Effective GDP optimization models for modular process synthesis

Qi Chen and Ignacio E. Grossmann*

Department of Chemical Engineering, Carnegie Mellon University, Pittsburgh, PA, USA

E-mail: grossmann@cmu.edu

Abstract

In this paper, we propose an optimization-based strategy to systematically evaluate tradeoffs associated with modular alternatives for the multi-period design of a chemical processing network. We give a general formulation as a Generalized Disjunctive Program (GDP) and discuss a linearizing reformulation that exploits structure common to modular design problems. By modeling the GDP in the Pyomo algebraic modeling language, we gain access to a flexible set of automatic reformulations and solution algorithms, from which the best tool may be selected to optimize a given model. We apply the design strategy to a set of illustrative case studies, including capacity expansion, bioethanol processing, and heat exchange network design. The results show that the proposed design strategy is able to solve modular design problems and provide general insights on tradeoffs between investment and transportation costs in which incorporation of modular facility constructions may or may not prove to be advantageous.

Introduction

The past few years bear witness to growing interest in modular chemical plants, which offer improved flexibility, quality, and schedule efficiency characteristics over traditional plant constructions.[1] In a modular plant, major pieces of processing equipment are assembled as standardized modules rather than custom-designed and constructed on-site. A module includes the processing equipment and associated control instruments, piping, valves, and interconnection points mounted in a structural steel framework. Each module forms a self-contained processing unit, which can be built and tested in a controlled environment at the manufacturer's workshop. In this way, modular design can be seen as a move towards greater standardization in the chemical process industry.

Modular design is not a new idea in manufacturing[2] or chemical processing,[3] but its combination with the ideas of distributed manufacturing[4] and process intensification[5,6] comprises a compelling solution to modern chemical process industry challenges.[7] However, modular plant design may not be appropriate in all places and situations.[1] Therefore, our main challenge is the assessment of modular design as a partial or complete replacement for existing processes and the exploration of new applications that are enabled by modular design. Mathematical programming can be a powerful tool for this assessment. Through optimization-based process synthesis,[8] we can systematically evaluate trade-offs between the advantages and disadvantages of modular process design.

The first step of an effective analysis is development of the appropriate physical and cost models. Several authors have investigated modular design as a central element of Europe's Industry 4.0 initiative, with flexibility and time to market identified as key benefits of modular designs.[9–11] Lier and Grünewald provide a net present value (NPV) analysis of modular versus conventional constructions, showing that for rapid market growth scenarios, modular plants can outperform conventional designs due to their flexibility and responsiveness.[12] The European literature also explores the value of flexibility in providing extra value in the presence of uncertainty.[13,14] Other authors have examined modular design in conjunction with specific applications. Modular chemical facilities appear well-suited to address the recent issue of stranded natural gas processing.[15,16] The distributed nature of gas sources in some regions makes constructing pipelines to a traditional centralized processing facility economically challenging. Instead, small scale modular facilities could be sited close to active wells and relocated based on shifting production patterns. These facilities would process the gas into a more transportable product, extracting value that would otherwise be lost and improving environmental performance through reduced flaring.

To consider the broad range of potential benefits and tradeoffs when introducing the modular process concept to a chemical enterprise, it is desirable to design a modular processing network with the help of an optimization framework—one that explicitly considers time-to-market and system flexibility advantages alongside module design. The resulting optimization problem involves not only discrete decisions in siting and allocation, but also nonlinear relations arising from concave cost functions and process unit performance descriptions. As such, it is in general a nonlinear, nonconvex discrete-continuous optimization problem. To solve this difficult class of problems, it is often advantageous to exploit special structure when it exists. There is therefore a need for an optimization framework that is able to pose and address these modular design problems.

In this paper, we develop a strategy for the simultaneous planning and design of a chemical processing network to incorporate modular process units. The problem is formulated as a multiperiod facility location, allocation, and design model which explicitly considers the time-to-market implications of selecting a modular versus conventional construction approach. The objective is to determine the optimal facility locations, respective designs, and transportation links for a chemical enterprise, taking into consideration raw material and product shipping costs, modular process alternatives, and module relocation. In the remainder of this paper, we present the general problem statement, its formulation as a Generalized Disjunctive Programming (GDP) model,[17] and explore tailored reformulation and solution strategies such as induced linearity[18] that suit the modular design problem. We also illustrate through a set of case studies how the GDP model enables the investigation of a range of modular design problems.

Problem statement

Given is a set of time periods $t \in T = \{0, 1, 2, \ldots, N_T\}$ with project life N_T, a set of suppliers $i \in I = \{1, \ldots, N_I\}$, a set of markets $k \in K = \{1, \ldots, N_K\}$, and their respective fixed locations (x_i, y_i) and (x_k, y_k). Suppliers have a time-variant raw material availability $\Psi_{i,t}$ and cost $Cost_{i,t}^{feed}$ while markets have their respective demand levels $\Phi_{k,t}$ and product sale values $Price_{k,t}^{prod}$. Given are also a set of potential processing facility sites $j \in J = \{1, \ldots, N_J\}$. At each site j we are able to construct a set of $n \in N_j$ potential units, which may also include modular units composed of $m \in M_n$ module types, in addition to conventional units.

Conventional units have a fixed cost $C_n^{f,conv}$ with an additional concave cost function given by a base cost $C_n^{s,conv}$ for a default size S_n^0 with an exponential scaling factor γ. Modular units feature a fixed cost $C_n^{f,mod}$ with a per-unit variable cost $C_{m,t}$ associated with the purchase of each unit type having size S_m.

We explicitly consider the conventional unit construction time τ_c as well as a modular setup time τ_m before each may start production. Modules may be repurposed or salvaged for a fraction ϕ_m of their original purchase value. Similarly, a salvage value fraction of ϕ_c can be ascribed to conventional facilities.

Transportation costs for raw material and product shipments are also incurred for connecting processing facilities with suppliers and markets, with fixed costs $TC_{i,j}^{f,feed}$,$TC_{j,k}^{f,prod}$ and variable costs TC_t^{feed},TC_t^{prod} for each, respectively. The transfer of modules between sites also incurs a per mile transportation cost $TC_{m,t}^{mod}$.

The main decisions in this problem include the locations of the potential facilities (x_j, y_j), production levels $p_{j,t}$ at each facility for each period of time, shipment quantities of raw material between suppliers and sites $f_{i,j,t}^{feed}$, shipment quantities of product between sites and customer markets $f_{j,k,t}^{prod}$, and Boolean decisions such as existence or absence of sites Y_j^{site}, of units within sites $Y_{j,n}^{unit}$, and conventional $Y_{j,n}^{conv}$ versus modular $Y_{j,n}^{mod}$ construction of units. Further Boolean decisions dictate existence or absence of a transportation links between suppliers and sites $Y_{i,j}^{feed}$, between sites and markets $Y_{j,k}^{prod}$, and between pairs of sites $Y_{j,j'}^{mod}$.

The facility locations determine distances between suppliers and sites $D_{i,j}$, sites and markets $D_{j,k}$ and between pairs of sites $D_{j,j'}$. At each facility, the production level is a function $f(r_{j,t}, S_{j,n,t}, \hat{x}_j)$ of the raw material supply $r_{j,t}$, the facility unit sizes $S_{j,n,t}$ and internal state variables $\hat{x}_j$. The unit sizes at each facility are in turn determined by either the time-invariant conventional unit size $S_{j,n}$ or the number of active modules of each type $n_{m,j,t}$. In each time period, modular units may increase or decrease in size due to module transfers between sites $nt_{m,j,j',t}$, new module purchases $np_{m,j,t}$, and module sales/salvage $ns_{m,j,t}$.

In this work, we assume that payments for capital investments take place at the beginning of the construction period, and that production commences only at the end of the construction period. In general, we also assume perfect knowledge of the demand and/or supply profiles for the multiperiod problem. That is, we do not explicitly treat uncertainty. However, note that accounting for uncertainty tends to move results in favor of modular designs.[14]

Formulation

The overall Generalized Disjunctive Programming (GDP) formulation is given by Equations (1)-(15).

$$
\begin{aligned}
\max NPV = & \sum_t \sum_k \sum_j Price_{k,t}^{prod} f_{j,k,t}^{prod} && \text{Product sales} \\
& + \sum_t \sum_j \sum_m C_{m,t} \phi_m ns_{m,j,t-\tau_m} && \text{Module sales} \\
& + \sum_j \sum_n C_{j,n}^{conv} \phi_c \delta_{N_T} && \text{Conventional salvage} \\
& - \sum_t \sum_j \sum_i Cost_{i,t}^{feed} f_{i,j,t}^{feed} && \text{Feed purchase} \\
& - \sum_t \sum_j \sum_i TC_t^{feed} f_{i,j,t}^{feed} D_{i,j} - \sum_i \sum_j TC_{i,j}^{f,feed} && \text{Feed transport} \\
& - \sum_t \sum_k \sum_j TC^{prod} f_{j,k,t} D_{j,k} - \sum_j \sum_k TC_{j,k}^{f,prod} && \text{Product transport} \\
& - \sum_j \sum_n C_{j,n}^{conv} && \text{Conventional investment} \\
& - \sum_t \sum_m C_{m,t} np_{m,j,t} && \text{Modular investment} \\
& - \sum_j \sum_n C_{j,n}^{f} && \text{Unit fixed cost} \\
& - \sum_t \sum_j \sum_{j' \in J \setminus j} \sum_m TC_{m,t}^{mod} nt_{m,j,j',t} D_{j,j'} && \text{Module transport}
\end{aligned}
\tag{1}
$$

The objective is to maximize the net present value of the processing network. Revenue arises from product sales at markets. Investment cost recovery also contributes to the profit via module sales and facility salvage value at the end of the project. The primary costs include raw material purchase costs as well as fixed and variable transportation costs and unit construction costs. Note that prices and costs are expressed here in time-variant terms which account for an assumed annual discount rate r such that $P(t) = P_0(1 + \frac{dr}{n_P})^{-t/n_P}$, where P_0 is the base price/cost and n_P is the number of time periods in a year. The discount factor δ_t can then be calculated as the ratio of present value to base value, $\delta_t = (1+\frac{dr}{n_P})^{-t/n_P}$.

