# A Spatial Superstructure Approach to the Optimal Design of Modular Processes and Supply Chains

Yue Shao, Jiaze Ma, and Victor M. Zavala

Department of Chemical and Biological Engineering
University of Wisconsin-Madison, 1415 Engineering Dr., Madison, WI 53706, USA

**Abstract**

Modularity is a design principle that aims to provide flexibility for spatio-temporal assembly/disassembly and reconfiguration of systems. This design principle can be applied to multiscale (hierarchical) manufacturing systems that connect units, processes, facilities, and entire supply chains. Designing modular systems is challenging because of the need to capture spatial interdependencies that arise between system components due to product exchange/transport between components and due to product transformation in such components. In this work, we propose an optimization framework to facilitate the design of modular manufacturing systems. Central to our approach is the concept of a spatial superstructure, which is a graph that captures all possible system configurations and interdependencies between components. The spatial superstructure is a generalization of the notion of a superstructure and of a p-graph used in process design, in that it encodes spatial (geographical) context of the system components. We show that this generalization facilitates the simultaneous design and analysis of processes, facilities, and of supply chains. Our framework aims to select the system topology from the spatial superstructure that minimizes design cost and that maximizes design modularity. We show that this design problem can be cast as a mixed-integer, multi-objective optimization formulation. We demonstrate these capabilities using a case study arising in the design of a plastic waste upcycling supply chain.

**Keywords:** modularity; design; supply chains

## 1 Introduction

Modularity is a design principle that aims to provide flexibility for spatio-temporal assembly, disassembly, and reconfiguration of systems. This design principle can be applied to multiscale manufacturing systems that connect equipment units/technologies, processes (collections of units/technologies) [1], facilities (collections of processes) [2], and entire supply chains (collections of facilities) [3]. Modular manufacturing systems are typically built from small-scale and standardized technologies (equipment units) that perform specific tasks and that are coupled together using sparse interfaces [4, 5]. Small dimensions and sparse interfaces facilitate system assembly/disassembly and reconfiguration (e.g., migration of technologies to a different location and expansion of capacity). This logistical flexibility helps systems adapt to fast-changing markets and other externalities (e.g., climate, resource availability, policy) [6, 7] and enable the recovery of resources that are highly distributed and potentially short-lived [8–10]. Modularity principles have been recently explored in diverse industrial sectors such as power generation, data centers, and chemical processes [11–13].

Modular design principles are often applied at the process synthesis stage. For instance, processes that compose a facility (collection of processes at a given geographical location) can be interpreted as modules and products exchanged between such modules give rise to a facility. At this level of an organization, modular design is usually coupled with process intensification that aims to improve energy and process efficiency by combining difficult unit operations in a single-process equipment [14]. Researchers have proposed generalized modular representation framework based on mass/heat-transfer principals [2], and analyzed the impacts of key factors on operability and control of intensified/modular designs such as process constraints, numbering up vs. scaling up, and dynamic/periodic operation [15]. These analysis helped develop systematic framework for the synthesis of operable process intensification systems such as reactive separation systems [1], and further provided novel ideas on discovering intensification/modularization opportunities at the process design stage [16,17]. Recently, researchers have also used graph theory to study modular processes from a pure connectivity point of view [18–20]. This perspective separates the concept of modular process and intensified process by identifying equipment modules that are tightly connected. Specifically, the connectivity induced from products exchanged between processes and from product transformation in such processes induces a degree of modularity of the system. For instance, facilities that have dense product interdependencies are less modular than those that have sparse interdependencies. The degree of product interdependency affects flexibility, as facilities that are tightly coupled are typically more difficult to reconfigure.

The graph-theoretic perspective to modular process design allows us to apply similar ideas at a higher organization level. For instance, facilities that compose a supply chain can be interpreted as modules that exchange products across geographical locations (e.g., via long-distance transport). This indicates that a supply chain can be seen as a distributed network of processes (a distributed facility with processes placed at different geographical locations), while a typical facility can be seen as a centralized network of processes (all processes are placed at the same geographical location). Similar to the case of a facility, the modularity of a supply chain is affected by the connectivity induced from product transport across components, from product transformation in its components, and from its ability to be reconfigured (e.g., movable processes). Recent work in power grid and natural gas networks has revealed that deploying distributed data centers, batteries, vehicle charging stations, gas-fired power plants, and manufacturing facilities can add flexibility, relieve network congestion, and enhance system-wide performance [21–23]. This flexibility can be used to absorb fluctuations of wind and solar power and can help withstand externalities (e.g., extreme weather events, policy, equipment failures). Researchers have found that the ability of the supply chain modules to relocate helps improves the efficiency [24] and the sustainability of the process [25,26], especially for processes that involve spatially-distributed materials such as the biomass-to-energy supply chains [27,28], and meets highly variable product demands in a cost-efficient manner [29].

Computational frameworks have been proposed to optimize modular supply chains considering demand uncertainty to manage risk [30,31], and to determine production schedules of supply chains with modular production units [32,33]. A key observation of these works is that modularity of the supply chain means that the system includes modular (small) production units. In other words, these works do not use modularity in a graph-theoretic sense. From a graph-theoretic perspective a modular supply chain is a system that contains modules (collections of nodes) with tight connectivity.

In other words, just like a modular chemical process where more than one unit operations are integrated to form a module to achieve better energy efficiency and process flexibility, a supply chain can be modularized by grouping and locating tightly connected production units together to improve transportation efficiency of materials and manage risk under product uncertainty.

Rigid/inflexible processes, facilities, and supply chains are vulnerable and can face significant risk due to changing markets and climate, shortages of resources at a specific location (e.g., water), and changes in the policy landscape (e.g., carbon emissions). On the other hand, economies of scale tend to benefit large, centralized systems due to favorable scaling of throughput with equipment size [7]. Industrial organizations typically evolve into a mixed state in which certain processing tasks are performed in small dispersed modular systems, while others are performed in large centralized facilities. This arrangement induces a hierarchical reorganization of individual production facilities and of associated supply chains. Inducing modularity by design in such organizations is difficult, as one must consider complex physical and geographical dependencies between processing tasks and products as well as trade-offs between economies of scale and flexibility provided by modularity.

Maximal p-graph structures and superstructures are system representations that have been widely used for the design of chemical processes and of mass/energy recovery networks for single facilities [34,35]. A maximal p-graph structure encodes all possible feasible paths between primary products, processing tasks, and intermediate and final products [36]. A superstructure encodes all possible configurations of equipment units and product flows that perform tasks defined by the maximal p-graph (i.e., multiple units might perform the same task) [37,38]. While these representations provide a powerful framework to investigate systems at a process level, they do not encode spatial information, which is necessary to capture how design affects flexibility at higher organization levels (facilities and supply chains).

In this work, we propose an optimization framework to facilitate the design of modular processes, facilities, and supply chains. Central to our approach is the concept a spatial superstructure, which is a graph that encodes all possible dependencies between components. We show that the spatial superstructure is a generalization of the superstructure and p-graph used for process design in that it encodes spatial (geographical) context. Moreover, we show that this generalization enables the simultaneous design of processes, facilities, and of supply chains in a unified manner. Specifically, a spatial superstructure is a superstructure under which technologies and flows encode positional (geographical) context and that encodes product dependencies that arise from transformation (as in a p-graph). This allows us to represent standard centralized processes and facilities (under which technologies are placed at the same geographical location) and a spatially-distributed process (under which technologies are distributed over multiple geographical locations) by using the same graph topology. The graph representation reveals that a key distinction between a spatial superstructure and other representations is in how product transportation is accounted for. For instance, short-range transport (inside a process or facility) might use pipelines while long-range transport might use truck hauling or railways.

The proposed approach leverages the graph representation of the spatial superstructure to identify topologies that minimize system design cost and that maximize design modularity. We show that this design problem can be cast as a mixed-integer, multi-objective optimization formulation and allows us to capture interdependencies between primary products (raw materials), intermediate products, and final products that arise from product transformation and transport across components. We also leverage the topology of the spatial superstructure to accelerate the optimal design search by restricting such search along feasible paths that obtain desired products from primary products. This approach contrasts with standard superstructure optimization approaches that search over individual technologies/units. We demonstrate the capabilities for the design of a plastic waste upcycling supply chain.

We highlight that, compared to existing literature, our work has the following contributions:

- Previous work has focused on the the design of supply chains that leverage modular production units to provide flexibility. In our framework, we treat the supply chain as a system that can be modularized (in a graph-theoretic sense) based on spatial connectivity among different production units. This connectivity provides flexibility to the system and can be induced by a combination of large, medium, and large production systems.
- We also propose the concept of a spatial superstructure to simultaneously design modular facilities and supply chains under a unified framework; previous superstructure and p-graph paradigms focus on facility design (single-site). Our generalization is enabled by taking a graph-theoretical view of the problem.

## 2 Concepts and Graph Representations

In this section, we revisit the concept of a p-graph, maximal p-graph, and superstructure and provide a unifying graph-theoretic perspective. We use these concepts to propose a spatial superstructure that will be used to guide the design of modular systems.

### 2.1 P-graph and Maximal p-graph

In the context of chemical processes, graphs have been used to analyze interdependencies between products and technologies (unit operations) in a process and with this unravel a number of fundamental systems properties such as topological feasibility (e.g., ability to reach a set of products from a set of primary products).

A process can be represented/modeled as a p-graph (short for *process graph*). In a p-graph, nodes represent technologies (processing tasks or unit operations) and products (primary products as well as intermediate and final products) while edges represent dependencies between products and technologies. Here, the concept of product is general in that it can capture general resources such as energy (e.g., electricity). In addition, we note that technologies induce complex interdependencies because they conduct transformation of products into other products (e.g., a chemical reactor or a separation unit). Under the p-graph abstraction, it is possible to derive a *maximal p-graph* (max p-graph for short) that encodes all possible technologies and required primary products and intermediate products that can be used to obtain reach a desired set of final products. This representation is powerful and insightful because any possible process configuration that connects primary products, technologies, and intermediate/final products is embedded in the maximal p-graph. A specific process realization is derived by selection of specific nodes and edges (which form a path between primary products (i.e., primary products), intermediate products, and final products desired). As we will see, superstructure representations inherit the topology of max p-graphs.

Suppose that a process involves a set of intermediate/final products $\mathcal{P}$ and a set of primary products $\mathcal{R}$ from which intermediate/final products are derived (via technologies). Furthermore, we define a set of all products involved in the process as $\mathcal{I}$. A product can potentially be generated by different types of technologies (techs for short), and we define a set of possible techs as $\mathcal{T}$.

Associated with each tech $t \in \mathcal{T}$, there is a set of output products $\Omega_t \in \mathcal{P}$, a set of input products $\mathcal{K}_t \in \mathcal{I}$, and a tech type $\theta_t$. For convenience, we categorize techs by products and types using the subsets $\mathcal{T}_{i,i',j} \subseteq \mathcal{T}$ with $\mathcal{T}_{i,i',j} := \{t | i \in \Omega_t, i' \in \mathcal{K}_t, \theta_t = j\}$.

We model the *max p-graph* as a directed graph $\mathcal{G}^p = (\mathcal{N}^p; \mathcal{E}^p)$ where $\mathcal{N}^p$ is its set of nodes (vertices) and $\mathcal{E}^p$ is its set of edges. The set of nodes include product nodes and tech nodes. We define the set of nodes representing the supplies/sources of primary product as $\mathcal{S}^p \subseteq \mathcal{N}^p$). Associated with each node $s \in \mathcal{S}^p$ there is a type of primary product $\Omega_s \in \mathcal{R}$. For convenience, we categorize suppliers as $\mathcal{S}^p_i \subseteq \mathcal{S}^p$ with $\mathcal{S}^p_i := \{s | \Omega_s = i\}$. Similarly, we define the set of nodes representing demands/sinks of final products as $\mathcal{D}^p \subseteq \mathcal{N}^p$. Associated with each $d \in \mathcal{D}^p$, there is a type of product $\mathcal{K}_d \in \mathcal{P}$; we categorize demand nodes as $\mathcal{D}^p_i \subseteq \mathcal{D}^p$ with $\mathcal{D}^p_i := \{d | \mathcal{K}_d = i\}$.

We define the set of nodes representing the techs as $\mathcal{U}^p \subseteq \mathcal{N}^p$. For each node $u \in \mathcal{U}^p$, there is a tech $\tau_u \in \mathcal{T}$ associated with it, and we classify tech nodes as $\mathcal{U}^p_t \subseteq \mathcal{U}^p$ with $\mathcal{U}^p_t := \{u | \tau_u = t\}$. Because each tech $t$ is also associated with an product sets $\Omega_t, \mathcal{K}_t$ and a type $\theta_t$, we have that each node $u \in \mathcal{U}^p$ is associated with a set of products $\Omega_{\tau_u} \in \mathcal{P}$ that the tech generates, a set of products $\mathcal{K}_{\tau_u} \in \mathcal{I}$ that enter the tech, and a type of tech $\theta_{\tau_u}$. For convenience, we use the short-hand notation $\Omega_u$, $\mathcal{K}_u$, and $\theta_u$. We define the subsets $\mathcal{U}^p_{i,i',j} := \{u | i \in \Omega_u, i' \in \mathcal{K}_u, \theta_u = j\}$. We highlight that, in the max p-graph representation, the tech node set $\mathcal{U}^p$ contains only attributes of tech $t \in \mathcal{T}$, so they are defined similarly. In other words, node $\mathcal{U}^p_t = \mathcal{U}^p_{i,i',j}$ corresponds to the tech $t = \mathcal{T}_{i,i',j}$. Finally, the set of all nodes is:

$$\mathcal{N}^p = \mathcal{S}^p \cup \mathcal{D}^p \cup \mathcal{U}^p. \tag{2.1}$$

Figure 1 provides an illustration of a max p-graph and showcases how complex interdependencies between products and techs arise. In this example, the set $\mathcal{P}$ contains products $i_3$ and $i_5$, set $\mathcal{R}$ contains primary products $i_1$, $i_2$, $i_3$, $i_4$ and $i_5$, and the set $\mathcal{I}$ contains the final product $i_5$. The intermediate product $i_3$ and final product $i_5$ are also included in the primary products because we consider the possibility of satisfying the demand by purchasing it from an external market. The set $\mathcal{T}$ contains a couple of tech types producing product $i_3$ from either $i_1$ or $i_1$ and $i_2$, and 2 types of techs producing product $i_5$ from either $i_3$ or $i_3$ and $i_4$, represented as $\{\mathcal{T}_{i_3,i_1,j_1}, \mathcal{T}_{i_3,\{i_1,i_2\},j_2}, \mathcal{T}_{i_5,\{i_3,i_4\},j_1}, \mathcal{T}_{i_5,i_3,j_2}\}$. Nodes representing supplies of primary products are on the left, shown as nodes $S^p_{i_1}$, $S^p_{i_2}$, $S^p_{i_3}$, $S^p_{i_4}$, and $S^p_{i_5}$ in set $\mathcal{S}^p$; nodes representing technologies are in the middle, shown as node $U^p_{i_3,i_1,j_1}$, $U^p_{i_3,\{i_1,i_2\},j_2}$, $U^p_{i_5,\{i_3,i_4\},j_1}$ and $U^p_{i_5,i_3,j_2}$ in set $\mathcal{U}^p$; nodes representing the demand are on the right, shown as node $D^p_{i_5}$

in set $\mathcal{D}^p$. Nodes and edges are highlighted based on the associated product; for example, node $S^p_{i_4}$ and the edge that carries the product $i_4$ to technology $U^p_{i_5,\{i_3,i_4\},j_1}$ have the same color. The product hierarchy of the process is also displayed in this max p-graph representation; looking from top to bottom, we can see that products or techs that are involved in the early stage of the process are on the top and those that are involved later in the process are on the bottom. Moreover, the max p-graph tells us that primary products $i_1, i_2$, and $i_4$ can only be purchased but not produced. Product $i_3$ is an intermediate product that can be produced by either type of tech or purchased from the external market; this product is also fed to the techs that produce product $i_5$ (which is the final product).

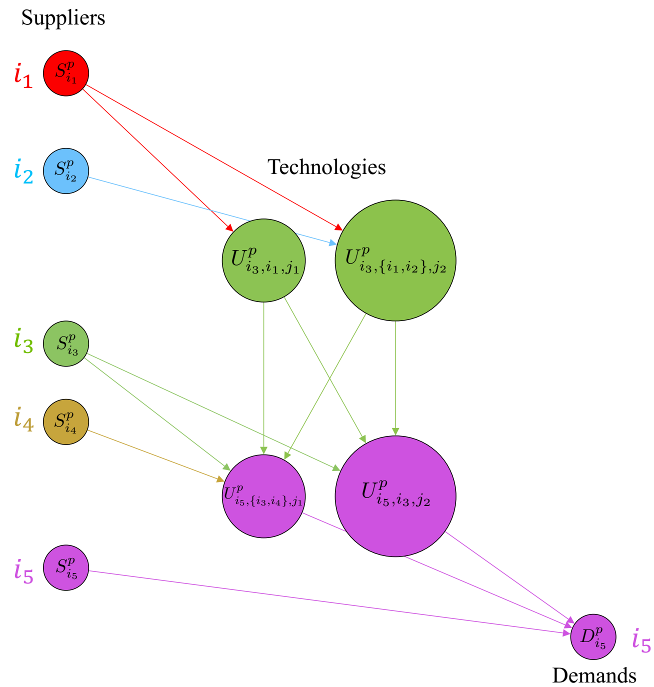


Figure 1: Illustration of a max p-graph showing dependencies between products and technologies.

