

MIPSYN-Global: Process Synthesis Enabled by Graphical Modelling

Miloš Bogataj^a, Zdravko Kravanja^{a,*}, Aleksander Soršak^a, Bojan Slemnik^b,
 Uroš Klemenčič^b, Simon Jurič^b

^a Faculty of Chemistry and Chemical Engineering, University of Maribor, Smetanova ulica 17, SI-2000 Maribor, Slovenia

^b INOVA IT, Jadranska ulica 25a, SI-2000, Maribor, Slovenia
 zdravko.kravanja@um.si

We introduce MIPSYN-Global - a unique computer-based process synthesizer. It is built on the foundations of its predecessor MIPSYN, using the knowledge and experience gained from decades of research in the field of PSE. One of its main features is the newly developed Graphical User Interface - MIPSYN-Global Modeler (MGM), which is specifically designed for fast modeling of process superstructures and representation of results. In addition, the modeler generates an Aspen Plus process simulator input file and runs the Aspen Plus simulation in which more rigorous thermodynamic and process unit models can be used. MGM is evidently the most notable new feature in the development of MIPSYN-Global. However, it is important to note that MIPSYN-Global inherits and builds upon the capabilities of its predecessor, making it a versatile and robust platform for solving process synthesis problems from engineering fields beyond chemical engineering. Nevertheless, since graphical modeling capability is a rather rare approach in equation-oriented synthesis, we present MGM's capabilities with an illustrative example of simple reactor network synthesis.

1. Introduction

Process synthesis approaches based on mathematical programming have proven to be valuable tools for identifying innovative solutions to complex engineering problems. These problems can span orders of magnitude in temporal and spatial scales, i.e., from nm and ms (molecular design) to km and y (supply networks). For example, Jonuzaj et al. (2018) presented an approach to design optimal mixtures from atom groups, while Cassiolato et al. (2019) presented an approach for optimization of water distribution networks. Approaches for synthesis and optimization of industrial-scale processes (Zhang et al., 2019) are commonly based on superstructural representation and generally formulated as mixed-integer nonlinear programming problems (MINLP). The core advantage of a MINLP is that it simultaneously captures the nonlinear relationships (e.g., descriptions of process units) and discrete decisions (e.g., existence or nonexistence of a particular process unit). On the other hand, encoding the mathematical representation of the superstructure in the chosen software syntax is time-consuming and often error-prone. It can be a significant hurdle, especially for inexperienced users. Moreover, the augmentation and reusability of whole or parts of hard-coded models is often very difficult.

Clearly, an environment that would enable, at least in the early stages of superstructure development, a workflow similar to that in modern process simulators by avoiding the unnecessary coding phase would be of enormous benefit. Such an environment would need to support functionalities such as drag and drop, allow the user to easily combine library and user-defined process unit models within a superstructure (flowsheet), enable automated compilation into a MINLP during the build phase, and allow retrieval and graphical representation of the obtained solutions.

Despite many contributions from the PSE community (Grossmann and Harjunkoski, 2019), there are no professional tools and hardly any academic tools specialized in providing such innovative solutions. One notable initiative is PYOSYN - an open-source framework for systematic superstructure-based process synthesis (Chen et al., 2021). Built upon MIPSYN (Kravanja, 2010), MGM in conjunction with MIPSYN-Global addresses these challenges and was designed to complement MIPSYN-Global, primarily by improving the user experience. In

the remainder of this manuscript, we introduce MGM in the context of MIPSYN-Global framework and present its capabilities with an example of a simple process synthesis problem.

2. MIPSYN-Global Modeler

MIPSYN-GLOBAL Modeler (MGM) is a graphical user interface primarily intended for rapid development of superstructure-based process models (flowsheets). It is a user-oriented environment that allows interaction between the user and MIPSYN-Global (MINLP Synthesis Environment), as well as interaction between MIPSYN-Global and Aspen Plus process simulator (Figure 1). The visual aspects and flowsheeting functionalities such as drag and drop, process units, streams, connectivity of units and input forms were built using open-source libraries from the DWSIM Process Simulator (DWSIM, 2021).

