# Pricing Interprocess Streams Using Slack Auctions

**Rinaldo A. Jose and Lyle H. Ungar**
Dept. of Chemical Engineering, University of Pennsylvania, Philadelphia, PA 19104

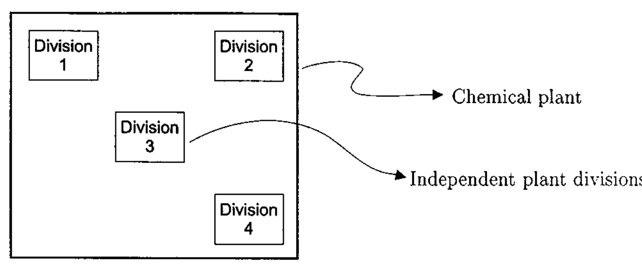
*When plantwide optimization problems must be decomposed and solved in pieces due to numerical complexity or practical concerns, it is natural to view each plant division as a separate profit maximizer. For a division to maximize its profit, however, it must know the prices of interprocess streams. It is claimed that the correct way of finding these prices is to define slack resources over the process streams and price them using slack auctions. When slack auctions terminate properly, it can be shown that the plant divisions' solutions are globally optimal. How slack auctions are related to Lagrangean relaxation and how they can overcome some of the problems that standard decomposition and relaxation techniques have are discussed.*

## Introduction

Centralized optimization may be the preferred way of solving chemical engineering problems, but there are cases where decentralized approaches are more appropriate. Detailed refinery problems, for instance, are still too large to be solved on-line all at once—in practice, each division in a refinery is still optimized separately (Jerome, 1998). Also, even if sufficient computing power were available, different and incompatible software is often used to model and operate different unit operations, making integration and centralization difficult. In addition, when unit operations are managed by different people (and especially by different companies), it is natural to optimize them with respect to existing divisions. This is motivated by more than aesthetics. From a practical perspective, a division prefers to conceal as much of its private information as possible, otherwise, it is exposed to potential manipulation. Decentralized optimization allows plant divisions to restrict the amount of visible private information.

If plant divisions are all independent, then it is trivial to solve a plantwide optimization problem using a decentralized approach: the plantwide solution $x^*$ is simply the solutions of all the component divisions taken together (see Figure 1). In a chemical plant, however, this is rarely the case because the divisions in a plant are not independent; they interact with one another via interprocess streams (see Figure 2). If the goal of each division is to maximize its own profit, then each division must known the "prices" of those streams (see Figure 3). One way of framing this problem is to use a *supply chain* metaphor. A supply chain is simply a network of divisions that buy and sell resources between each other. To solve a plantwide optimization problem using this approach, one must define the appropriate interprocess "resources" and find prices which lead the divisions to a globally optimal state.

In this article we argue that the appropriate interprocess resources are *slack resources*—the absolute difference between the desired stream states of the upstream and downstream units—and we claim that these slack resources can be priced using *slack auctions*. When slack auctions terminate with *equilibrium prices* which balance the demand for slack with its supply, the plant is globally optimized regardless of




$x^* = (x_1^*, x_2^*, x_3^*, x_4^*)$

**Figure 1. Modular optimization for independent plant divisions.**

Correspondence concerning this article should be addressed to R. A. Jose at his new address: Electro Scientific Industries, Inc., 13900 NW Science Park Drive, Portland, OR 97229.

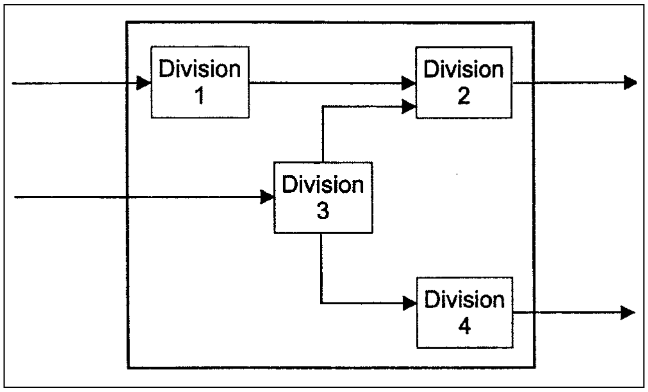



**Figure 2. Plant divisions interact because of interprocess streams.**

the convexity or continuity of the divisions' underlying optimization problems. In addition, we discuss how slack auctions are related to Lagrangean relaxation methods and why slack auctions can solve a class of common problems where standard relaxation techniques fail.

## Decentralized Optimization Using a Supply Chain Metaphor

The supply chain idea originated in manufacturing and operations research as a way of managing resource flows between companies (Axelsson and Easton, 1992; Vollmann et al., 1997). Supply chain management in all its detail can be quite involved: companies must forecast demands, decide what products to produce, when to make them, and how to get them to their customers. A typical supply chain spans many companies and multiple industries. In principle, though, a supply chain can also be defined over the divisions in a single chemical plant. We call these supply chains *microsupply chains* to distinguish them from the standard variety.

By viewing a plant as a microsupply chain, interactions between plant divisions can be framed as competition for limited resources. Mathematically, all resources are associated

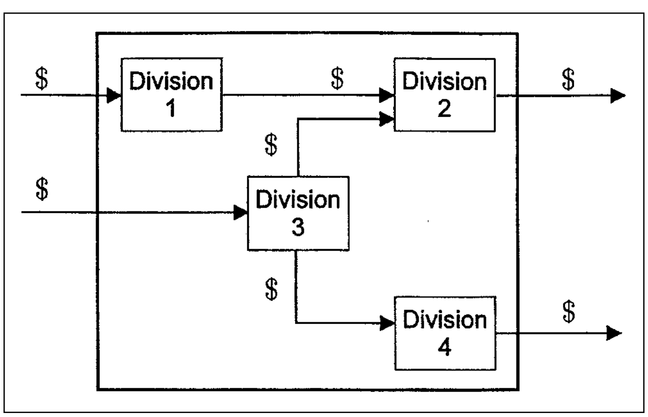



**Figure 3. Plant divisions need to know interprocess prices.**

with constraints of the form

$$D_1 + \cdots + D_n \leq S, \tag{1}$$

where $D_i$ is the resource demand of Division $i$, and $S$ is the total resource supply. Divisions in a supply chain interact when their overall resource demands exceed the available supply. This interaction can be managed by charging the divisions based on how much resource they desire: high prices reduce demand, while low prices encourage it. When Division $i$ is charged based on how much resource it uses, its goal is to solve

$$\begin{aligned} &\max_{x_i} \quad f_i(x_i) - p \cdot D_i(x_i) \\ &\text{subject to:} \\ &\quad \text{Division's private constraints,} \end{aligned} \tag{2}$$

where $x_i$ is Division $i$'s decision variable vector, $f_i(x_i)$ is its nominal objective function, $p$ is a resource price vector, and $D_i(x_i)$ is the division's resource demand vector. If resource prices can be found which satisfy following equilibrium conditions

$$\sum_{i=1}^{n} D_i \leq S \tag{3}$$

$$\sum_{j=1}^{m} p_j \cdot (D_j - S_j) = 0, \tag{4}$$

$$p_j \geq 0 \qquad j \in \{1, \ldots, m\}, \tag{5}$$

then the sum of the divisions' objective is globally maximized with respect to the available resources (see the Appendix for a proof of this result).

In order to apply this price-driven approach to plantwide optimization problems, however, one must define the appropriate resources over the interprocess streams. Although this seems trivial, it is not. The key problem is that the resources in a microsupply chain are different from those in a typical supply chain. In an ordinary supply chain, every company is free to buy as much or as little as it wants; in a microsupply chain, this is impossible. A plant division cannot dispose of "resources" sent to it via process streams—it has no choice but to accept the material regardless of how much it actually demands. Interprocess streams cannot really be viewed as resources. Mathematically, the problem is that interprocess constraints do not "look like" resource constraints—they are essentially physical consistency constraints like

$$C_{A,\text{upstream}} = C_{A,\text{downstream}}. \tag{6}$$

That is, the state of a stream leaving an upstream unit must be the same as its state entering a downstream unit (barring energy losses and so on). This is irrelevant for the purpose of defining resources, and it also complicates the pricing of interprocess streams.