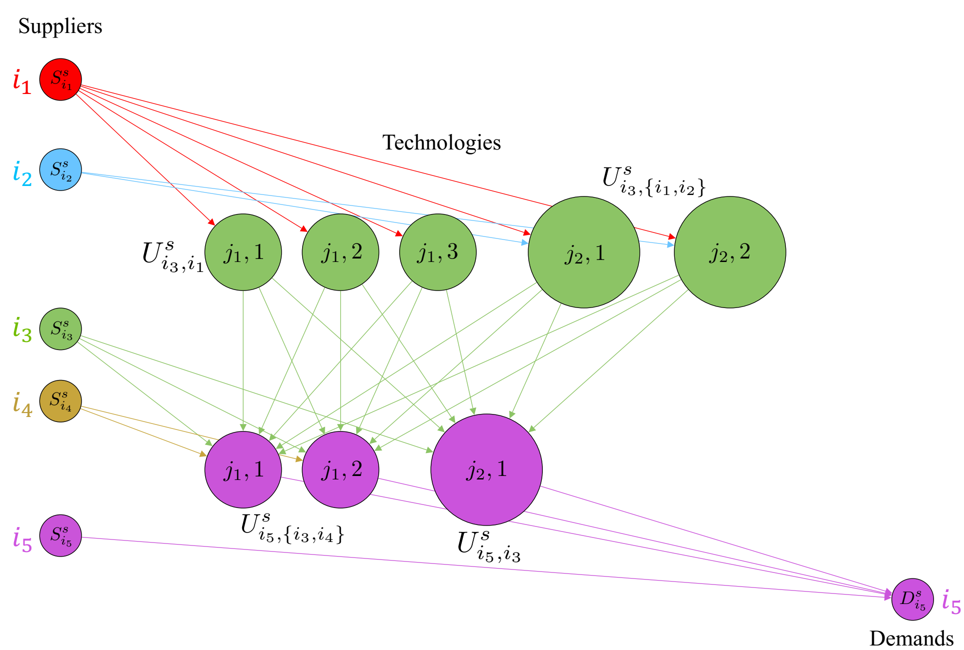


Figure 2: Illustration of a superstructure (associated with max p-graph in Figure 1) showing dependencies between products and technologies.

### 2.2 Superstructure

A superstructure is a system representation that inherits properties of a max p-graph (product-tech connectivity) but also accounts for the possibility of having multiple units/copies of techs and also account for additional attributes of techs (e.g., capacities). It is thus important to highlight that a superstructure can be derived from the topology of a max p-graph. The key difference is that, in a max p-graph, techs are interpreted as unit operations while, in a superstructure, techs represent equipment units.

A superstructure can also be represented as a graph; specifically, it can be represented as a directed graph $\mathcal{G}^s = (\mathcal{N}^s; \mathcal{E}^s)$, where $\mathcal{N}^s$ is its set of nodes (vertices) and $\mathcal{E}^s$ is its set of edges. As in the p-graph, primary products are defined as $\mathcal{R}$; intermediate/final products are defined as $\mathcal{P}$; and all products are defined as $\mathcal{I}$; techs are defined as $\mathcal{T}$ with the same attributes categories as in the max-p graph. Nodes representing supplies and demands are defined as $\mathcal{S}^s$ and $\mathcal{D}^s$, and the set of nodes $\mathcal{U}^s$ that represents techs. In this representation, each tech $u \in \mathcal{U}^s$ has an additional attribute that represents the unit number $\eta_u$.

We categorize the nodes $\mathcal{U}^s$ as $\mathcal{U}^s_{t,h} = \mathcal{U}^s_{i,i',j,h} \subseteq \mathcal{U}^s$ with $\mathcal{U}^s_{t,h} := \{u|\tau_u = t, \eta_u = h\}$ and $\mathcal{U}^s_{i,i',j,h} := \{u|i \in \Omega_u, i' \in \mathcal{K}_u, \theta_u = j, \eta_u = h\}$. In the case of a superstructure, the set $\mathcal{U}^s$ adds another layer of information on top of the set $\mathcal{T}$ to indicate that multiple copies of the same tech might be available. In other words, node $\mathcal{U}^s_{t,h} = \mathcal{U}^s_{i,i',j,h}$ corresponds to the $h$ unit/copy of tech $\mathcal{T}_{i,i',j}$. The set $\mathcal{N}^s$ for all nodes in the superstructure is $\mathcal{N}^s = \mathcal{S}^s \cup \mathcal{D}^s \cup \mathcal{U}^s$.

Using the same example shown in Figure 1, we illustrate a superstructure representation of this system in Figure 2. Here, we labeled the set $\mathcal{U}^s_{i,i'} := \{u|i \in \Omega_u, i' \in \mathcal{K}_u\}$ on the side and labeled the attribute $\{\theta_u, \eta_u\}$ on the node for each $u \in \mathcal{U}^s$. Comparing to the max p-graph representation, only the notation for nodes representing techs has changed. For instance, the notation $U^s_{i_3,i_1,j_1,1}$ denotes the first unit/copy of tech $\mathcal{T}_{i_3,i_1,j_1}$ that is used to obtain the intermediate product $i_3$. Multiple units/copies of this tech are available to satisfy the demand of final product $i_5$ and the same applied for other techs. We can also observe that the product-tech connectivity of the max p-graph is inherited by the superstructure. Moreover, we see that the superstructure graph is much denser than that of the max p-graph (due to the availability of multiple tech units).

### 2.3 Spatial Superstructure

We now proceed to generalize the notion of a max p-graph and of a superstructure to capture spatial context. Capturing spatial context is necessary to represent flexibility provided by modularity at different scales (process, facilities, supply chains). The *key observation* is that a supply chain can be seen as a distributed facility that exchanges products between processes (placed at different geographical locations). Similarly, a centralized facility can be seen as a supply chain with a single geographical location. This *unifying view* of a system will reveal interesting insights that can be exploited to derive a general graph-theoretic framework that explains how modularity emerges in a system design. Specifically, we will see that the topology of the spatial superstructure directly inherits the topology of a superstructure, which in turn inherits the product-tech topology of a max p-graph. Exploiting this topological dependencies is key in building superstructures and in identifying feasible system designs. Our final aim will be to derive optimization formulations identify a subgraph from the spatial superstructure (a design) to obtain a supply chain (composed of processes and facilities of different sizes and at potentially multiple locations) that minimizes system-wide cost and that maximizes modularity.

A spatial superstructure is a superstructure under which techs and connections encode positional context. This allows us to represent a standard single-site process/facility (under which equipment units are located at the same geographical location) and a spatially-distributed process/facility (under which units are distributed over multiple geographical locations) by using the same graph topology. The spatial superstructure allows us to capture transportation modes for the products and associated constraints and costs; for instance, short-range transport (inside a location) might use pipelines while long-range transport (across locations) might use trucks or railways.

The graph representation of the spatial superstructure is inherited from that of the superstructure. We model the spatial superstructure as a directed graph $\mathcal{G}^q = (\mathcal{N}^q; \mathcal{E}^q)$ where $\mathcal{N}^q$ is its set of nodes (vertices) and $\mathcal{E}^q$ is its set of edges. We define a set of potential spatial locations for placing technologies as $\mathcal{G}_t$, a set of potential locations for suppliers as $\mathcal{G}_s$, and a set of potential locations for demands as $\mathcal{G}_d$. We then define a set of all locations as $\mathcal{G} = \mathcal{G}_t \cup \mathcal{G}_s \cup \mathcal{G}_d$.

As in the max p-graph representation, primary products are defined as $\mathcal{R}$; intermediate and final products are defined as $\mathcal{P}$; all products are defined as $\mathcal{I}$; techs are defined as $\mathcal{T}$ with the same attributes and nested representation. Nodes representing supplies are defined as $\mathcal{S}^q$, and for nodes $s \in \mathcal{S}$, there is a new attribute $\phi_s \in \mathcal{G}_s$ representing the location of the supplies/sources of primary products. We define subsets to categorize suppliers by location and product as $\mathcal{S}^q_{i,g} \subseteq \mathcal{S}^q_i \subseteq \mathcal{S}^q$ with $\mathcal{S}^q_{i,g} := \{s|\Omega_s = i, \phi_s = g\}$ and $\mathcal{S}^q_i := \{s|\Omega_s = i\}$. Nodes representing demands are defined as $\mathcal{D}^q$, and for node $d \in \mathcal{D}^q$, there is a new location attribute $\phi_d \in \mathcal{G}_d$. We define the categorization as subsets $\mathcal{D}^q_{i',g} \subseteq \mathcal{D}^q_{i'} \subseteq \mathcal{D}^q$ with $\mathcal{D}^q_{i',g} := \{d|\mathcal{K}_d = i', \phi_d = g\}$ and $\mathcal{D}^q_{i'} := \{d|\mathcal{K}_d = i'\}$.

For each tech node $u \in \mathcal{U}^q$ there is a new location attribute $\phi_u \in \mathcal{G}_t$, and the subsets are written as $\mathcal{U}^q_{t,h,g} = \mathcal{U}^q_{i,i',j,h,g} \subseteq \mathcal{U}^q$ with $\mathcal{U}^q_{t,h,g} := \{u|\tau_u = t, \eta_u = h, \phi_u = g\}$ and $\mathcal{U}^q_{i,i',j,h,g} := \{u|i \in \Omega_u, i' \in \mathcal{K}_u, \theta_u = j, \eta_u = h, \phi_u = g\}$. Specifically, node $\mathcal{U}^q_{t,h,g} = \mathcal{U}^q_{i,i',j,h,g}$ corresponds to the $h$ copy of tech $\mathcal{T}_{i,i',j}$ that is located at location $g$. The set of all nodes is $\mathcal{N}^q = \mathcal{S}^q \cup \mathcal{D}^q \cup \mathcal{U}^q$.

Using the same example shown in Figure 2 and a couple of potential locations $A$ and $B$, we illustrate the spatial superstructure in Figure 3. Here, we labeled the set $\mathcal{U}^q_{i,i'}$ for the nodes on the side while labeled the attribute $\{\theta_u, \eta_u, \phi_u\}$ on the nodes. We note that there might be multiple locations for suppliers and demands. We also note that edges that connect nodes at the same locations or across different locations have different meaning. For example, we can choose to install the first copy of tech $\mathcal{T}_{i_3,i_1,j_1}$ that produces $i_3$ at location A (represented as node $U^q_{i_3,i_1,j_1,1,A}$) and the first copy of tech $\mathcal{T}_{i_5,i_3,j_2}$ that produces $i_5$ at location B (represented as node $U^q_{i_5,i_3,j_2,1,B}$), and the edge connecting them represents the transportation of product $i_3$ from location A to location B. If these techs are both placed at location A, an edge connecting them represents short-range (on-site) transport. We observe that the topology of the spatial superstructure is inherited from that of the superstructure, which in turn inherits the product-tech connectivity from the max p-graph. We also note that the spatial superstructure is much *larger and denser* than the superstructure and can becomes difficult (if not impossible) to express and visualize, due to the potential need to capture many geographical locations and tech units at such locations. Therefore, deriving an automatic approach that generates and analyzes the connectivity of the spatial superstructure is necessary.

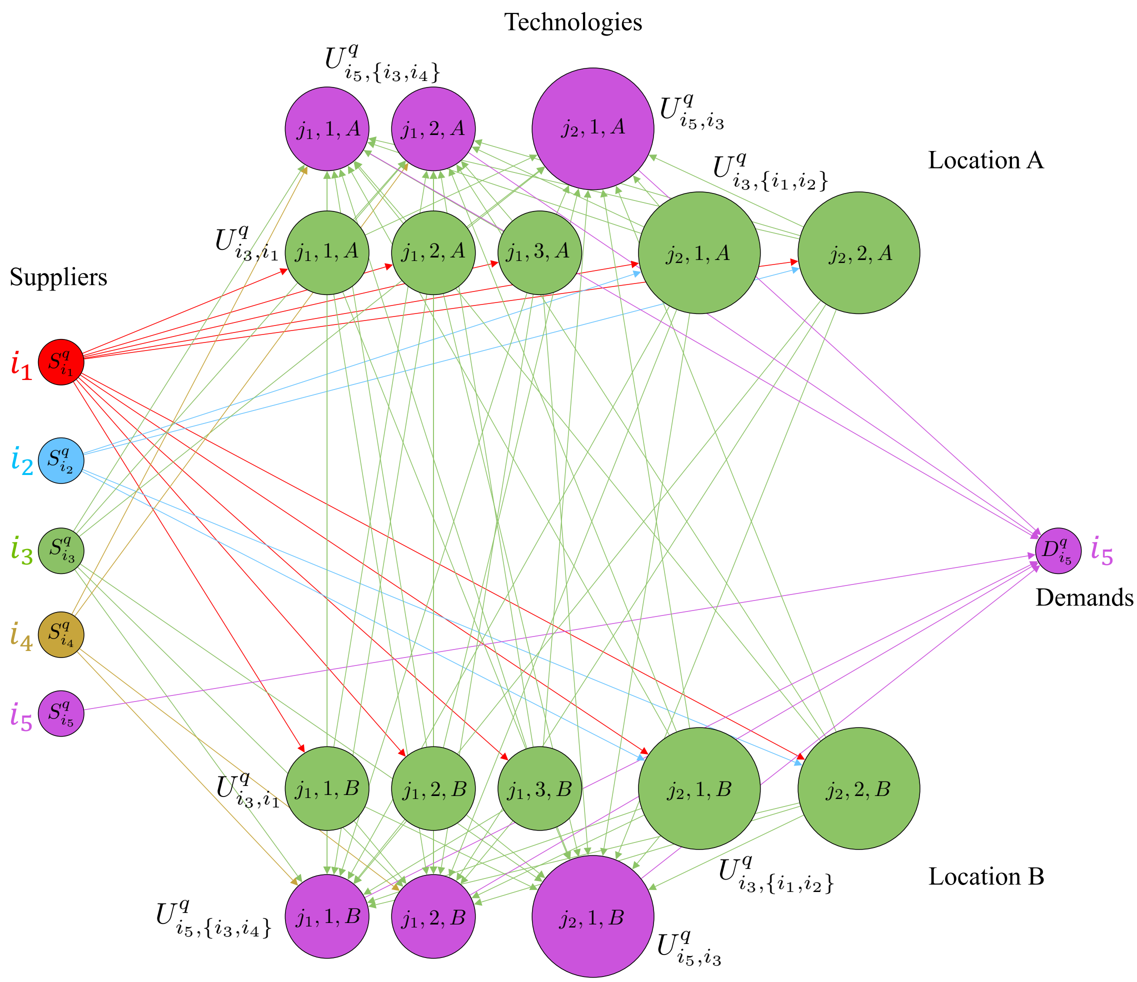


Figure 3: Illustration of a spatial superstructure (associated with max p-graph of Figure 1 and superstructure of Figure 2) showing dependencies between products and technologies across geographical locations.

### 2.4 Feasible Paths

A feasible path is a collection of nodes and edges that enables reaching final products from primary products. A feasible path can be derived by a reduction of nodes and edges of a max p-graph or superstructure by leveraging graph-theoretic concepts. For example, a feasible path is a subgraph $\mathcal{G}^f = (\mathcal{N}^f, \mathcal{E}^f)$ of the superstructure graph $\mathcal{G}^s$ (i.e., $\mathcal{G}^f \subseteq \mathcal{G}^s$) where $\mathcal{N}^f$ is a subset of $\mathcal{N}^s$ and $\mathcal{E}^f$ is a subset of $\mathcal{E}^s$. We present a feasible path obtained from an example max p-graph and superstructure in Figure 4. Compared to the number of possible paths derived from a max p-graph, the number of feasible paths from a superstructure is much larger because we now consider multiple copies of technologies (which enables more combinations). It is also important to highlight that, any feasible path in a superstructure (and spatial superstructure), has to be a feasible path for the max p-graph (because the superstructure inherits the product-tech connectivity). This observation is key in building superstructures that avoid spurious (infeasible) paths.