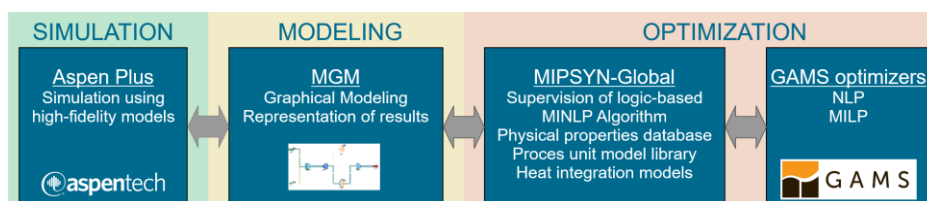


Figure 1: MGM and MIPSYN-Global framework

The MGM GUI is presented in Figure 2. Its main components are:

- **Canvas** (superstructure flowsheeting)
- **Model library** (library of models in GAMS syntax)
- **Library of pure component properties** (currently 480 components)
- **Project explorer** (project files)
- **Model library palette** (graphical representation of models):
 - Material nodes (feed, product)
 - Energy nodes (heating, cooling)
 - System nodes (Heat integration models: a) HINTS (Duran and Grossmann, 1986), b) HENY (Yee and Grossmann, 1990))
 - Interconnection nodes (Single and multiple-choice splitters and mixers)
 - Process unit nodes (reactor, component separator, heater, cooler, distillation column, membrane, pump, compressor, flash, valve, etc.)
 - Tasks (conceptual models on the level of tasks: pressure change, temperature change, reaction, separation)
 - Objectives (objective functions)
- **Modes toolbar:**
 - Superstructure (selection of components and values of global parameters and variable bounds)
 - Nodes (selection of material, energy, interconnection and process unit nodes in represented superstructure, drag and drop)
 - Streams (node interconnections, aggregation)
 - Binary variables (assignment of binary variables to process units and/or streams, identification of substructures for model-decomposition)
 - MIPSYN and Aspen (build optimization model and run optimization (automated, interactive), build Aspen Plus input file and run simulation, representation of results)

2.1 Workflow in MGM Graphical user interface

The workflow in MGM follows the order of elements in Modes Toolbar, guiding the user from selection of components, generation of superstructure to running the optimization model and representation of results. The remainder of this section briefly describes each of the toolbar elements and their functionality.

Superstructure Mode

In Superstructure Mode, the user selects components and, if necessary, changes the default values of scaling parameters and values of global (general) bounds on variables.

Nodes Mode

In Nodes Mode, the user drags and drops unit models from the Model Library Palette onto the canvas and enters the required inputs for the unit models (e.g., upper and lower bounds on model-specific variables, fixed operating

parameters, economic parameters etc.). These inputs are entered via input forms, which become available by double-clicking on the model located on the canvas.

Streams Mode

In Streams Mode, the user interconnects the unit model with streams into a superstructure.

Binary Variable Mode

In Binary Variable Mode, the user allocates binary variables to either streams or unit models (feeds, products reactors etc.) or both by drag and drop.

Substructure Mode

Before the model can be run, the user must identify an initial flowsheet (primary substructure) and all the remaining substructures. This is a necessary step for obtaining outer approximations of the entire superstructure through Lagrangian suboptimization.

MIPSYN and Aspen Mode

Finally, in MIPSYN and Aspen Mode, the user can run the optimization in either automated or interactive mode and run Aspen Plus simulation of the identified integer solutions (optimal and/or suboptimal).