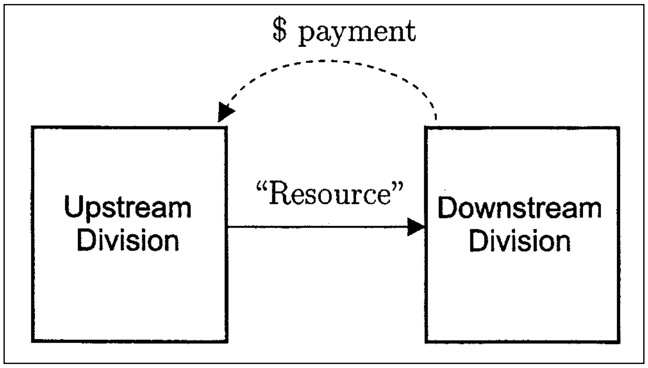



**Figure 4. Improperly defined stream resources.**

## Difficulty in Pricing Interprocess Streams

The apparent way of pricing interprocess streams is to treat the streams themselves as resources and have downstream units pay the upstream ones based on the streams' flow rates (see Figure 4). While intuitively plausible, the reasoning behind this is flawed. For one, because there is no resource constraint of the form of Eq. 1, the proposed stream resource is meaningless. Furthermore, because the downstream units cannot dispose of any of the upstream material piped to them, there is no notion of resource demand. Without meaningful resources and resource demands, there can be no equilibrium prices.

### *Lagrangean substitution*

One approach to pricing interprocess streams is to ignore resources altogether and apply a standard decomposition technique like Lagrangean substitution (see Findstein et al., 1980)—this, however, has its own problems. Consider the following problem. Suppose a reactor division sends its output to a separations division in order to purify some product $A$ (see Figure 5). Because the separations division produces material for direct sale, it has strong preferences over what the $A$ concentration of Stream $S_2$ is. The reactor division, on the other hand, is only concerned with its operating costs; it has no preference over the actual quality of its output stream. The objective functions for the reactor and separations divisions ($f_R(C_A)$ and $f_S(C_A)$, respectively) are shown in Figure 6. The overall optimization problem is to maximize the sum of the units' objectives subject to their private constraints

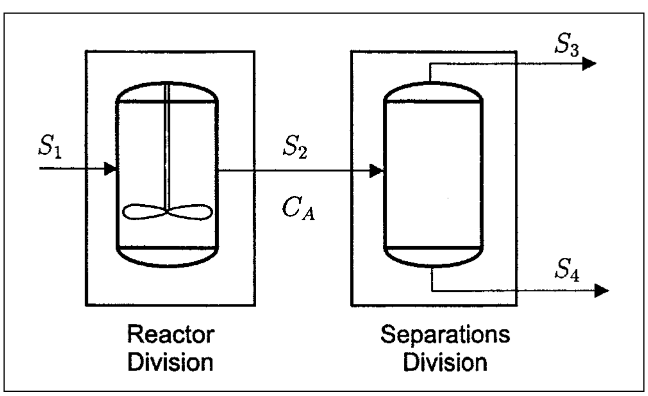



**Figure 5. Reactor/separations system.**

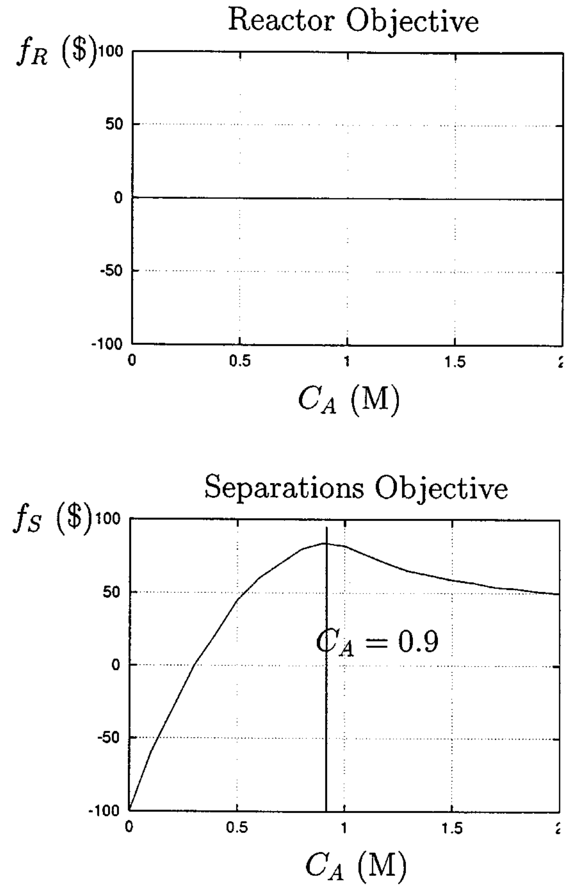



**Figure 6. Objective functions for the reactor and separations divisions.**

$$\begin{aligned} &\max_{C_A} \quad f_R(C_A) + f_S(C_A) \\ &\text{subject to:} \\ &\quad \text{Reactor division's private constraints} \\ &\quad \text{Separations division's private constraints.} \end{aligned} \tag{7}$$

In Lagrangean substitution, a microsupply chain is broken up into independent divisions by tearing process streams. Here, the stream connecting the reactor and separations divisions is torn to create two variables: $C_{A,R}$ for the reactor division and $C_{A,S}$ for the separations division (see Figure 7). The overall optimization problem (Eq. 7) is then expressed in terms of these new variables, but with the additional physical consistency constraint that $C_{A,R} = C_{A,S}$

$$\begin{aligned} &\max_{(C_{A,R}, C_{A,S})} \quad f_R(C_{A,R}) + f_S(C_{A,S}) \\ &\text{subject to:} \\ &\quad \text{Reactor division's private constraints} \\ &\quad \text{Separations division's private constraints} \\ &\quad C_{A,R} = C_{A,S}. \end{aligned} \tag{8}$$

Next, the consistency constraint is relaxed using a Lagrange

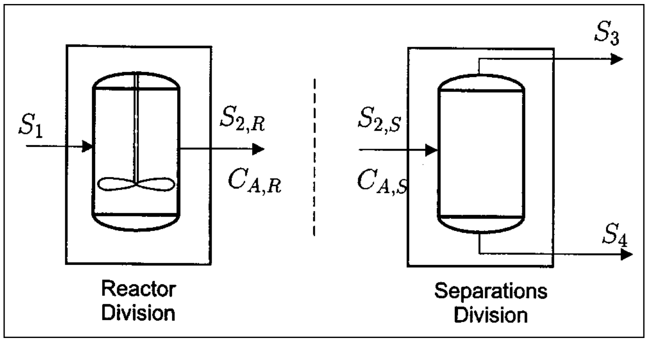



**Figure 7. Tearing a process stream to create upstream and a downstream versions of $C_A$.**

multiplier

$$\max_{(C_{A,R},C_{A,S})} \quad f_R(C_{A,R}) + f_S(C_{A,S}) - \lambda(C_{A,S} - C_{A,R})$$
$$\text{subject to:} \qquad (9)$$
$$\text{Reactor division's private constraints}$$
$$\text{Separations division's private constraints.}$$

The problem is then decomposed into two subproblems. The reactor division's problem is

$$\max_{C_{A,R}} \quad f_R(C_{A,R}) + \lambda C_{A,R}$$
$$\text{subject to:} \qquad (10)$$
$$\text{Reactor division's private constraints,}$$

and the separations division's problem is

$$\max_{C_{A,S}} \quad f_S(C_{A,S}) - \lambda C_{A,S}$$
$$\text{subject to:} \qquad (11)$$
$$\text{Separations division's private constraints.}$$

$\lambda$ is set to some initial value, and the subproblems are solved to determine values of $C_{A,R}$ and $C_{A,S}$. The Lagrange multiplier $\lambda$ is repeatedly adjusted until the consistency constraint is satisfied. Although this is a standard technique, it cannot solve this problem: there is no $\lambda$ which causes the consistency constraint to hold. When Lagrangean substitution is applied; it will not converge (see Figure 8). We discuss this failure in more detail in the section on slack auctions and Lagrangean relaxation.

## Slack Resources

The difficulties in pricing interprocess streams occur because interprocess resources are not well-defined. One way to overcome this is to remodel the microsupply chain, so that meaningful resource constraints are evident. Consider the consistency constraint in the Lagrangean substitution approach

$$C_{A,R} = C_{A,S}. \qquad (12)$$

This can be rewritten so that it has the form of a resource constraint

$$C_{A,R} = C_{A,S} \qquad (13)$$

$$C_{A,R} - C_{A,S} = 0 \qquad (14)$$

$$|C_{A,R} - C_{A,S}| \le 0 \qquad (15)$$

$$\underbrace{1/2|C_{A,R} - C_{A,S}|}_{D_R} + \underbrace{1/2|C_{A,R} - C_{A,S}|}_{D_S} \le 0. \qquad (16)$$

The lefthand side of Eq. 16 can be interpreted as an aggregate demand for resources, $D_R$ being the demand of the reactor division and $D_S$ being the demand for the separations division; the righthand side can be interpreted as a resource supply. Since $|C_{A,R} - C_{A,S}|$ is the amount of slack in Eq. 16 (actually, it is the negative of the slack), we call these new resources *slack resources*. In general, Eq. 16 can be written as

$$1/2|C_{A,R} - C_{A,S}| + 1/2|C_{A,R} - C_{A,S}| \le S, \qquad (17)$$

where $S > 0$ is the slack resource supply. These slack resources alter the structure of the plantwide microsupply chain. Instead of one division being a seller and the other being a buyer, *both* divisions are buyers—the seller of the slack is, in some sense, a representative of the plant itself (see Figure 9).

