq Bf_{in} \leq B_i^{max} Wf_{in} \qquad i > 0, 0 < n < N \qquad (3.5.15)$$

$$t_{jn} \leq \sum_{i \ in \mathcal{I}_j} (\alpha_i Wp_{in} + \beta_i Bs_{in}) \qquad 0 < n < N \qquad (3.5.16)$$

$$S_{sn} = S_{s(n-1)} + \sum_{\substack{i \in \mathcal{I}_s^p \\ i \neq 0}} \sum_{j \in \mathcal{J}_i} \rho_{is} Bf_{in} - \sum_{\substack{i \in \mathcal{I}_s^c \\ i \neq 0}} \sum_{j \in \mathcal{J}_i} \rho_{is} Bp_{in} \qquad \forall s \in \mathcal{S}, \forall n \qquad (3.5.17)$$

$$SL_n \leq \max_j \left[ \max_{i \in \mathcal{I}_j} (\alpha_i + \beta_i B_i^{max}) \right] \qquad \forall n \qquad (3.5.18)$$

$$t_{jn} \leq \max_{j \in \mathcal{J}_i} (\alpha_i + \beta_i B_i^{max}) \qquad \forall j \in \mathcal{J}, \forall n \qquad (3.5.19)$$

$$S_{sn} \leq S_s^{max} \qquad \forall s \in \mathcal{S}, \forall n \qquad (3.5.20)$$

$$Bp_{in}, Bs_{in}, Bf_{in} \leq B_i^{max} \qquad \forall i \in \mathcal{I}, \forall n \qquad (3.5.21)$$

# 4 GUI Design

This chapter describes the basic structure of the instance builder web page and the various assertions and conditions required for a well defined scheduling instance.

## 4.1 Schematic of communication

## 4.2 Instance structure

To facilitate compatibility with backend C++ code, the JavaScript instance object is converted to a JSON string. JSON (JavaScript Object Notation) is a text format that is completely language independent but uses conventions that are familiar to many different programming languages, including C, C++, Java, JavaScript, Python and others. These properties make JSON an ideal data-interchange format.

JSON is built on two universal data structures:

- A collection of name/value pairs. In various languages, this is realized as an object, record, struct, dictionary, hash table, keyed list, or associative array.
- An ordered list of values. In most languages, this is realized as an array, vector, list, or sequence.

Virtually all programming languages support these structures in one form or other. In JSON, they take on these forms:

An *object* is an unordered set of name/value pairs. An object begins with { (left brace) and ends with } (right brace). Each name is followed by : (colon) and the name/value pairs are separated by , (comma).

## 4.3 Instance object

The structure of the Instance JSON object is as shown in Figure 4.1. The empty arrays `[]` for `"Units"`, `"States"`, `"Tasks"`, `"Utilities"` and `"Orders"` in the structure contain the JSON objects as specific to the instance. `"Name"` is a string denoting the given name for the instance. `"Horizon"` is an integer or float. `"isCompleteInstance"` is a boolean `true` or `false` value denoting if an instance is completely specified.

```json
{
    "Name": "Scheduling_Instance",
    "Horizon": 8,
    "Units": [],
    "States": [],
    "Orders": [],
    "Utilities": [],
    "Tasks": [],
    "isCompleteInstance": false
}
```


*Fig. 4.1:* Instance JSON


```json
{
    "Name": "Unit1",
    "MaximumCapacity": 100
}
```


*Fig. 4.2:* Unit JSON object

An instance is deemed 'complete' if:

1. At least one valid unit
   - Positive maximum capacity
2. At least two states
   - Positive max capacity
3. At least one valid task
   - Non zero processing time
4. Positive horizon and at least one state to be sold with positive price OR At least one state in demand with positive demand amount

### 4.3.1 Units

The structure of the JSON object for Units is as follows:

```
{
    "StateName": "State1",
    "StateInitialLevel": 100,
    "StateMaxLevel": 100,
    "IsZeroWait": true,
    "IsUIS": false,
    "Price": 10
}
```

*Fig. 4.3:* State JSON object

```
{
    "Name": "Utility1",
    "MaximumAvailability": 250
}
```

*Fig. 4.4:* Utility JSON object

### 4.3.2 States

### 4.3.3 Utilities

### 4.3.4 Tasks

### 4.3.5 Orders

## 4.4 Instance builder

Figure 4.7 shows the elements in the instance builder GUI.