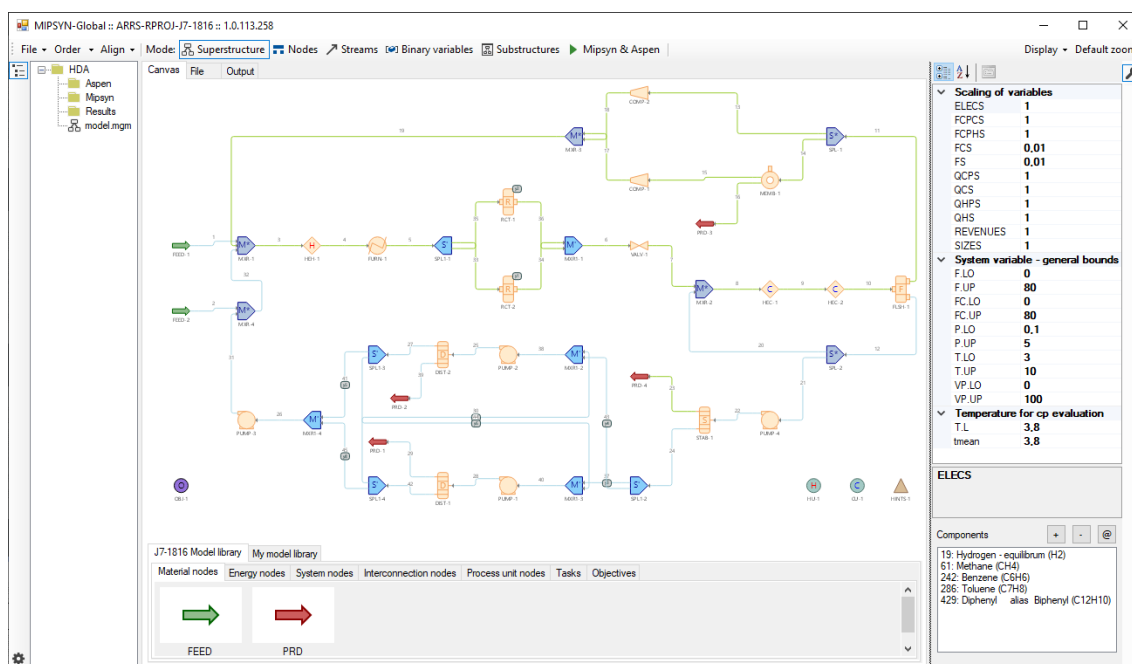


Figure 2: MGM Graphical User Interface

3. Illustrative Example

The goal of the example - a variation on the one originally presented by Kocis and Grossmann (1989) - is to identify an optimal process for production of 1 kmol/s of methanol ($C^{\text{PROD}} = 8,000 \text{ k€}/(\text{kmol} \cdot \text{y})$) from syngas. Two feeds are available with different compositions and prices of syngas, and two reactors with different efficiency, as well as variable and fixed cost. The objective function (Eq. 1) to be maximized is total annual cost (TAC):

$$\min TAC = C^{\text{PROD}} \cdot F_{\text{CH}_3\text{OH}}^{\text{PROD}} - \sum_{i \in \text{FEED}} C_i \cdot \sum_{j=1}^4 F_{i,j} - \sum_{k \in \text{RCT}} y_k \cdot C_k^F - \sum_{k \in \text{RCT}} C_k^V \cdot V_k \quad (1)$$

where C represents cost coefficients, F represents component molar flowrates, y is binary variable denoting (non)existence of reactor and V is reactor volume.

The discrete decisions are related to selection of feeds and reactors. In the case studied, only one feed and one reactor may be selected. Composition of feeds and cost factors are given in Table 1. Although the example is trivial in the sense that one could simply determine the optimal process by sequentially optimizing the four possible structures, it is its simplicity that makes it suitable for demonstrating the process synthesis enabled by graphical modelling. We start building the synthesis problem by first selecting the components (hydrogen, carbon monoxide, hydrogen, methane and methanol) from the component database. In the next step, as shown in Figure 3a, we populate the canvas with material nodes (2 feeds (FEED-1, FEED-2), 1 product (PRD-1)), interconnection nodes (2 single-choice mixers (MXR1-1, MXR1-2) and a single-choice splitter (SPL1-1)), and

process units (2 reactors (RCT1, RCT-2)). At this stage, we define the feeds (composition, pressure, temperature), we enter the data required by the reactor models (upper bounds on reactor volume, bounds on inlet and outlet temperatures) and define the product requirements (targeted methanol flowrate). In the next step, we interconnect the elements on the canvas with streams (1–8) and assign binary variables to reactors (y_1, y_2) and feeds (y_3, y_4). Note that MGM automatically assigns names to all the elements on the canvas. These are later, during the build phase, used to define and populate GAMS sets.