Disjunctions

(a) Selection of sites

$$\begin{bmatrix} Y_j^{site} \text{ site selected} \\ r_{j,t} = \sum_i f_{i,j,t}^{feed} \quad \forall t \\ p_{j,t} = \sum_k f_{j,k,t}^{prod} \quad \forall t \\ p_{j,t} \leq f(r_{j,t}, S_{j,n,t}, \hat{x}_j) \quad \forall t \end{bmatrix} \veebar \begin{bmatrix} \neg Y_j^{site} \text{ site not selected} \\ \sum_t \sum_n S_{j,n,t} = 0 \\ \sum_t p_{j,t} = 0 \\ \sum_t \sum_m ns_{m,j,t} = 0 \end{bmatrix} \qquad \forall j \in J \tag{2}$$

(b) Selection of units

$$Y_j^{site} \implies \begin{bmatrix} Y_{j,n}^{unit} \text{ unit selected} \\ g_{j,n,t}(\hat{x}) \leq 0 \quad \forall t \end{bmatrix} \veebar \begin{bmatrix} \neg Y_{j,n}^{unit} \text{ unit not selected} \\ \sum_t \sum_{m \in M_n} n_{m,j,t} = 0 \\ \sum_t \sum_{m \in M_n} ns_{m,j,t} = 0 \\ \sum_t \sum_{m \in M_n} np_{m,j,t} = 0 \\ \sum_t \sum_{j' \in J \setminus j} \sum_{m \in M_n} nt_{m,j,j',t} = 0 \\ \sum_t S_{j,n,t} = 0 \quad \forall t \end{bmatrix} \qquad \forall n \in N_j, \forall j \in J \tag{3}$$

$$
Y^{unit}_{j,n} \implies \left(\begin{bmatrix} Y^{\mathrm{conv}}_{j,n} \text{ unit conventional} \\ S_{j,n,t} = S_{j,n} \quad \forall t \\ C^{conv}_{j,n} = C^{s,conv}_{n} \left(\dfrac{S_{j,n}}{S^{0}_{n}} \right)^{\gamma} \\ C^{f}_{j,n} = C^{f,conv}_{n} \\ \displaystyle\sum_{t} \sum_{m \in M_n} n_{m,j,t} = 0 \\ \displaystyle\sum_{t} \sum_{m \in M_n} ns_{m,j,t} = 0 \\ \displaystyle\sum_{t} \sum_{m \in M_n} np_{m,j,t} = 0 \\ \displaystyle\sum_{t} \sum_{j' \in J \setminus j} \sum_{m \in M_n} nt_{m,j,j',t} = 0 \end{bmatrix} \veebar \begin{bmatrix} Y^{\mathrm{mod}}_{j,n} \text{ unit modular} \\ S_{j,n,t} = \displaystyle\sum_{m \in M_n} S_m n_{m,j,t} \quad \forall t \\ C^{f}_{j,n} = C^{f,mod}_{n} \\ \begin{aligned} n_{m,j,t} = {} & n_{m,j,t-1} \\ & + \sum_{j' \in J \setminus j} \left(nt_{m,j',j,t-\tau_m} - nt_{m,j,j',t} \right) \\ & + np_{m,j,t-\tau_m} \\ & - ns_{m,j,t} \end{aligned} \qquad \forall m \in M_n \\ C^{f}_{j,n} = C^{f,mod}_{n} \end{bmatrix} \right) \quad \forall n \in N_j, j \in J \tag{4}
$$

(c) Selection of route

$$\begin{bmatrix} Y_{i,j}^{sup} \text{ supply route exists} \\ D_{i,j} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} \\ TC_{i,j}^{f,feed} = TF^{feed} \end{bmatrix} \veebar \begin{bmatrix} \neg Y_{i,j}^{sup} \text{ supply route absent} \\ f_{i,j,t}^{feed} = 0 \quad \forall t \\ TC_{i,j}^{f,feed} = 0 \end{bmatrix} \quad \forall i \in I, j \in J \tag{5}$$

$$\begin{bmatrix} Y_{j,k}^{mkt} \text{ product route exists} \\ D_{j,k} = \sqrt{(x_j - x_k)^2 + (y_j - y_k)^2} \\ TC_{j,k}^{f,prod} = TF^{prod} \end{bmatrix} \veebar \begin{bmatrix} \neg Y_{j,k}^{mkt} \text{ product route absent} \\ f_{j,k,t}^{prod} = 0 \quad \forall t \\ TC_{j,k}^{f,prod} = 0 \end{bmatrix} \quad \forall j \in J, k \in K \tag{6}$$

$$\begin{bmatrix} Y_{j,j'}^{move} \text{ module transfer possible} \\ D_{j,j'} = \sqrt{(x_j - x_{j'})^2 + (y_j - y_{j'})^2} \end{bmatrix} \veebar \begin{bmatrix} \neg Y_{j,j'}^{move} \text{ no module transfer} \\ \sum_t \sum_m nt_{m,j,j',t} = 0 \end{bmatrix} \quad \forall j \in J, j' \in J \setminus j \tag{7}$$

(d) Flow constraints

$$\sum_j f_{j,k,t}^{prod} \leq \Phi_{k,t} \quad \forall k, t \tag{8}$$

$$\sum_j f_{j,k,t}^{prod} \geq \Phi_{k,t} \quad \forall k, t \tag{9}$$

$$\sum_j f_{i,j,t}^{feed} \leq \Psi i, t \quad \forall i, t \tag{10}$$

(e) Logic constraints

$$Y_{j+1}^{site} \implies Y_j^{site} \quad \forall j \tag{11}$$

$$Y_j^{site} \iff \bigvee_i Y_{i,j}^{feed} \quad \forall j \tag{12}$$

$$Y_j^{site} \iff \bigvee_k Y_{j,k}^{prod} \quad \forall j \tag{13}$$

$$\neg Y_j^{site} \implies \bigwedge_{j'} \neg Y_{j,j'}^{move} \quad \forall j \tag{14}$$

$$\neg Y_j^{site} \implies \bigwedge_{j'} \neg Y_{j',j}^{move} \quad \forall j \tag{15}$$

Disjunction (2) determines whether a given potential facility exists or not. For an active site, the implication (3) gives a disjunction between whether each potential facility unit exists or not. This is logically equivalent to a nested disjunction in which the disjunction of (3) is a statement within the Y_j^{site} disjunct of disjunction (2). However, we present the logic here in this expanded form due to formatting limitations. Similarly, for an active unit $Y_{j,n}^{unit}$, the disjunction (4) governs selection of a conventional or modular unit construction. The disjunctions (5) and (6) determine the existence or absence of transport links between facilities to suppliers and markets, respectively. Finally, the disjunction (7) determines whether module transfer is possible between two site pairs.

General constraints are also present to enforce mass balances and to describe logical relations. Equation (8) enforces the maximum demand at each market. If market demand needs to be satisfied exactly, then equation (9) should be added to the formulation. On the supply side, equation (10) enforces availability of raw material at each supplier.

Equation (11) is a symmetry-breaking constraint for site activation, so that activation takes place sequentially among the set of potential sites. Note that in the special case of fixed site locations, this constraint should not be included.

The logic constraints (12)-(15) relate the activation of sites with the transportation links between sites and to suppliers and markets.

Induced linearity reformulation

A central tenet of modular design is *numbering up*[19] rather than scaling up equipment sizes in order to achieve efficiencies from standardization and to expedite production scale-up. As such, modular designs are characterized by the replacement of traditionally continuous decisions such as unit size with selection among a discrete set of alternatives.

This change from a continuous to discrete domain for key decision variables potentially can be exploited to improve solution performance. In the context of batch process design, it has been shown that mixed-integer linear reformulations are possible for nonlinear programs when continuous sizes are restricted to discrete values.[20] We identify this structure and its corresponding linear formulation as an *induced linearity* reformulation since the introduction of discrete variable domains induces linearity in nonlinear constraints of the original problem. The same induced linearity structure arises for modular design problems. Grossmann et al.[18] describes a reformulation for bilinear constraints of the form $h(x, v, w) \leq 0$ where

$$h_{q,r} = \alpha_{q,r} u_q v_r - \beta_{q,r} w_r \leq 0 \tag{16}$$

$\alpha_{q,r} \neq 0$ and $\beta_{q,r} \neq 0$ are nonzero coefficients. v_r and w_r are continuous state variables. u_q are the original continuous design variables, each of which for discrete sizes take a value from its corresponding discrete set $DS_q = \{d_{q,1}, d_{q,2}, \dots, d_{q,N_q}\}$. We designate these as *effectively discrete* variables. Introducing binary variables $z_{q,s}$ corresponding to each potential value of the effectively discrete variables, we can express these variables u_q using

$$u_q = \sum_s d_{q,s} z_{q,s} \tag{17}$$

$$\sum_s z_{q,s} = 1 \quad \forall q \tag{18}$$

From equations (16) and (17), we can obtain

$$g_{q,r} = \alpha_{q,r} \sum_{s} d_{q,s} z_{q,s} v_r - \beta_{q,r} w_r \leq 0 \tag{19}$$

Now, introducing auxiliary continuous variables $\hat{v}_{q,r,s}$, we can obtain a linear reformulation for the original equation (16). The auxiliary continuous variables are defined as

$$v_r = \sum_{s} \hat{v}_{q,r,s} \quad \forall r \in R_q, q \in Q \tag{20}$$

$$v_r^L z_{q,s} \leq \hat{v}_{q,r,s} \leq v_r^U z_{q,s} \quad \forall r \in R_q, s \in DS_q, q \in Q \tag{21}$$

where v_j^L and v_j^U are valid lower and upper bounds.

Using these auxiliary variables, it is now possible to rewrite equation (19) as the linear constraint (22).

$$\alpha_{q,r} \sum_{s} d_{q,s} \hat{v}_{q,r,s} - \beta_{q,r} w_r \leq 0 \quad r \in R_q, q \in Q \tag{22}$$

In that way the nonlinear inequality in (16) is replaced by the linear constraints (20)-(22).

Equivalence to GDP basic step with Hull Reformulation The induced linearity reformulation can be seen as taking advantage of additional information from disjunction (23), the choice among alternatives for each effectively discrete variable u_q.

$$\bigvee_{s} [u_q = d_{q,s}] \quad \forall q \in Q \tag{23}$$

By performing an improper basic step between the equation (16) and disjunction (23), and substituting $u_q = d_{q,s}$ within each disjunct of the result, we obtain the disjunction (24). The induced linearity reformulation is equivalent to the Hull Reformulation[17] applied to disjunction (24).

$$\bigvee_{s} [\alpha_{q,r} d_{q,s} v_r - \beta_{q,r} w_r \leq 0] \quad \forall q \in Q \tag{24}$$

Flexible reformulation options Note that the induced linearity reformulation may be applied to a nonlinear GDP model either before or after reformulation to an MINLP. The most common use is after the reformulating the GDP into an MINLP. The resulting induced linearity in the MILP involves more variables and inequalities than the original MINLP, but it is often an easier problem to solve due to the reliability of commercial MILP solvers compared to MINLP codes. For the special case that $w_r = 1$, $\forall r$, a smaller reformulation is possible.[18] However, as this case is less frequently observed, we do not discuss it in the context of this work. Note as well that when other nonlinear expressions exist in the original MINLP, the induced linearity transformation may result in an MINLP rather than an MILP; however, this new MINLP will have more linear structure than the original formulation.

If the induced linearity reformulation is applied before the nonlinear GDP reformulation to MINLP, then care needs to be taken to account for logical relations within the disjunctions. The result will still be a GDP, but with some nonlinear relationships replaced with linear ones. Several solution strategies are now possible for the reformulated GDP. It can be transformed as before into an MINLP, or it can be directly solved using logic-based decomposition algorithms.[21]

Automatic detection and reformulation

To make detection and reformulation of induced linearity structure more accessible to modelers, we also present an initial implementation of automatic detection and reformulation as a contributed package in the Pyomo algebraic modeling language.[22] Pyomo is an open-source python library that provides the ability to specify sets, variables, parameters, constraints, and higher-level modeling constructs such as disjunctions, along with interfaces to various solvers and meta-solvers. Pyomo offers users a familiar interface for specifying their optimization problem and allows developers of custom reformulations the ability to manipulate modeling components programmatically, allowing the development of automatic reformulations.

Given a Pyomo model, automated induced linearity detection and reformulation can be invoked with a single line of code:

```
TransformationFactory('contrib.induced_linearity').apply_to(model)
```

The induced linearity automatic reformulation consists of the following steps:

1. Search through all constraints to identify continuous variables u_q which are involved in linear constraints of the form $u_q = \sum_{s'} \tilde{d}_{s'}$, where $\tilde{d}_{s'}$ are discrete variables. These continuous variables are effectively discrete due to these linear relationships with discrete variables.

2. Determine for each effectively discrete variable u_q the set of valid values DS_q.

3. Find nonlinear expressions $h(u, v, w) \leq 0$ in which effectively discrete variables participate.

4. Apply the induced linearity reformulation, introducing auxiliary variables and constraints.

The code to support this capability is publicly available via the Pyomo Github repository, and comes packaged with a standard Pyomo developer installation. Once the induced linearity reformulation is applied to a Pyomo model, interfaces to Pyomo automatic reformulations from GDP to MINLP,[21] as well as commercially available MINLP codes (e.g. BARON, DICOPT, SCIP, see review[23]) and custom solvers such as MindtPy[24] and GDPopt[21] allow the user to solve the corresponding MINLP or GDP problem.

Capacity expansion case study

We present a capacity expansion case study to illustrate the applicability of the problem formulation on a wide range of modular process design considerations.