### 4.4.1 Error messages/assertions

### 4.4.2 List of errors

## 4.5 Submission page

### 4.5.1 Options structure

### 4.5.2 List of errors/assertions

```json
{
    "TaskName": "Task1",
    "CompatibleUnits": [
        {
        "UnitName": "Unit1",
        "alpha": 5,
        "beta": 1
        }
    ],
    "ConsumedStates": [
        {
        "ConStateName": "State1",
        "consRatio": 1
        }
    ],
    "ProducedStates": [
        {
        "ProdStateName": "State2",
        "prodRatio": 1
        }
    ],
    "ConsumedUtilities": [
        {
        "ConsUtilName": "Utility1",
        "CompUnit": "Unit1",
        "gamma": 2,
        "delta": 0.2
        }
    ]
}
```

*Fig. 4.5:* Task JSON object

```json
{
    "StateName": "Product1",
    "Amount": 100
}
```

*Fig. 4.6:* Order (demand) JSON object

*Table 4.1:* Uncertainty frameworks and adjustable variables

| | | | Adjustable variables | | |
|---|---|---|---|---|---|
| | SRO | ARO | T | S | B + S |
| $\alpha$ | ✓ | ✓ | ✓ | ✗ | ✓ |
| $\alpha + \beta$ | ✓ | ✓ | ✓ | ✗ | ✗ |
| $\rho_p$ | ✗ | ✓ | ✗ | ✓ | ✗ |
| $\alpha + \rho_p$ | ✗ | ✓ | ✓ | ✓ | ✗ |

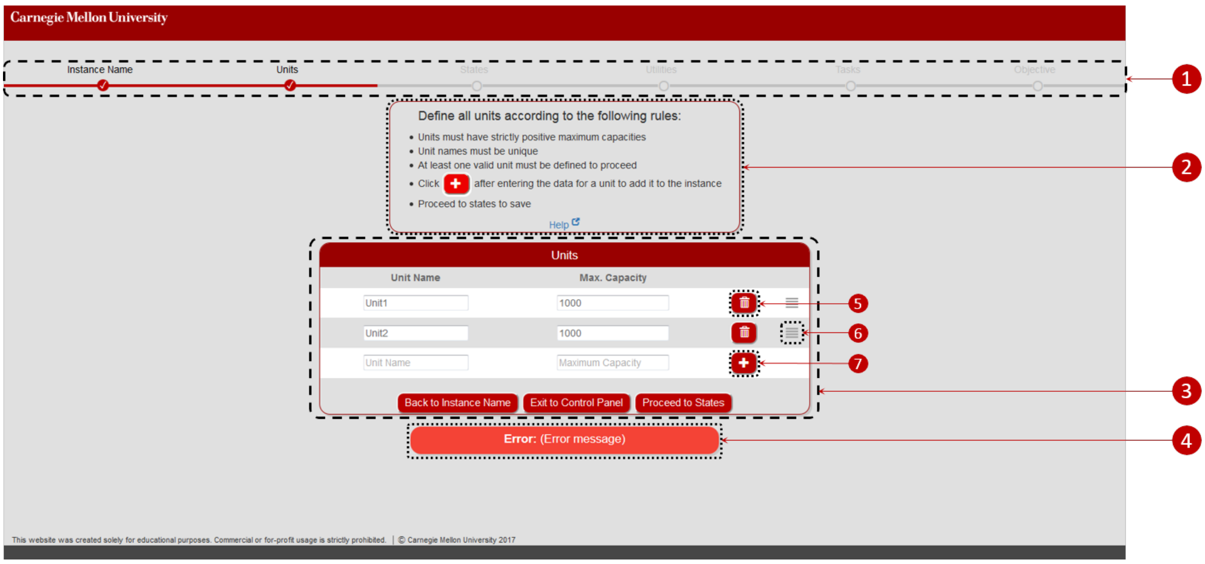


*Fig. 4.7:* Instance builder elements: 1. Progress bar 2. Instructions 3. Input table 4. Error message(s) 5. Delete item 6. Drag and drop to rearrange 7. Add new item