Table 1: Feed stream and reactor data

| Parameter | Feed 1 | Feed 2 | Parameter | Reactor 1 | Reactor 2 |
|----------------------------------|--------|--------|-------------|-----------|-----------|
| $T(K)$ | 400 | 400 | $C^F(k€/y)$ | 100 | 200 |
| $p(MPa)$ | 2.5 | 2.5 | $C^V(k€/y)$ | 5 | 10 |
| x_{H_2} | 0.60 | 0.65 | | | |
| x_{CO} | 0.25 | 0.30 | | | |
| x_{CH_4} | 0.15 | 0.05 | | | |
| $C^{FEED} (k€/((kmol \cdot y)))$ | 500 | 600 | | | |

The final required step before the build phase can be executed is the identification of the initial flowsheet and substructures. Although the procedure could be automated, at least in principle, this step requires user input. The main reason for this is that the initial flowsheet and substructures may not be unique for a given superstructure. For example, the superstructure in Figure 3b has four potential initial flowsheets, where an initial flowsheet is defined as any continuous sequence of elements on the canvas that starts with feed(s) and ends with product(s). In other words, the initial flowsheet is a self-containing part of a superstructure that can be simulated. The one shown in Figure 4a starts at FEED-1, continues through MXR1-2, SPL1-1, RCT-2, MXR1-1 and ends at PRD-1. An alternative initial flowsheet would be a sequence containing RCT-1 instead of RCT-2. Clearly, the other two remaining initial flowsheet alternatives start at FEED-2 and follow similar paths through the superstructure (i.e., FEED-2 \rightarrow MXR1-2 \rightarrow SPL1-1 \rightarrow RCT-2 \rightarrow MXR1-1 \rightarrow PRD-1 and FEED-2 \rightarrow MXR1-2 \rightarrow SPL1-1 \rightarrow RCT-1 \rightarrow MXR1-1 \rightarrow PRD-1).

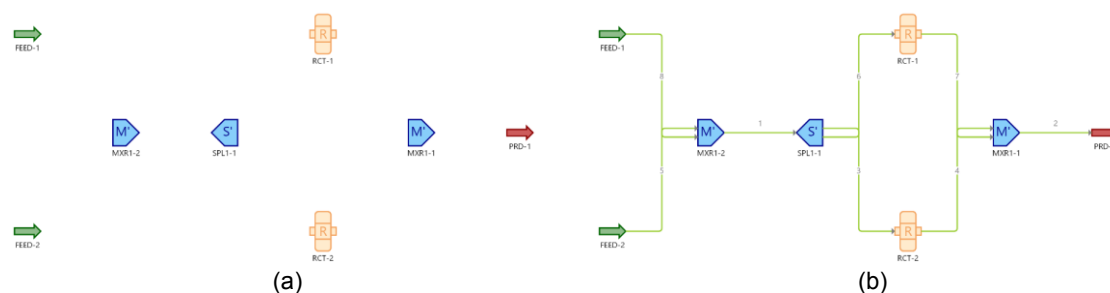


Figure 3: Constructing the methanol synthesis superstructure

Once we have selected the initial flowsheet, we identify all potential substructures (partial flowsheets) that are present in the superstructure. In general, these substructures are the branches that represent alternatives to the initial flowsheet configuration. According to the initial flowsheet in Figure 4a, the two substructures are shown in Figures 4b and 4c. The first one represents an alternative to the feed selection and the second one represents an alternative to the reactor selection.

Finally, the MINLP problem can be built and run. MGM builds the MIPSYN-Global input files by combining the information from the canvas (structure, connectivity, user inputs) and models of the nodes and objectives represented in the superstructure from the model library. The input files are compiled into a GAMS file and passed to GAMS solvers (NLP and MILP) by MIPSYN-Global, which supervises the logic-based MINLP solution procedure. The procedure comprises initialization of variables, simulation of initial flowsheet, suboptimization of substructures, generation of linearizations and the solution algorithm based on Outer Approximation / Equality Relaxation (OA/ER). For a more comprehensive description of MIPSYN-Global inner workings, the user is referred to the initial article on the tool (Kravanja and Grossmann, 1990) and the presentation of the updated tool (Kravanja, 2010).