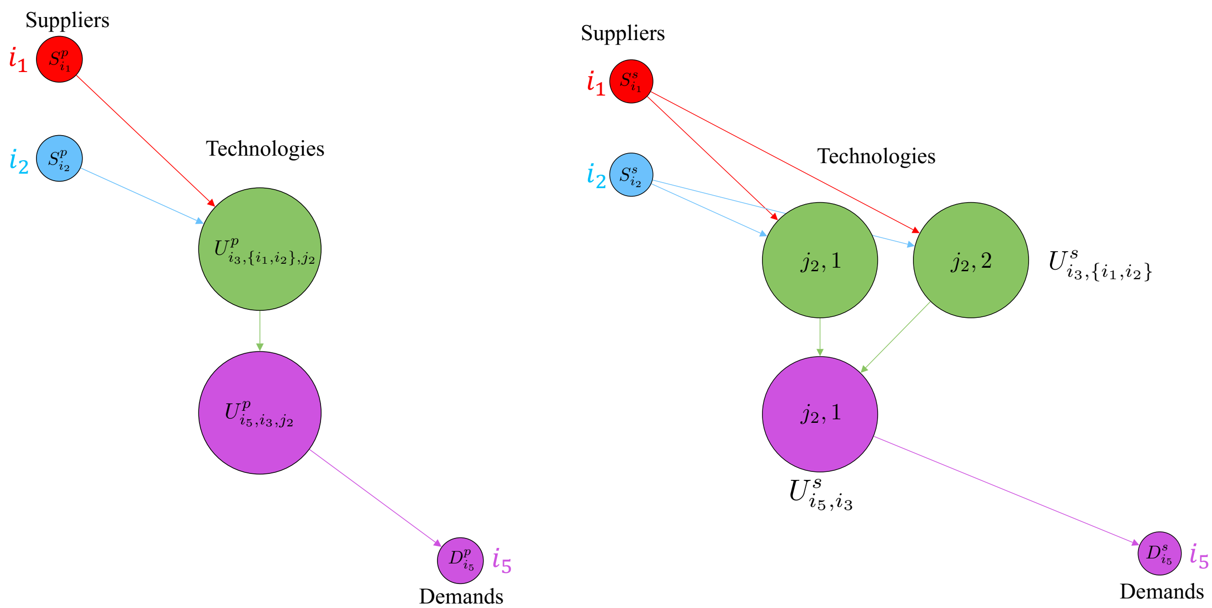


Figure 4: Example of a feasible path obtained from a max p-graph (left) and from a superstructure (right).

We can similarly derive any feasible paths from a spatial superstructure between primary products, techs, and final products as shown in Figure 5. Here, a couple of copies of the same technology $\mathcal{T}_{i_3,i_1,j_1}$ that produces product $i_3$ is located at location B (nodes $U^q_{i_3,i_1,j_1,1,B}$ and $U^q_{i_3,i_1,j_1,2,B}$), and two other technologies $\mathcal{T}_{i_3,\{i_1,i_2\},j_2}$ (node $U^q_{i_3,\{i_1,i_2\},j_2,1,A}$) and $\mathcal{T}_{i_5,\{i_3,i_4\},j_2}$ (node $U^q_{i_5,\{i_3,i_4\},j_2,1,A}$) that produces $i_3$ and final product $i_5$ are located at location B. As we add spatial information to the superstructure, combinations of techs across different locations are now possible and thus the number of possible feasible paths becomes even larger. Among all the feasible paths, an optimal design is a feasible path that takes into account tech and transport costs and modularity. As we consider complicated interdependencies between products and different capital and transportation cost due to potential locations, finding an optimal design is not immediately obvious from the spatial superstructure. Therefore, one needs to rely on optimization techniques to identify optimal paths, as we describe next.

## 3 Optimization Formulation for Finding Optimal System Designs

In this section, we derive an optimization formulation that aims to identify the hierarchy of products in the system. This step is essential for computing the amount of each product and the number of technologies required. This information is in turn required to generate superstructures and spatial superstructures. We then proceed to introduce an optimization formulation to obtain an optimal system design.

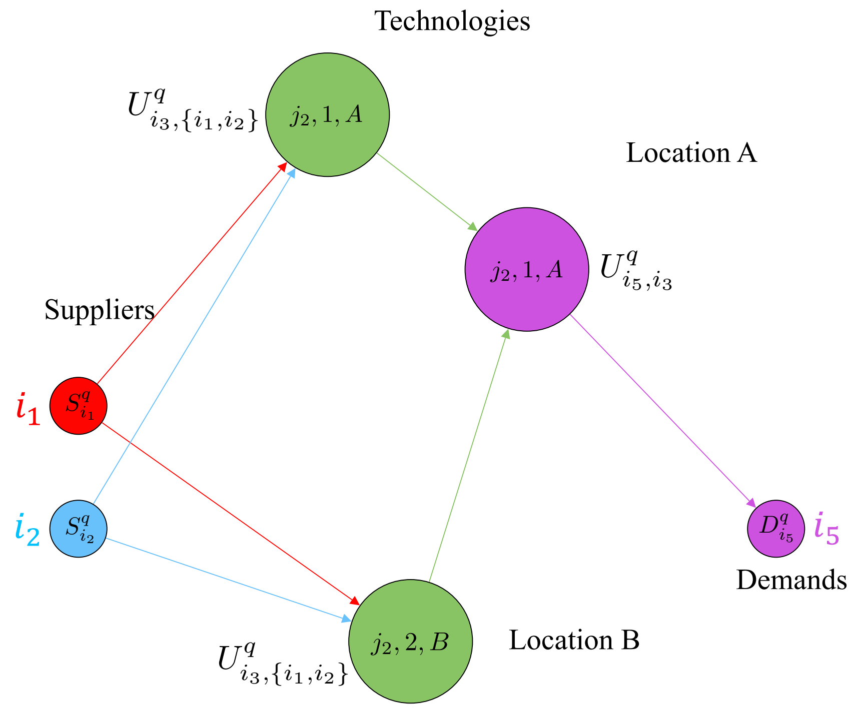


Figure 5: Example of a feasible path obtained from a spatial superstructure.

### 3.1 Computing a Product Hierarchy

Generating a superstructure and a spatial superstructure requires that we compute the number of all possible pathways based on their product-tech connectivity encoded in the max p-graph. Here, we assume that there are no cycles in the dependencies in the underlying max p-graph. For instance, if producing product $i_1$ requires some $i_2$, producing $i_2$ cannot require product $i_1$. With this, we can define a product hierarchy that moves from primary products to intermediate products and to final products. We use $\beta^{i'}_{i,t}, i' \in \mathcal{I}, i \in \Omega_t, t \in \mathcal{T}$ to represent that producing a unit of product $i \in \Omega_t$ requires $\beta^{i'}_{i,t}$ units of product $i'$. These quantities can be interpreted as techn yield/transformation factors.

Table 1: Example of product dependencies in technologies.

| | $i_3$ | | $i_5$ | |
|---|---|---|---|---|
| | $\mathcal{T}_{i_3,i_1,j_1}$ | $\mathcal{T}_{i_3,\{i_1,i_2\},j_2}$ | $\mathcal{T}_{i_5,\{i_3,i_4\},j_1}$ | $\mathcal{T}_{i_5,i_3,j_2}$ |
| $i_1$ | 2 | 1 | 0 | 0 |
| $i_2$ | 0 | 1.5 | 0 | 0 |
| $i_3$ | 0 | 0 | 0.8 | 1 |
| $i_4$ | 0 | 0 | 1 | 0 |

Taking the example form the previous section, we consider the product-tech dependencies of a system that involves supplies of $i_1$ and $i_2$ and produces the intermediate product $i_3$ using techs $\mathcal{T}_{i_3,i_1,j_1}$ and $\mathcal{T}_{i_3,\{i_1,i_2\},j_2}$ and a system that involves supplies of $i_3$ and $i_4$ and produces the final product $i_5$ using techs $\mathcal{T}_{i_5,\{i_3,i_4\},j_1}$ and $\mathcal{T}_{i_5,i_3,j_2}$. The product dependencies are shown in Table 1. Rows in this table represent input products while the columns represent output products. Specifically, producing $i_3$ requires 2 units of $i_1$ using $\mathcal{T}_{i_3,i_1,j_1}$ ($\beta^{i_1}_{i_3,\mathcal{T}_{i_3,i_1,j_1}} = 2$) or 1 unit of $i_1$ and 1.5 unit of $i_2$ using $\mathcal{T}_{i_3,\{i_1,i_2\},j_2}$ ($\beta^{i_1}_{i_3,\mathcal{T}_{i_3,\{i_1,i_2\},j_1}} = 1$, and $\beta^{i_2}_{i_3,\mathcal{T}_{i_3,\{i_1,i_2\},j_1}} = 1.5$), and it is represented in entries 2 and 1 in the first row ($i_1$ as the feed)and first two columns ($i_3$ as the product) in the table. Producing $i_5$ requires 0.8 unit of $i_3$ and 1 unit of $i_4$ using $\mathcal{T}_{i_5,\{i_3,i_4\},j_1}$ ($\beta^{i_3}_{i_5,\mathcal{T}_{i_5,\{i_3,i_4\},j_1}} = 0.8$ and $\beta^{i_4}_{i_5,\mathcal{T}_{i_5,\{i_3,i_4\},j_1}} = 1$) or 1 unit of $i_3$ using $\mathcal{T}_{i_5,i_3,j_2}$ ($\beta^{i_3}_{i_5,\mathcal{T}_{i_5,i_3,j_1}} = 1$), and it is represented in entries $0.8$ and $1$ in the third row ($i_3$ as the feed) and last columns ($i_5$ as the product) of the table. Here, $i_1$, $i_2$ and $i_4$ can be seen as a primary product, $i_3$ can be seen as an intermediate product and $i_5$ can be treated as a product. In this case, $i_3$ has a higher hierarchy than $i_1$ and $i_2$ because producing it depends on these products. Product $i_5$ has the highest hierarchy since no other products depend on it. Because there are no dependencies between products $i_1$, $i_2$, and $i_4$, the hierarchy among them can be arbitrary. Therefore, a possible hierarchy of these five products is $\{i_1 : 3, i_2 : 3, i_3 : 2, i_4 : 3, i_5 : 1\}$. Obtaining the product hierarchy allows us to estimate the amount of each product and the number of techs needed for the system to satisfy a set of demands; this information is necessary for generating the superstructure and spatial superstructure. Unfortunately, when the product-tech dependency becomes complicated, such product hierarchical order is not easy to observe; therefore, we formulate an optimization problem that determines the hierarchical level of each product.

We define a positive integer variable $x_i, i \in \mathcal{I}$ that represents the hierarchy of each product $i$. The optimization formulation that computes the product hierarchy is:

$$\min_{x} \sum_{i \in \mathcal{I}} x_i \tag{3.2a}$$

$$\text{s.t. } x_i \leq x_{i'} - 1, t \in \mathcal{T}, i \in \Omega_t, i' \in \mathcal{K}_t \tag{3.2b}$$

$$x_i \geq 1, i \in \mathcal{I}, \tag{3.2c}$$

Minimizing the objective (3.2a) ensures that the hierarchies for all products are consecutive numbers. Constraint (3.2b) indicates that, if producing product $i$ requires product $i'$, the hierarchy of $i$ should be higher than the hierarchy of $i'$. The final constraint makes sure that the highest hierarchy starts with a value of one. This computation of the hierarchical level for each product aids the computation of the number of each techs possibly required, which is necessary information for generating the superstructures. Then, without further computations, we are able to derive a connectivity matrix (adjacency matrix) for all the nodes (products and techs) based on the information of products involved and their interdependencies, and thus build the graph representation of the superstructures.

We define the adjacency matrix for the superstructure and spatial superstructure as $\mu_{k,k'}, k, k' \in \mathcal{N}^s$ and $\mu_{k,k'}, k, k' \in \mathcal{N}^q$, respectively. The adjacency matrix is a fundamental quantity that encodes the topology of the superstructures.

## 3.2 Computing Optimal Designs from Superstructures

We first derive an optimization formulation to identify an optimal design from a superstructure that minimizes cost and maximizes modularity; this will generate a modular process design, as a superstructure does not encode locational context. We will then extend the formulation to identify an optimal design for a supply chain from a spatial superstructure.

### 3.2.1 Cost-Minimizing Optimal Design

A feasible path (a process design) is obtained by extracting a subgraph (nodes and edges) from the superstructure graph. Our goal is that this feasible path minimizes cost and maximizes modularity). The graph representation of the superstructure will allow us to derive an intuitive (and computable) measure for modularity, this measure will implicitly capture logistical flexibility (e.g., by capturing module sizes and connectivity).

We consider the overall system cost is the net present value for the annualized capital and operational cost. Associated with each tech $t \in \mathcal{T}$, we define the installation and operational cost as $\alpha_t^\xi$ and $\alpha_t^o$, and we define the capacity of tech $t$ that produces product $i \in \Omega_t$ as $\xi_{t,i}$. For each node $u \in \mathcal{U}^s$ and its associated tech $\tau_u$, we define the capital cost, operational cost, and capacity as $\alpha_{\tau_u}^\xi$, $\alpha_{\tau_u}^o$, and $\xi_{\tau_u,\Omega_u}$ and we use the short-hand notation $\alpha_u^\xi$, $\alpha_u^o$, and $\xi_{u,i}, i \in \Omega_u$ (the capital cost is annualized with factor $\epsilon_a$). The unit cost of each material is defined as $\alpha_i^\rho, i \in \mathcal{R}$ and the required amount of final product $i$ is $\delta_i, i \in \mathcal{P}$. For simplicity, we assume that the cost for every connection/transport, denoted as $\alpha^f$, is the same regardless of the product and scales linearly with the amount of product that it carries. The disposal cost of any excess product is denoted as $\alpha_i^d, i \in \mathcal{I}$; this disposal cost allows us to capture potential environmental impacts (e.g., carbon emissions).

We define a collection of continuous variables $f_{k,k'}, k, k' \in \mathcal{N}^s$ representing the flow of product from node $k$ to node $k'$. We define a continuous variable $v_i, i \in \mathcal{R}$ that represents the amount of each primary product purchased from suppliers. We define a binary variable $y_u, u \in \mathcal{U}^s$ such that $y_u = 1$ if node $u$ (a unit) is selected as part of the design, and $y_u = 0$ otherwise.

Under these definitions, the total annualized cost is:

$$C = \sum_{u \in \mathcal{U}^s} (\epsilon_a \cdot \alpha_u^\xi + \alpha_u^o) \cdot y_u + \sum_{s \in \mathcal{S}^s, i \in \Omega_s} \alpha_i^\rho \cdot v_i + \sum_{k,k' \in \mathcal{N}^s} \alpha^f \cdot f_{k,k'} + \sum_{u \in \mathcal{U}^s, i \in \Omega_u} (\xi_u \cdot y_u - \sum_{k \in \mathcal{N}^s \text{ if } i \in \mathcal{K}_k} f_{u,k}) \cdot \alpha_i^d. \tag{3.3}$$

The total cost captures installation cost, the cost of purchasing primary products, the cost of transport, and the cost of waste disposal. The installation cost implicitly captures economies of scale (as it captures technology cost based on size/capacity).

There are a couple of constraint sets that contribute to the formulation. The first set of constraints ensures that the feasible path is derived from the superstructure and can be expressed as:

$$f_{k,k'} \leq M \cdot \mu_{k,k'}, k, k' \in \mathcal{N}^s \tag{3.4}$$

where $M$ is a sufficiently large coefficient; this constraint reduces the feasible region of the problem. The second set of constraints are the product balances at the graph nodes:

$$\sum_{k \in \mathcal{N}^s} f_{s,k} \leq v_i, s \in \mathcal{S}^s, i \in \Omega_s \tag{3.5a}$$

$$\sum_{k' \in \mathcal{N}^s} f_{u,k'} \leq \xi_{u,i} \cdot y_u, u \in \mathcal{U}^s, i \in \Omega_u \tag{3.5b}$$

$$\sum_{k \in \mathcal{N}^s} f_{k,u} = \xi_{u,i'} \cdot \beta_{i,\tau_u}^{i'} \cdot y_u, u \in \mathcal{U}^s, i' \in \mathcal{K}_u, i \in \Omega_u \tag{3.5c}$$

$$\sum_{k \in \mathcal{N}^s} f_{k,d} = \delta_{i'}, d \in \mathcal{D}^s, i' \in \mathcal{K}_d. \tag{3.5d}$$

Constraint (3.5a) is the product balance for supplier nodes, (3.5b) and (3.5c) ensure the inlet and outlet balance for tech nodes, and constraint (3.5c) is the balance for the demand node.

To illustrate the definition of the variables and constraints in the optimization formulation, we will use the same example discussed in the previous section. Consider that an optimal design is derived from a superstructure as shown in Figure 6. Nodes in the design on the right are marked from $k_1$ through $k_5$, where $k_1$ and $k_2$ are suppliers, $k_3$, $k_4$ and $k_5$ are technologies and $k_6$ is demand. If nodes $k_3$ and $k_4$ (copies of the same technology) take 400 units of primary product $i_1$ and 200 units of $i_2$ and produces 200 units of $i_3$. Product streams $i_3$ are then fed into node $k_5$ that produces 300 units of final product $i_5$ that is required by the market. The non-zero entries for each decision variables $f$, $v$, and $y$ are highlighted.

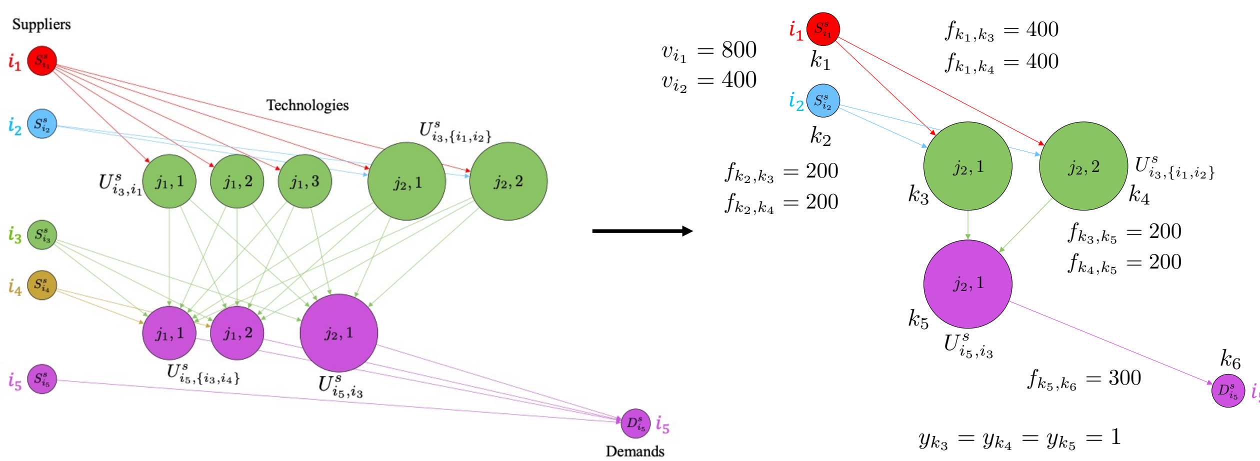


Figure 6: Illustration of notation for optimal design from superstructure.