# 5 Case Study

## 5.1 Instance description

In this chapter, we present a case study of a well known scheduling instance from Kondili et al. (1993), optimized using the described online tool. The production of two products 1 and 2 from three feed stocks A, B and C takes place according to the STN representation given in Fig. 5.1. This instance has four intermediate states and five tasks.

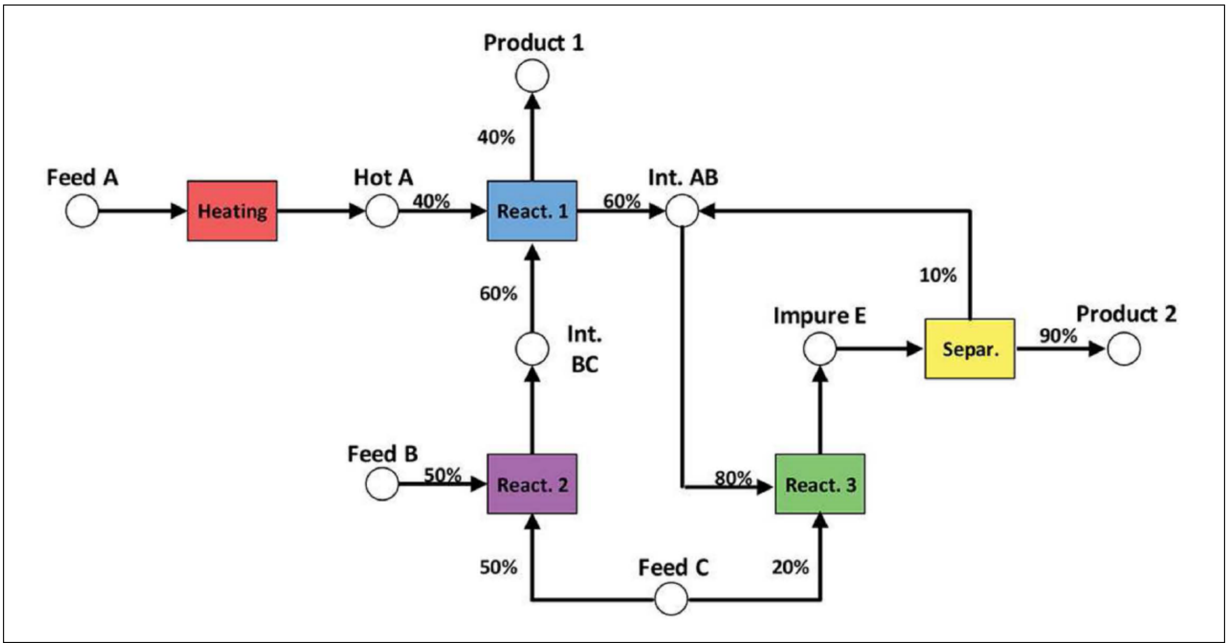


*Fig. 5.1:* State task network for the example instance

Table 5.1 shows the state maximum capacity and initial load data for the problem. The available unit, task compatibility and processing time data is given in Table 5.2.

## 5.2 Adding instance to webtool

### 5.2.1 Defining units

Unit names and maximum capacities from Table 5.2 are input into the units table of the web tool as shown in Fig. 5.2

### 5.2.2 Defining states

State names, maximum storage capacities and initial levels from Table 5.1 are input into the units table of the web tool as shown in Fig. 5.3.