Slack resources are the appropriate way of viewing microsupply chain resources, because they capture the idea of resource demand, resource supply, and resource scarcity. Slack resources becomes scarce when the slack demand exceeds the slack supply—when the reactor and separations divisions want different values of $C_A$. And the slack supply is essentially the plant's tolerance for how different $C_{A,R}$ and $C_{A,S}$ can be. The slack demand can be controlled by charging the divisions based on how much slack they desire. The appropriate prices can be found using slack resource auctions.

## Slack Resource Auctions

The goal of any auction is to find the prices of resources with respect to a set of buyers (such as the divisions in a plant). An auctioneer adjusts resource prices until the equilibrium conditions (Eqs. 3, 4, and 5) are satisfied. When resources are sold at equilibrium prices, they are optimally allocated in the sense that the sum of the divisions' objectives is maximized (see the Appendix).

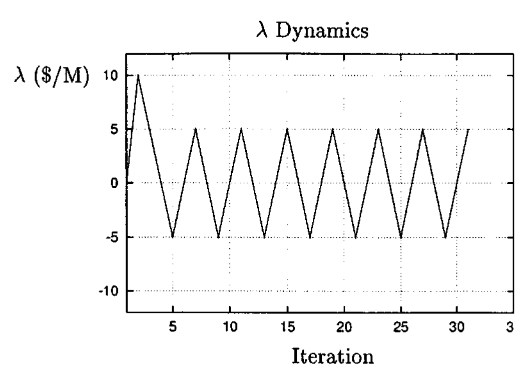



**Figure 8. Lagrangean substitution has convergence problems.**

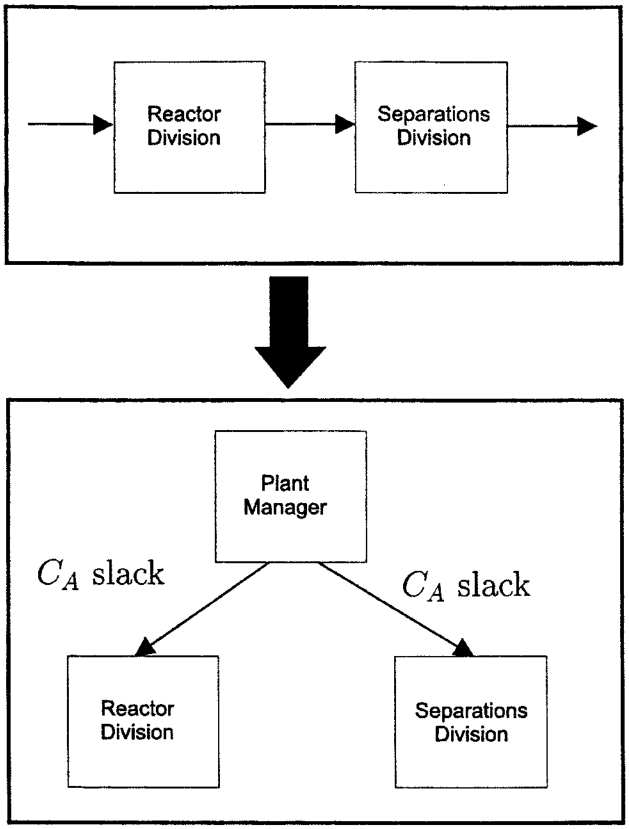



**Figure 9. Remodeling the plantwide microsupply chain using slack resources.**

Consider the reactor division's slack demand

$$D_R = 1/2|C_{A,R} - C_{A,S}|. \tag{18}$$

In order for the reactor division to compute this demand, it must know the value of $C_{A,S}$. Therefore, when an auctioneer announces prices, it must also announce some reference value of $C_{A,S}$ for the reactor division to use. Denote this reference as $C^0_{A,S}$ and call it the *slack pole* of the reactor division. Given a slack pole and a price $p$, the reactor division solves the following problem to compute its slack demand

$$\begin{aligned} &\max_{C_{A,R}} \quad f_R(C_{A,R}) - p \cdot 1/2|C_{A,R} - C^0_{A,S}| \\ &\text{subject to:} \\ &\quad \text{Reactor division's private constraints.} \end{aligned} \tag{19}$$

The slack auction procedure is shown in Table 1. Suppose the supply of concentration slack is $S = 0.0001$ M, and suppose for practical purposes that the auctioneer considers any price less than \$0.001/M to be sufficiently close to \$0/M. Following the procedure in Table 1, the auctioneer sets the slack price $p = \$0/\text{M}$. The separations division then responds with its optimal operating point $C^*_{A,S} = 0.90$ M (see Figure 6). The reactor division being indifferent to $C_A$ responds with some feasible value, say $C^*_{A,R} = 0.50$ M. For these proposed values of $C_A$, the slack resource constraint is violated

**Table 1. Slack Auction Procedure for Illustrative Problem**

The auctioneer sets the slack price $p = \$0/\text{M}$.
The divisions solve their problems and respond with $C^*_{A,R}$ and $C^*_{A,S}$.
If equilibrium conditions hold:
  Stop. The global solution has been found
Until equilibrium conditions hold or if $p$ is sufficiently close to \$0/M:
  The auctioneer sets $C^0_{A,S} = C^*_{A,R}$.
  The auctioneer sets $C^0_{A,R} = C^*_{A,S}$.
  The auctioneer adjusts $p$.
  The auctioneer announces $p$, $C^0_{A,R}$, and $C^0_{A,S}$ to the divsions.
  The divisions solve their problems and respond with $D_R$, $C^*_{A,R}$ and $D_S$, $C^*_{A,S}$.
Stop. The global solution has been found.

$$1/2|0.90 - 0.50|M + 1/2|0.90 - 0.50|M = 0.4M \nleq 0.0001M. \tag{20}$$

Since the resource constraint does not hold, the auction must continue. In the second iteration, the auctioneer sets $C^0_{A,S} = C^*_{A,R} = 0.50$ M and $C^0_{A,R} = C^*_{A,S} = 0.90$ M. It raises the slack price and announces $p$, $C^0_{A,R}$, and $C^0_{A,S}$ to the divisions. The reactor division responds with $D_R = 0$, $C^*_{A,R} = 0.90$ M, and the separations division responds with $D_S = 0.20$ M, $C^*_{A,S} = 0.90$ M. Since the slack resource constraint is still violated, the auctioneer must raise prices again. The remaining iterations are shown in Table 2. The slack auction terminates at an equilibrium price of \$0.001/M ≈ \$0/M and at $C_{A,R} = C_{A,S} = 0.90$ M, which is the global solution to this problem. Although this is a simple optimization problem, recall that Lagrangean substitution was unable to solve it.

### *Remarks*

One might argue, from a computational perspective, that slack resources should be defined using a square function rather than an absolute value. In other words, that Eq. 16 should be expressed as

$$1/2(C_{A,R} - C_{A,S})^2 + 1/2(C_{A,R} - C_{A,S})^2 \leq 0. \tag{21}$$

While square functions are easier for numerical solvers to handle (because of smoothness), there are several reasons for defining slack resources as absolute differences. First of all, when the divisions' proposed slack poles are moderately far apart, a squared difference tends to exaggerate the required demand in the same way that the integral of the squared error (ISE) attaches proportionally more weight to moderate controller error. Likewise, as the slack demand decreases, a

**Table 2. Slack Auction Results for Reactor/Separations Problem**

| Iter. | p (\$/M) | $C^*_{A,R}$ (M) | $D_R$ (M) | $C^*_{A,S}$ (M) | $D_S$ (M) |
|---|---|---|---|---|---|
| 1 | 0.0 | 0.50 | — | 0.90 | — |
| 2 | 1.0 | 0.90 | 0.00 | 0.90 | 0.20 |
| 3 | 10.0 | 0.90 | 0.00 | 0.90 | 0.00 |
| 4 | 1.0 | 0.90 | 0.00 | 0.90 | 0.00 |
| 5 | 0.1 | 0.90 | 0.00 | 0.90 | 0.00 |
| 6 | 0.01 | 0.90 | 0.00 | 0.90 | 0.00 |
| 7 | 0.001 | 0.90 | 0.00 | 0.90 | 0.00 |

squared difference tends to underestimate the actual demand, leading to slack prices which are artificially depressed.