With these definitions and constraints, we formulate the optimal design problem:

$$\min_{f,v,y} \ C \tag{3.6a}$$

$$\text{s.t. } f_{k,k'} \leq M \cdot \mu_{k,k'}, k, k' \in \mathcal{N}^s \tag{3.6b}$$

$$\sum_{k \in \mathcal{N}^s} f_{s,k} \leq v_i, s \in \mathcal{S}^s, i \in \Omega_s \tag{3.6c}$$

$$\sum_{k' \in \mathcal{N}^s \text{ if } i \in \mathcal{K}_{k'}} f_{u,k'} \leq \xi_{u,i} \cdot y_u, u \in \mathcal{U}^s, i \in \Omega_u \tag{3.6d}$$

$$\sum_{k \in \mathcal{N}^s} f_{k,u} = \xi_{u,i'} \cdot \beta^{i'}_{i,\theta_u} \cdot y_u, u \in \mathcal{U}^s, i' \in \mathcal{K}_u, i \in \Omega_u \tag{3.6e}$$

$$\sum_{k \in \mathcal{N}^s} f_{k,d} = \delta_{i'}, d \in \mathcal{D}^s, i' \in \mathcal{K}_d \tag{3.6f}$$

$$f_{k,k'} \geq 0, k, k' \in \mathcal{N}^s \tag{3.6g}$$

This formulation uses the superstructure connectivity to reduce the feasible space of the problem and is concise to read and easy to understand. We will now expand this formulation by incorporating a modularity measure as a trade-off of cost.

### 3.2.2 Cost-Minimizing Optimal Modular Design

The modularity measure that we adopt is a modified version of that presented in [20]. This measure is directly derived from graph-theoretical principles, can be computed using mixed-integer optimization techniques, and captures aspects of relevance in the context of manufacturing systems and supply chains. Specifically, the measure captures dimension (size) of modules, which enables capturing the fact that such modules should be transportable. The modularity measure proposed is computationally more suitable for system design.

The modularity measure is ccomputed using concept of graph coverage; this is done by formulating a mixed integer optimization problem that minimizes the number of intermodular edges relative to the total number of edges (intra- and inter-modular). In other words, the measure aims to capture the ability of assembling/disassembling a system. The measure is defined as $M_n$ with $n$ being the predefined number of modules. The measure $M_n$ has a range has a range of $[0, 1]$; $M_n = 1$ being the most modular system possible and $M_n = 0$ represents the least modular system possible.

In addition to the previous attributes of tech $t \in \mathcal{T}$, we define a dimension (physical size) of the tech represented as $\gamma_t$. Similarly, for each node $u \in \mathcal{U}^s$ and its associated tech $t = \tau_u$, the dimension is represented as $\gamma_{\tau_u}$, abbreviated as $\gamma_u$. The set of modules is defined as $\mathcal{L} = \{1, 2, ..., n\}$ and $|\mathcal{L}| = n$ (the number of modules). We impose dimensionality constraints for each module by defining $\bar{D}$ and $\underline{D}$ as upper and lower limits.

To compute the *modularity measure* of a given feasible path (a potential design), we define the collection of binary parameters $a_{u,u'}, u, u' \in \mathcal{U}^s$, which represent the adjacency matrix of the subgraph associated to the feasible path. We define the collection of binary variables $a^m_{u,u',l}, u, u' \in \mathcal{U}^s, l \in \mathcal{L}$; here, $a^m_{u,u',l} = 1$ indicates that nodes $u, u'$ are connected and they are both in module $l$. We define the binary variable $y_{u,l}, u \in \mathcal{U}^s, l \in \mathcal{L}$ such that $y_{u,l} = 1$ if node (unit) $u$ exists in the design and appears in module $l$.

The measure that we propose is computed by minimizing the following function for a pre-defined number of modules $n$:

$$M_n = \frac{\displaystyle\sum_{u,u' \in \mathcal{U}^s, l \in \mathcal{L}} a^m_{u,u',l}}{\displaystyle\sum_{u,u' \in \mathcal{U}^s} a_{u,u'}}. \tag{3.7}$$

We note that this modularity measure is different from that reported in [20], which is:

$$M_n = \frac{\displaystyle\sum_{u,u' \in \mathcal{U}^s} \pi_{u,u'} \cdot a_{u,u'}}{\displaystyle\sum_{u,u' \in \mathcal{U}^s} a_{u,u'}}. \tag{3.8}$$

where $\pi$ is the membership variable matrix. Here, we have that the entry $\pi_{u,u'}$ is 1 if node $u$ and $u'$ are in the same module. We highlight that, in the work of [20], the adjacency $a_{u,u}$ is a fixed parameter and $\sum_{u,u' \in \mathcal{U}^s} a_{u,u'} = 2m$, where $m$ is the number of edges in the graph. However, in the design context discussed here, the adjacency $a_{u,u}$ is a variable (affected by the design selection). As such, the second modularity measure would be computationally difficult to implement. This motivates our desire to use the first modularity measure (we will see that this is easier to implement).

We now proceed to show that the modularity measures are equivalent; we establish this result by showing that:

$$\sum_{l \in \mathcal{L}} a^m_{u,u',l} = \pi_{u,u'} \cdot a_{u,u'}. \tag{3.9}$$

In other words, we aim show that the numerators of both modularity measures are equivalent. Specifically, we would like to show that both numerators are binary and that, when the numerator on the left takes a value of 1, the numerator on the right also takes a value of 1.

The right numerator is binary because both terms are binary and thus their product is binary. For the left numerator we have that, by definition, there are no overlapping modules (each node can exist only in one module). For a module $l' \in \mathcal{L}$ we thus have that:

$$a^m_{u,u',l'} = 1 \Longleftrightarrow \sum_{l \in \mathcal{L}} a^m_{u,u',l} = 1 \tag{3.10}$$

and

$$\sum_{l \in \mathcal{L}} a^m_{u,u',l} \leq 1. \tag{3.11}$$

Because the variable $a^m$ is also binary, we have that it can only take values of 0 and 1. We then have that, by definition, the term $a^m_{u,u',l'}$ takes a value of 1 if and only if node $u$ connects to node $u'$ and they are both in module $l$. Therefore, for a module $l' \in \mathcal{L}$, we have:

$$a^m_{u,u',l'} = 1 \Longleftrightarrow a_{u,u'} = 1 \text{ and } \pi_{u,u'} = 1. \tag{3.12}$$

Combining these expressions:

$$\sum_{l \in \mathcal{L}} a^m_{u,u',l} = 1 \Longleftrightarrow a_{u,u'} = 1 \text{ and } \pi_{u,u'} = 1. \tag{3.13}$$

Therefore, we can rewrite the above expression as:

$$\sum_{l \in \mathcal{L}} a^m_{u,u',l} = a_{u,u'} \cdot \pi_{u,u'} \tag{3.14}$$

which establishes the equivalence.

Note that the new modularity measure proposed is nonlinear (fractional) and would make the design formulation intractable if this is added directly as an objective function. Interestingly, however, when the measure is used as a constraint, this can be reformulated in linear form as:

$$\sum_{u,u' \in \mathcal{U}^s, l \in \mathcal{L}} a^m_{u,u',l} \geq \epsilon_M \cdot \sum_{u,u' \in \mathcal{U}^s} a_{u,u'}, \tag{3.15}$$

where $\epsilon_M$ is a desired threshold value for the modularity measure. This observation is relevant because the design formulation is multi-objective (minimize cost and maximize modularity). Using an $\epsilon$-constrained method to compute the Pareto frontier thus provides a natural approach to deal with the modularity measure.

In addition to the modularity constraint and the constraints in formulation (3.6), new constraints are added the impact of the assignment of nodes into modules on process variables. The logic between continuous flow variables and the binary adjacency variables is:

$$f_{k,k'} \leq M \cdot a_{k,k'}, k, k' \in \mathcal{N}^s \tag{3.16}$$
$$a_{k,k'} \leq f_{k,k'}, k, k' \in \mathcal{N}^s. \tag{3.17}$$

The logic for variable $y$ (a node can only exist in one module), variable $a^m$, and $a$ (a node $u$ connects to any node $u'$ and they all belong to a module $l$ only if node $u$ and $u'$ both exist) are:

$$\sum_{l \in \mathcal{L}} y_{u,l} \leq 1, u \in \mathcal{U}^s \tag{3.18}$$
$$\sum_{u' \in \mathcal{U}^s} a^m_{u,u',l} + \sum_{u' \in \mathcal{U}^s} a^m_{u',u,l} \leq M \cdot y_{u,l}, u \in \mathcal{U}^s, l \in \mathcal{L} \tag{3.19}$$
$$\sum_{l \in \mathcal{L}} a^m_{u,u',l} \leq a_{u,u'}, u, u' \in \mathcal{U}^s \tag{3.20}$$

The constraint that governs the upper and lower bound for the dimensionality of each module is:

$$\underline{D} \leq \sum_{u \in \mathcal{U}^s} y_{u,l} \cdot \gamma_u \leq \bar{D}, l \in \mathcal{L}. \tag{3.21}$$

Figure 7 presents an example to demonstrate the definitions of variables and constraints. Here, we show a modular division of the optimal design where $k_3$ and $k_5$ are in module $l_1$ and $k_4$ is in module $l_2$. It this worth noticing is that only one entry of variable $a^m$ is 1 since only the edge that connects node $k_3$ and $k_5$ is counted as the edge within modules.

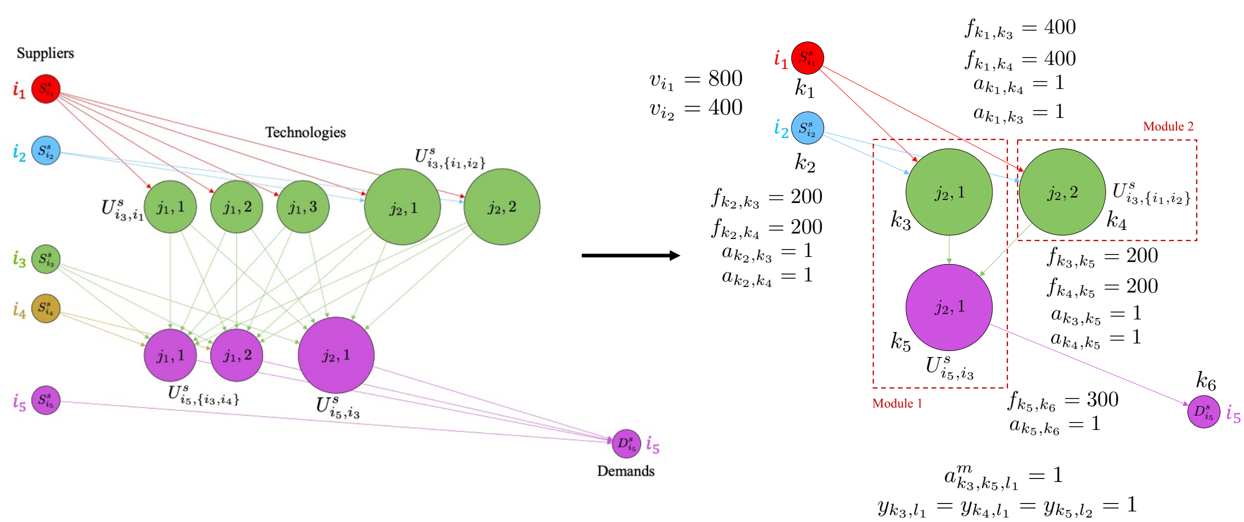


Figure 7: Illustration of notation for optimal design (left) obtained from superstructure (right).

The optimization formulation to select the cost-optimal design given a modularity threshold $\epsilon_M$ is:

$$
\begin{aligned}
\min_{f,v,a,a^m,y} \quad & C && \text{(3.22a)}\\
\text{s.t.} \quad & f_{k,k'} \leq M \cdot \mu_{k,k'}, k, k' \in \mathcal{N}^s && \text{(3.22b)}\\
& \sum_{k \in \mathcal{N}^s} f_{s,k} \leq v_i, s \in \mathcal{S}^s, i \in \Omega_s && \text{(3.22c)}\\
& \sum_{k' \in \mathcal{N}^s} f_{u,k'} \leq \xi_{u,i} \cdot \sum_{l \in \mathcal{L}} y_{u,l}, u \in \mathcal{U}^s, i \in \Omega_u && \text{(3.22d)}\\
& \sum_{k \in \mathcal{N}^s} f_{k,u} = \xi_{u,i'} \cdot \beta_{i,\tau_u}^{i'} \cdot \sum_{l \in \mathcal{L}} y_{u,l}, u \in \mathcal{U}^s, i' \in \mathcal{K}_u, i \in \Omega_u && \text{(3.22e)}\\
& \sum_{k \in \mathcal{N}^s} f_{k,d} = \delta_{i'}, d \in \mathcal{D}^s, i' \in \mathcal{K}_d && \text{(3.22f)}\\
& f_{k,k'} \leq M \cdot a_{k,k'}, k, k' \in \mathcal{N}^s && \text{(3.22g)}\\
& a_{k,k'} \leq f_{k,k'}, k, k' \in \mathcal{N}^s && \text{(3.22h)}\\
& \sum_{l \in \mathcal{L}} y_{u,l} \leq 1, u \in \mathcal{U}^s && \text{(3.22i)}\\
& \sum_{u' \in \mathcal{U}^s} a^m_{u,u',l} + \sum_{u' \in \mathcal{U}^s} a^m_{u',u,l} \leq M \cdot y_{u,l}, u \in \mathcal{U}^s, l \in \mathcal{L} && \text{(3.22j)}\\
& \sum_{l \in \mathcal{L}} a^m_{u,u',l} \leq a_{u,u'}, u, u' \in \mathcal{U}^s && \text{(3.22k)}\\
& \underline{D} \leq \sum_{u \in \mathcal{U}^s} y_{u,l} \cdot \gamma_u \leq \bar{D}, l \in \mathcal{L} && \text{(3.22l)}\\
& \sum_{u,u' \in \mathcal{U}^s, l \in \mathcal{L}} a^m_{u,u',l} \geq \epsilon_M \cdot \sum_{u,u' \in \mathcal{U}^s} a_{u,u'} && \text{(3.22m)}\\
& f_{k,k'} \geq 0, k, k' \in \mathcal{N}^s && \text{(3.22n)}
\end{aligned}
$$

Incorporating the modularity measure in the design formulation increases the number of variables and constraints, which are required for the calculation of intra-modular edges. However, this formulation gives us interesting insights on connectivity, decentralization, and transportability of the process design. Formulation (3.6) and (3.22) find the optimal modular design based on the superstructure graph (that does not account for spatial information). We will now introduce formulations that find the optimal feasible path with spatial information, and we would like to show that with the concept of spatial superstructure, these formulations are similar to formulations (3.6) and (3.22).

## 3.3 Computing Optimal Designs from Spatial Superstructures

We then consider the situation that we not only assign technologies to modules, but also put them at different locations. Here, we consider two potential factors that might change the geological preference for different technologies. First, the installation cost may be different for the same technology at different locations due to the different cost of land. Second, the transportation cost from one location to another may be different for the same product. Specifically, if technologies connected to each other are installed at the same location, the short-range transport of product can be achieved using pipelines or other simple methods. If they are placed at different locations, trucks or trains may be utilized to long-range transport of the product. Alternative for short- and long-range transport options are directly captured by our formulation. Also, the primary products are usually supplied at some certain locations and the final products are usually transported to the location of demand. We further assume that technologies within the same module should be placed together in the same location and multiple modules can be placed at the same location. We first formulate the problem that solves for the cost-minimizing optimal spatial feasible path and then incorporate the modularity measure for an optimal modular design.

### 3.3.1 Cost-Minimizing Optimal Design with Spatial Information

The graph-theoretic representation of the spatial superstructure makes the notation of the optimization formulation directly analogous to that of the superstructure. As such, we briefly discuss all the necessary definitions for the formulation.

The capacity associated with technology $t = \tau_u$ for node $u \in \mathcal{U}^q$ are abbreviated as $\xi_u$. The installation cost associated with each technology $t \in \mathcal{T}$ at location $g \in \mathcal{G}_t$ is defined as $\alpha^{\xi}_{t,g}$. For the node $u \in \mathcal{U}^q$ associated with tech$\tau_u$, the installation cost can be represented as $\alpha^{\xi}_{\tau_u,\phi_u}$, abbreviated similarly as $\alpha^{\xi}_u$. The unit cost of each product is $\alpha^{\rho}_i$ and the demand of the market for each product $i \in \mathcal{P}$ is defined as $\delta_i$, and the cost of disposal for each product is defined as $\alpha^{d}_i$. We redefine the cost for connection/flow of product $i$ from location $g$ to location $g'$ which stands for the cost of transportation as $\alpha^{f}_{i,g,g'}, i \in \mathcal{I}, g, g' \in \mathcal{G}$. Note that if $g = g'$, we obtain the short-term transport cost (as in a typical process).