*Table 5.1:* Problem data (states)

| **State** | **Capacity** | **Initial load** | **Price (per unit)** |
|---|---|---|---|
| Feed A | 1000 | 1000 | - |
| Feed B | 1000 | 1000 | - |
| Feed C | 1000 | 1000 | - |
| Hot A | 100 | - | - |
| Int. BC | 200 | - | - |
| Int. AB | 150 | - | - |
| Impure E | 200 | - | - |
| Product 1 | 1000 | - | 10 |
| Product 2 | 1000 | - | 10 |

*Table 5.2:* Problem data (units & tasks)

| Unit | Heater | | Reactor 1 | | Reactor 2 | | Separator | |
|---|---|---|---|---|---|---|---|---|
| Maximum Load | 100 | | 50 | | 80 | | 200 | |
| Tasks | $\alpha$ | $\beta$ | $\alpha$ | $\beta$ | $\alpha$ | $\beta$ | $\alpha$ | $\beta$ |
| Heating | 0.667 | 0.007 | | | | | | |
| Reaction 1 | | | 1.334 | 0.027 | 1.334 | 0.017 | | |
| Reaction 2 | | | 1.334 | 0.027 | 1.334 | 0.017 | | |
| Reaction 3 | | | 0.667 | 0.013 | 0.667 | 0.008 | | |
| Separation | | | | | | | 1.334 | 0.007 |