Squared differences are less natural and less physically meaningful. When a plant operator wants a process stream to be hotter, he does not think in terms of increasing the "squared temperature," nor does a plant manager think to increase the "squared production rate" of a particular compound. It is more natural to think in terms of absolute values, and it is easier to interpret the associated slack prices when slack resources are defined this way. Squared differences add a layer of artificiality to the procedure.

Using squared differences also leads to something called *price discrimination*. Price discrimination occurs when the amount that one person pays for a resource is different than what another pays for the same resource. From a numerical point of view, price discrimination is unimportant. When dealing with multiple decision-makers, however, price discrimination can encourage strategic behavior like lying and cheating, which can complicate the solving of the optimization problem.

## Optimizing a Cyclohexane Plant by Pricing Interprocess Streams

Consider a plant that produces a liquid cyclohexane by hydrogenating benzene. The plant receives benzene and hydrogen feeds at no charge from another plant in the same company, but has the option of buying more hydrogen from an external market (see Figure 10). The price of market hydrogen is fixed at \$0.28/mol, but the cyclohexane price depends on the mole fraction of cyclohexane according to

$$p_C(x_C) = 0.66\$/\text{kg} + 654.53\$/\text{kg} \times (x_C - 0.9970), \quad (22)$$

where $p_C$ is the market price of cyclohexane at a cyclohexane mole fraction $x_C$. Market hydrogen and plant hydrogen are mixed in Mixer $M1$ and sent to Heater $H1$ along with a benzene stream for preheating. This combined stream is sent to Reactor $R1$ and reacted to equilibrium at the preheated temperature. The outlet stream is then cooled and sent to a flash vessel whose liquid outlet is taken as the cyclohexane product.

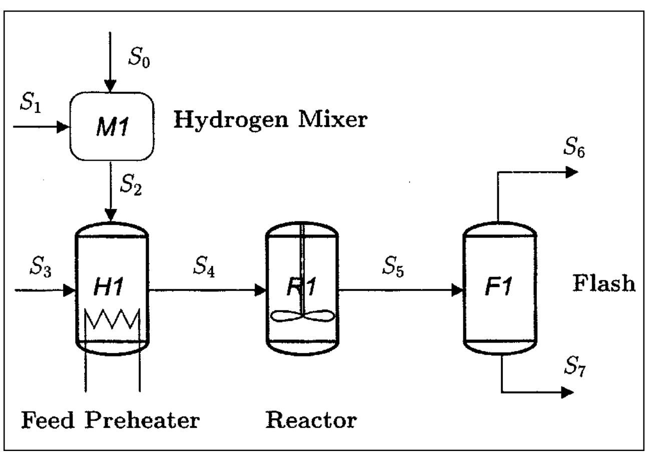



**Figure 10. Cyclohexane plant.**

**Table 3. Stream Descriptions**

| Stream | $T$ (K) | $P$ (atm) | Description |
|---|---|---|---|
| S0 | 422.0 | 10.0 | Plant hydrogen: 300 mol/h |
| S1 | 422.0 | 10.0 | Market hydrogen |
| S2 | 422.0 | 10.0 | Combined hydrogen feed |
| S3 | 297.0 | 1.0 | Plant benzene: 100 mol/h |
| S4 | 477.6–560.9 | 1.0 | Reactor feed: hydrogen and benzene |
| S5 | 477.6–560.9 | 1.0 | Reactor outlet: cyclohexane, hydrogen, and benzene |
| S6 | 297.0–366.5 | 1.0 | Flash vapor stream: hydrogen, benzene, cyclohexane |
| S7 | 297.0–266.5 | 1.0 | Product stream: cyclohexane, benzene |

The heater $H1$ can operate between 477.6 and 560.9 K; the flash vessel can operate between 297.0 and 366.5 K. The basis for calculation is 100 mol/h benzene. Vapor pressures are estimated using the Antoine equation; the optimizer FSQP (Zhou and Tits, 1997) is used to solve the reactor and flash models. All units use water with a 5.5 K rise for cooling and a 400 lb steam for heating. The input stream properties are shown in Table 3, and the utility and material costs are shown in Table 4.

### *Problem description*

The goal of this problem is to find a steady operating state which maximizes the plantwide profit. In most centralized approaches, one explicitly defines decision variables and constraints for every state variable of every stream and every unit operation in a plant, potentially leading to hundreds of thousands of variables and constraints. Many of these are *aggravation* variables and aggravation constraints which are necessary internally, but which add nothing useful to a high-level description of the problem. In truth, the number of actual decision variables is usually quite small; it is always equal to the number of degrees of freedom in the design problem. If there are no degrees of freedom, then a plant is "optimized" by solving the equations making up the plant model. Only when the design problem is underspecified does plantwide optimization make any sense. In this problem, there are dozens of aggravation variables and constraints, but there are only *three* degrees of freedom. We use $F_1$, $T_4$, and $T_7$ as our decision variables (see Table 5 for a description of the notation for this section).

The plantwide optimization problem is to maximize the overall profit

$$\max_{F_1,T_4,T_7} \underbrace{p_C(x_7(F_1,T_4,T_7)) \cdot F_7(F_1,T_4,T_7) \cdot MW_C}_{\text{Revenue from cyclohexane product}} - \underbrace{p_H \cdot F_1}_{\text{Hydrogen cost}} - \underbrace{p_S \cdot Q_{H1}(F_1,T_4)}_{\text{Preheater steam cost}} - \underbrace{p_W \cdot Q_{R1}(F_1,T_4)}_{\text{Reactor cooling cost}} - \underbrace{p_{\text{water}} \cdot Q_{F1}(F_1,T_4,T_7)}_{\text{Flash cooling cost}}$$

**Table 4. Utility and Material Costs**

| Description | Price |
|---|---|
| 400 lb steam | $p_S$ = \$5.51 per 1,000 kg |
| Cooling water | $p_W$ = \$0.013 per 1,000 L |
| Market hydrogen | $p_H$ = \$0.28/mol |

**Table 5. Notation for Cyclohexane Example**

| | |
|---|---|
| $F_k$ = | total molar flow rate of stream $S_k$ ($F_k \in \Phi_{\mathrm{mol/h}}$) |
| $T_k$ = | temperature of stream $S_k$ ($T_k \in \tilde{\Phi}_T$) |
| $x_k$ = | mole fraction of cyclohexane in stream $S_k$ ($x_k \in \Phi_{\Re}$) |
| $MW_C$ = | molecular weight of cyclohexane |
| $Q_{\mathrm{H1}}$ = | preheater's heating load in terms of steam ($Q_{\mathrm{H1}} \in \Phi_{\mathrm{kg}}$) |
| $Q_{\mathrm{R1}}$ = | reator's cooling load in terms of water ($Q_{\mathrm{R1}} \in \Phi_1$) |
| $Q_{\mathrm{F1}}$ = | flash's cooling load in terms of water ($Q_{\mathrm{F1}} \in \Phi_1$) |
| $\hat{y}$ = | mole fraction in the reactor outlet on a hydrogen-free basis ($\hat{y} \in \Phi_{\Re}$) |
| H:B = | hydrogen to benzene ratio in the reactor feed (H:B $\in \Phi_{\Re}$) |

subject to

$$0.0 \text{ mol/hr} \le F_1 < \infty \text{ mol/hr}$$
$$477.6 \text{ K} \le T_4 \le 560.9 \text{ K}$$
$$297.0 \text{ K} \le T_7 \le 366.5 \text{ K}$$

The objective is nonlinear and continuous, all the decision variables are continuous, and the problem constraints are linear and convex. The objective is evaluated by simulating the cyclohexane plant for given values of $F_1$, $T_4$, and $T_7$. The solution to this problem is shown in Table 6.

### *Defining the microsupply chain*

Defining the units in a microsupply chain and deciding on the slack resources is a *hard* problem. There is no one right way to do this. Our main goal here is to show how microsupply chains can be optimized, not necessarily how to define them. Therefore, let us define the microsupply chain in the obvious way: tear $S4$ and $S5$ to create a feed division, a reactor division, and a separations division (see Figure 11). The slack resources we have chosen to use are defined over $T_4$, H:B, and $\hat{y}$.