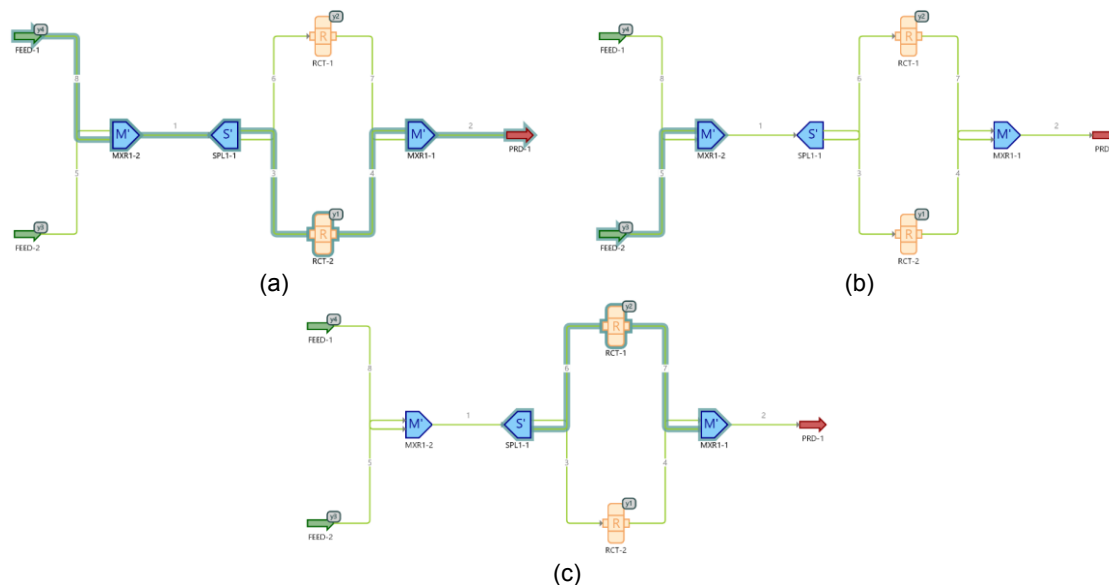


Figure 4: Initial flowsheet and substructures for logic-based MINLP algorithm with Lagrangian suboptimization

3.1 Representation of the optimization and simulation results

The O/AER algorithm generally identifies multiple integer solutions. These solutions differ in values of binary and continuous variables and represent unique flowsheet configurations. For each of the identified solutions, the user can generate an Aspen Plus input file and run the simulation.

The results of all identified integer solutions are available to the user via GAMS listing files and/or user-defined text files. However, the most common and useful results, such as topology, total flowrates, component flowrates, temperatures, and pressures, can be displayed directly in the canvas (see Figure 5).

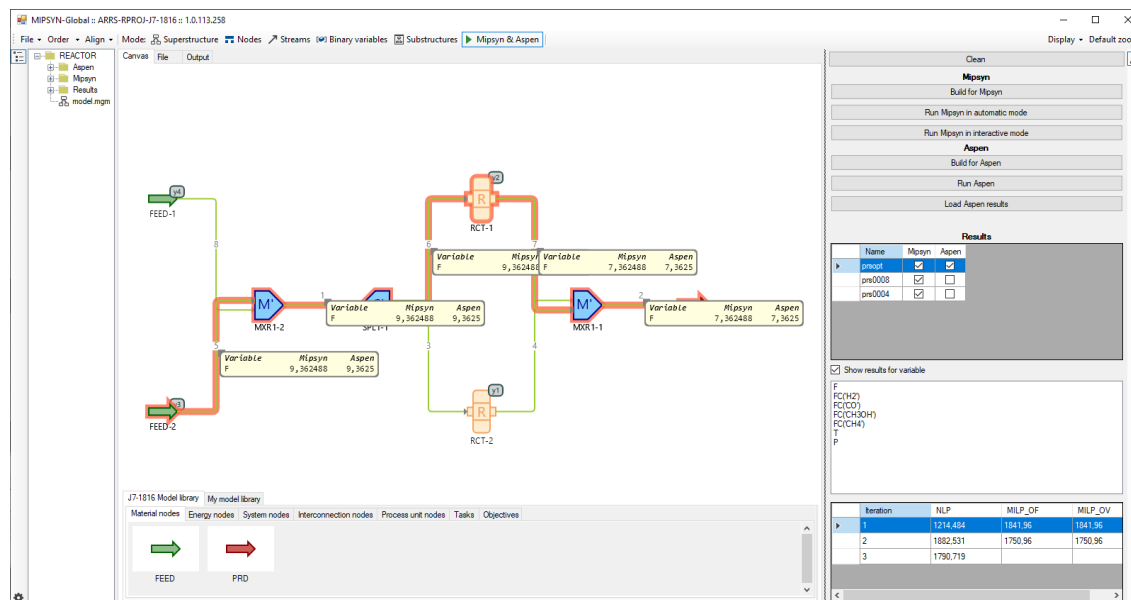


Figure 5: Optimization and simulation results

The topology of the selected integer solution is highlighted and represents a complete process flowsheet. The data regarding stream properties are displayed besides the streams. If Aspen Plus simulation results are available, they are displayed along with the optimization results.