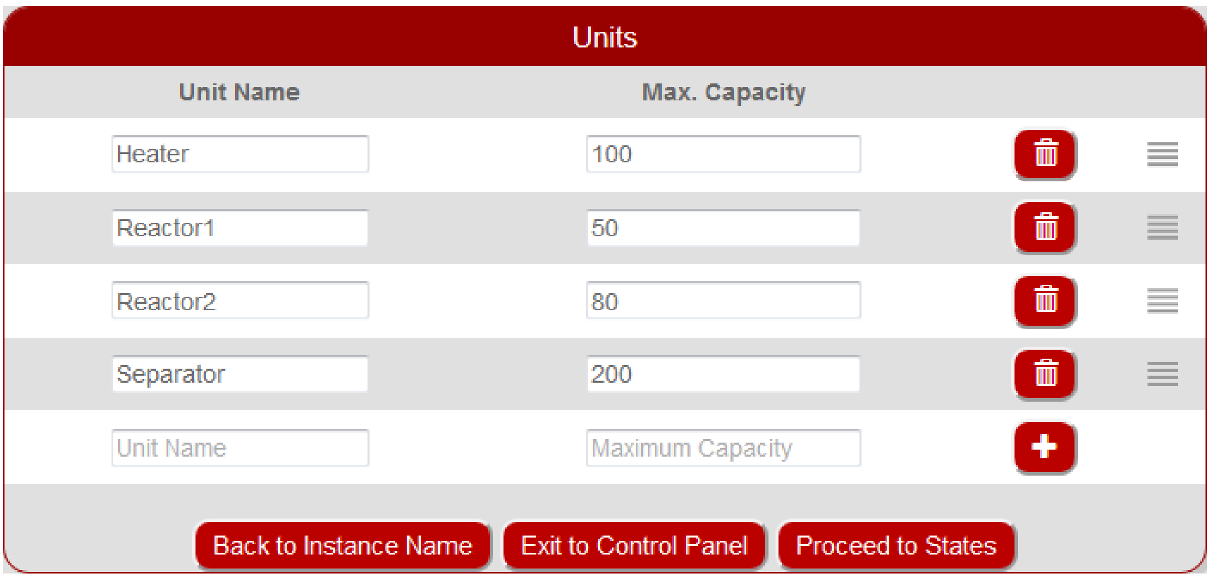


*Fig. 5.2:* Units input into webtool

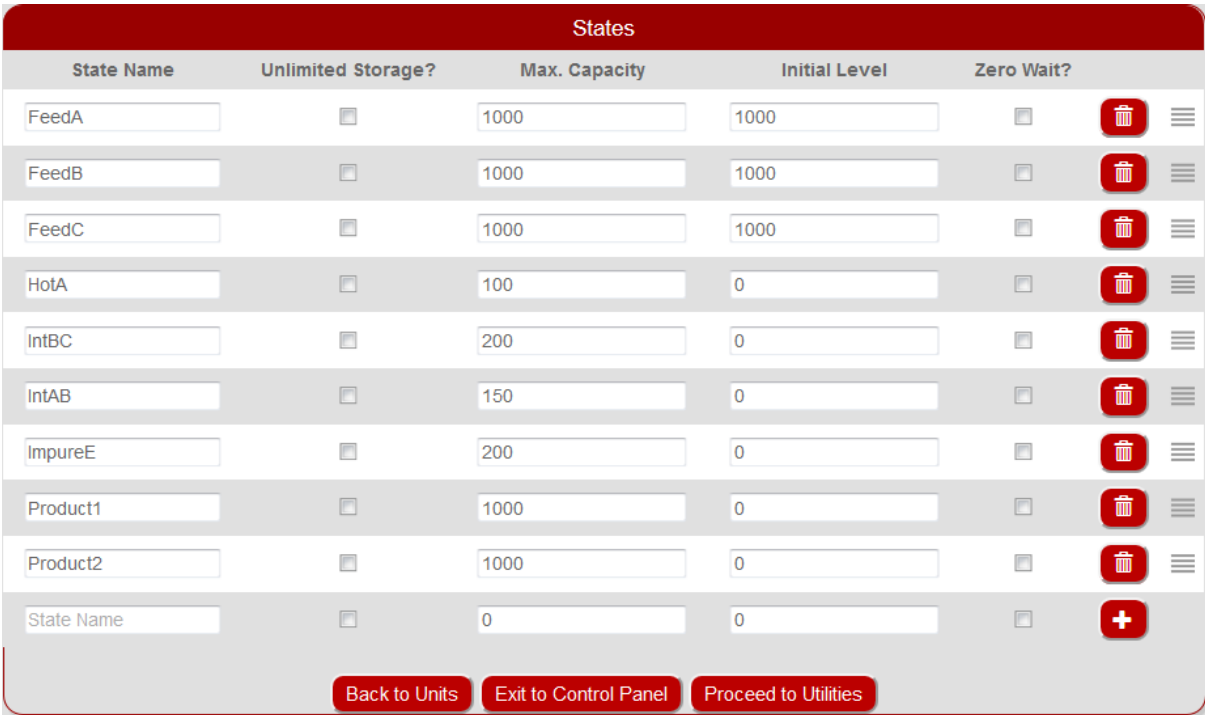


*Fig. 5.3:* States input into webtool

### 5.2.3 Defining tasks

This instance does not involve utilities. Hence we can skip utilities input. For each task in Table 5.2, the values of $\alpha$ and $\beta$ are input after selecting the appropriate compatible unit(s). Fig. 5.4 shows the tasks table after successful input of tasks Heating and Reaction 1.