During the auction, the feed division's problem is to solve

$$\max_{F_1, T_{4F}, \mathrm{H{:}B}_F} \; -p_H \cdot F_1 - p_s \cdot Q_{H1}(F_1, T_{4F}) - p_{T_4} \cdot 1/2 |T_{4F} - T_{4R}^0| - p_{\mathrm{H:B}} \cdot 1/2 \mathrm{H{:}B}_F(F_1) - \mathrm{H{:}B}_R^0| \quad (23)$$

subject to

$$\text{Feed division's modeling equations}$$
$$0.0 \text{ mol/hr} \le F_1 < \infty \text{ mol/hr}$$
$$477.6 \text{ K} \le T_{4F} \le 560.9 \text{ K}$$

The reactor division's problem is

$$\max_{T_{4R}, \mathrm{H{:}B}_R} \; - p_S \cdot Q_{R1}(F_1(\mathrm{H{:}B}_R), T_{4R}) - p_{T_4} \cdot 1/2 |T_{4R} - T_{4F}^0| - p_{\mathrm{H:B}} \cdot 1/2 |\mathrm{H{:}B}_R - \mathrm{H{:}B}_F^0| - p_{\hat{y}} \cdot 1/2 |\hat{y}_R(T_{4R}, \mathrm{H{:}B}_R) - \hat{y}_S^0| \quad (24)$$

**Table 6. Solution to Cyclohexane Problem**

| Variable | Value |
|---|---|
| $F_1$ | 10.70 mol/h |
| $T_4$ | 477.6 K |
| $T_7$ | 310.65 K |
| Objective | 12.49 $/h |

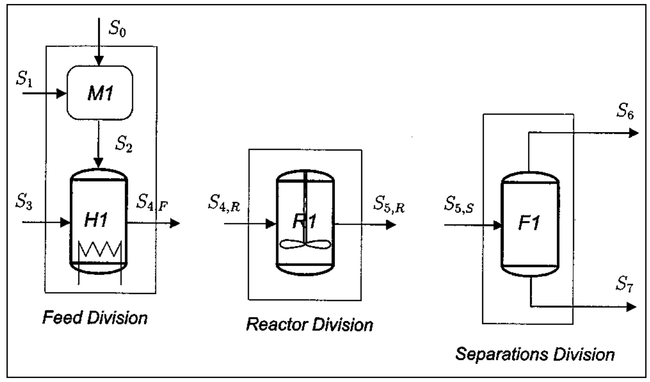



**Figure 11. Decomposing the cyclohexane plant.**

subject to

$$\text{Reactor division's modeling equations}$$
$$0.0 \text{ mol/hr} \le F_1 < \infty \text{ mol/hr}$$
$$477.6 \text{ K} \le T_{4R} \le 560.9 \text{ K}.$$

And the separations division's problem is

$$\max_{\hat{y}_S, T_7} \; p_C(x_7(\hat{y}_S, T_7)) \cdot F_7(\hat{y}_S, T_7) \cdot MW_C - p_W \cdot Q_{F1}(\hat{y}_S, T_7) - p_{\hat{y}} \cdot 1/2 |\hat{y}_S - \hat{y}_R^0| \quad (25)$$

subject to

$$\text{Separations division's modeling equations}$$
$$0.0 \text{ mol/hr} \le F_1 < \infty \text{ mol/hr}$$
$$297.0 \text{ K} \le T_7 \le 366.5 \text{ K}.$$

The slack auction procedure for this problem is shown in Table 7—we assume that the slack supplies are all zero. When a slack auction is applied to this problem, it terminates in 13 iterations; the final slack poles and slack prices are shown in Table 8, and the corresponding solution is shown in Table 9. Note that the slack auction solution is identical to the global solution as prescribed by Theorem 2.

**Table 7. Slack Auction Procedure for Cyclohexane Problem**

The auctioneer sets $p_{T_4}$, $p_{\mathrm{H:B}}$, and $p_{\hat{y}}$ all to zero.
The feed division responds with $T_{4F}^*$ and $\mathrm{H{:}B}_F^*$.
The reactor division responds with $T_{4R}^*$, $\mathrm{H{:}B}_R^*$, and $\hat{y}_R^*$.
The separations division responds with $\hat{y}_S^*$.
If equilibrium conditions hold:
  Stop. The global solution has been found
Until equilibrium conditions hold:
  The auctioneer sets $T_{4R}^0 = T_{4F}^*$ and $T_{4F}^0 = T_{4R}^*$.
  The auctioneer sets $\mathrm{H{:}B}_R^0 = \mathrm{H{:}B}_F^*$ and $\mathrm{H{:}B}_F^0 = \mathrm{H{:}B}_R^*$.
  The auctioneer sets $\hat{y}_R^0 = \hat{y}_S^*$ and $\hat{y}_S^0 = \hat{y}_R^*$.
  The auctioneer adjusts $p_{T_4}$, $p_{\mathrm{H:B}}$, and $p_{\hat{y}}$.
  The auctioneer announces prices and slack poles to the divisions.
  The feed division responds with $D_{T_{4F}}$, $T_{4F}^*$ and $D_{\mathrm{H:B}_F}$, $\mathrm{H{:}B}_F^*$.
  The reactor division responds with $D_{T_{4R}}$, $T_{4R}^*$; $D_{\mathrm{H:B}_R}$, $\mathrm{H{:}B}_R^*$; and $D_{\hat{y}_R}$, $\hat{y}_R^*$.
  The separations division responds with $D_{\hat{y}_S}$, $\hat{y}_S^*$.
Stop. The global solution has been found.

**Table 8. Auction Results**

| Slack Resource | Price | Upstream Pole | Downstream Pole |
|---|---|---|---|
| T | \$1.44/K/h | 477.6 K | 477.6 K |
| H:B | \$0.64/h/pph | 3.107 | 3.107 |
| y | \$7.42/h/ppt | 0.99950 | 0.99950 |

*Remarks: recycle loops*

It is natural at this point to ask whether the presence of recycle loops poses any difficulties to a slack auction. In short, it does not. Recycle loops can only lead to problems when the decisions a unit makes at some point in time eventually result in its making amplified versions of those decisions later on (that is, when there is the possibility of positive feedback). This should not happen during slack auctions for two main reasons. In the first place, a slack auction starts by having the units find their optimal operating points independent of the other units in the system. This initial state cannot be improved upon during the course of the auction, because the slack prices are always nonnegative. Increasing the slack price in a recycle loop can only drive the state of a unit away from its optimum—it cannot lead to a state of positive feedback. Secondly, the feedback of slack resource information in a recycle loop is different from the feedback of ordinary information. The feedback of ordinary information may lead to instability. Slack resources, however, are always defined between two units with competing interests. (For the degenerate case, where there is only a single unit, the "competing" interests are one and the same, and the slack demand in the recycle loop will always be zero.) The presence of slack resources tend to "pull" the units away from their optimal operating states—it can never enhance them. At equilibrium, the slack prices in a recycle loop balance the benefits of a unit's actions vs. the cost of their effect on the unit in the form of feedback. The authors have demonstrated the application of slack auctions to problems with recycle loops elsewhere (see Jose and Ungar, 1998).

## Slack Auctions and Lagrangean Relaxation

Slack auctions fall under the heading of decomposition and Lagrangean relaxation in optimization theory (Mesarović, et al., 1970; Lasdon, 1971; Bertsekas, 1976). In slack auctions, plantwide optimization problems are decomposed by relaxaing interprocess resource constraints; this is manifested as a slack resource charge that each unit pays during the auction. Although slack auctions are a kind of Lagrangean relaxation, they are different from traditional techniques in that they relax only *resource* constraints. In general, relaxation techniques like Lagrangean substitution relax constraints even if they are not resource constraints. This leads to improperly defined resources and to potential convergence problems as

**Table 9. Slack Auction and Global Solutions**

| Variable | Auction | Global |
|---|---|---|
| Reactor Temp. | 477.6 K | 477.6 K |
| Market Hydrogen | 10.70 mol/h | 10.70 mol/h |
| Flash Temp. | 310.65 K | 310.65 K |
| Objective | \$12.49/h | \$12.49/h |

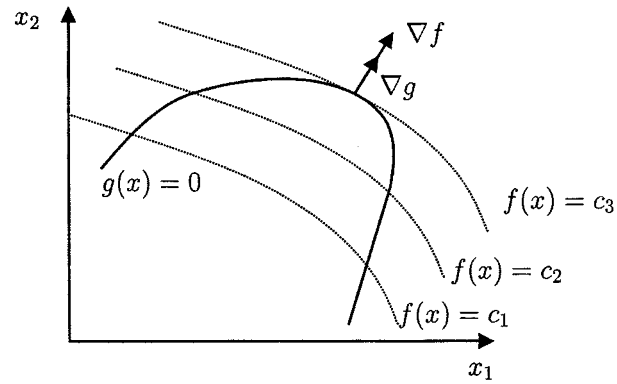



**Figure 12. Method of Lagrange multipliers.**

we saw in the section on difficulty in pricing interprocess streams. In this section, we discuss why Lagrangean substitution has difficulty solving some simple, but common problems.

*Case 1: indifferent divisions*

When one or more divisions are indifferent to the value of an interprocess stream state, Lagrangean substitution will fail. This failure can be explained using the Method of Lagrange Multipliers.