Single market capacity expansion

For the initial case, we consider a hypothetical case involving a single facility serving a market with time-variant demand $\Phi_{k,t}$ and a discount rate of $dr = 8\%$. For this simple case study, only one site exists, and it is fixed to be active. Therefore, disjunction (2) simplifies to enforce only the constraints in the left-hand-side disjunct. Similarly, we assume that the facility consists of only one processing unit (conventional) or one module type, which is always active. A monthly time period t is used with a project life of 10 years, $N_t = 120$. We do not consider in this case the supplier relationship, so feed considerations are neglected. We assume that one market exists with distance 0 from the site. Therefore, transportation-related costs are also zero. Finally, we do not need to satisfy demand exactly here, so equation (9) is excluded from the model.

With these assumptions, the key decision involves selection of production levels at each time point. For the conventional case, a capacity must be selected. For the modular case, the number of modules to be purchased or sold in each time period must be determined.

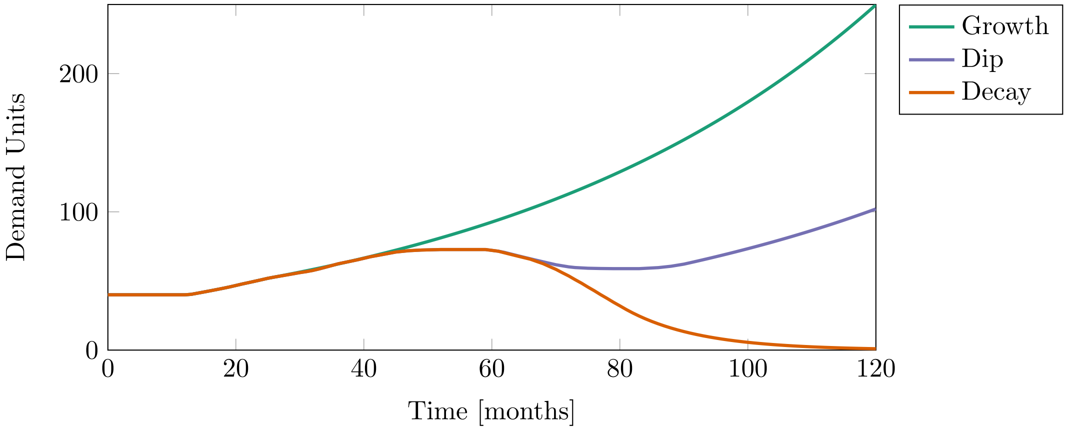


Figure 1: Demand profile scenarios for the capacity expansion case study.

In this illustration, we assume that demand begins at a value of 40 units per month in the first time period $t = 0$, and we evaluate three different demand profile scenarios: a "Growth" scenario in which demand increases monotonically to just under 250 units per month by the end of the project life, a "Dip" scenario in which demand experiences a brief dip before continuing to grow to around 100 units per month by the end of the project life, and a "Decay" scenario in which demand subsides after an initial growth period (see Fig. 1). The "Growth" scenario may be viewed as the nominal outcome, with the two adverse realizations representing potential downside market risks. The sale value of the product is \$583 per unit, adjusted over time by the discount rate. For this example, we assume that a process with production capacity of 25 units per month is available at a cost of \$1 million. For the conventional case, we assume a scaling factor of $\gamma = 0.6$ for a larger capacity facility. For the modular case, a single module type with a capacity of 25 units per month is available at a cost of \$1 million per module.

The problem is formulated for each demand scenario as a nonlinear GDP with 488 variables, 486 constraints, and 1 disjunction. After fixing the logical decision in the GDP model for the conventional case, an NLP is obtained with 486 variables and 259 constraints. The NLP is solved in less than 1 second by using the BARON 18.5.8 solver via GAMS 25.1.3. Enforcement of the modular choice leads to a linear GDP. After automatic application of the hull reformulation, the resulting MILP has 972 variables (363 integer) and 1337 constraints. Gurobi solves this MILP in less than 1 second.

From the solution of their respective formulations, we obtain a NPV profit of \$2.47 million for the modular facility compared to \$3.59 million for the conventional facility in the nominal "Growth" scenario. Under the "Dip" demand scenario net present values of \$1.99 and \$2.10 million are obtained for the modular and conventional cases, respectively. Finally, the "Decay" scenario features a NPV profit of \$0.85 and \$0.72 million for the respective modular and conventional constructions. These results show that for a single site capacity expansion in which demand may be reliably forecast, competing against conventional economies of scale presents a significant challenge to modular designs in the nominal case. Even in the case of market downturns, as in the “Dip” scenario, the conventional design slightly outperforms the modular design. However, in the face of adverse market conditions or transient demand, as in the “Decay” scenario, the flexibility of the modular alternative gives it the advantage.

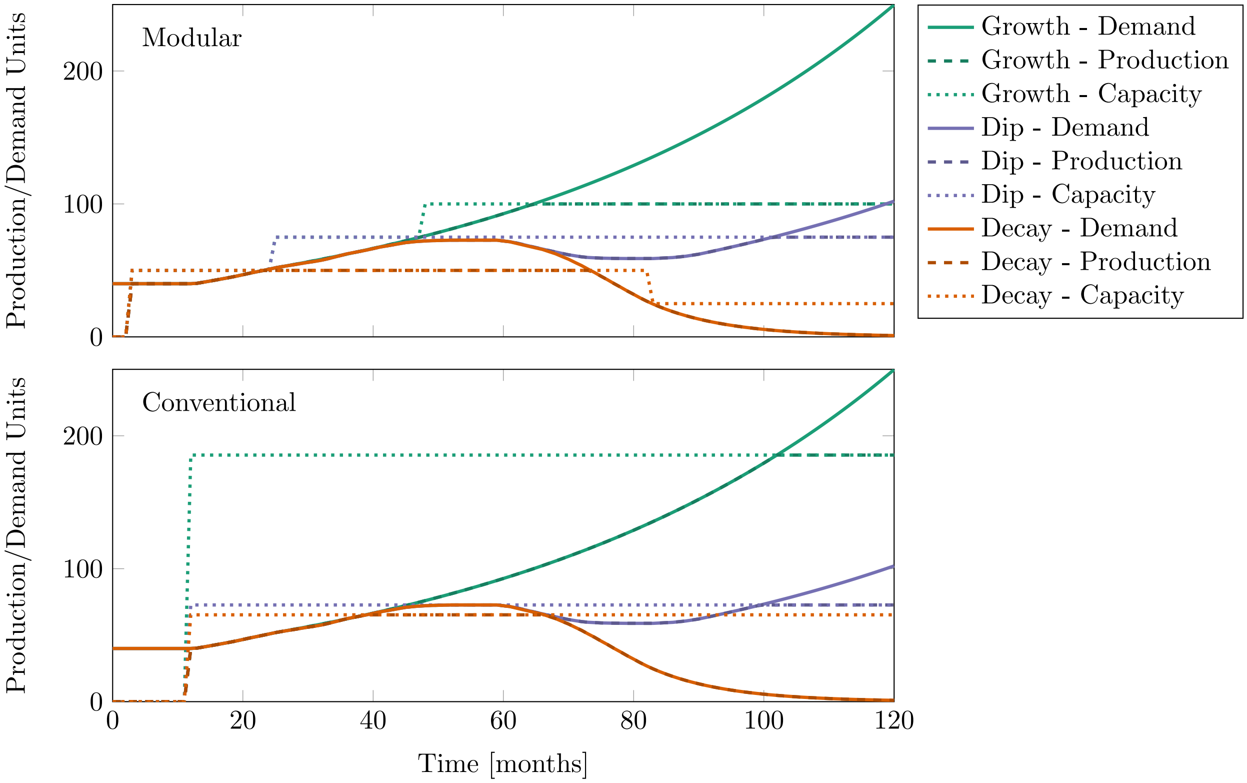


Figure 2: Capacity expansion case study production and capacity under Growth, Dip, and Decay demand profile scenarios.

Fig. 2 illustrates how production and capacity change in relation to demand over the project lifetime for both the modular and conventional facilities in each of the three demand scenarios. Based on assumed construction times, the modular facility begins production in the third month, before the conventional facility completes construction in the 12th month. Initial modular production yields a present value revenue of $210 thousand before the completion of the conventional facility, reflecting the time-to-market advantage afforded by faster module assembly and engineering. This accounts for 4% to 9% of total modular present value revenue, depending on the demand scenario. Subsequent module purchases are made in the “Growth” and “Dip” scenarios to accommodate increased demand. In the “Decay” scenario, a production module is sold before the end of the project life, freeing up unneeded capacity. Note that in all scenarios, the optimization chooses not to fully satisfy demand, as the marginal revenue does not outweigh the marginal cost of purchasing an additional production module, even after accounting for salvage value.

The conventional facility features a constant capacity of 186, 73, or 65 units per month for the Growth, Dip, and Decay scenarios, respectively, available following construction for the remainder of the project lifetime. Note that the production profile for the conventional case assumes the ability to operate the facility at production levels well below nameplate capacity. In practice, this may introduce inefficiencies or complexities, but for simplicity, we do not account for these effects in this example.

Comparing the modular and conventional production profiles, we notice that the conventional facility exploits economies of scale to provide comparable or higher production capacity across each scenario. Table 1 gives the revenue and cost breakdowns for each scenario, where it can be observed that construction costs for the conventional facility are consistently cheaper, even given its capacity advantage. Economies of mass production[12] has been proposed as a means to counter this advantage in scaling up; however, it is not considered in this work.

From the “Dip” scenario results, hints of modular advantages can be seen even though the conventional construction yields a better overall profit. The modular present value revenue of \$4.14 million exceeds the conventional revenue of \$3.90 million, demonstrating the advantage of faster time-to-market in a modular construction given similar production capacities. The salvage value of the modular facility is also higher, at \$840 thousand compared to \$10 thousand for the conventional facility. This is due to the higher modular salvage value recovery fraction, assumed to be $\phi_m = 0.30$ versus $\phi_c = 0.05$ for the conventional plant. As production modules are easier to re-purpose, perhaps by relocating them to another project site, they retain more of their value at the end of the project life. Furthermore, modules may be sold prior to the project conclusion to enable faster value recapture. The combination of these factors allows the modular construction to have a superior overall present value profit in the "Decay" scenario.

Table 1: Capacity expansion case profit and cost factors

	Growth		Dip		Decay	
Factor [million $]	Conv.	Modular	Conv.	Modular	Conv.	Modular
Construction cost	3.33	3.96	1.90	2.99	1.78	2.00
Market Revenue	6.77	5.31	3.90	4.14	2.41	2.28
Salvage value	0.16	1.12	0.01	0.84	0.01	0.57
Total NPV	3.59	2.47	2.10	1.99	0.72	0.85

Despite the conventional facility's capacity advantage, much of it ends up under-utilized, particularly in the "Growth" scenario. This over-sizing can present a liability. Note that the model assumes perfect knowledge of future demand profiles. In the presence of forecasting uncertainty, the modular facility would be more capable of quickly adapting, giving it a further advantage compared to the conventional case. To illustrate this more clearly, we optimize a two-stage stochastic programming (TSSP) formulation for the conventional case, considering the facility sizing investment decision at the first stage and the production profile at the second stage. Given the set of scenarios $\theta \in \{\text{Growth}, \text{Dip}, \text{Decay}\}$, we assume scenario weights of $wt_{\text{Growth}} = 50\%$, $wt_{\text{Dip}} = 25\%$, and $wt_{\text{Decay}} = 25\%$. We include a copy of the deterministic problem for each scenario, including the non-anticipativity constraint Eqn. (25).

$$S_{j,n}^{\text{Growth}} = S_{j,n}^{\text{Dip}} = S_{j,n}^{\text{Decay}} \tag{25}$$

The profit for each scenario is given by NPV^{θ}, with the overall objective the maximization of Eqn. (26).

$$\max NPV^{*} = \sum_{\theta} wt_{\theta} NPV^{\theta} \tag{26}$$

The combined TSSP formulation is a NLP with 365 variables and 727 constraints. BARON 18.5.8 solves the problem in less than 1 second.