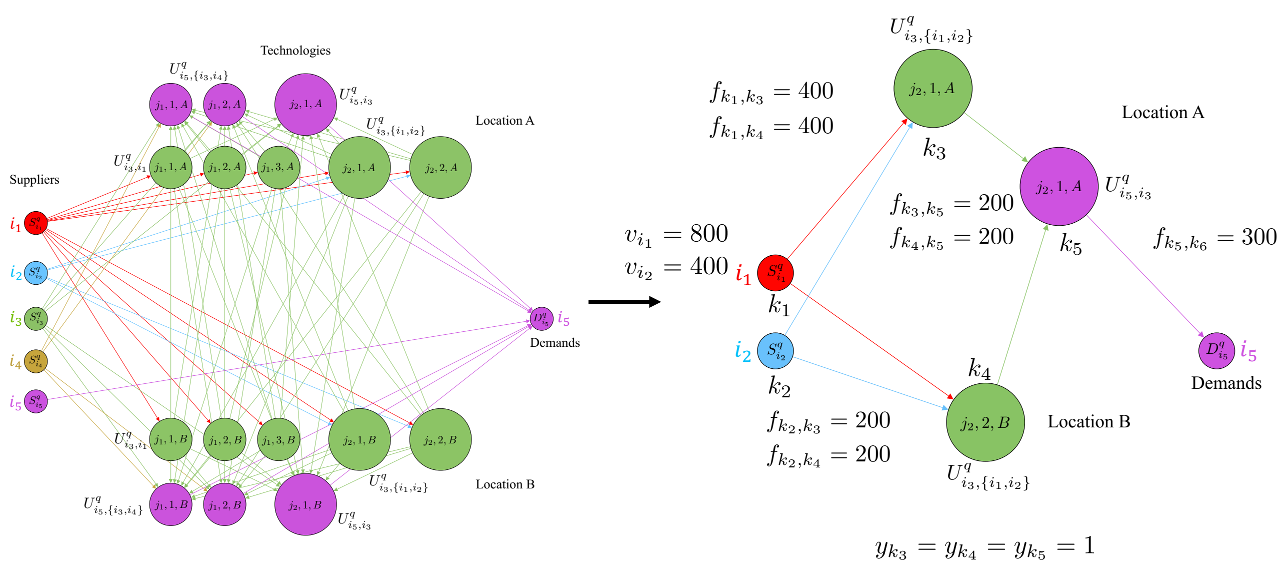


Figure 8: Illustration of notation for optimal design (right) obtained from spatial superstructure (left).

The variables defined for the spatial superstructure problem have a similar interpretation as those of the superstructure but we can now attribute locational context. Specifically, we define a continuous variable $f_{k,k'}, k, k' \in \mathcal{N}^q$ representing the flow of product from node $k$ at location $\phi_k$ to node $k'$ at location $\phi_{k'}$. We also define a continuous variable for the purchase of each primary product as $v_i, i \in \mathcal{R}$. We define a binary variable matrix $y_u, u \in \mathcal{U}^q$ such that $y_u = 1$ if node $u$ is selected as part of the design located at location $\phi_u$, and $y_u = 0$ otherwise. Therefore, the total annualized cost of the system can be written as:

$$C = \sum_{u \in \mathcal{U}^q} (\epsilon_a \cdot \alpha^\xi{}_u + \alpha^o_u) \cdot y_u + \sum_{s \in \mathcal{S}^q, i \in \Omega_s} \alpha^\rho_i \cdot v_i + \sum_{i \in \Omega_k, k \in \mathcal{N}^q, k' \in \mathcal{N}^q} \alpha^f_{i,\phi_k,\phi_{k'}} \cdot f_{k,k'} + \sum_{u \in \mathcal{U}^q, i \in \Omega_u} (\xi_u \cdot y_u - \sum_{k \in \mathcal{N}^q \text{ if } k \in \mathcal{K}_k} f_{u,k}) \cdot \alpha^d_i. \tag{3.23}$$

The only difference in this cost function is that we can now capture transport cost. An illustration of the problem variables is provided in Figure 8.

The formulation to obtain a cost-minimizing design from the spatial superstructure is:

$$\min_{f,v,y} \; C \tag{3.24a}$$
$$\text{s.t. } f_{k,k'} \leq M \cdot \mu_{k,k'}, k, k' \in \mathcal{N}^q \tag{3.24b}$$
$$\sum_{k \in \mathcal{N}^q} f_{s,k} \leq v_i, s \in \mathcal{S}^q, i \in \Omega_s \tag{3.24c}$$
$$\sum_{k' \in \mathcal{N}^q \text{ if } i \in \mathcal{K}_{k'}} f_{u,k'} \leq \xi_{u,i} \cdot y_u, u \in \mathcal{U}^q, i \in \Omega_u \tag{3.24d}$$
$$\sum_{k \in \mathcal{N}^q} f_{k,u} = \xi_{u,i'} \cdot \beta^{i'}_{i,\tau_u} \cdot y_u, u \in \mathcal{U}^q, i' \in \mathcal{K}_u, i \in \Omega_u \tag{3.24e}$$
$$\sum_{k \in \mathcal{N}^q} f_{k,d} = \delta_{i'}, d \in \mathcal{D}^q, i' \in \mathcal{K}_d \tag{3.24f}$$
$$f_{k,k'} \geq 0, k, k' \in \mathcal{N}^q \tag{3.24g}$$

The spatial superstructure is encoded $\mu_{k,k'}, k, k' \in \mathcal{N}^q$. This optimization formulation is directly analogous to that in (3.6) but can be computationally more challenging to solve because one can account for multiple possible locations for techs, suppliers, and demands.

### 3.3.2 Cost-Minimizing Optimal Modular Design with Spatial Information

The modularity measure is directly analogous to the one defined previously but we need to specify additional information to account for location of modules. The dimension associated with technology $t = \tau_u$ for node $u \in \mathcal{U}^q$ are abbreviated as $\gamma_u$. We use set $\mathcal{L} = \{1, 2, ..., n\}$ for the set of modules, and $\bar{D}$ and $\underline{D}$ to represent the upper and lower limits of the dimensionality requirements. We use the binary variable matrix $a_{k,k'}, k, k' \in \mathcal{N}^q$ to represent the adjacency matrix of the feasible path (subgraph of the spatial superstructure). The binary variable $a^m_{u,u',l}, u, u' \in \mathcal{U}^q, l \in \mathcal{L}$ represents the relationship between node $u$, $u'$ and module $l$. We define the binary variable $y_{u,l}, u \in \mathcal{U}^q, l \in \mathcal{L}$ such that $y_{u,l} = 1$ if node $u$ belongs to module $l$ at location $\phi_u$ and $y_{u,l} = 0$ otherwise. Finally, we define the binary variable collection $z_{l,g}, l \in \mathcal{L}, g \in \mathcal{G}_t$ such that if module $l$ is placed at location $g$, $z_{l,g} = 1$ and $z_{l,g} = 0$ otherwise.

The modularity measure with predefined number of modules $n$ modules is:

$$M_n = \frac{\sum_{u,u' \in \mathcal{U}^q, l \in \mathcal{L}} a^m_{u,u',l}}{\sum_{u,u' \in \mathcal{U}^q} a_{u,u'}}. \tag{3.25}$$

A couple of additional constrains (compared to formulation (3.22)) are added due to the newly defined variable $z$. The first ensures that one module can only be placed at one location, and the second one ensures the logic between variable $y$ and $z$ (a node $u$ that exists in module $l$ only if the module $l$ exists at location of node $u$), and they are expressed as:

$$\sum_{g \in \mathcal{G}_t} z_{l,g} \leq 1, l \in \mathcal{L} \tag{3.26}$$

$$y_{u,l} \leq z_{l,\phi_u}, u \in \mathcal{U}^q, l \in \mathcal{L}. \tag{3.27}$$

An illustration of the defined variables is shown in Figure 9; note that a couple of entries for the newly defined variable $z$ are 0, as module $l_1$ exists at location A and module $l_2$ exists at location B.

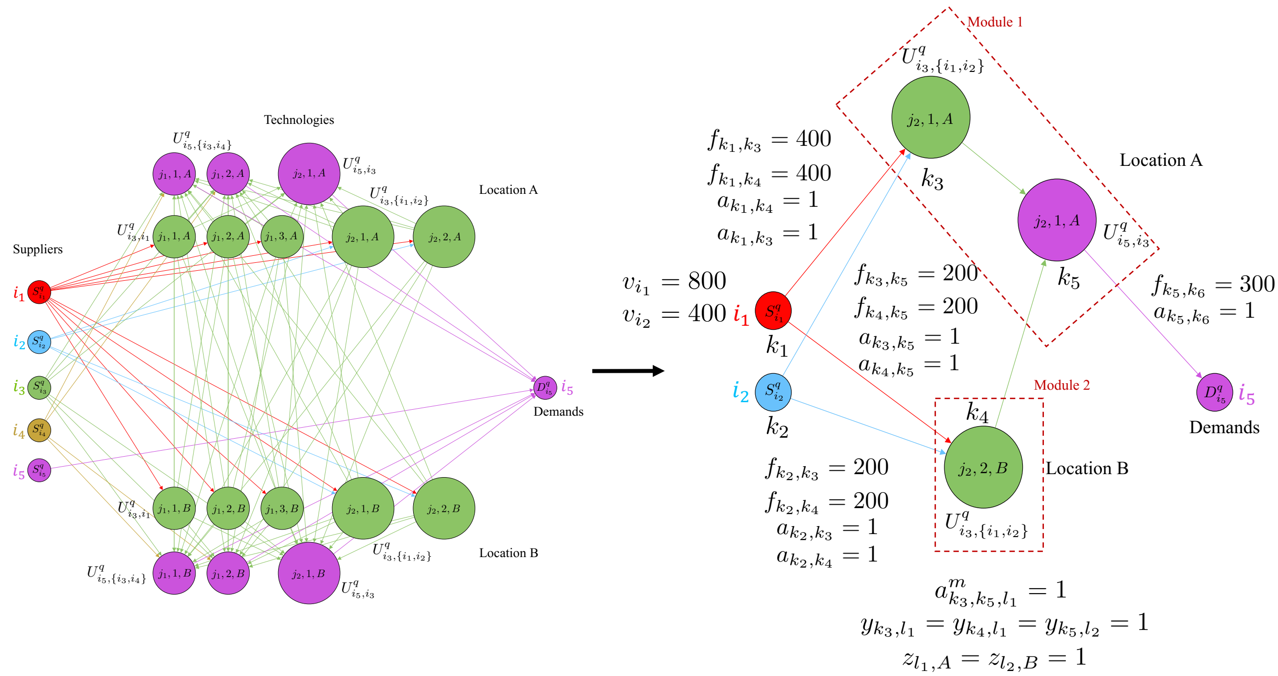


Figure 9: Illustration of notation for optimal modular design from spatial superstructure

With the previous definitions, the optimization formulation is:

$$\min_{f,v,a,a^m,y,z} \quad C \tag{3.28a}$$

$$\text{s.t. } f_{k,k'} \leq M \cdot \mu_{k,k'}, k, k' \in \mathcal{N}^q \tag{3.28b}$$

$$\sum_{k \in \mathcal{N}^q} f_{s,k} \leq v_i, s \in \mathcal{S}^q, i \in \Omega_s \tag{3.28c}$$

$$\sum_{k' \in \mathcal{N}^q} f_{u,k'} \leq \xi_{u,i} \cdot \sum_{l \in \mathcal{L}} y_{u,l}, u \in \mathcal{U}^q, i \in \Omega_u \tag{3.28d}$$

$$\sum_{k \in \mathcal{N}^q} f_{k,u} = \xi_{u,i'} \cdot \beta^{i'}_{i,\tau_u} \cdot \sum_{l \in \mathcal{L}} y_{u,l}, u \in \mathcal{U}^q, i' \in \mathcal{K}_u, i \in \Omega_u \tag{3.28e}$$

$$\sum_{k \in \mathcal{N}^q} f_{k,d} = \delta_{i'}, d \in \mathcal{D}^q, i' \in \mathcal{K}_d \tag{3.28f}$$

$$f_{k,k'} \leq M \cdot a_{k,k'}, k, k' \in \mathcal{N}^q \tag{3.28g}$$

$$a_{k,k'} \leq f_{k,k'}, k, k' \in \mathcal{N}^q \tag{3.28h}$$

$$\sum_{l \in \mathcal{L}} y_{u,l} \leq 1, u \in \mathcal{U}^q \tag{3.28i}$$

$$\sum_{u' \in \mathcal{U}^q} a^m_{u,u',l} + \sum_{u' \in \mathcal{U}^q} a^m_{u',u,l} \leq M \cdot y_{u,l}, u \in \mathcal{U}^q, l \in \mathcal{L} \tag{3.28j}$$

$$\sum_{l \in \mathcal{L}} a^m_{u,u',l} \leq a_{u,u'}, u, u' \in \mathcal{U}^q \tag{3.28k}$$

$$\underline{D} \leq \sum_{u \in \mathcal{U}^q} y_{u,l} \cdot \gamma_u \leq \bar{D}, l \in \mathcal{L} \tag{3.28l}$$

$$\sum_{u,u' \in \mathcal{U}^q, l \in \mathcal{L}} a^m_{u,u',l} \geq \epsilon_a \cdot \sum_{u,u' \in \mathcal{U}^q} a_{u,u'} \tag{3.28m}$$

$$f_{k,k'} \geq 0, k, k' \in \mathcal{N}^q \tag{3.28n}$$

$$\sum_{g \in \mathcal{G}_t} z_{l,g} \leq 1, l \in \mathcal{L} \tag{3.28o}$$

$$y_{u,l} \leq z_{l,\phi_u}, u \in \mathcal{U}^q, l \in \mathcal{L} \tag{3.28p}$$

This formulation is more comprehensive that the one based on a superstructure in that it delivers an optimal system design that not only captures techs needed but also their geographical location. In other words, this formulation simultaneously designs a supply chain and associated processes.

# 4 Case Study

We present a case study to illustrate how our optimization formulations can help automate the generation of superstructures and spatial superstructures and to identify optimal system designs with desired modularity. The study tries to identify an optimal supply chain design for plastic waste upcycling that takes municipal solid waste (MSW) as the input and produces ethylene, propylene and hydrogen as final products. The MILPs were solved using Gurobi (version 9.0.3) and were implemented in the Julia-based JuMP modeling framework. We use Gephi for graph manipulation and visualization. All optimization formulations are solved using a commercial laptop and the solving time is referred to the wall clock time. All code needed to reproduce the results can be found in https://github.com/zavalab/JuliaBox/tree/master/ModularDesign.

## 4.1 Problem Setup and Material Hierarchy of the Process

A high-level view of the processing tasks involved in plastic waste upcycling is provided in Figure 10. Here, MSW (denoted as $i_1$) collected from households is fed into a product recovery facility (MRF) that obtains a plastic bale ($i_2$), the plastic bale goes through a reprocessing facility (RF) that cleans the bale and converts it into plastic flakes ($i_3$), a pyrolysis process (PP) takes the plastic flakes and converts these into pyrolysis gas ($i_4$) and pyrolysis oil ($i_5$), a steam cracking (SC) process obtains the final products, given by ethylene ($i_6$), propylene ($i_7$), and hydrogen ($i_8$). For this system, techs that produce the same products have the same interdependencies between products and they only differ in their capacities. Therefore, we eliminate the attribute of techs and the product dependencies between the different techs is shown in Table 2. Producing 1 unit of $i_2$ requires 7.69 unit of $i_1$; producing 1 unit of $i_3$ requires 1 unit of $i_2$; producing 1 unit of $i_5$ requires 1.29 unit of $i_3$; finally, 1 unit of $i_6$, $i_7$

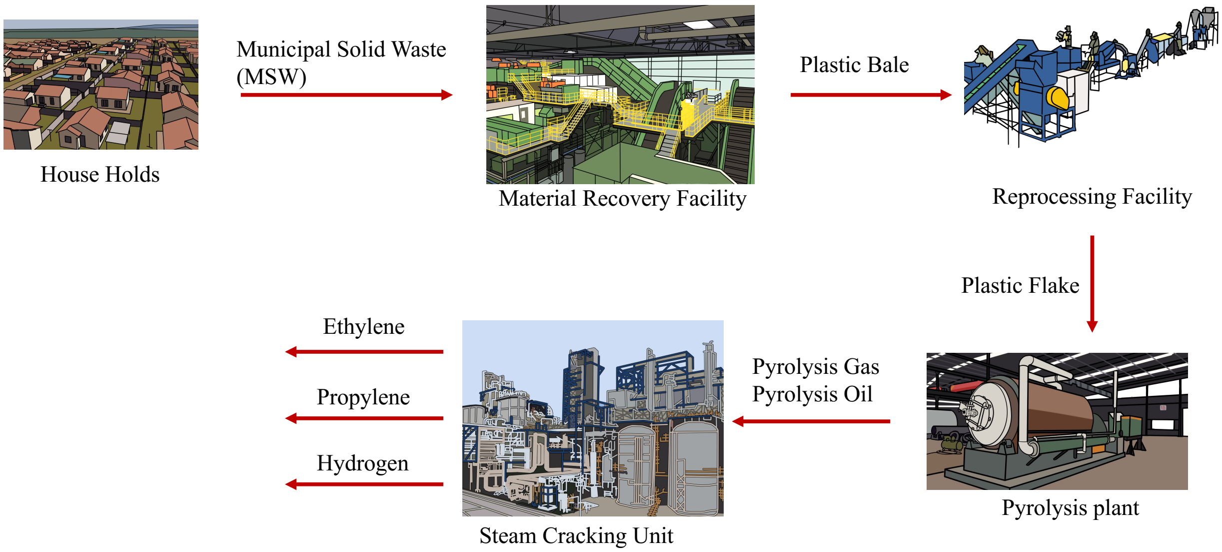


Figure 10: High-level view of processing tasks involved in plastic waste upcycling.

or $i_8$ requires 3.81, 6.16 or 125 unit of $i_5$ respectively. Note that intermediate product $i_4$ is not used in the following process and is therefore considered a waste. Additional information is summarized in Table 3.