Consider the following optimization problem

$$\begin{aligned} &\max_{x_1, x_2} \quad f(x_1, x_2) \\ &\text{subject to:} \\ &\quad g(x_1, x_2) = 0, \end{aligned} \tag{26}$$

The solution to this problem can be found by plotting the constraint $g(x) = 0$ and the level sets of $f(x)$ on a single graph. The solution occurs where the objective is tangent to the constraint curve—where the gradients of the objective and of the constraint point in the same direction (see Figure 12). This condition can be concisely stated as

$$\nabla f(x^*) = \lambda \nabla g(x^*), \tag{27}$$

where $\lambda$ is the Lagrange multiplier and $(x^*)$ is the optimal solution.

Now consider the reactor/separation problem (Eq. 8) before it is decomposed into subproblems

$$\begin{aligned} &\max_{C_{A,R}, C_{A,S}} \quad f_R(C_{A,R}) + f_S(C_{A,S}) \\ &\text{subject to:} \\ &\quad \text{Reactor division's private constraints} \\ &\quad \text{Separations division's private constraints} \\ &\quad C_{A,R} = C_{A,S}. \end{aligned} \tag{28}$$

Plot the constraint $C_{A,R} - C_{A,S} = 0$ and the level sets of the objective function together (see Figure 13). Because the level sets are never tangent to the constraint line, the only place where $\nabla f = \lambda \nabla g$ is true is where $\nabla f = 0$ and $\lambda = 0$. This is the

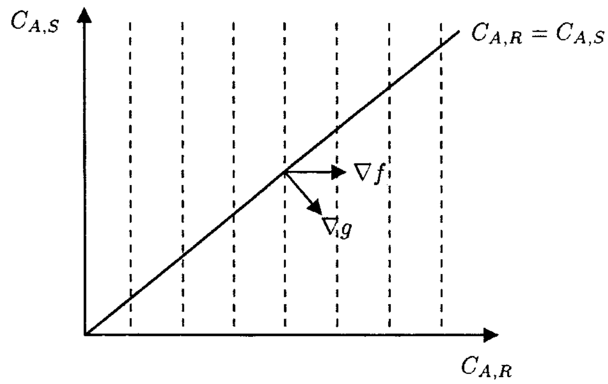



**Figure 13. Gradients of the objective and the constraints are never parallel.**

correct answer for the problem before it is decomposed, but after decomposition, a Lagrange multiplier of zero is useless. A multiplier of zero implies that the consistency constraint can be ignored. This is only true when both divisions choose the same value of $C_A$ independently of each other. In general, this is not the case, and ignoring the consistency constraint results in infeasible candidate solutions and convergence failure. Thus, Lagrangean substitution cannot be applied to these types of problems.

*Case 2: divisions with linear objectives*

Consider the following optimization problem

$$\begin{aligned} &\max_{x_1, x_2} \quad k_1 x_1 + k_2 x_2 \\ &\text{subject to:} \\ &\quad \text{Division 1's private constraints} \\ &\quad \text{Division 2's private constraints} \\ &\quad x_1 = x_2, \end{aligned} \tag{29}$$

where $k_1, k_2 > 0$ and $x_1, x_2 \in \Re$. Applying Lagrangean substitution yields the following subproblems

$$\begin{aligned} &\max_{x_1} \quad k_1 x_1 + \lambda x_1 \\ &\text{subject to:} \\ &\quad \text{Division 1's private constraints,} \end{aligned} \tag{30}$$

and

$$\begin{aligned} &\max_{x_2} \quad k_2 x_2 - \lambda x_2 \\ &\text{subject to:} \\ &\quad \text{Division 2's private constraints.} \end{aligned} \tag{31}$$

If $\lambda > -k_1$, then Division 1 proposes its maximum feasible value of $x_1$; if $\lambda < -k_1$, then Division 1 proposes its minimum feasible value. Likewise, if $\lambda > k_2$, then Division 2 proposes its minimum feasible value of $x_2$; if $\lambda < k_2$, it proposes its maximum feasible value. Since, in general, all of these possible proposals are different, there is no $\lambda$ which can cause the relaxed constraint $x_1 = x_2$ to hold. Thus, Lagrangean substitution will not converge here either.

*Case 3: divisions with piecewise linear objectives*

Now consider the general case where both divisions have piecewise linear objectives. It can be argued that the global optimum occurs at one of the breakpoints of one of the division's objective functions (see Figure 14). If not, then the optimum must lie between adjacent breakpoints of both divisions' utility functions. However, since the objectives for both divisions are linear over both intervals, their sum must also be linear. Thus, if the sum is not zero, there must be a direction where the sum of the objectives increases. However, this contradicts the assumption that the optimum did not occur at a breakpoint; therefore, the global optimum must lie at one of the divisions' breakpoints. (In the case where the sum of the divisions' objectives over the interval is zero, then at least one of the divisions' breakpoints must be no worse than any point in the interval—pick that breakpoint to be the global optimum.)

It follows that the divisions will only propose stream states lying at breakpoints in their objectives. Adjusting $\lambda$ only causes the divisions to change the breakpoint they propose—it can never make them select points in the interval between two adjacent breakpoints. Since, in general, divisions have no breakpoints in common, there is no $\lambda$ that causes both divisions to propose the same stream states and Lagrangean substitution will fail.

Lagrangean substitution cannot force divisions to select points lying between breakpoints in their objectives; slack auctions do this naturally. Slack auctions allow divisions to propose points which are not breakpoints of their nominal objective functions by temporarily adding new kinks or breakpoints to their objectives via the slack resource charge. If the divisions' objectives are concave, then, at the end of a slack auction, the final kink will coincide with the system solution.

## Conclusion

In this article, we described how modular plantwide optimization could be accomplished by pricing interprocess streams. The initial difficulty with pricing streams, however, is that they cannot be treated as traditional resources in a mathematical programming sense. By remodeling chemical plants using a supply chain metaphor with an emphasis on meaningful resources, it became evident that slack resources are the appropriate way of viewing interprocess streams. These slack resources were priced by running slack auctions;

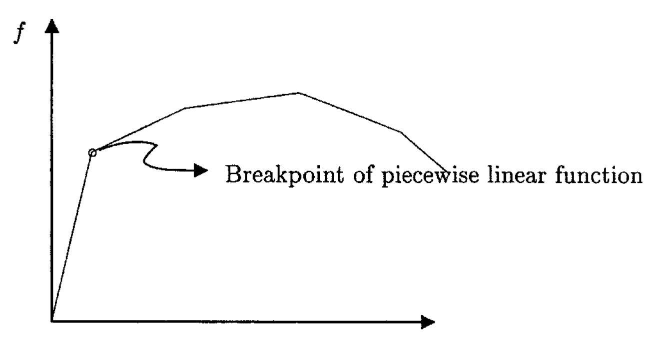



**Figure 14. Piecewise linear objective function.**

when slack auctions terminate with equilibrium slack prices, the plant can be proved to operate at a globally optimal state.

Equilibrium slack prices can be used for more than the decentralized optimization of chemical plants. These prices can also be used to identify areas which may benefit from process redesign by estimating the return on capital investments. For instance, a temperature slack price measures the benefit of adding a heat exchanger to a process stream, and a concentration slack price determines the highest price a plant manager should pay for intermediates from another company or on the open market. Because slack prices can measure the costs of interaction between plant divisions, it is conceivable that they can also be used to define cost-based analogs of standard controller performance indices such as the integrals of the squared and absolute error (ISE/IAE). Rather than minimizing the controller error, one could minimize the *cost* of controller error. Control systems designed with this goal should outperform those based on generic measures.