The resulting solution yields an expected NPV profit of $2.03 million, with a production

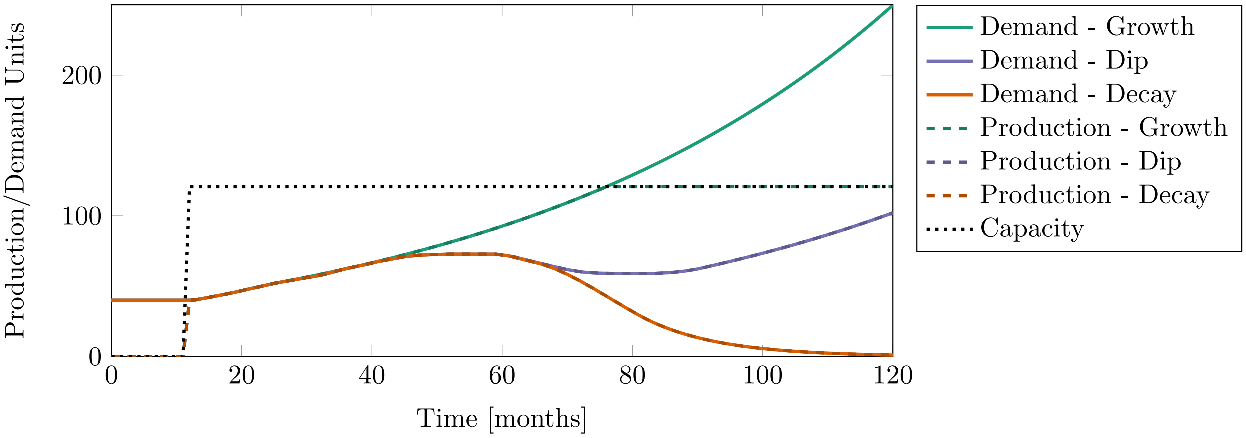


Figure 3: Capacity expansion case study stochastic programming conventional facility capacity and scenario production profiles.

capacity of 121 units per month (see Fig. 3). This capacity value is significantly lower than the nominal value of 186 units per month in the "Growth" scenario, so the solution hedges against purchasing excess capacity; however, this is still sufficient to recover 84% of the nominal revenue if the "Growth" scenario is realized. The cost breakdown for the TSSP solution can be found in Table 2. Note that by assuming that all modular design decisions are made in the second stage, the equivalent modular expected net present value is $1.95 million, within 4% of the conventional case. If the more conservative scenario weighting is used such that all scenarios have the same weight, the resulting expected net present value will be $1.72 million and $1.77 million for the conventional and modular case, respectively. Under this risk-adverse assumption, the modular option outperforms the conventional alternative.

Table 2: Capacity expansion case study stochastic programming results.

Factor [million $]	Expected Value	Growth	Dip	Decay
Construction cost	2.57	-	-	-
Market Revenue	4.48	5.68	4.07	2.50
Salvage value	0.12	-	-	-
Total NPV	2.03	3.22	1.62	0.04

Even with this simple case study, we can gain insights into the trade-offs between modular and conventional chemical facility construction. Modular construction is most relevant in the presence of volatile markets where it is important to gain an early time-to-market advantage and to hedge against downside risks. Modular construction also benefits when a secondary market exists so that modules better retain their value for re-purposing or resale past the project lifetime. When these conditions do not exist, then it can be difficult to recommend a modular design due to the cost scaling advantages of conventional economies of scale.

Multiple markets with module relocation

Next, we consider a given product with multiple markets and potential facility sites. In this case study, the objective is to minimize total system cost while enforcing satisfaction of demand at each of five customer markets, subject to facility construction costs as well as transportation costs. Here, rather than allowing the sale of modules, we allow for their relocation between production sites, subject to a transportation cost relative to the distance between the sites. Relocated modules also incur a transport time of 3 months, during which they do not contribute to site production. The capital cost for conventional facilities is assumed to be \$1 million for a production capacity of 10 units per month, with a scaling factor of $\gamma = 0.6$ for a larger capacity facility. For modular facilities, a single module type is available with a capacity of 10 production units per month, costing \$1 million each. The transportation cost per unit of product per mile is assumed to be \$17 thousand. Transportation costs for module relocation are assumed to be \$1 million per mile. Note that these values are given to illustrate a scenario in which transportation costs are dominant.

The markets have fixed locations on a two-dimensional grid as illustrated in Fig. 4. The axes denote miles from an origin point at (0, 0). The demand profiles corresponding to each market, as well as the total demand, are given in Fig. 5.

Note that peaks in the demand profiles for each market location occur at different times from each other, with markets 1 and 2 initially providing the most demand, but in later time periods, demand has shifted to market 3, then markets 4 and 5. Since demand must be fulfilled exactly in this case study, we do not consider the product sale price, letting $Price^{prod}_{k,t} = 0$. We also do not consider supplier relationships here, assuming that the raw

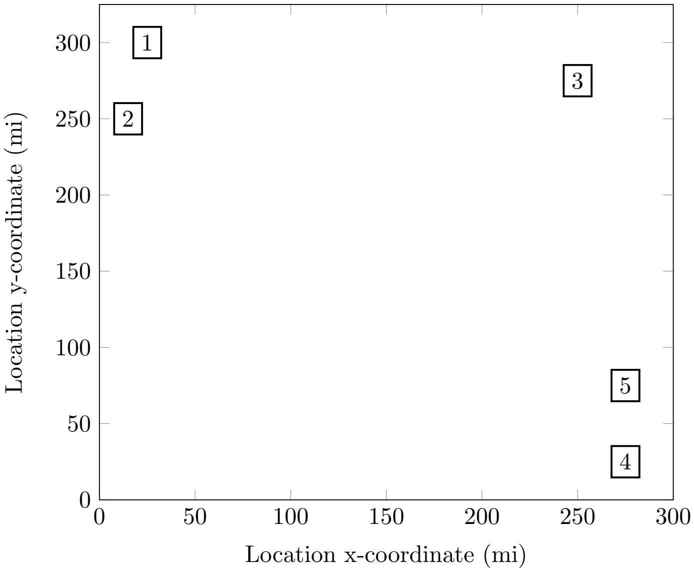


Figure 4: Market locations on a 2-D spatial grid.

material is readily available at any potential facility, as in the case of air separation.

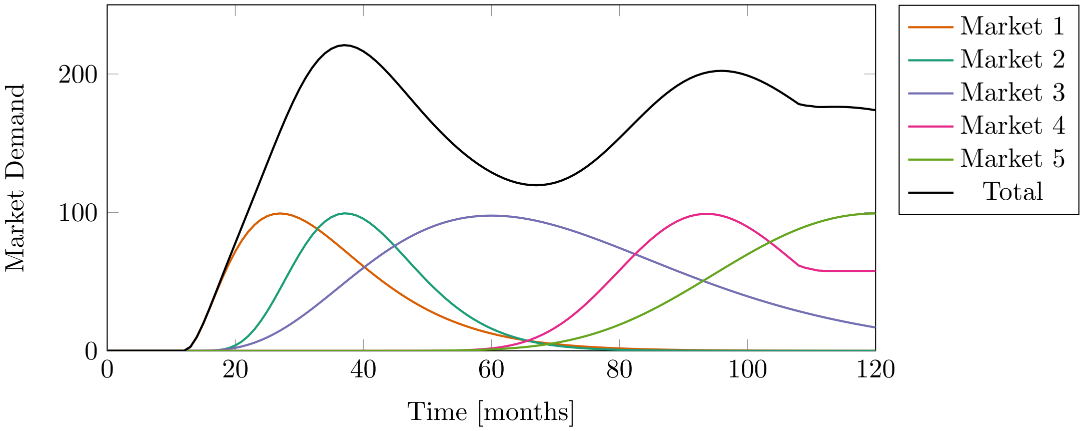
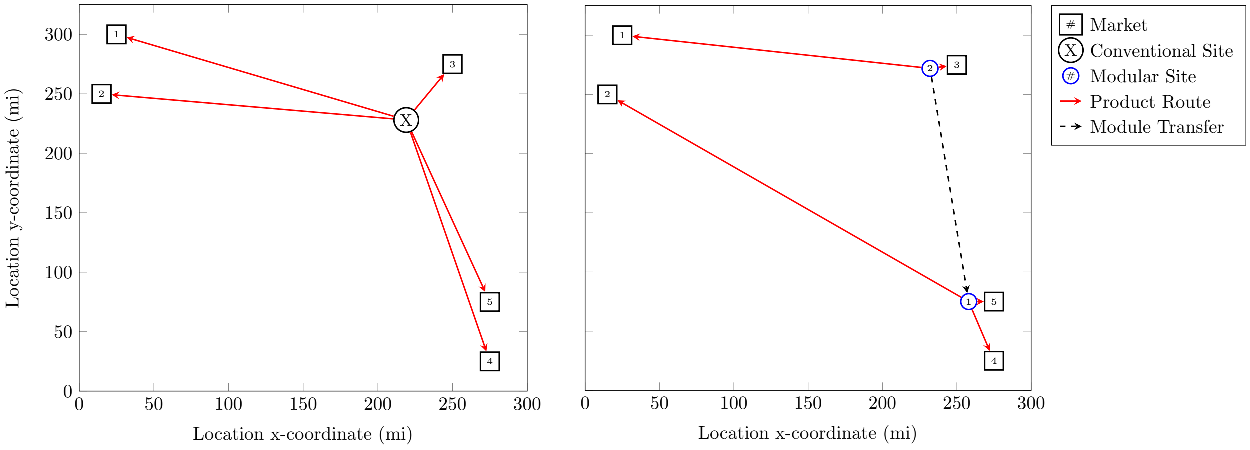


Figure 5: Demand over time at individual markets and in total.

In the conventional single-facility case, there is one site available, which necessarily is linked to each of the market locations. The location of the site is set by the result of the optimization to minimize costs. With these restrictions, the general GDP formulation simplifies to an NLP, with 735 variables and 853 constraints. Solving with IPOPT, a solution is obtained in less than 1 second.

The conventional facility with capacity 221 units per month is constructed at coordinate location (219, 228) at a cost of $4.2 million. Adding in product transportation costs of $42 million, the total system cost is $46.2 million over the project scope. As seen in Fig. 6b, the optimal solution chooses to situate the conventional facility slightly off-center towards market 3. This minimizes the transportation costs, which constitute the largest cost factor in the case study.



(a) Conventional solution

(b) Modular solution

Figure 6: Multiple markets case study solution. Conventional facility site is indicated by X. Numbered circles are modular facility sites. Solid lines indicate product shipment routes. Dashed line indicates module transfers from one site to another.

The modular problem is formulated as a nonlinear GDP with 2224 variables (718 integer), 1387 constraints, and 13 disjunctions. This is then automatically transformed to an MINLP using the big-M reformulation and solved in 7 seconds using the DICOPT solver[25,26] via the Pyomo-GAMS solver interface and GAMS version 25.1.3.[27]

The distributed, modular case chooses to use both potential facility sites, with site 1 located at coordinate (270, 75) and site 2 at (246, 272). Site 1 supplies markets 2, 4, and 5, while site 2 supplies markets 1 and 3. Fig. 6b shows the site locations with respect to the market sites, as well as the transportation links that exist. The placement of site 2 near market 3 rather than markets 1 and 2 may appear to be an odd decision at first, but examining the cumulative demand based on Fig. 5 reveals that market 3 is dominant over the project lifetime. Therefore, placement of a facility close to that market is more advantageous than one close to markets 1 and 2.