### 5.2.4 Selecting Objective

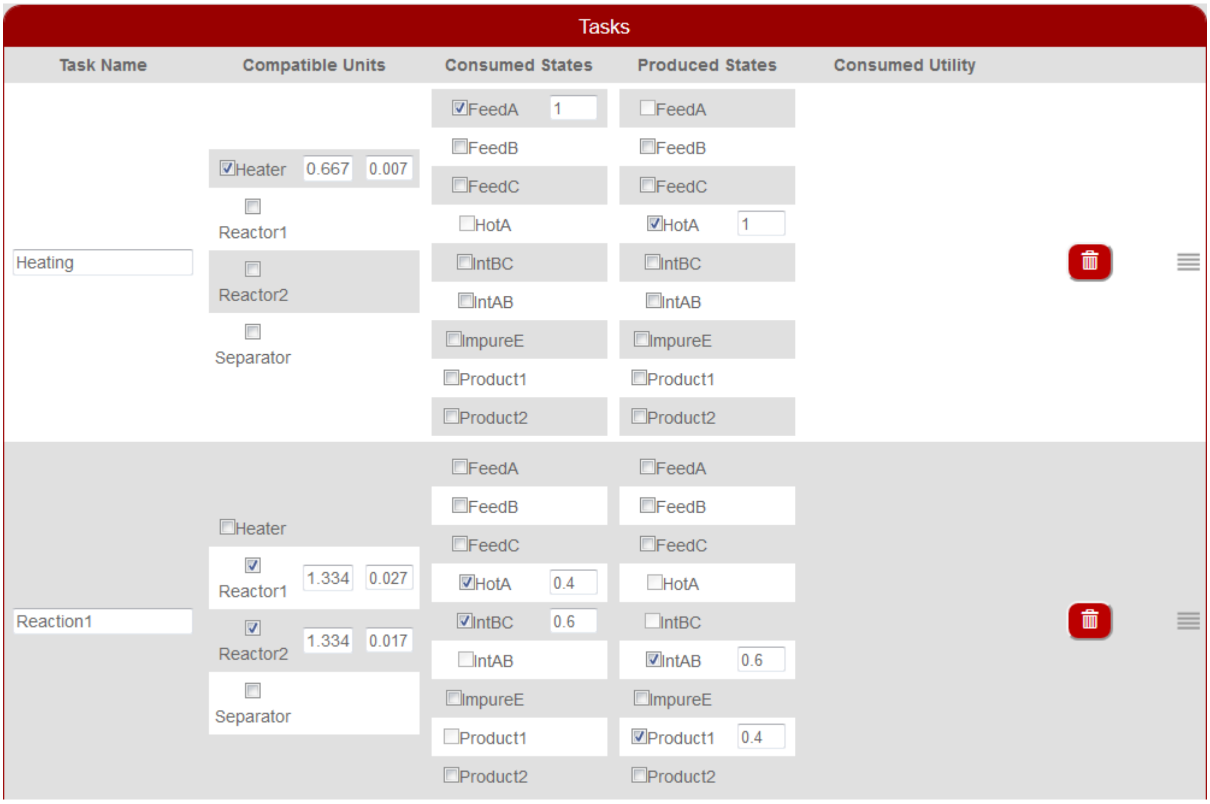


*Fig. 5.4:* Tasks input into webtool


*Fig. 5.5:* Objective selection

## References

Harjunkoski, I., Maravelias, C. T., Bongers, P., Castro, P. M., Engell, S., Grossmann, I. E., Hooker, J., Mndez, C., Sand, G., and Wassick, J. Scope for industrial applications of production scheduling models and solution methods. *Computers & Chemical Engineering*, 62(Supplement C):161 – 193 (2014).

Janak, S. L., Lin, X., and Floudas, C. A. A new robust optimization approach for scheduling under uncertainty: II. Uncertainty with known probability distribution. *Computers & Chemical Engineering*, 31(3):171 – 195 (2007).

Kondili, E., Pantelides, C., and Sargent, R. A general algorithm for short-term scheduling of batch operations – I. MILP formulation. *Computers & Chemical Engineering*, 17(2):211 – 227 (1993).

Lappas, N. H. and Gounaris, C. E. Multi-stage adjustable robust optimization for process scheduling under uncertainty. *AIChE Journal*, 62(5):1646–1667 (2016).

Li, Z. and Ierapetritou, M. G. Robust Optimization for Process Scheduling Under Uncertainty. *Industrial & Engineering Chemistry Research*, 47(12):4148–4157 (2008).

Lin, X., Janak, S. L., and Floudas, C. A. A new robust optimization approach for scheduling under uncertainty:: I. Bounded uncertainty. *Computers & Chemical Engineering*, 28(6):1069 – 1085 (2004), FOCAPO 2003 Special issue.

Méndez, C. A., Cerdá, J., Grossmann, I. E., Harjunkoski, I., and Fahl, M. State-of-the-art review of optimization methods for short-term scheduling of batch processes. *Computers & Chemical Engineering*, 30(6):913 – 946 (2006).