Table 2: Product interdependencies between techs.

| | $i_2$ | $i_3$ | $i_5$ | $i_6$ | $i_7$ | $i_8$ |
|---|---|---|---|---|---|---|
| $i_1$ | 7.69 | 0 | 0 | 0 | 0 | 0 |
| $i_2$ | 0 | 1 | 0 | 0 | 0 | 0 |
| $i_3$ | 0 | 0 | 1.29 | 0 | 0 | 0 |
| $i_5$ | 0 | 0 | 0 | 3.81 | 6.16 | 125 |

Each installation cost, operating cost tech dimension and tech capacity is associated with each technology in the above row respectively. For example, for technology $T_{\{i_4,i_5\},i_3,j_1}$, its installation cost is $\$\ 0.46 \times 10^8$, operating cost is $\$\ 14$ per unit of input $i_3$, dimension is 2, and capacities for producing product $i_4$ and $i_5$ are 169,000 tons and 39,000 tons, respectively. We assume that the operating cost for techs producing the same products but of different sizes/kinds is the same. It is worth emphasizing that, to capture economies of scale, the estimation of the installation cost is based on the so-called "2/3 rule" that is prevalent in cost estimation. Specifically, the rule that applies to the different technologies producing product $i_2$ can be expressed as:

$$\left(\frac{\xi_{T_{i_2,i_1,j_1}}}{\xi_{T_{i_2,i_1,j_2}}}\right) = \left(\frac{\alpha^{\xi}_{T_{i_2,i_1,j_1}}}{\alpha^{\xi}_{T_{i_2,i_1,j_2}}}\right)^{\frac{3}{2}}. \tag{4.29}$$

Table 3: Data for plastic upcycling system.

| Parameters | Values |
|---|---|
| Supplied Materials, $\mathcal{R}$ | $[i_1, i_2, i_3, i_5]$ |
| Products, $\mathcal{P}$ | $[i_2, i_3, i_4, i_5, i_6, i_7, i_8]$ |
| All Materials, $\mathcal{I}$ | $[i_1, i_2, i_3, i_4, i_5, i_6, i_2, i_8]$ |
| Technologies , $\mathcal{T}$ | $[T_{i_2,i_1,j_1}, T_{i_2,i_1,j_2}, T_{i_2,i_1,j_3}]$<br>$[T_{i_3,i_2,j_1}, T_{i_3,i_2,j_2}, T_{i_3,i_2,j_3}]$<br>$[T_{\{i_4,i_5\},i_3,j_1}, T_{\{i_4,i_5\},i_3,j_2}, T_{\{i_4,i_5\},i_3,j_3}]$<br>$[T_{\{i_6,i_7,i_8\},i_5,j_1}, T_{\{i_6,i_7,i_8\},i_5,j_2}, T_{\{i_6,i_7,i_8\},i_5,j_3}]$ |
| Installation Cost, $\alpha^{\xi}$ ($\times 10^8$ \$) | [0.27, 0.46, 0.70]<br>[0.13, 0.30, 0.56]<br>[0.46, 0.79, 1.20]<br>[6.05, 9.17 13.90] |
| Operating Cost, $\alpha^{u}$ (\$ / unit of input) | 8.87<br>44.19<br>14<br>71.8 |
| Technology Dimension, $\gamma$ | [3, 5, 8]<br>[3, 5, 8]<br>[2, 4, 8]<br>[3, 5, 8] |
| Technology Capacity, $\xi$ ( $\times 10^4$ tons) | [24.2, 60.5, 120.9]<br>[20.8, 52, 104]<br>[[16.9, 3.9], [42.3, 9.8], [112.8, 26.3]]<br>[[13.1, 8.1, 0.4], [26.1, 16.2, 0.80], [52.3, 32.3, 1.6]] |
| Purchasing Unit Cost, $\alpha^{\rho}_i, i \in \mathcal{R}$, (\$/unit) | [0, 250, 1300, 1100] |
| Disposal Cost, $\alpha^{d}_i, i \in \mathcal{I}$, (\$/ton) | [50, 40, 40, 400, 800, 0, 0, 0] |
| Required Production, $\delta_i, i \in \mathcal{I}$, (tons) | [0, 0, 0, 0, 0, 150000, 100000, 5000] |
| Project Duration, $t_p$ (yrs) | 20 |
| Discount Rate, $r$ | 0.06 |
| Annualization Factor, $\epsilon_a$ | 0.087 |
| Module Dimension Limits, $[\underline{D}, \bar{D}]$ | [2, 12] |

In order to generate the superstructures of the system, we first use the formulation (3.2) to solve for the hierarchy of all products based on their interdependencies. The result shows that $i_4$, $i_6$, $i_7$, and $i_8$ have the highest hierarchy followed by $i_5$, $i_3$, $i_2$ and $i_1$ respectively. This makes sense because products $i_6$, $i_7$, and $i_8$ are the final product of the process, and for intermediate product $i_4$, even though it is the product of an intermediate process, it is the waste of the process that no other products depend on. Therefore, it has the highest hierarchy but it is not a required product in the process. Then, with the hierarchy of each product, we can compute the total amount of each product needed for the system and the number of all possible techs needed for the process. The information is summarized in Table 4.

Table 4: Results for hierarchy of products, quantity of products, and number of technologies.

| Materials | Hierarchy | Required Amount (tons) | Number of Technologies |
|---|---|---|---|
| $i_1$ | 5 | $6.22 \times 10^6$ | – |
| $i_2$ | 4 | $8.09 \times 10^5$ | $T_{i_2,i_1,j_1}$: 4<br>$T_{i_2,i_1 j_2}$: 2<br>$T_{i_2,i_1,j_3}$: 1 |
| $i_3$ | 3 | $8.09 \times 10^5$ | $T_{i_3,i_2,j_1}$: 4<br>$T_{i_3,i_2,j_2}$: 2<br>$T_{i_3,i_2,j_3}$: 1 |
| $i_5$ | 2 | $6.25 \times 10^5$ | $T_{\{i_4,i_5\},i_3,j_1}$: 4<br>$T_{\{i_4,i_5\},i_3,j_2}$: 2<br>$T_{\{i_4,i_5\},i_3,j_3}$: 1 |
| $i_4$ | 1 | 0 | / |
| $i_6$ | 1 | $1.5 \times 10^5$ | $T_{\{i_6,i_7,i_8\},i_5,j_1}$: 2 |
| $i_7$ | 1 | $1 \times 10^5$ | $T_{\{i_6,i_7,i_8\},i_5,j_2}$: 1 |
| $i_8$ | 1 | $5 \times 10^3$ | $T_{\{i_6,i_7,i_8\},i_5,j_3}$: 1 |

With above information, we are now ready to generate the superstructures of the system and then identify an optimal feasible path. We first consider the design using superstructure without spatial information and solve the problem using formulation (3.6) and (3.22). Then, we solve for the spatial optimal design using formulation (3.24) and (3.28).

## 4.2 Optimal System Design without Spatial Information

The superstructure of this system is shown in Figure 11. Looking from top to bottom, we have the product hierarchy of the system, which takes MSW as the input, generates plastic bale/flakes and pyrolysis gas/oil as intermediate products, and generates ethylene, propylene and hydrogen as final outputs. Looking from left to right, we have four nodes representing the supplies of the four primary products, and in the middle nodes with different sizes represent techs of different capacities. The number of copies of each techs coincides with the number shown in Table 4. Finally, the three nodes on the right represent the demands. As expected, the superstructure is dense due to the large number of possible techs.

We assume that the connectivity cost $\alpha^f$ is \$ $0.01$ per unit of product that an edge carries. First, we used formulation (3.6) to solve for the cost-minimizing optimal supply chain design and the result is

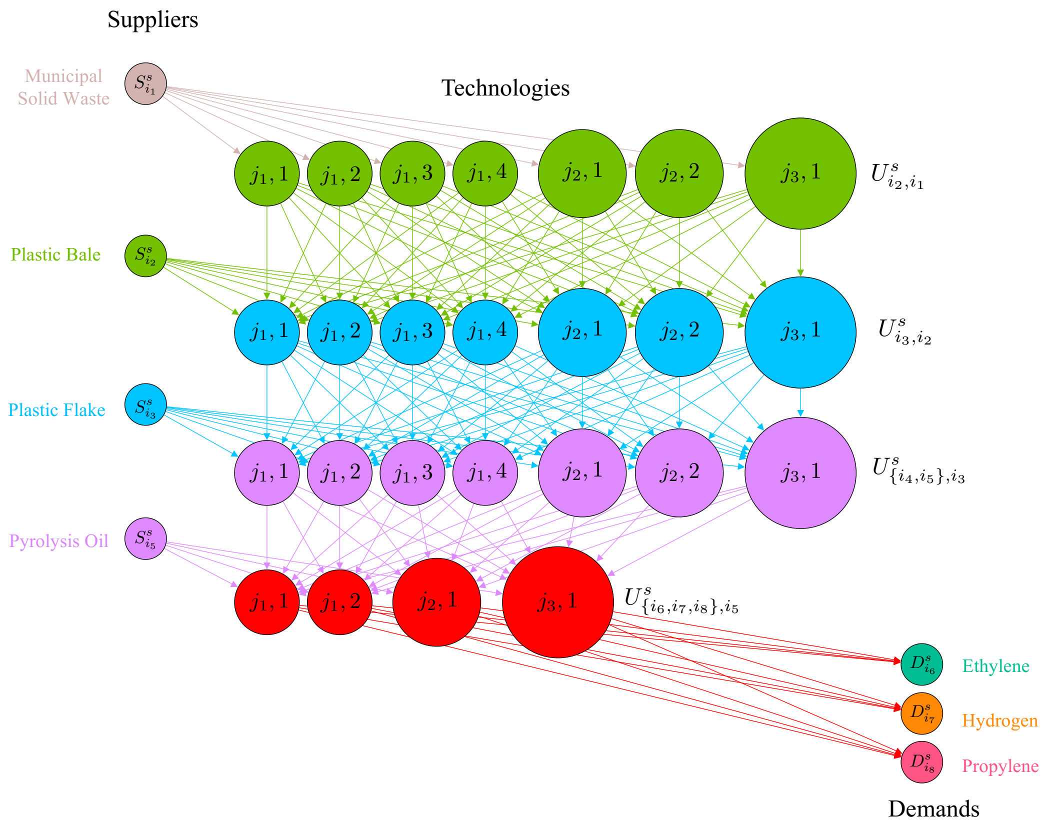


Figure 11: Superstructure for plastic upcycling system (no spatial information).

shown in Figure 12 . The design problem contains 1028 continuous variables and 25 binary variables, and contains 1111 constraints. This problem tooks less than 0.01 second to solve. The result shows that the optimal design design contains 7 tech units and achieves an annualized cost of \$ $6.56 \times 10^8$. We can see that it chooses the tech with largest capacity for processes MRF, RF and PP as they are the most cost efficient units to satisfy the required amount of products.

We then use optimization formulation (3.22) to solve for the cost-minimizing design that also achieves a certain degree of modularity. This problem has 1028 continuous variables, 3624 integer variables, and 3918 constraints. A couple of designs that correspond to different levels of modularity are shown in Figure 13. Module division in both cases are grouped by red dashed rectangles and nodes within 4 modules for each case are summarized in Table 5. Note that the optimal design shown on the left in Figure 13 contains identical techs as in Figure 12. This means that the cost-minimizing design without considering modularity achieves a modularity measure of 0.3. This is the maximum level of modularity that a valid system design can achieve, while achieving the minimum level of the cost (and we see that this value is quite low). We can also see that, as we require a higher degree of modularity, medium- and small-sized technologies start to be considered in the design. We summarize the results for different combinations of modularity measure and cost in Table 6.

We can see that, as we increase the required degree of modularity, the cost increases and the number of techs installed also increases. This demonstrates that, to achieve a higher modularity measure, smaller techs need to be installed and there exists a trade-off between cost and the degree of

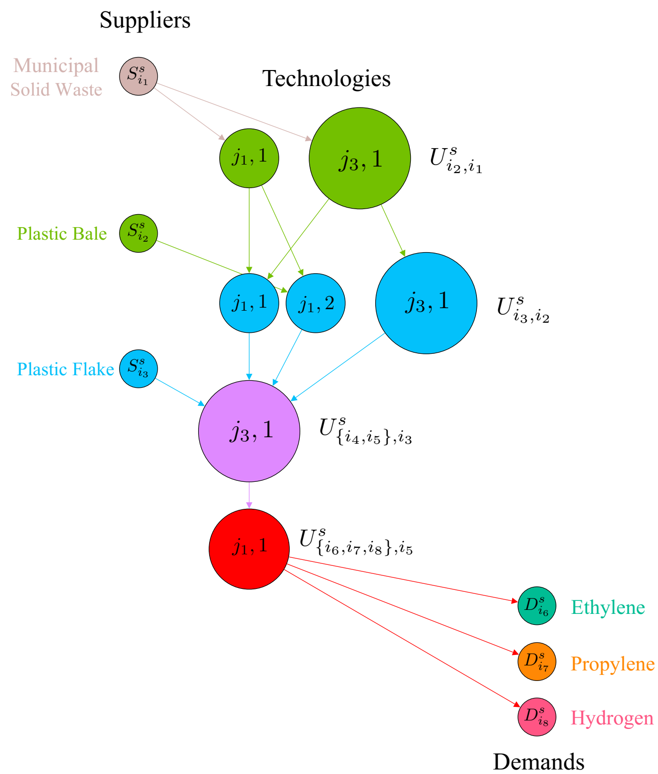


Figure 12: Cost-minimizing optimal design of plastic upcycling system (no spatial information).