The total modular system cost is $45.2 million, of which product transportation accounts for $26.1 million, module purchases $11.9 million, and module transfers $7.2 million. Table 3 gives a comparison between the modular and conventional system costs, showing the two to have very similar total cost. However, it is clear that the conventional solution in this case has lower investment cost than the modular solution, but the modular design has lower transportation cost. Fig. 7 shows the capacity and production levels at each site over time.

Table 3: Multiple market case profit and cost factors

Cost [million $]	Conventional	Modular
Construction	4.2	11.9
Product Transportation	42.0	26.1
Module Transfer	-	7.2
Total NPV	45.7	45.2

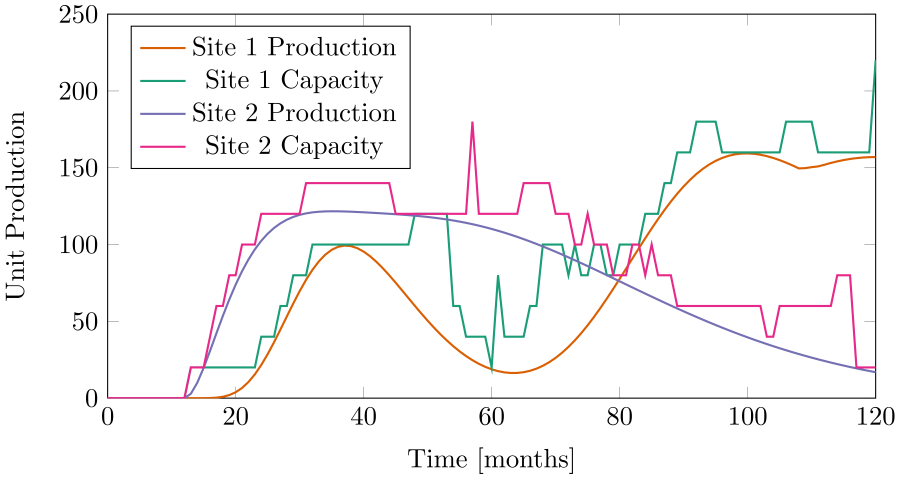


Figure 7: Distributed modular capacity expansion production and demand

As seen in Fig. 7, the capacity of site 2 grows over time in the initial capacity expansion phase, but then as demand falls in markets 1 and 3, modules are transferred from site 2 to site 1, increasing site 1 production closer to the more active markets. These transfers in the modular case allow production to be more easily moved into areas of demand growth, preserving module production value subject to transportation costs.

Fig. 7 also offers an indication that a better mathematical solution may exist. At times, additional capacity is added at a site, whether via module purchase or module transfer, even when it is not needed to support the required production level. The solution for the modular network in this case study was obtained using the DICOPT solver, which requires convexity to guarantee mathematical optimality. Used on a non-convex problem such as this case study, it may find good quality solutions such as the one presented, but cannot guarantee the globally optimal solution. Application of the BARON[28] global optimization solver did not yield a feasible solution after 10 minutes.

Quarterly time periods

In order to reduce the problem complexity and seek an improved modular solution, we aggregate the time periods for the distributed modular case into quarters rather than months. As a result, we reduce the problem size in the time domain. Consequently, we now consider up to a three-site modular design. Aggregation of time periods slightly modifies the discount rate compounding, as it takes place quarterly instead of monthly; however, this is not likely to significantly affect the result. Indeed, the conventional facility NPV using quarterly time periods improves slightly to $43.1 million, but is otherwise unchanged (see Table 4). The topology of the demand markets also remains the same from the previous case study (see Fig. 4).

Table 4: Multiple market case profit and cost factors, quarterly time periods

Cost [million $]	Conventional	Modular
Construction	4.2	11.6
Product Transportation	38.9	6.6
Module Transfer	-	2.2
Total NPV	43.1	20.4

The time-aggregated modular formulation now involves 1314 variables (486 integer), 672 constraints, and 21 disjunctions. Using BARON, the best solution obtained after 10 minutes is presented. The total modular system cost is $20.4 million, with product transportation costs accounting for $6.6 million, module purchases $11.6 million, and module transfers $2.2 million. Site 1 is situated at (275, 30) serving markets 1, 4, and 5. Site 2 is at (246, 273) with product shipments to markets 2 and 3. Finally, site 3 is located at (25, 295), serving markets 1, 2, and 4. The resulting network structure is shown in Fig. 8.

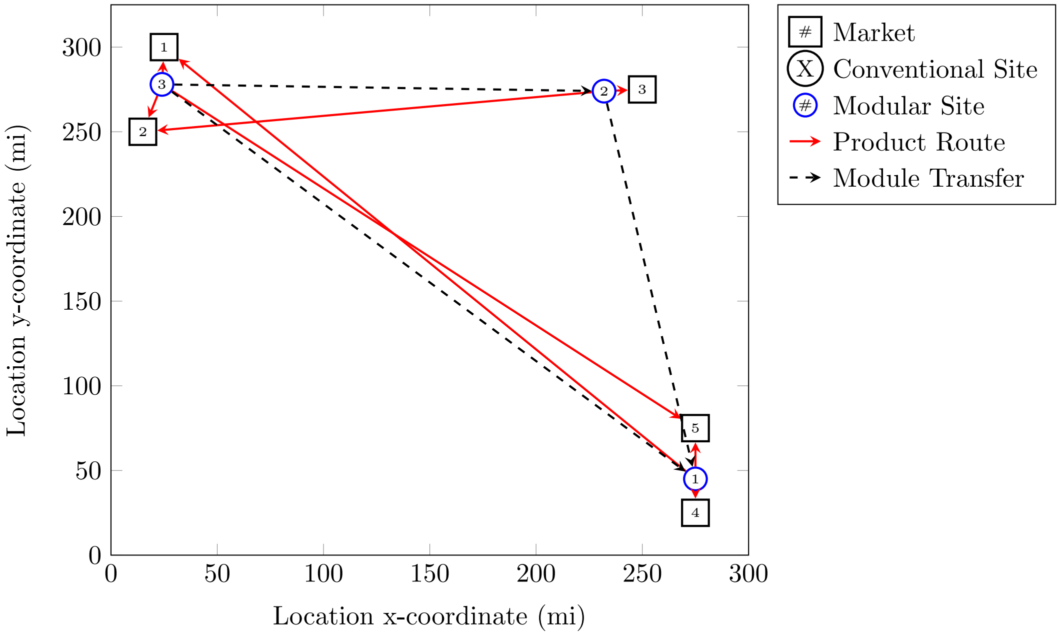


Figure 8: Distributed modular capacity expansion production and demand, quarterly time periods

Already, we see an improvement through the use of a more rigorous search technique. The transportation costs are reduced, and although module purchase costs remain higher, the module transfer costs are also decreased. Fig. 8 shows that the network behavior is more complex, with more interconnections formed. In order to simplify the analysis, we examine the quarterly shipments from each of the three modular facilities separately.

Fig. 9 plots the quarterly shipments corresponding to each facility. Facility 1 primarily serves markets 4 and 5. However, earlier on, it also contributes to demand satisfaction at market 1. Similarly, site 2 primarily serves market 3, with additional minor contributions to markets 2. Site 3 is the simplest, serving exclusively markets 1 and 2.

The minor contributions can be seen as backfill production provided by a more distant modular facility, while the more proximal facility expands its capacity to meet the rising demands. Fig. 10 shows that near the beginning, site 3 production is capacity-limited as it is more cost-effective to ship in product from sites 1 and 2 to satisfy marginal demand above

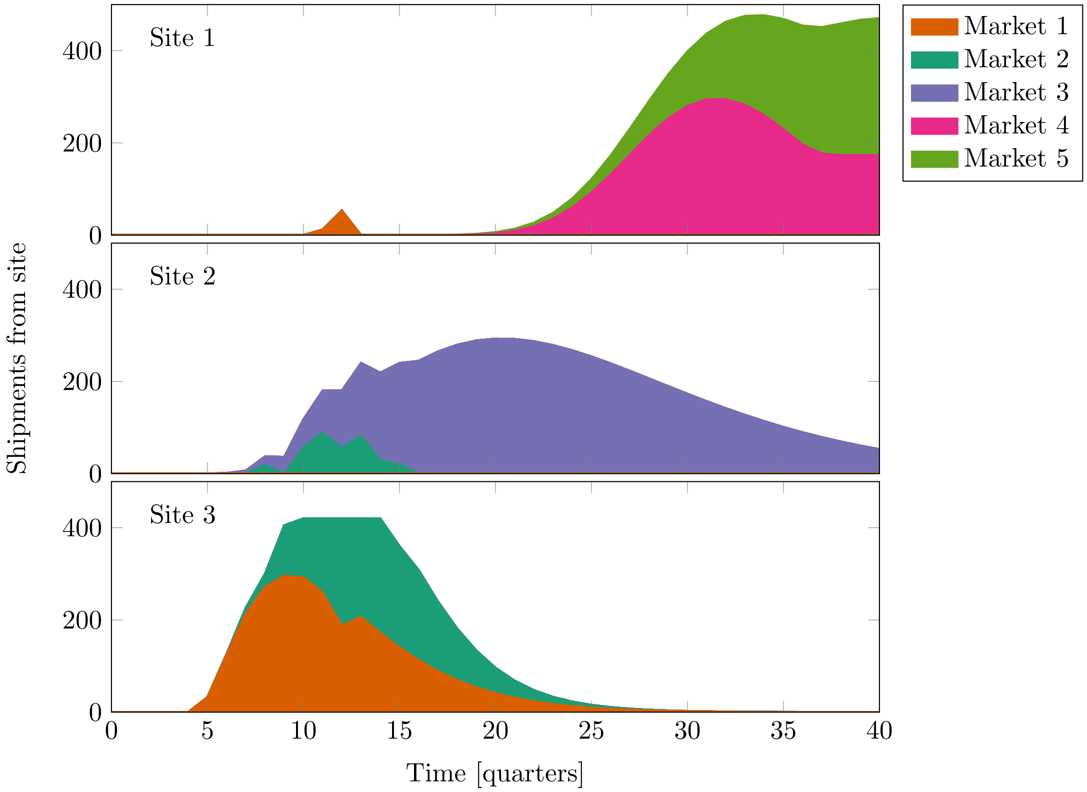


Figure 9: Quarterly shipment quantities from modular sites

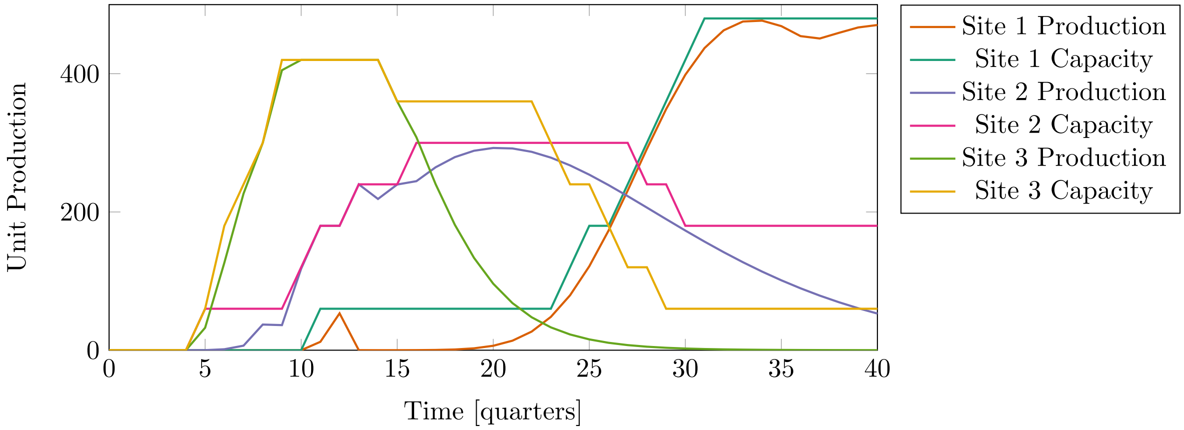


Figure 10: Quarterly production and capacity for modular sites

site 3 capacity than it is to purchase additional modules. In later time periods, demand from markets 1 and 2 diminishes, so modules are relocated from site 3 to aid capacity expansion in sites 1 and 2, alleviating the need for new module purchases. Therefore, this case study demonstrates the value that module relocation affords in terms of flexibility in the presence of significant transportation costs and demand variability, and the ability of the presented design framework to support these considerations.

