Quantity Competition in Multi-tier Supply Chain Networks

Tao Jiang* Young-San Lin † Thành Nguyen‡

November 14, 2018

Abstract

We consider a model of multi-stage competition over multi-tier supply chain networks and show that in particular, for parallel serial networks, there is a unique equilibrium. We provide a polynomial time algorithm to compute the equilibrium and study the impact of the network structure to the total trade flow at equilibrium. Our results shed light on the trade-off between competition, production cost and double marginalization.

1 Introduction

Supply chain networks in practice are multi-tier and heterogeneous. A firm's decision influences not only other firms within the same tier but also across. The literature on game theoretical models of supply chain networks, however, has largely focused on two extreme cases: heterogeneous 2-tier networks (bipartite graph) Kranton and Minehart [2001], Bimpikis et al. [2014] and a linear chain of n-tier firms Wright and Wong [2014], Nguyen [2017]. One main reason for this is that most models of sequential decision making in multi-tier supply chain networks are intractable. Sequential decision making is a well observed phenomenon in supply chains because firms at the top tier typically need to make decisions on the quantity to sell to firms in the next tier and the buying firms then decide how much to buy from which suppliers, and continue to pass on the goods by determining the quantity for firms at the next level.

To study such models, one needs to analyze subgame perfect equilibria in which a firm needs to internalize all the decision of all the firms downstream and compete with all the firms

*Krannert School of Management, Purdue University, jiang282@purdue.edu
†Department of Computer Science, Purdue University, nilnamuh@gmail.com
‡Krannert School of Management, Purdue University, nguye161@purdue.edu

of the same tier at the same time. Another factor that further complicates models of general supply chain networks is that even the basic concept of tiers is ambiguous because there are often multiple routes of different length that goods are traded from the original producers to the consumers. Our paper studies a model of sequential network game motivated by supply chain network applications. Our main goal is to understand the effect of network structure on the efficiency of the system.

When considering the efficiency of a supply chain network, there is a trade-off between the length and the number of trading routes. On the one hand, a large variety of options to trade indicates a high degree of competition, which leads to a more efficient system. On the other hand, a long trading path causes double, triple and higher degree marginalization problems. In this paper, to capture these ideas, we consider a sequential game theoretical model for a special class of networks: series-parallel graphs. We focus our analysis on these networks because they are rich enough for studying the trade-off described above and simple enough for characterizing the equilibrium outcomes. In particular, series-parallel networks have a natural decomposition of parallel and serial insertions. A parallel insertion, which merges two different sub-networks at the source and the sink, can capture the increase in competition. A serial insertion, which attaches two sub-networks sequentially, corresponds to the increase in the length of trading paths.

Our first contribution is a result showing that the equilibrium is unique in these networks. Furthermore, we provide a polynomial time algorithm to compute the equilibrium. Our algorithm is nontrivial and combines a dynamic program capturing the backward induction of an equilibrium computation and a convex quadratic programming technique for computing the flow and price functions.

Our second contribution is a set of comparative analysis on the influence of the network structure and the two operations in series-parallel graphs to the total flow at equilibrium. For example, we show that:

- Parallel insertion increases total flow, while serial insertion decreases total flow.
- Given two networks N_1 and N_2 the order of serial insertion to obtain N_1N_2 or N_2N_1 network matters only when the production cost of at least one component is positive. The total flow is larger if the component with higher production cost is closer to the source.
- In parallel insertion, adding a component to a longer range increases the flow more than adding it to a shorter range. This means increasing competition globally is more beneficial than increasing competition locally.

- An upstream firm that controls all the flow of goods of another downstream firm has a location advantage. The utility of this upstream firm is at least twice as much as the dominated downstream firm.

Finally, we show that extending the series parallel graph to a slightly more general class of network, series parallel graphs with multiple producers or markets, the problem may become intractable. With multiple producers and single market our technique extends to construct the unique equilibrium of the game. However, with multiple producers and markets, there may exist multiple or no pure strategy equilibria.

The paper is organized as follows. In section 2 we introduce the model of competition and parallel serial networks together with the composition. In section 3 we provide the algorithm to compute the unique equilibrium. Section 4 uses the network composition and the algorithm to analyze comparative analysis on how network structure influences the efficiency measured by the total trade flow. Section 5 discusses extensions to other classes of networks and shows that pure equilibrium might not exist in general networks.