Table 5: Module details for optimal system designs with different degrees of modularity.

| Modularity Measure, $M_4$ | Annualized Cost, $C$ (\$) | Module 1 | Module 2 | Module 3 | Module 4 |
|---|---|---|---|---|---|
| 0.3 | 6.56 $\times 10^8$ | $U^s_{i_2,i_1,j_1,1}$ $U^s_{i_3,i_2,j_3,1}$ | $U^s_{i_2,i_1,j_3,1}$ $U^s_{i_3,i_2,j_1,1}$ | $U^s_{i_3,i_2,j_1,2}$ $U^s_{\{i_4,i_5\},i_3,j_3,1}$ | $U^s_{\{i_6,i_7,i_8\},i_5,j_1,1}$ |
| 0.6 | 6.95 $\times 10^8$ | $U^s_{i_2,i_1,j_1,1}$ $U^s_{i_2,i_1,j_1,2}$ $U^s_{i_3,i_2,j_1,1}$ $U^s_{i_3,i_2,j_1,2}$ | $U^s_{i_2,i_1,j_1,3}$ $U^s_{i_3,i_2,j_1,3}$ $U^s_{\{i_4,i_5\},i_3,j_2,2}$ | $U^s_{i_2,i_1,j_2,1}$ $U^s_{\{i_4,i_5\},i_3,j_2,1}$ | $U^s_{\{i_4,i_5\},i_3,j_2,1}$ $U^s_{\{i_6,i_7,i_8\},i_5,j_2,1}$ |

modularity of the system. In addition, we observe that, we increase the level of modularity measure from 0 to 0.6, the cost increase is relatively small (∼2% for 0.1 increase in modularity). However, as we further increase the modularity (from 0.6 and above), the cost increases sharply in order to achieve the same amount of modularity increase (∼10% for 0.1 increase in modularity). This means that there exists a threshold for the process design when increasing modularity measure becomes too expensive and makes no economic sense; as such, decision-makers should choose any optimal design within that threshold to achieve a balanced tradeoff between cost and modularity. As we increase the modularity measure, the solution time generally becomes longer since the smaller technologies become relevant here, increasing the possible number of designs and therefore harder for the solver to find an optimal solution. When we are at a high modularity measure (0.8 in this case), the design

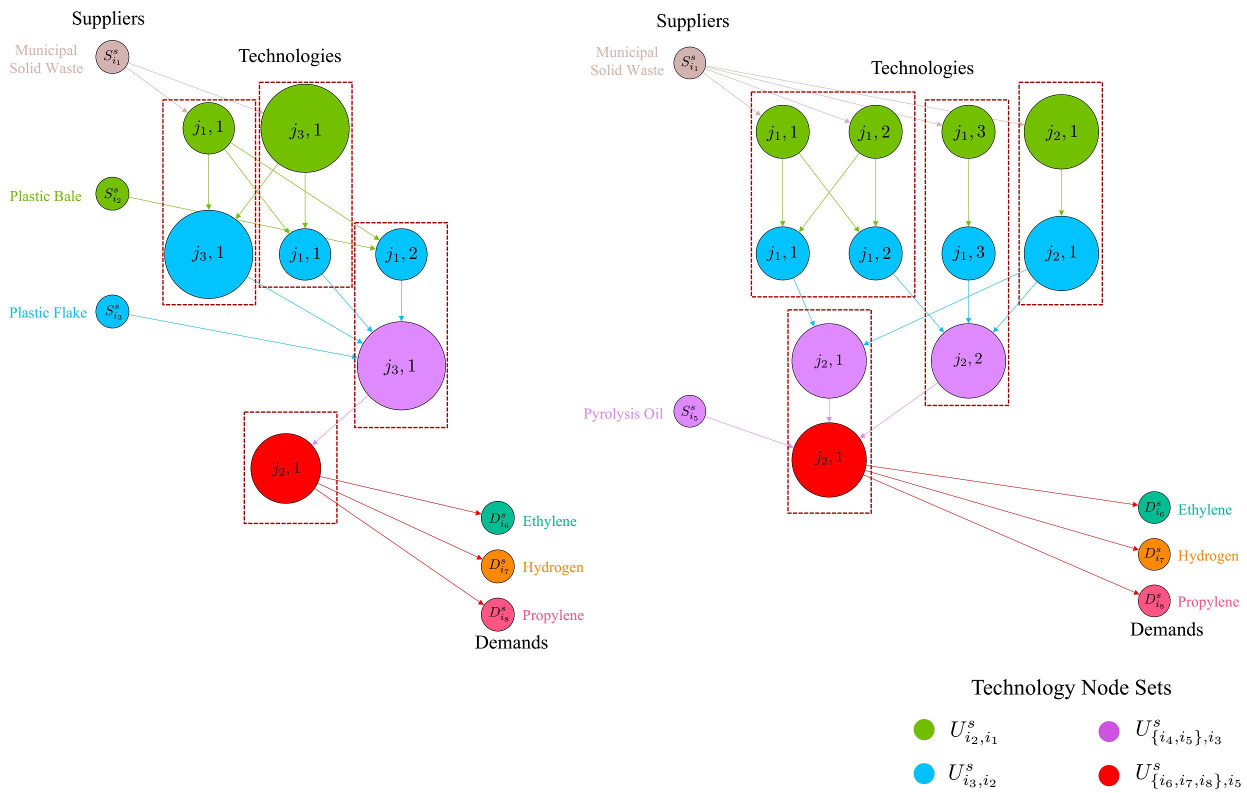


Figure 13: Cost-minimizing optimal modular designs with $M_4 \geq 0.3$ (left) and $M_4 \geq 0.6$ (right)

requirements become too strict, and therefore reduce the number of possible designs available. This explains the reduction on the solution time for the case when the modularity measure is 0.8.

Table 6: Trade-offs between system cost and modularity for optimal designs (no spatial information).

| Modularity Measure | Cost ($) | # of Technology | Solving Time (s) |
|---|---|---|---|
| 0 | $6.56 \times 10^8$ | 7 | 0.27 |
| 0.3 | $6.56 \times 10^8$ | 7 | 1.42 |
| 0.4 | $6.71 \times 10^8$ | 9 | 22.33 |
| 0.5 | $6.83 \times 10^8$ | 10 | 16.99 |
| 0.6 | $6.95 \times 10^8$ | 11 | 22.52 |
| 0.7 | $7.65 \times 10^8$ | 11 | 48.51 |
| 0.8 | $8.38 \times 10^8$ | 12 | 17.71 |

### 4.3 Optimal System Design with Spatial Information

When we extend the problem to include spatial information, all the case settings defined in Table 2 and Table 3 remain valid. We define the additional spatial information such as potential locations to install techs and the transportation cost across different locations in Table 7.

Here, a couple of potential locations ($B$ and $D$) are available to install technologies. Locations

Table 7: Spatial information for plastic upcycling system.

| Parameters | Values |
|---|---|
| Potential Locations to Install Technologies, $\mathcal{G}_t$ | $[B, D]$ |
| Locations for Suppliers of Materials, $\mathcal{G}_s^i, i \in \mathcal{R}$ | $[B, C, B, D]$ |
| Locations for Demand of Materials, $\mathcal{G}_d^i, i \in \mathcal{P}$ | $[C, D, B, C, B, B]$ |
| Set of All Locations, $\mathcal{G}$ | $[B, C, D]$ |
| Scale of Total Cost at $[B, D]$ | $T_{i_2}$: [0.9, 1.1] |
| | $T_{i_3}$: [1, 1.1] |
| | $T_{\{i_4,i_5\}}$: [1.1, 1] |
| | $T_{\{i_6,i_7,i_8\}}$: [0.9, 1.1] |
| Transportation Cost, $\alpha^f_{i,B,g'}, i \in \mathcal{I}, g' \in \mathcal{G}$ ($/unit) | [0.01, 0.1, 0.14] |
| Transportation Cost, $\alpha^f_{i,C,g'}, i \in \mathcal{I}, g' \in \mathcal{G}$ ($/unit) | [0.14, 0.01, 0.12] |
| Transportation Cost, $\alpha^f_{i,D,g'}, i \in \mathcal{I}, g' \in \mathcal{G}$ ($/unit) | [0.14, 0.11, 0.01] |

for suppliers and demands corresponds to the sequences of product $i$ in set $\mathcal{R}$ and $\mathcal{P}$ as shown in Table 3. For example, supply of product $i_1$ is at location $B$ and demand of product $i_6$ is at location $C$. Installation costs for location $B$ and $D$ are represented as a scale times the cost defined in Table 3. For instance, installation costs for tech $t \in T_{i_2}$ (technologies $T_{i_2,i_1,j_1}$, $T_{i_2,i_1,j_2}$, and $T_{i_2,i_1,j_3}$) at locations $B$ equals to the scale $0.9$ times their original installation costs $[0.19, 0.45, 0.89]$ as defined in Table 3, and the costs at location $D$ equals to the scale $1.1$ times their original costs. Finally, we assume that transport costs for different products are the same but they are different across different locations. For example, transportation cost for any product from location $B$ to $D$ is 0.14 and from location $C$ to $D$ is 0.12. Note that transportation costs at the same location (from location $B$ to $B$) are much smaller than those across different locations.

The spatial superstructure of the system is shown in Figure 14. Nodes on the top represent potential technologies at location B while nodes on the bottom represent their installation at location D. This spatial superstructure is much denser than the superstructure in Figure 11; therefore, we have a much larger optimization problem.

We first use formulation (3.24) to solve for the cost-minimizing spatial design and the result is shown in Figure 15 . This problem has 3253 continuous variables, 50 binary variables, and 3416 constraints, and required 0.16 seconds to solve. The result shows that the optimal design contains 7 techs, with 6 of them being placed at location B while the only pyrolysis plant is placed at location D. This configuration makes sense, as the optimal design tries to put technologies at locations with less installation cost, which is the major cost of the process. This cost-minimizing design achieves an annualized cost of $\$\ 6.30 \times 10^8$ and similarly, it selects the technology with largest capacity for processes MRF, RF, and PP as they are the most cost-efficient units to satisfy the required amount of products. We then use optimization formulation (3.28) to solve for the cost-minimizing design with spatial information that also achieves a certain level of modularity.

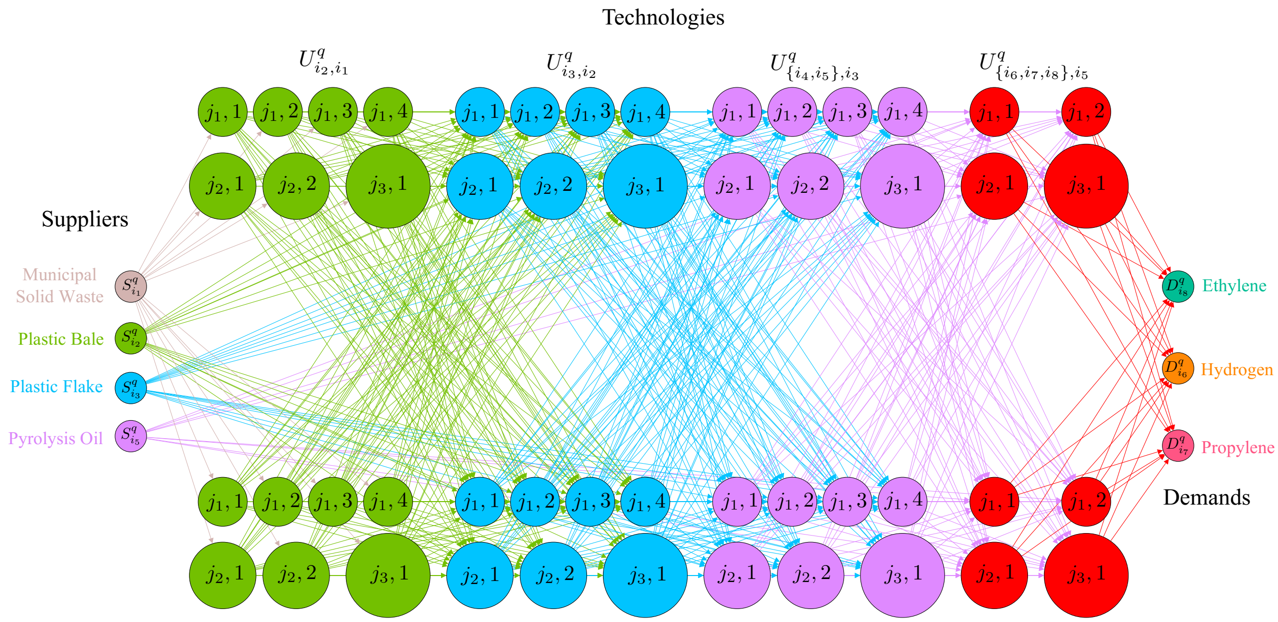


Figure 14: Spatial superstructure for plastic waste upcycling system. Technologies on the top are for location B and technologies on the bottom are for location D.

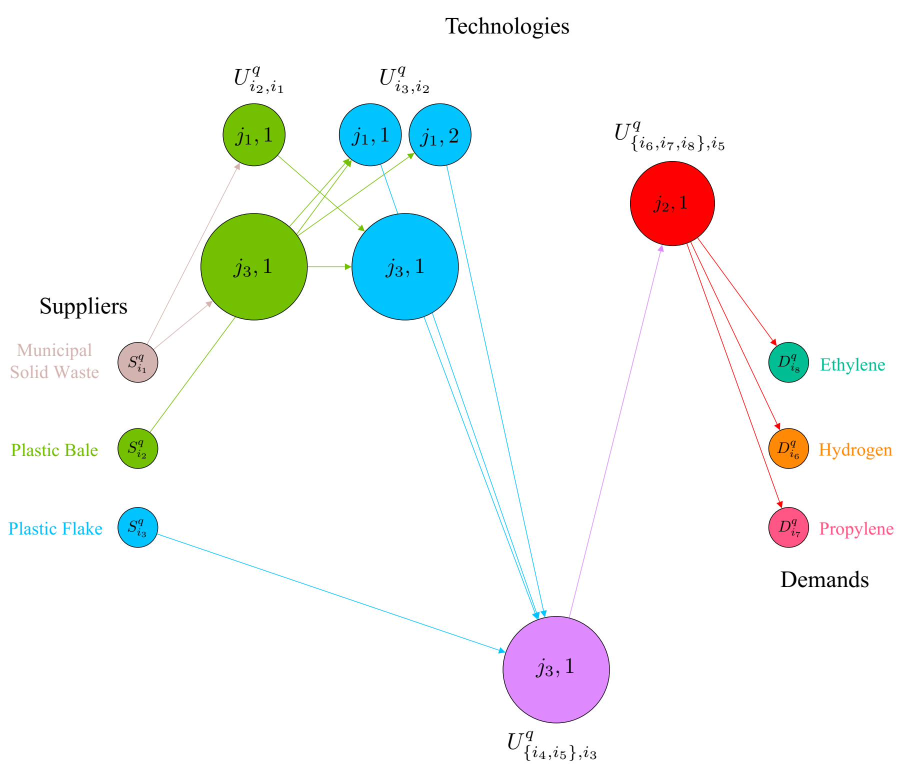


Figure 15: Cost-minimizing optimal spatial system design.

The optimal designs that correspond to two different levels of modularity are shown in Figure 16.

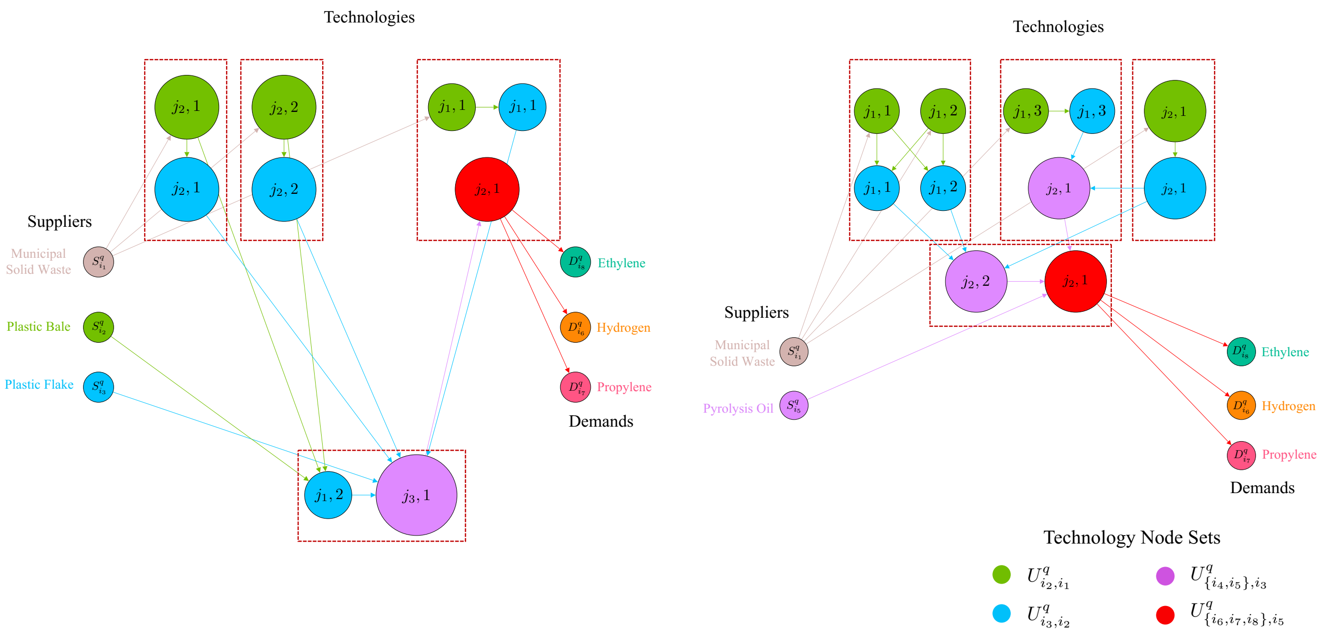


Figure 16: Cost-minimizing optimal modular system design (with spatial information) for modularity $M_4 \geq 0.4$ (left) and $M_4 \geq 0.6$ (right)

Table 8: Details for a couple of modular designs.

| Modularity Measure, $M_4$ | Annualized Cost, $C$ (\$) | Module 1 | Module 2 | Module 3 | Module 4 |
|---|---|---|---|---|---|
| 0.4 | 6.46 $\times 10^8$ | $U^q_{i_2,i_1,j_2,1}$ $U^q_{i_3,i_2,j_2,1}$ | $U^q_{i_2,i_1,j_2,2}$ $U^q_{i_3,i_2,j_2,2}$ | $U^q_{i_2,i_1,j_1,1}$ $U^q_{i_3,i_2,j_1,1}$ $U^q_{\{i_6,i_7,i_8\},i_5,j_2,1}$ | $U^q_{i_3,i_2,j_1,2}$ $U^q_{\{i_4,i_5\},i_3,j_3,1}$ |
| 0.6 | 6.74 $\times 10^8$ | $U^q_{i_2,i_1,j_1,1}$ $U^q_{i_2,i_1,j_1,2}$ $U^q_{i_3,i_2,j_1,1}$ $U^q_{i_3,i_2,j_1,2}$ | $U^q_{i_2,i_1,j_1,3}$ $U^q_{i_3,i_2,j_1,3}$ $U^q_{\{i_4,i_5\},i_3,j_2,1}$ | $U^q_{i_2,i_1,j_2,1}$ $U^q_{i_3,i_2,j_2,1}$ | $U^q_{\{i_4,i_5\},i_3,j_2,2}$ $U^q_{\{i_6,i_7,i_8\},i_5,j_2,1}$ |

Module divisions in both cases are grouped by red dashed rectangles and nodes within four modules for each case are summarized in Table 8. We can see that, when requirement for modularity is low, the large pyrolysis plant in purple takes the advantage of low installation costs at location D and is grouped with a small reprocessing facility in black. The module (facility) that these units form is placed at location D, while all other units and modules are placed at a facility in location B. On the right we can see that, as we require a higher degree of modularity, all units and modules are placed at location B. Here, a low installation cost is outweighed by the high transportation cost due to increased connectivity across locations and therefore no units are placed at location D. We can also see that smaller techs are utilized in the optimal design with higher modularity. We also summarize the results for different combinations of modularity measure and cost in Table 9. A similar trend can be observed in Table 6; specifically, a threshold exists around a modularity measure of 0.6 where further increasing the measure causes the cost of design to rise sharply, and the solving time increases dramatically as we increase the requirement for modularity measure initially, and eventually drops when the requirement is too high.