Bioethanol case study

We next present a bioethanol case study, adapted from literature,[29] to illustrate adaptation and use of the proposed general formulation. The original case study considers a continuous location-allocation model to compare centralized versus distributed process network arrangements.[29] In this work, we propose a multi-period extension in which the potential site locations are predetermined. This case study attempts to find the minimum cost network layout and production allocation in order to satisfy market demands. To preserve feasibility given the 12 month construction time required for the conventional facility, the demands are modified so that they are zero until month 12. After this initial period, the demand levels at each market remain constant. However, we introduce variability in the availability of raw material from various suppliers. Some suppliers initially have no raw material availability, but then come online during the project life span, while others may go offline after a fixed amount of time. This time-variance of the suppliers is known and fixed ahead of time. The spatial layout of the processing network is shown in Fig. 11.

We are given a base process with a cost of \$268.4 million, capable of producing 120 thousand tons of bioethanol product per year.[29] For the conventional case, we allow for continuous facility sizes at each site with a scaling factor of $\gamma = 0.7$. For the modular case, we take this base cost and production capacity as the default module, with additional production capacity available at each site through additional module purchases. As given

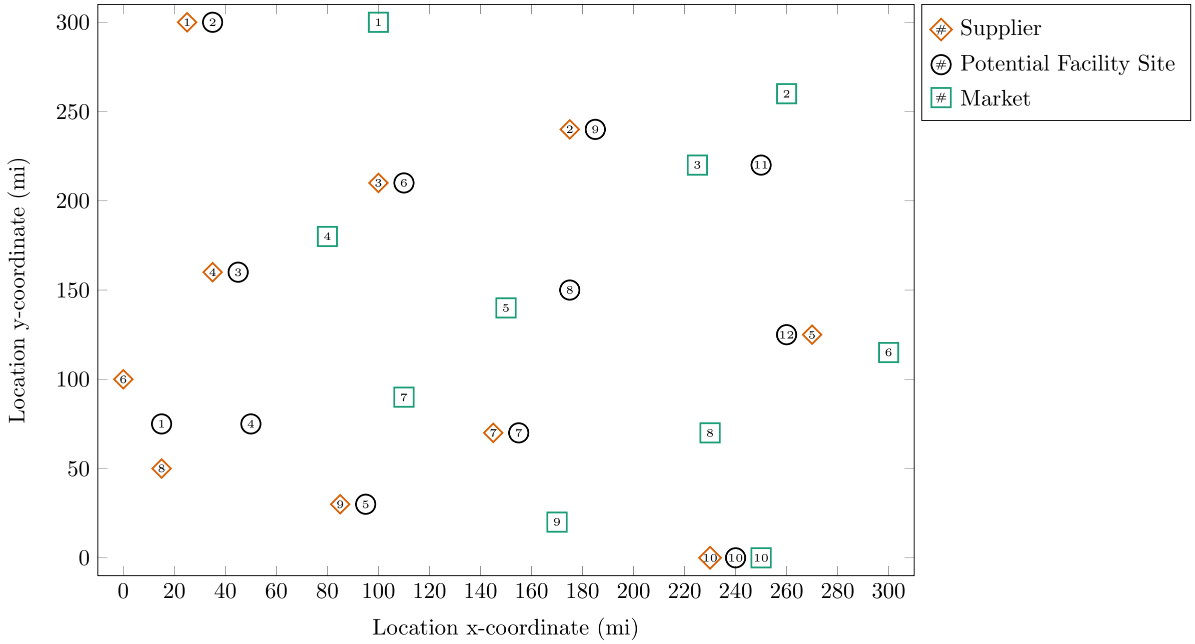


Figure 11: Bioethanol production processing network site layout. Suppliers are indicated by diamonds. Potential processing facilities are indicated by circles. Customer markets are indicated by rectangles.

in the literature example, conversion of raw material to product is 0.26 on a mass basis. With fixed supplier, facility, and market locations, the transportation distances for this case study are predetermined. Therefore, the modular case is a linear GDP problem which is automatically reformulated using big-M as an MILP involving 36,768 variables (504 binary, 4,320 integer) and 13,388 constraints.

After 20 minutes, Gurobi yields a solution of $4.9 billion with an optimality gap of 0.7%. The topology of the modular process network alternative is shown in Fig. 12, involving 11 modules at 10 processing sites. Each selected site is assigned one module, except for site 1, which hosts two modules after a purchase in the 60th month.

The conventional case is a nonconvex MINLP due to the concave cost functions for each of the sites. It involves 36792 variables (504 binary, 4320 integer) and 10676 constraints (24 nonlinear). This problem was solved using SCIP[30] via the Pyomo-GAMS interface using GAMS 25.1.3. After 441 seconds, a solution was obtained with an optimality gap of 0.61%. The conventional process network topology involving 10 plants is displayed in Fig. 13.

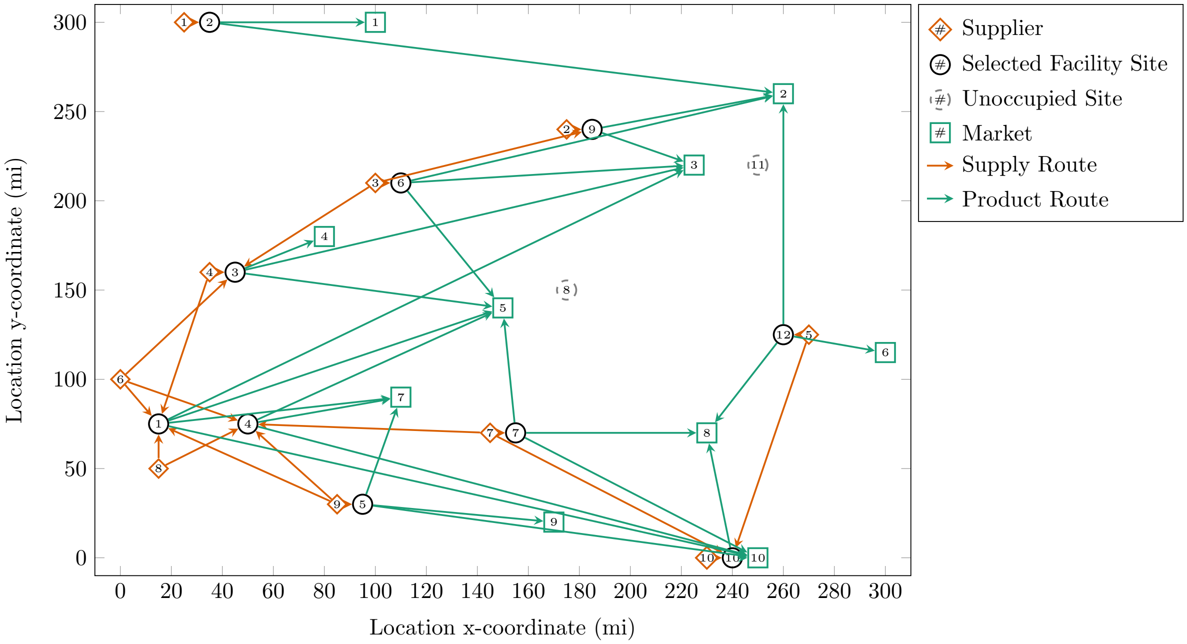


Figure 12: Modular processing network topology for bioethanol production

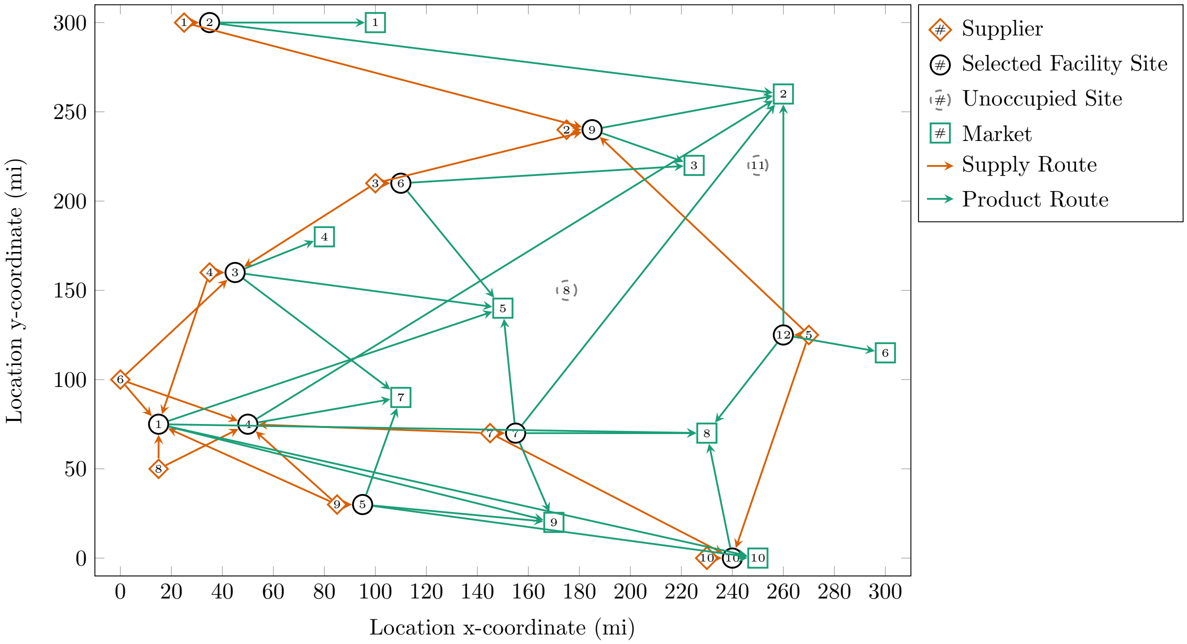


Figure 13: Conventional processing network topology for bioethanol production

In this case study, the modular alternative gives a better solution than the conventional option due to a combination of effects (see Table 5). The sum total of these differences leads to a present value cost of \$4.9 billion versus \$4.1 billion for the conventional and modular cases, respectively. First, note that the construction costs for the conventional and modular cases are similar in this example. Due to the dominance of raw material transportation costs in the case study, a centralized processing facility is not competitive. Therefore, even in the conventional case, multiple smaller distributed facilities need to be constructed near different suppliers to minimize raw material transportation distances. This implicit size restriction blunts the impact of traditional economies of scale and allows the modular construction, while still more expensive, to be more competitive. Second, we assume in this case study, as in the capacity expansion example, that the salvage value recovery fractions for the conventional and modular facilities are $\phi_c = 0.05$ and $\phi_m = 0.30$, respectively. This allows the modular design to retain more of its value at the end of the project, accounting for a \$705 million difference in the network present value. The ability to relocate modular production to take advantage of transient supply or demand shifts can also be seen, with the modular network allocating production closer to suppliers to reduce raw material transportation costs by \$372 million compared to the conventional case. In this example, the effect even outweighs the conventional advantage in construction cost. The modular case also allows for delayed capital expenditure due to faster assembly time. This decreases the NPV cost of purchasing a modular system. However, this has only a minor impact, accounting for a \$21 million difference in the present value construction cost.