## Notation

$\Phi_\alpha$ = metric space consisting of all numbers with the same physical type as $\alpha$
$n$ = number of divisions in the system
$m$ = number of resource types
$i$ = index referring to a generic division
$j$ = index referring to a generic resource
$X_i$ = set of options available to Division $i$
$x_i$ = one of Division $i$'s options ($x_i \in X_i$)
$X$ = Cartesian product of all $X_i$ ($X = \Pi_{i=1}^{n} X_i$)
$x$ = tuple of the divisions' options ($x \in X$)
$\Phi_\$$ = metric space representing money
$f_i$ = objective function for Division $i$ ($f_i: X_i \to \Phi_\$$)
$R_j$ = metric space whose elements represent amounts of Resource $j$
$R$ = Cartesian product of all resource types ($R = \Pi_{j=1}^{m} R_j$)
$d_i$ = function that gives Division $i$'s resource demand for pursuing an option in its feasible set ($d_i: X_i \to R$)
$d_{i,j}$ = function that gives Division $i$'s demand for Resource $j$ ($d_{i,j}: X_i \to R_j$)
$D_j = D_j = \sum_{i=1}^{n} d_{i,j}(x_i)$, a function that returns the total demand for Resource $j$ ($D_j: X \to R_j$)
$P_j$ = metric space representing prices of Resource $j$
$p_j$ = nonnegative price for Resource $j$ ($p_j \in P_j$)
$P$ = Cartesian product of all prices ($P = \Pi_{j=1}^{m} P_j$)
$p$ = nonnegative price vector ($p \in P$)

## Literature Cited


Axelsson, B., and G. Easton, eds., *Industrial Networks: A New View of Reality*, Routledge, New York (1992).
Backx, T., O. Bosgra, and W. Marquardt, "Towards Intentional Dynamics in Supply Chain Conscious Process Operations," FOCAPO98, Snow Bird, UT (1998).
Bailey, J. K., A. N. Hyrmak, S. S. Treiber, and R. B. Hawkins, "Nonlinear Optimization of a Hydrocracker Fractionation Plant," *Computers and Chem. Eng.*, **17**, 123 (1993).
Bertsekas, D. P., "On Penalty and Multiplier Methods for Constrainted Minimization," *SIAM J. Control Opt.*, **12**, 216 (1976).
Dadebo, S. A., and K. B. McAuley, "Dynamic Optimization of Constrained Chemical Engineering Problems Using Dynamic Programming," *Computers and Chem. Eng.*, **19**, 513 (1995).
Dantzig, G. B., and P. Wolfe, "Decomposition Principle for Linear Programs," *Oper. Res.*, **8**, 101 (1960).
Darby, M. L., and D. C. White, "On-Line Optimization of Complex Process Units," *Chem. Eng. Prog.*, **10**, 51 (1988).
Everett, H., "Generalized Lagrange Multiplier Method for Solving Problems of Optimum Allocation of Resources," *Oper. Res.*, **11**, 399 (1963).
Findstein, W., F. N. Bailey, M. Brdyś, K. Malinowski, P. Tatjewski, and A. Woźniak, *Control and Coordination in Hierarchical Systems*, Wiley, New York (1980).
Gibson, W. D., "Getting a Grip on the Supply Chain," *Chem. Eng.*, **11**, 35 (1998).
Jerome, N., Personal communication, Amoco headquarters, Chicago (1998).
Jose, R. A., and L. H. Ungar, "Auction-Driven Coordination for Plantwide Optimization," FOCAPO98, Snow Bird, UT (1998).
Jung, B. S., W. Mirosh, and W. H. Ray, "Large Scale Process Optimization Techniques Applied to Chemical and Petroleum Processes," *Can. J. of Chem. Eng.*, **49**, 844 (1971).
Kocis, G. R., and I. E. Grossman, "Relaxation Strategy for the Structural Optimization of Flow-Sheets," *Ind. Eng. Chem. Res.*, **26**, 1869 (1987).
Koppel, L., "Smart Manufacturing and Integration," FOCAPO98, Snow Bird, UT (1998).
Lasdon, L. S., *Optimization Theory for Large Systems*, Macmillan, New York (1970).
Mekarapiruk, W., and R. Luus, "Optimal Control of Final State Constrained Systems," *Can. J. of Chem. Eng.*, **75**, 806 (1997).
Mesarović, M. D., D. Macko, and Y. Takahara, *Theory of Hierarchical Multilevel Systems*, Academic Press, New York (1970).
Rudd, D., and C. Watson, *Strategy of Process Engineering*, Wiley, New York (1968).
Stephanopoulos, G., and A. W. Westerberg, "The Use of Hestenes' Method of Multipliers to Resolve Dual Gaps in Engineering System Optimization," *JOTA*, **15**, 285 (1975).
Vollmann, T., C. Cordon, and H. Raabe, "Supply Chain Management," The Complete MBA Companion, Pitman Publishing, London, p. 316 (1997).
Wang, W. P., P. B. Luh, D. L. Kleinman, and S. C. Chang," Coordination Mechanisms: A View from Dual Variables," *Proc. 30th Conference on Decision and Control*, Sponsored by IEEE Control Systems Society (1991).
Watanabe, N., Y. Nishimura, and M. Mastubara, "Decomposition in Large System Optimization using the Method of Multipliers," *JOTA*, **25**, 181 (1978).
Wilde, D. J., "Production Planning of Large Systems," *Chem. Eng. Prog.*, **59**, 46 (1963).
Zhou, J. L., and A. Tits, "User's Guide for FFSQP Version 3.7: A Fortran Code for Solving Optimization Programs with General Inequality Constraints," Technical Report SRC-TR-92-107r5, Institute for Systems Research, Univ. of Maryland, College Park (1997).


## Appendix: Global Optimality of Slack Auctions

The auction formulation essentially converts an optimization problem of the form

$$\begin{aligned} &\max_{x_1,\ \dots,\ x_n} \sum_{i=1}^{n} f_i(x_i) \\ &\text{subject to:} \\ &\quad \sum_{i=1}^{n} d_i(x_i) \leq S \\ &\quad x_i \in X_i \qquad \text{for } i \in \{1,\ \dots,\ n\}. \end{aligned} \tag{A1}$$

into a *separable* problem by relaxing the resource constraints; the first step is to prove that separable problems can be solved by optimizing each piece of the problem separately. The next step is to show that at a given set of prices, the solution to the relaxed problem solves a variant of the original problem. Under certain conditions, this variant is identical to the original problem and the solution to the relaxed problem therefore solves the original problem; these conditions are exactly the auctioneer's equilibrium conditions Eqs. 3, 4, and 5. The final step is to invoke the separability results to show that the auction formulation yields solutions which are globally optimal.

### Optimizing separable problems

A function $f(x_1, x_2)$ is called *separable* if it can be expressed as the sum of two functions—one depending only on $x_1$ and one depending only on $x_2$

$$f(x_1, x_2) = f_1(x_1) + f_2(x_2). \tag{A2}$$

A feasible set $F \subseteq X_1 \times X_2$ is separable only if $F = X_1 \times X_2$ —in other words, when $x_1$ and $x_2$ can be chosen independently of one another and always be feasible. An optimization problem is called separable if its objective is separable and if its feasible set is separable. For instance, consider the following problem

$$\begin{aligned} &\max_{x_1, x_2} \quad f(x_1, x_2) \\ &\text{subject to:} \\ &\quad (x_1, x_2) \in F. \end{aligned} \tag{A3}$$

This problem is separable if it can be expressed as

$$\begin{aligned} &\max_{x_1, x_2} \quad f_1(x_1) + f_2(x_2) \\ &\text{subject to:} \\ &\quad (x_1, x_2) \in F = X_1 \times X_2. \end{aligned} \tag{A4}$$

If an optimization problem is separable, then its solution can be found by solving pieces of the problem separately.

*Lemma 1: The solution to the problem.*

$$\begin{aligned} &\max_{x_1, x_2} \quad f_1(x_1) + f_2(x_2) \\ &\text{subject to:} \\ &\quad (x_1, x_2) \in X_1 \times X_2. \end{aligned} \tag{A5}$$

*is* $(x_1^*, x_2)$, *where* $x_i^*$ *solves*

$$\begin{aligned} &\max_{x_i} \quad f_i(x_i) \\ &\text{subject to:} \\ &\quad x_1 \in X_i. \end{aligned} \tag{A6}$$

*Proof.* Suppose not. Then there is some $(\hat{x}_1, \hat{x}_2)$ such that

$$f_1(x_1^*) + f_2(x_2^*) < f_1(\hat{x}_1) + f_2(\hat{x}_2). \tag{A7}$$

By definition, $f_1(x_1^*) \geq f_1(\hat{x}_1)$ and $f_2(x_2^*) \geq f_2(\hat{x}_2)$. Summing these two inequalities yields

$$f_1(x_1^*) + f_2(x_2^*) \geq f_1(\hat{x}_1) + f_2(\hat{x}_2). \tag{A8}$$

But this contradicts Eq. A7, and the lemma is proved.

*Theorem 1. The solution to the problem*

$$\begin{aligned} &\max_{x_1, \ldots, x_n} \quad \sum_{i=1}^{n} f_i(x_i) \\ &\text{subject to:} \\ &\quad (x_1, \ldots, x_n) \in \prod_{i=1}^{n} X_i, \end{aligned} \tag{A9}$$

*is* $(x_1^*, \ldots, x_n^*)$, *where* $x_i^*$ *solves*

$$\begin{aligned} &\max_{x_i} \quad f_i(x_i) \\ &\text{subject to:} \\ &\quad x_i \in X_i. \end{aligned} \tag{A10}$$

**Proof.** The proof is by induction. Let $S(k)$ be the proposition that the solution to the problem

$$\begin{aligned} &\max_{x_1, \ldots, x_k} \quad \sum_{i=1}^{k} f_i(x_i) \\ &\text{subject to:} \\ &\quad (x_1, \ldots, x_k) \in \prod_{i=1}^{k} X_i, \end{aligned} \tag{A11}$$

is given by $(x_1^*, \ldots, x_k^*)$. If $S(k)$ is true, then $S(k+1)$ is true because Lemma 1 can be applied to the problem

$$\begin{aligned} &\max_{(x_1, \ldots, x_k, x_{k+1})} \left( \sum_{i=1}^{k} f_i(x_i) \right) + f_{k+1}(x_{k+1}) \\ &\text{subject to:} \\ &\quad (x_1, \ldots, x_k, x_{k+1}) \in \prod_{i=1}^{k} X_i \times X_{k+1}. \end{aligned} \tag{A12}$$

Since $S(1)$ is true by definition, the theorem is proved.