Table 9: Trade-off between cost and modularity for optimal system design (with spacial information).

| Modularity Measure | Cost ($) | # of Technologies | Solving Time (s) |
|---|---|---|---|
| 0 | $6.30 \times 10^8$ | 7 | 0.92 |
| 0.3 | $6.32 \times 10^8$ | 7 | 4.89 |
| 0.4 | $6.46 \times 10^8$ | 9 | 287.51 |
| 0.5 | $6.61 \times 10^8$ | 10 | 708.48 |
| 0.6 | $6.74 \times 10^8$ | 11 | 719.71 |
| 0.7 | $7.44 \times 10^8$ | 11 | 3074.36 |
| 0.8 | $8.17 \times 10^8$ | 12 | 1312.20 |

## Conclusions and Future Work

We have presented an optimization framework that facilitates the design of modular systems. Central to our development is the introduction of the concept of a spatial superstructure, which is a generalization of a superstructure that encodes spatial context. This generalization enables the simultaneous design of processes, facilities, and supply chains with desired modularity properties. The spatial superstructure is used to derive a multi-objective, mixed-integer optimization formulation for identifying optimal designs that trade-off cost and modularity. We demonstrate the capabilities of the proposed framework by using a case study arising in the design of a plastic upcycling supply chain. As part of future work, we are interested in exploring the use of strategies to address computational tractability issues and to capture higher fidelity in the design (e.g., detailed physical models).

## Acknowledgments

We acknowledge funding from NSF CAREER award CBET-1748516 for the development of the methodology. The development of the plastic waste case study was supported by the U.S. Department of Energy, Office of Energy Efficiency and Renewable Energy, Bioenergy Technologies Office under Award Number DEEE0009285.

## Nomenclature

### Abbreviations

MSW - Municipal solid waste
MRF - Material/Product recovery facility
RF - Reprocessing facility

PP - Pyrolysis process
SC - Steam cracking

### Indices and Sets

$i \in \mathcal{R}$ - Primary products
$i \in \mathcal{P}$ - Intermediate/Final products
$i \in \mathcal{I}$ - All products
$t \in \mathcal{T}$ -Technologies
$\mathcal{G}^p$ - Max p-graph
$k \in \mathcal{N}^p$ - Nodes in max p-graph
$s \in \mathcal{S}^p$ - Supplier nodes in max p-graph
$d \in \mathcal{D}^p$ - Demand nodes in max p-graph
$u \in \mathcal{U}^p$ - Technology nodes in max p-graph
$\mathcal{E}^p$ - Edges in max p-graph
$\mathcal{G}^s$ - Superstructure
$k \in \mathcal{N}^s$ - Nodes in superstructure
$s \in \mathcal{S}^s$ - Supplier nodes in superstructure
$d \in \mathcal{D}^s$ - Demand nodes in superstructure
$u \in \mathcal{U}^s$ - Technology nodes in superstructure
$\mathcal{E}^s$ - Edges in superstructure
$\mathcal{G}^q$ - Spatial superstructure
$k \in \mathcal{N}^q$ - Nodes in spatial superstructure
$s \in \mathcal{S}^q$ - Supplier nodes in spatial superstructure
$d \in \mathcal{D}^q$ - Demand nodes in spatial superstructure
$u \in \mathcal{U}^q$ - Technology nodes in spatial superstructure
$\mathcal{E}^q$ - Edges in spatial superstructure
$g \in \mathcal{G}_t$ - Locations for technologies in spatial superstructure
$g \in \mathcal{G}_s$ - Locations for suppliers in spatial superstructure
$g \in \mathcal{G}_d$ - Locations for demands in spatial superstructure
$\mathcal{G}^f$ - Feasible path
$k \in \mathcal{N}^f$ - Nodes in feasible path
$\mathcal{E}^f$ - Edges in feasible path
$l \in \mathcal{L}$ - Modules

### Attributes

$\Omega_t$ - Technology $t$ output products set $\Omega$
$\mathcal{K}_t$ - Technology $t$ input products set $\mathcal{K}$
$\theta_t$ - Technology $t$ technology type $\theta$
$\gamma_t$ - Technology $t$ physical size
$\Omega_s$ - Supplier node $s$ output product $\Omega$
$\phi_s$ - Supplier node $s$ location $\phi$
$\mathcal{K}_d$ - Demand node $d$ input product $\mathcal{K}$
$\phi_d$ - Demand node $d$ location $\phi$

$\tau_u$ - Technology node $u$ technology $\tau$
$\Omega_{\tau_u}/\Omega_u$ - Technology $\tau$ of technology node $u$ output products set $\Omega$
$\mathcal{K}_{\tau_u}/\mathcal{K}_u$ - Technology $\tau$ of technology node $u$ input products set $\mathcal{K}$
$\theta_{\tau_u}/\theta_u$ - Technology $\tau$ of technology node $u$ technology type $\theta$
$\gamma_{\tau_u}/\gamma_u$ - Technology $\tau$ of technology node $u$ physical size $\gamma$
$\eta_u$ - Technology node $u$ copy $\eta$
$\phi_u$ - Technology node $u$ location $\phi$

### Derived Subsets

$\mathcal{T}_{i,i',j}$ - Technologies $t|i \in \Omega_t, i' \in \mathcal{K}_t, \theta_t = j$
$\mathcal{S}_i^p$ - Supplier node $s|\Omega_s = i$ for max-p graph
$\mathcal{S}_{i,g}^q$ - Supplier node $s|\Omega_s = i, \phi_s = g$ for spatial superstructure
$\mathcal{D}_i^p$ - Demand node $d|\mathcal{K}_d = i$ for max-p graph
$\mathcal{D}_{i,g}^q$ - Demand node $d|\mathcal{K}_d = i, \phi_d = g$ for spatial superstructure
$\mathcal{U}_t^p$ - Technology node $u|\tau_u = t$ for max-p graph
$\mathcal{U}_{i,i',j}^p$ - Technology node $u|i \in \Omega_u, i' \in \mathcal{K}_u, \theta_u = j$ for max-p graph
$\mathcal{U}_{t,h}^s$ - Technology node $u|\tau_u = t, \eta_u = h$ for superstructure
$\mathcal{U}_{i,i',j,h}^s$ - Technology node $u|i \in \Omega_u, i' \in \mathcal{K}_u, \theta_u = j, \eta_u = h$ for superstructure
$\mathcal{U}_{t,h,g}^q$ - Technology node $u|\tau_u = t, \eta_u = h, \phi_u = g$ for spatial superstructure
$\mathcal{U}_{i,i',j,h,g}^q$ - Technology node $u|i \in \Omega_u, i' \in \mathcal{K}_u, \theta_u = j, \eta_u = h, \phi_u = g$ for spatial superstructure

### Parameters

$\beta_{i,t}^{i'}$ - Amount of units of product $i'$ required to produce a unit of product $i$ using technology $t$
$\mu_{k,k'}$ - Superstructure/Spatial superstructure connectivity
$\alpha_t^\xi$ - Technology $t$ installation cost (for superstructure)
$\alpha_{t,g}^\xi$ - Technology $t$ installation cost at location $g$ (for spatial superstructure)
$\alpha_t^o$ - Technology $t$ operational cost
$\xi_{t,i}$ - Technology $t$ capacity of product $i$
$\alpha_{\tau_u}^\xi/\alpha_u^\xi$ - Technology $\tau$ of technology node $u$ installation cost
$\alpha_{\tau_u}^o/\alpha_u^o$ - Technology $\tau$ of technology node $u$ operational cost
$\xi_{\tau_u,\Omega_u}/\xi_{u,i}$ - Technology $\tau$ of technology node $u$ capacity of product $i \in \Omega_u$
$\epsilon_a$ - Capital cost annualization factor
$\epsilon_M$ - Modularity measure threshold
$\alpha_i^\rho$ - Unit cost of product $i$
$\delta_i$ - Required amount of final product $i$
$\alpha^f$ - Uniform connection/transport cost (for superstructure)
$\alpha_{i,g,g'}^f$ - Transportation cost of product $i$ from location $g$ to $g'$ (for spatial superstructure)
$\alpha_i^d$ - Disposal cost for product $i$
$M$ - Big M parameter
$n$ - Number of modules
$\bar{D}$ - Dimensionality upper limit
$\underline{D}$ - Dimensionality lower limit

### Variables

$x_i$ - Product $i$ hierarchy
$f_{k,k'}$ - Flow of product from node $k$ to node $k'$
$v_i$ - Primary product $i$ purchase quantity
$y_u$ - Technology node $u$ selection
$y_{u,l}$ - Technology node $u$ selection with module information
$C$ - Total annualized cost
$M_n$ - Modularity measure with predefined $n$ modules
$a_{u,u'}$ - Connectivity of technology nodes associated to the feasible path
$a^m_{u,u',l}$ - Connectivity of technology nodes with module information
$\pi_{u,u'}$ - Membership variable matrix
$z_{l,g}$ - Module $l$ location information

## References

[1] Yuhe Tian, Iosif Pappas, Baris Burnak, Justin Katz, and Efstratios N. Pistikopoulos. A systematic framework for the synthesis of operable process intensification systems – reactive separation systems. *Computers and Chemical Engineering*, 134(4):106675, 2020.

[2] Katerina P. Papalexandri and Efstratios N. Pistikopoulos. Generalized modular representation framework for process synthesis. *AIChE Journal*, 42(4):1010–1032, 1996.

[3] Sudipta Chakraborty, Bill Kramer, and Benjamin Kroposki. A review of power electronics interfaces for distributed energy systems towards achieving low-cost modular design. *Renewable and Sustainable Energy Reviews*, 13(9):2323 – 2335, 2009.

[4] Christian Bramsiepe Tim Seifert, Stefan Sievers and Gerhard Schembecker. Small scale, modular and continuous: a new approach in plant design. *Chemical Engineering and Processing: Process Intensification*, 52, 2012.

[5] T Seifert C Bramsiepe, S Sievers et al. Low-cost small scale processing technologies for production applications in various environmentsmass produced factories. *Chemical Engineering and Processing: Process Intensification*, 51, 2012.

[6] Ramchandran Jaikumar. Postindustrial manufacturing. *Harvard Business Review*, 64(6):69–76, 1986.

[7] S. Rajagopalan. Flexible versus dedicated technology: A capacity expansion model. *International Journal of Flexible Manufacturing Systems*, 5(2):129–142, 1993.

[8] Andrew Allman and Qi Zhang. Dynamic location of modular manufacturing facilities with relocation of individual modules. *European Journal of Operational Research*, 2020.

[9] Qi Chen and Ignacio E Grossmann. Economies of numbers for a modular stranded gas processing network: Modeling and optimization. In *Computer Aided Chemical Engineering*, volume 47, pages 257–262. Elsevier, 2019.

[10] Neal Davis. Natural gas flaring in north dakota has declined sharply since 2014. https://www.eia.gov/todayinenergy/detail.php?id=26632, 2016. Accessed: 2020-07-05.

[11] Michal Frivaldsky, Pavol Spanik, Jan Morgos, and Michal Pridala. Control strategy proposal for modular architecture of power supply utilizing lcct converter. *Energies*, 11(12):3327, 2018.

[12] Michel Berthélemy and Lina Escobar Rangel. Nuclear reactors' construction costs: The role of lead-time, standardization and technological progress. *Energy Policy*, 82:118 – 130, 2015.

[13] DiPippo R. Small geothermal power plants: design, performance and economics. *GHC bulletin*, 20(2), 1999.

[14] Yong ha Kim, Lydia K. Park, Sotira Yiacoumi, and Costas Tsouris. Modular chemical process intensification: A review. *Annual Review of Chemical and Biomolecular Engineering*, 8:359–380, 2017.

[15] Efstratios N. Pistikopoulos, Yuhe Tian, and Rahul Bindlish. Operability and control in process intensification and modular design: Challenges and opportunities. *AIChE Journal*, 67(5):e17204, 2021.

[16] Yuhe Tian, Salih Emre Demirel, MM Faruque Hasan, and Efstratios N Pistikopoulos. An overview of process systems engineering approaches for process intensification: State of the art. *Chemical Engineering and Processing-Process Intensification*, 133:160–210, 2018.

[17] Michael Baldea. From process integration to process intensification. *Computers and Chemical Engineering*, 81(4):104–114, 2015.

[18] Prodromos Daoutidis, Wentao Tang, and Sujit S Jogwar. Decomposing complex plants for distributed control: Perspectives from network theory. *Computers & Chemical Engineering*, 114:43–51, 2018.

[19] Wentao Tang and Prodromos Daoutidis. Network decomposition for distributed control through community detection in input–output bipartite graphs. *Journal of Process Control*, 64:7–14, 2018.

[20] Yue Shao and Victor M. Zavala. Modularity measures: Concepts, computation, and applications to manufacturing systems. *AIChE Journal*, 66(6):e16965, 2020.

[21] Kibaek Kim, Fan Yang, and Victor M. Zavalaand Andrew A Chien. Data centers as dispatchable loads to harness stranded power. *IEEE Transactions on Sustainable Energy*, 8(1):208 – 218, 2017.

[22] Ramteen Sioshansi, Paul Denholm, Thomas Jenkin, and Jurgen Weiss. Estimating the value of electricity storage in pjm: Arbitrage and some welfare effects. *Energy economics*, 31(2):269 – 277, 2009.

[23] Nai-Yuan Chiang and Victor M. Zavala. Large-scale optimal control of interconnected natural gas and electrical transmission systems. *Applied Energy*, 168:226 – 235, 2016.

[24] Tristan Becke, Bastian Bruns, Stefan Lier, and Brigitte Werners. Decentralized modular production to increase supply chain efficiency in chemical markets. *Journal of Business Economics*, 91:867–895, 2021.

[25] Amalia Nikolopoulou and Marianthi G. Ierapetritou. Optimal design of sustainable chemical processes and supply chains: A review. *Computers and Chemical Engineering*, 44:94–103, 2012.

[26] Qi Zhang, NilayShah, John Wassick, Rich Helling, and Peter van Egerschotc. Sustainable supply chain optimisation: An industrial case study. *Computers and Industrial Engineering*, 74:68–83, 2014.

[27] Andrew Allman, Che Lee, Mariano Martin, and Qi Zhang. Biomass waste-to-energy supply chain optimization with mobile production modules. *Computers and Chemical Engineering*, 150:107326, 2021.

[28] Andrew Allman, Che Lee, Mariano Martin, and Qi Zhang. On the benefit of modular and mobile production units in biomass waste-to-energy supply chains. *Computer Aided Chemical Engineering*, 48:1363–1368, 2020.

[29] Andrew Allman and Qi Zhang. Dynamic location of modular manufacturing facilities with relocation of individual modules. *European Journal of Operational Research*, 286(2):494–507, 2020.

[30] Vishal Goyal and Marianthi G. Ierapetritou. Multiobjective framework for modular design generation incorporating demand uncertainty. *Industrial and Engineering Chemistry Research*, 44(10):3594–3606, 2005.

[31] Atharv Bhosekar, Oluwadare Badejo, and Marianthi Ierapetritou. Modular supply chain optimization considering demand uncertainty to manage risk. *AIChE Journal*, 67(11):e17367, 2021.

[32] R. Cory Allen, Styliani Avraamidou, and Efstratios N. Pistikopoulos. Production scheduling of supply chains comprised of modular production units. *IFAC PapersOnLine*, 53(2):11452–11457, 2020.

[33] Atharv Bhosekar and Marianthi Ierapetritou. A framework for supply chain optimization for modular manufacturing with production feasibility analysis. *Computers and Chemical Engineering*, 145:107175, 2021.

[34] YW Huang F Friedler, K Tarjan and LT Fan. Graph-theoretic approach to process synthesis: axioms and theorems. *Chemical Engineering Science*, 47(8):1973–1988, 1992.

[35] Amy R Ciric Christodoulos A Floudas and Ignacio E Grossmann. Automatic synthesis of optimum heat exchanger network configurations. *AIChE Journal*, 32(2):276–290, 1986.

[36] Hector Yeomans and Ignacio E Grossmann. A systematic modeling framework of superstructure optimization in process synthesis. *Computers and Chemical Engineering*, 23(6):709–731, 1999.

[37] Mahmoud M El-Halwagi and Vasilios Manousiouthakis. Synthesis of mass exchange networks. *AIChE Journal*, 35(8):1233–1244, 1989.

[38] AJ Isafiade and DM Fraser. Interval based minlp superstructure synthesis of combined heat and mass exchanger networks. *Chemical Engineering Research and Design*, 87(11):1536–1542, 2009.