Table 5: Bioethanol production process network cost factors

Factor [million $]	Conventional	Modular
Construction cost	2684	2931
Salvage value	126	831
Raw material transport cost	2225	1853
Product transport cost	98	114
Total net cost	4881	4067

Modular HENS case study

The general formulation also supports investigation of subsystems for potential facilities. There has been recent[31] as well as historical[3] interest in the area of modular heat exchanger networks. We present a modular heat exchanger network synthesis (HENS) case study. Here, we are given a set of 2 hot and 2 cold streams within a potential site. The units involved are potential matches between the hot and cold streams, which can be activated or deactivated by $Y_{j,n}^{unit}$. A 2-stage superstructure is used for the HENS.[32] Furthermore, each potential match may be satisfied by a conventional exchanger or modular units. With this case study, we also illustrate how induced linearity can be applied to the linearization of the overall heat transfer calculation, equation (27), given discrete module sizes.

The problem data is adapted from literature,[32] with stream inlet/outlet temperatures and heat content given in Table 6. Process steam and cooling water are available at a cost of \$80 per kW and \$20 per kW, respectively. The overall heat transfer coefficient U is 1.2 $kWm^{-1}K^{-1}$ for heaters and 0.8 for all other exchangers. Conventional heat exchangers are available with a fixed investment cost of \$2000, with a subsequent concave variable investment cost with a factor of $C_n^{s,conv} = 1000$ (1200 for heaters), base size of $S_n^0 = 1$, and exponential scaling factor of $\gamma = 0.6$. For modular exchangers, fixed module sizes of 10, 50, and 100 square meters are available. We assume no fixed investment cost for modular exchangers as a proxy for their relative ease of setup, but the subsequent module unit cost for each size is determined by a higher cost factor of 1100 (1300 for heaters). Since these sizes are determined before the optimization, the corresponding per-module costs are pre-calculated for each size.

In this case study, we consider only a single time period, assuming that the network will not change over the design period. We also assume that appropriate partial bypasses are in place such that the overall heat transfer calculation, equation (27), does not need to be satisfied exactly. Therefore, the exchanger area at the match provides an upper limit to the amount of heat transfer for a given temperature driving force, but can have excess capacity.

Table 6: Stream information for HENS case study

Stream	T_{in} [K]	T_{out} [K]	FC_p [kW/K]
H1	443	333	30
H2	423	303	15
C1	293	408	20
C2	353	413	40
steam	450	450	
water	293	313	

This is important to allow for feasibility of the modular case, since only discrete sizes are possible.

$$\mathcal{Q} \leq UA\,\text{LMTD} \tag{27}$$

For a modular heat exchanger network design, the available heat exchanger surface area at a match between a hot stream and cold stream will be the sum of the discrete areas for all modules installed for the match. Therefore, the heat exchange surface area A is *effectively discrete*, allowing the application of induced linearity. A therefore substitutes in for u_q in equation (17), with DS_q from section equal to the discrete set of possible heat exchange surface areas at a given match q:

$$A_q = \sum_m d_{q,m} z_{q,m} \tag{28}$$

$$\sum_m z_{q,m} = 1 \tag{29}$$

We can use equation (28) to induce linearity in the bilinear overall heat transfer calculation, equation (27). Here, we introduce auxiliary variables for the log mean temperature difference at each stream match LMTD_q, corresponding to equations (20) and (21):

$$\text{LMTD}_q = \sum_m \text{LMTD}_{q,m} \tag{30}$$

$$\text{LMTD}_q^L z_{q,m} \leq \text{LMTD}_{q,m} \leq \text{LMTD}_q^U z_{q,m} \tag{31}$$

Adding these constraints to the formulation, we can therefore rewrite equation (27) for a given match q as the linear inequality:

$$\mathcal{Q}_q \leq U \sum_m d_{q,m} \text{LMTD}_{q,m} \tag{32}$$

We consider a variety of different cases: use of conventional exchangers, use of a single module type, use of multiple module types, and use of a hybrid network with both conventional and modular exchangers. In the conventional case, we examine the classic HENS design with no modular exchangers. The single module case produces a heat exchanger network design using only one standardized exchanger module size, in pursuit of increased module standardization and economies of mass production. For the multiple module case, we allow a mix of each possible module size. Finally, for the hybrid case, we allow each match to be either implemented with a conventional exchanger or with modular exchangers. The corresponding GDP model sizes for each case is shown in Table 7. Details for both the base formulation and induced linear reformulation (ILR) of each modular case are shown.

Table 7: HENS case study GDP problem sizes

	Variables	Binary	Integer	Constraints	Disjunctions
Conventional	208	24	0	250	32
Single module type	307	27	96	265	33
Single module ILR	3071	2107	0	6006	33
Multiple module types	108	36	36	262	44
Multiple module ILR	1108	324	0	1486	32
Hybrid	120	48	36	322	44
Hybrid ILR	1008	348	0	2122	44

After formulation, Pyomo offers multiple options for solving the GDP model. We apply three solvers with the HENS case study: GDPopt,[21] DICOPT,[25] and BARON.[28] For DICOPT and BARON, a big-M automatic reformulation[17] is applied from GDP to MINLP prior to sending the model to the solver. For the latter two solvers, a five minute (300 second) time limit was set, and for GDPopt, a Python implementation of the Logic-based Outer Approximation algorithm,[33] no run exceeded the same time limit. The total system cost results are given in Table 8, with the final lower bound values for BARON also reported.

Table 8: HENS case study cost minimization results, thousands $

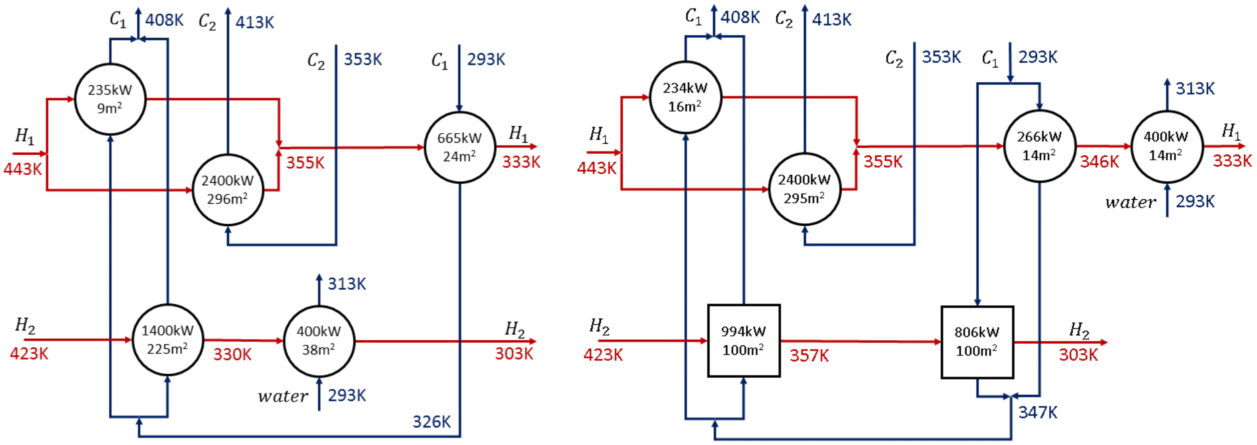
	GDPopt	DICOPT	BARON (LB)
Conventional	115	109	107 (106)
Single module type	166	-	135 (134)
Single module ILR	167	-	-
Multiple module types	134	-	120 (66)
Multiple module ILR	109	-	109 (90)
Hybrid	119	-	102 (40)
Hybrid ILR	119	-	105 (30)

From the results, several observations may be made. First, given 300 seconds, BARON find the best result in almost every instance. Though BARON is unable to close the bounds and establish global optimality for this case study, good solutions are still obtained. However, if the objective is to more quickly screen for good solutions, BARON may be less ideal, as it is not as fast as the non-global options. The two non-global solvers, GDPopt and DICOPT, give respectable solutions for the conventional case, with DICOPT slighly outperforming GDPopt, but for modular problems, GDPopt is able to find feasible solution where DICOPT is unable to. A dashed line indicates when a solver is not able to find a feasible solution within the set time limit. In the case of DICOPT, the solver often terminates before the time limit with an error message of "intermediate non-integer". This demonstrates the value of modeling with GDP: many different solution strategies become available, with the user able to select among them based on their efficacy and speed.

The induced linearity reformulation was shown to improve the performance of the GDPopt algorithm for the multiple module case, while yielding similar results for the other cases. For

BARON, the effect was more mixed. In the single module type case, the increase in problem size due to the ILR resulted in a failure to find a solution. However, in the case for multiple module types, use of the ILR resulted in an improved solution. Therefore, as the ILR is available as an automatic transformation in Pyomo, it has value as an optional preprocessing step for difficult modular process design problems.

The results show that the variable cost premium for strictly modular designs can lead to a cost disadvantage, even when fixed costs for each match are reduced by ease of installation in the modular case. However, the results also indicate that use of a hybrid system in which large exchangers are combined with smaller modular exchangers can result in cost savings beyond the base conventional case.



(a) Conventional HENS

(b) Hybrid HENS

Figure 14: Heat exchanger network designs. Circles indicate conventional heat exchangers. Box indicates modular heat exchanger. Heat transfer rate Q and exchanger area A are indicated within each exchanger.

Here, the hybrid network (see Fig. 14b) cost of \$102,000 outperforms the total cost of \$107,000 for the conventional network (see Fig. 14a).

Conclusion

This paper proposes a strategy for the design of chemical processing networks incorporating modular unit constructions. Modular designs show great promise due to their flexibility and other advantages. However, they may not be the best choice in all instances. A general logical (Generalized Disjunctive Programming) formulation is presented which examines several factors in modular design, including time-to-market considerations, module relocation, and hybrid conventional-modular configurations, to systematically consider the tradeoffs involved in selecting a modular versus conventional unit construction within a processing network. We also identify the *effectively discrete* problem structure that is characteristic of many modular design problems and describe the application of an induced linearity reformulation that allows some nonlinear constraints to be exactly represented with a linear-discrete set of equations. The applicability of the proposed general formulation is demonstrated using a set of illustrative as well as literature case studies. We also demonstrate the implementation of the design strategy as a set of inter-operable modeling tools within the Pyomo modeling language. After developing an initial GDP modular design model, several solution strategies are available, including automatic reformulations from GDP to MILP/MINLP, followed by the application of a commercial MINLP code, or the use of direct logic-based solution algorithms.

Though it would be desirable to examine more specific case studies, the representative examples presented offer a few generic insights into when the modular approach may be advantageous. When the project life is shorter, the ability of modular facilities to be assembled more quickly can provide a time-to-market advantage. If a firm is cash-flow constrained, a modular approach may realize positive cash flows more quickly. Similarly, if supply availability or demand profiles experience variations or cycling, the flexibility afforded by reallocatable and relocatable modules becomes desirable. This flexibility also allows modules to retain a higher salvage value when a secondary market or destination exists beyond the project life. Finally, modular designs can be more effective in distributed processing networks in which transportation costs prohibit use of a large centralized processing facility, blunting the conventional economies of scale effect. From the HENS example, we can conclude that modular constructions are relevant in subsystems when smaller modules allowing more efficient physical phenomena replace large units due to reduced fixed cost.

Despite progress in systematic modular process network design, several challenges remain. First, we note that in most cases presented, the value of the profit improvements or cost reductions is less than 10%. The conclusions are therefore sensitive to the values of the input data for the examples. For instance, the salvage value of modular units is often a key factor in the end result.

More work also remains to be done in quantifying the trade-offs between modular and conventional facilities. In practice, several factors related to modular construction such as improved reliability, safety, and quality are often considered only in a qualitative sense. However, to systematically evaluate trade-offs with optimization-based techniques, a quantitative model is necessary. In addition, for large-scale problems, computation of good solutions with mathematical optimality guarantees remains a significant challenge. Improved modeling techniques and solution strategies that take advantage of problem structure will continue to be valuable.