### Optimizing relaxed problems

The following lemma is a key result from the optimization literature regarding the solution to relaxed problems. Since the proof is short and instructive, it is presented here for the reader's benefit. The power of this result is that it is true for any optimization problem regardless of its convexity, continuity, discreteness, and so on.

**Lemma 2 (Everett, 1963)**. *If* $x^*$ *maximizes*

$$f(x) - \sum_{j=1}^{m} p_j D_j(x), \tag{A13}$$

*then* $x^*$ *solves*

$$\begin{aligned} &\max_{x} \quad f(x) \\ &\text{subject to:} \\ &\quad D(x) \leq D(x^*) \end{aligned} \tag{A14}$$

**Proof.** Let $\hat{X}$ be the feasible set for the problem above: $\hat{X} = \{x: D(x) \le D(x^*)\}$. By the definition of $x^*$, for any $x \in X$

$$f(x^*) - \sum_{j=1}^{m} p_j D_j(x^*) \ge f(x) - \sum_{j=1}^{m} p_j D_j(x), \quad \text{(A15)}$$

or

$$f(x^*) \ge f(x) + \sum_{j=1}^{m} p_j \big(D_j(x^*) - D_j(x)\big). \quad \text{(A16)}$$

Because Eq. A16 is true for any $x \in X$ it is also true for $\hat{x} \in \hat{X} \subseteq X$. Since $D_j(x^*) - D_j(\hat{x}) > 0$ by definition of $\hat{X}$ and since $p_j > 0$ by supposition, the summation on the righthand side of Eq. A16 is positive. However, this implies that $f(x^*) \ge f(\hat{x})$ for every $\hat{x} \in \hat{X}$, and the lemma is proved.

The above result shows that solutions to the relaxed problem at a given set of prices $p$ also solve a variant of the original problem, a variant in which the resource demand is constrained to be less than $D(x^*)$. An immediate corollary of this result is that if $p$ is chosen such that $D(x^*) = S$, then $x^*$ solves the original problem Eq. A1.

**Corollary 1.** *If $x^*$ maximizes the problem*

$$f(x) - \sum_{j=1}^{m} p_j D_j(x), \quad \text{(A17)}$$

*such that*

$$D(x^*) = S, \quad \text{(A18)}$$

*then $x^*$ solves Eq. A1.*

**Proof.** By Lemma 2, $x^*$ solves

$$\max_{x_1, \ldots, x_n} \sum_{i=1}^{n} f_i(x_i) \quad \text{subject to:} \quad D(x) \le D(x^*) = S \quad \text{(A19)}$$

Since this is exactly Eq. A1, the corollary is proved.

While this is a useful result, the conditions are somewhat restrictive. These restrictions can be loosened so that $D(x^*)$ may be less than $S$ by introducing complementary slackness.

**Lemma 3.** *If $x^*$ maximizes the function*

$$f(x) - \sum_{j=1}^{m} p_j D_j(x), \quad \text{(A20)}$$

*such that*

$$D(x^*) \le S, \quad \text{(A21)}$$

*and*

$$\sum_{j=1}^{m} p_j \cdot \big(D_j(x^*) - S_j\big) = 0, \quad \text{(A22)}$$

*then $x^*$ solves Eq. A1.*

**Proof.** Let $A = \{j: D_j(x^*) = S_j\}$. That is, $A$ is the set of all indices associated with the constraints that hold with equality (or that are *active*) at $x^*$. Define a variation of Eq. A1 as follows

$$\sum_{i=1}^{n} f_i(x_i) \quad \text{subject to} \quad \sum_{i=1}^{n} d_i'(x_i) \le S', \quad \text{(A23)}$$

where $d_i': X_i \to \Pi_{j \in A} R_j$ such that $d_{i,j}'(x_i) = d_{i,j}(x_i)$ for $j \in A$; and $S' \in \Pi_{j \in A} R_j$ such that $S_j' = S_j$ for $j \in A$. In other words, this is just Eq. A1 without the constraints which are inactive at $x^*$.

Let $x'^*$ maximize the function

$$f(x) - \sum_{j \in A} p_j D_j(x). \quad \text{(A24)}$$

By the defintion of $A$,

$$\sum_{i=1}^{n} d_i'(x_i'^*) = S'. \quad \text{(A25)}$$

Then, by Corollary 1, $x'^*$ solves Eq. A1.

Note that for $j \in A$, $D_j(x'^*) = S_j$, and for $j \in \{1, \ldots, m\} - A$, $D_j(x'^*) < S_j$. Therefore,

$$\sum_{i=1}^{n} d_i(x_i'^*) \le S, \quad \text{(A26)}$$

which is the same as Eq. A21. Moreover, given Eq. A22, Eq. A24 is equivalent to Eq. A20. Therefore, $x'^*$ also maximizes Eq. A20 such that Eqs. A21 and A22 are true, and we can identify it as $x^*$.

Since the feasible set $\hat{X}'$ of Eq. A23 contains the feasible set $\hat{X}$ of Eq. A1 as a subset and since the objectives of Eq. A23 and Eq. A1 are equal, $f(x^*) \ge f(x)$ for every $x \in \hat{X}$. However, because $x^*$ satisfies Eq. A21 $x^* \in \hat{X}$. Therefore, $x^*$ solves Eq. A1.

Note that if $j \in A$, by definition, $D_j(x^*) = S_j$, and if $j \notin A$, then $D_j(x^*) < S_j$ and $p_j = 0$. In other words, $x^*$ solves the original problem if it solves

$$\begin{aligned} &\max_x \; f(x) - \sum_{j=1}^{m} p_j D_j(x) \\ &\text{subject to:} \\ &D_j(x) = S_j \qquad \text{for } j \in A \\ &D_j < S_j \qquad \text{for } j \notin A, \\ &p_j = 0 \qquad \text{for } j \notin A, \end{aligned} \quad \text{(A27)}$$

or equivalently, such that

$$p_j \cdot \big(D_j(x) - S_j\big) = 0 \qquad \text{for } j \in \{1, \ldots, m\} \quad \text{(A28)}$$

$$D_j(x) \le S \qquad \text{for } j \in \{1, \ldots, m\}. \quad \text{(A29)}$$

However, since this is equivalent to

$$\sum_{j=1}^{m} p_j \cdot \left( D_j(x) - S_j \right) = 0 \quad \text{(A30)}$$

$$D(x) \le S, \quad \text{(A31)}$$

the lemma is proved.

### *Proof of the main result*

Now we have enough theoretical support to prove our main result.

*Theorem 2. For each Division's i, define the following problem for a given set of prices* $\{p_1, \ldots, p_m\}$

$$\begin{aligned} &\max_{x_i} \quad f_i(x_i) - \sum_{j=1}^{m} p_j d_{i,j}(x_i) \\ &\text{subject to:} \\ &\quad x_i \in X_i \end{aligned} \quad (32)$$

*and denote its solution as* $x_i^*$

*If*

$$\sum_{i=1}^{n} d_i(x_i^*) \le S, \quad \text{(A33)}$$

*and*

$$\sum_{j=1}^{m} p_j \cdot \left( D_j(x^*) - S_j \right) = 0, \quad \text{(A34)}$$

*then* $(x_1^*, \ldots, x_n^*)$ *solves Eq. A1.*

**Proof**. By Theorem 1, the tuple $(x_1^*, \ldots, x_n^*)$ solves the following problem.

$$\max_{x_1, \ldots, x_n} \sum_{i=1}^{n} \left[ f_i(x_i) - \sum_{j=1}^{m} p_j d_{i,j}(x_i) \right], \quad \text{(A35)}$$

or

$$\max_{x} \quad f(x) - \sum_{j=1}^{m} p_j D_j(x). \quad \text{(A36)}$$

Because Eqs. A33 and A34 are true, $(x_1^*, \ldots, x_n^*)$ also solves Eq. A1 (by Lemma 3). The theorem is therefore proved.

*Manuscript received Dec. 17, 1998, and revision received Oct. 28, 1999.*