4. Conclusions

We have presented MGM, a newly developed Graphical User interface for MIPSYN-Global. It was developed to enable rapid development of process synthesis models and to enable data flow between synthesis/optimization environment (MIPSYN-Global) and simulation environment (Aspen Plus). One of the main advantages of MGM is the generation of Aspen Plus input files based on the topologies of the obtained integer solutions. Although the unit models defined in these input files are by default the simplest unit models (e.g., component separator instead of the rigorous Radfrac model), the input files can be opened as flowsheets in Aspen Plus and edited as desired. For example, one could replace the process unit models with more rigorous models, or perhaps replace the thermodynamic model with a more appropriate one. In a sense, MGM creates Aspen Plus flowsheet templates, and this is what becomes useful in more complex cases, especially from the perspective of saving time and reducing errors.

In terms of development stages, MGM has gone through the proof-of-concept phase. However, like any technology, MGM needs to be developed further. Future work will focus on concurrent development of both MGM and MIPSYN-Global. On the MGM side, we would like to enhance the user experience by automating the validation of user inputs (e.g. check on sum of mole fractions), creating intuitive (nested) input forms that would guide the user through needed inputs and would only reveal additional inputs when necessary, generation of customized reports (e.g. html, pdf.) etc. On the side of MIPSYN-Global, we would like to develop libraries of models for optimization/synthesis of processes under uncertainty, develop a module for synthesis and optimization of microprocesses, and implement an algorithm for global optimization of large scale nonconvex MINLP models.

Acknowledgements

The authors would like to acknowledge the Slovenian Research Agency (Research Program P2-0032 and Project J7-1816) for financial support.

References

- Cassiolato G.H.B., Carvalho E.P., Caballero J.A., Ravagnani M.A.S.S., 2019. Water Distribution Networks Optimization Using Disjunctive Generalized Programming. *Chemical Engineering Transactions*, 76, 547–552.
- Chen Y.L., Seastream G., Sirola J.D., Grossmann I.E., 2021. Pyosyn: a new framework for conceptual design modeling and optimization, *Computers and Chemical Engineering*, doi: 10.1016/j.compchemeng.2021.107414.
- Duran M.A., Grossmann I.E., 1986. Simultaneous optimization and heat integration of chemical processes. *AIChE Journal*, 32(1), 123–138.
- DWSIM, 2021. DWSIM – Open Source Process Simulator, <<https://dwsim.inforside.com.br/new/>>, accessed 06. 06. 2021.
- Grossmann I.E., Harjunkski I., 2019. Process systems Engineering: Academic and industrial perspectives. *Computers and Chemical Engineering* 126, 474–484.
- Jonuzaj S., Gupta A., Adjiman C.S., 2018. The design of optimal mixtures from atom groups using Generalized Disjunctive Programming. *Computers and Chemical Engineering*, 116, 401–421.
- Kocis G.R., Grossmann I.E., 1989. A modelling and decomposition strategy for the MINLP optimization of process flowsheets. *Computers and Chemical Engineering*, 13(7), 797–819.
- Kravanja Z., 2010. Challenges in sustainable integrated process synthesis and the capabilities of an MINLP process synthesizer MipSyn. *Computers and Chemical Engineering*, 34, 1831–1848.
- Kravanja Z., Grossmann I.E., 1990. Prosyn—an MINLP process synthesizer. *Computers and Chemical Engineering*, 14(12), 1363–1378.
- Yee T.F., Grossmann I.E., 1990. Simultaneous optimization models for heat integration—II. Heat exchanger network synthesis. *Computers & Chemical Engineering*, 14(10), 1165–1184.
- Zhang L., Wang Y., Feng X., 2020. Design and Operation Optimization of Industrial Waste Heat Recovery for District Heating and Cooling, *Chemical Engineering Transactions*, 81, 511–516.