Nomenclature

δ_t	Discount factor at time period t
γ	Conventional unit cost exponential scaling factor
$\hat{v}$	Auxiliary continuous variables
$\hat{x}_j$	Internal state variables at site j
$\mathcal{Q}$	Heat transfer (kW)
LMTD	Log mean temperature difference (K)
ϕ_c	Conventional unit salvage value fraction
ϕ_m	Module salvage value fraction

$\Psi_{i,t}$ Raw material availability at supplier i in period t

N_T Project life length (periods)

τ_c Conventional unit construction time (periods)

τ_m Modular unit setup time (periods)

θ Demand profile scenarios

A Heat transfer surface area

$C_{j,n}^{conv}$ Conventional capital cost for unit n at site j

$C_n^{f,conv}$ Fixed installation cost for conventional unit of type n

$C_n^{f,mod}$ Fixed installation cost for modular unit of type n

$C_n^{s,conv}$ Base size cost for conventional unit of type n

$C_{m,t}$ Purchase cost for module of type m in period t

$Cost_{i,t}^{feed}$ Raw material cost at supplier i in period t

$D_{i,j}, D_{j,k}, D_{j,j'}$ Distance between suppliers and sites, sites and markets, and pairs of sites, respectively (miles)

$d_{q,s}$ Size of discrete element for variable q and size s

dr Annual discount rate

$f(r_{j,t}, S_{j,n,t}, \hat{x}_j)$ Production function

$f_{i,j,t}^{feed}$ Shipment quantities of raw material between supplier i and site j in period t

$f_{j,k,t}^{prod}$ Shipment quantities of product between site j and market k in period t

FC_p Heat content of process stream p (kW/K)

$g_{j,n,t}(\hat{x})$ Unit performance equations for unit n at site j in period t

$h(u, v, w)$ Bilinear constraints

$i \in I$ Set of suppliers

$j \in J$ Set of potential processing sites

$k \in K$ Set of markets

$m \in M_n$ Set of modular alternatives for unit of type n

$n \in N_j$ Set of potential units at site j

n_P Number of time periods in a year

$n_{m,j,t}$ Number of modules of type m at site j in period t

$np_{m,j,t}$ Number of modules of type m purchased for site j in period t

NPV^{θ} Net present value of scenario θ solution

$ns_{m,j,t}$ Number of modules of type m sold at site j in period t

$nt_{m,j,j',t}$ Number of modules of type m transferred from site j to site j' in period t

$P(t)$ Present value function

$p_{j,t}$ Production level at site j in period t

$Price^{prod}_{k,t}$ Product sale value at market k in period t

$q \in Q$ Set of effectively discrete design variables

$r \in R_q$ Set of continuous state variables multiplied with effectively discrete variable q

$r_{j,t}$ Raw material supply at site j in period t

$s \in DS_q$ Set of discrete sizes for variable with index q

$s' \in DS'$ Set of discrete variables

$S_{j,n}^{\theta}$ Conventional alternative size of unit n at site j in scenario θ

S_n^0 Base size for conventional unit of type n

$S_{j,n,t}$ Size of unit n at site j in period t

$S_{j,n}$ Conventional alternative size of unit n at site j

S_m Size of module of type m

$t \in T$ Set of time periods

T_{in}/T_{out} Stream inlet/outlet temperature (K)

$TC_{i,j}^{f,feed}$ Fixed transportation cost for raw material from supplier i to site j

$TC_{j,k}^{f,prod}$ Fixed transportation cost for product from site j to market k

TC_t^{feed} Variable transportation cost for raw material (\$ per unit feed per mile)

$TC_{m,t}^{mod}$ Transportation cost for module of type m (\$ per module per mile)

TC_t^{prod} Variable transportation cost for product (\$ per unit product per mile)

TF^{feed} Fixed cost for raw material transportation route

TF^{prod} Fixed cost for product transportation route

U Overall heat transfer coefficient ($kWm^{-1}K^{-1}$)

u Effectively discrete variable

v, w Continuous state variables

v_r^L/v_r^U Lower/upper bound for variable of index r

wt_{θ} Scenario probability weighting

x_i, y_i Spatial coordinates of supplier i

x_j, y_j Spatial coordinates of process site j

x_k, y_k Spatial coordinates of markets k

$Y_{j,n}^{conv}$ Boolean variable for selection of the conventional alternative for unit n at site j

$Y_{i,j}^{feed}$ Boolean variable for existence of a transportation link between supplier i and site j

$Y_{j,n}^{mod}$ Boolean variable for selection of a modular alternative for unit n at site j

$Y_{j,j'}^{move}$ Boolean variable for existence of a transportation link between site j and site j' for transferring modules

$Y_{j,k}^{prod}$ Boolean variable for existence of a transportation link between site j and market k

Y_{j}^{site} Boolean variable for existence of site j

$Y_{j,n}^{unit}$ Boolean variable for existence of unit n at site j

$z_{q,s}$ Binary variable for selection of size s for variable q

Acknowledgement

The authors thank Sulogna Roy of Zeton, Inc. for her helpful insights on modular design considerations. The authors also gratefully acknowledge support by the U.S. Department of Energy, Office of Fossil Energy as part of the Institute for the Design of Advanced Energy Systems (IDAES).

References

(1) Roy, S. Consider modular plant design. *Chemical Engineering Progress* **2017**, *113*, 28–31.

(2) Rogers, G. G.; Bottaci, L. Modular production systems: a new manufacturing paradigm. *Journal of Intelligent Manufacturing* **1997**, *8*, 147–156.

(3) Stephanopoulos, G.; Westerberg, A. W. Modular Design of Heat Exchanger Networks. *Chemical Engineering Communications* **1980**, *4*, 119–126.

(4) Shah, N. K.; Ierapetritou, M. G. Integrated production planning and scheduling optimization of multi-site, multi-product process industry. *21st European Symposium on Computer Aided Process Engineering* **2011**, *Volume 29*, 1015–1019.

(5) Moulijn, J. A.; Stankiewicz, A.; Grievink, J.; Górak, A. Process intensification and process systems engineering: A friendly symbiosis. *Computers & Chemical Engineering* **2008**, *32*, 3–11.

(6) Demirel, S. E.; Li, J.; Hasan, M. M. F. Systematic Process Intensification using Building Blocks. *Computers & Chemical Engineering* **2017**, *105*, 2–38.

(7) Baldea, M.; Edgar, T. F.; Stanley, B. L.; Kiss, A. A. Modular manufacturing processes: Status, challenges, and opportunities. *AIChE Journal* **2017**, *63*, 4262–4272.

(8) Chen, Q.; Grossmann, I. Recent Developments and Challenges in Optimization-Based Process Synthesis. *Annual Review of Chemical and Biomolecular Engineering* **2017**, *8*, 249–283.

(9) Lier, S.; Wörsdörfer, D.; Grünewald, M. Wandlungsfähige Produktionskonzepte: Flexibel, Mobil, Dezentral, Modular, Beschleunigt. *Chemie Ingenieur Technik* **2015**, *87*, 1147–1158.

(10) Brodhagen, A.; Grünewald, M.; Kleiner, M.; Lier, S. Erhöhung der Wirtschaftlichkeit durch beschleunigte Produkt- und Prozessentwicklung mit Hilfe modularer und skalierbarer Apparate. *Chemie Ingenieur Technik* **2012**, *84*, 624–632.

(11) Seifert, T.; Sievers, S.; Bramsiepe, C.; Schembecker, G. Small scale, modular and continuous: A new approach in plant design. *Chemical Engineering and Processing: Process Intensification* **2012**, *52*, 140–150.

(12) Lier, S.; Grünewald, M. Net Present Value Analysis of Modular Chemical Production Plants. *Chemical Engineering & Technology* **2011**, *34*, 809–816.

(13) Wörsdörfer, D.; Lier, S.; Crasselt, N. Real options-based evaluation model for transformable plant designs in the process industry. *Journal of Manufacturing Systems* **2017**, *42*, 29–43.

(14) Mothes, H. No-Regret-Lösungen - Modulare Produktionskonzepte für komplexe, unsichere Zeiten. *Chemie Ingenieur Technik* **2015**, *87*, 1159–1172.

(15) Tan, S. H.; Barton, P. I. Optimal dynamic allocation of mobile plants to monetize associated or stranded natural gas, Part I: Bakken shale play case study. *Energy* **2015**, *93*, 1581–1594.

(16) Yang, M.; You, F. Modular methanol manufacturing from shale gas: Techno-economic and environmental analyses of conventional large-scale production versus small-scale distributed, modular processing. *AIChE Journal* **2018**, *64*, 495–510.

(17) Grossmann, I. E.; Trespalacios, F. Systematic modeling of discrete-continuous optimization models through generalized disjunctive programming. *AIChE Journal* **2013**, *59*, 3276–3295.

(18) Grossmann, I. E.; Voudouris, V. T.; Ghattas, O. In *Recent Advances in Global Optimization*; Floudas, C., Pardalos, P., Eds.; Princeton Univ. Press, 1992; pp 475–512.

(19) Reay, D.; Ramshaw, C.; Harvey, A. *Process intensification engineering for efficiency, sustainability and flexibility*; Elsevier, 2008.

(20) Voudouris, V. T.; Grossmann, I. E. Mixed-integer Linear Programming Reformulations for Batch Process Design with Discrete Equipment Sizes. *Ind. Eng. Chem. Res.* **1992**, *31*, 1315–1325.

(21) Chen, Q.; Johnson, E. S.; Siirola, J. D.; Grossmann, I. E. Pyomo.GDP: Disjunctive Models in Python. Proceedings of the 13th International Symposium on Process Systems Engineering. San Diego, 2018.

(22) Hart, W. E.; Laird, C. D.; Watson, J.-P.; Woodruff, D. L.; Hackebeil, G. A.; Nicholson, B. L.; Siirola, J. D. *Pyomo âĂŤ Optimization Modeling in Python*, 2nd ed.; Springer Optimization and Its Applications; Springer International Publishing: Cham, 2017; Vol. 67.

(23) D'Ambrosio, C.; Lodi, A. Mixed integer nonlinear programming tools: an updated practical overview. *Annals of Operations Research* **2013**, *204*, 301–320.

(24) Bernal, D. E.; Chen, Q.; Gong, F.; Grossmann, I. E. Mixed-Integer Nonlinear Decomposition Toolbox for Pyomo (MindtPy). *Computer Aided Chemical Engineering* **2018**, *44*, 895–900.

(25) Viswanathan, J.; Grossmann, I. E. A combined penalty function and outer-approximation method for MINLP optimization. *Computers and Chemical Engineering* **1990**, *14*, 769–782.

(26) Duran, M. A.; Grossmann, I. E. An outer-approximation algorithm for a class of mixed-integer nonlinear programs. *Mathematical Programming* **1986**, *36*, 307.

(27) Brook, A.; Kendrick, D.; Meeraus, A. GAMS, a user's guide. *ACM SIGNUM Newsletter* **1988**, *23*, 10–11.

(28) Tawarmalani, M.; Sahinidis, N. V. A polyhedral branch-and-cut approach to global optimization. *Mathematical Programming* **2005**, *103*, 225–249.

(29) Lara, C. L.; Trespalacios, F.; Grossmann, I. E. Global optimization algorithm for capacitated multi-facility continuous location-allocation problems. *Journal of Global Optimization* **2018**, *71*, 871–889.

(30) Vigerske, S.; Gleixner, A. *SCIP: Global Optimization of Mixed-Integer Nonlinear Programs in a Branch-and-Cut Framework*; 2016.

(31) Eilermann, M.; Post, C.; Schwarz, D.; Leufke, S.; Schembecker, G.; Bramsiepe, C. Generation of an equipment module database for heat exchangers by cluster analysis of industrial applications. *Chemical Engineering Science* **2017**, *167*, 278–287.

(32) Yee, T.; Grossmann, I. Simultaneous optimization models for heat integrationâĂŤII. Heat exchanger network synthesis. *Computers & Chemical Engineering* **1990**, *14*, 1165–1184.

(33) Türkay, M.; Grossmann, I. E. Logic-based MINLP algorithms for the optimal synthesis of process networks. *Computers & Chemical Engineering* **1996**, *20*, 959–978.