Related work: In our paper, we assume the consuming nodes are Cournot markets. Thus, the structure of the game is closely related to the literature on Cournot games in networks. Bimpikis et al. [2014], Pang et al. [2017], for example, consider a Cournot game in two sided markets. Kannan and Nguyen [2018] study Cournot game in three-tier networks. However, the 2-tier structure of the network in these papers, and the assumption that only the middle tier make decision in Kannan and Nguyen [2018] assumes away the complex sequential decision making considered in our paper.

Nava [2015] studies a Cournot game in general networks, however, firms are assumed to make simultaneous decisions. As discussed above, simultaneous games are easier to analyze, but do not capture the essential element of sequential decision making of firms in supply chain networks.

Carr and Karmarkar [2005] considers assembly network where agents make sequential decision, but assumes a tree network. The analysis for a tree network is substantially simpler, because each firm has a single downstream node that it can sell the products to. In our game, the network is more general, each firm needs to make decision of the goods quantity to each firm that it is connected to. As we show, some of the quantities on some of the links can be zero. Such "inactive" links make the analysis more complicated.

Recently, Bimpikis et al. [2017] also considered a sequential game and use market clearing prices like our paper. The network considered in this paper is however symmetric and its structure is linear. The focus of Bimpikis et al. [2017] is on the uncertainty of yields, which is different from the motivation in our paper.

More broadly, our paper belongs to the growing literature of network games and their applications in supply chains, includingCorbett and Karmarkar [2001], Federgruen and Hu [2016], Nguyen et al. [2016], Nguyen [2017]. These papers, however, are different from ours in the main focus as well as the modeling approach. Corbett and Karmarkar [2001] for example, assume a linear structure of supply chains, Federgruen and Hu [2016] consider price competition in two-tier networks, and Nguyen et al. [2016], Nguyen [2017] analyze bargaining games in networks with simpler structures. The main contribution of our paper to this line of work is a tractable analysis of sequential competition model in series parallel graphs, which allows for a richer comparative analysis and deeper understanding of how basic network elements influence market outcomes.

2 Model

In this section, we introduce the sequential decision mechanism in a supply chain network and the definition of series parallel graph.

2.1 Sequential Decisional Game

Let $G = (V \cup \{s, t\}, E)$ be a simple directed acyclic network that represents an economy where s is the producer at the source, t is the sink market and V represents intermediary firms. The edges of G represent the possibility of trade between two agents. The direction of an edge indicates the direction of trade. The outgoing end of the edge corresponds to the seller, and the incoming end is the buyer, while s has only outgoing edges, and t has only incoming edges. The remaining vertices $i \in V$ representing intermediary firms has both incoming and outgoing edges. For a vertex i, $B(i)$ (buyer set) and $S(i)$ (seller set) are the sets of agents that can be buyers and sellers in a trade with i, respectively.

Assume every agent has full information about the structure of the network. Agents start deciding their order quantities and selling quantities after the output of their upstream suppliers is determined. Furthermore, the *market clearance price* at i is such that the total demand from i matches the total supply.

Each intermediate firm i decides on how much to buy from each of his sellers and how much to sell to each of his buyers. Specifically, i's decision includes:

- The buying quantity $x_{ki}^{in} \geqslant 0$ for every $k \in S(i)$;
- The selling quantity $x_{ij}^{out} \geqslant 0$ to every $j \in B(i)$.

while the source only initializes the supplying amount and the sink will take all the goods at the market price. Fig. 1 shows an example of decisions in the supply chain:

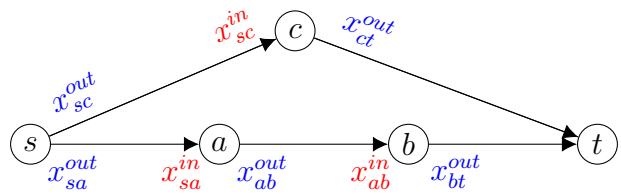


Figure 1: Decisions in a Supply Chain

For producer s, his unit cost of production p_s is given and assumed to be an affine function on X_s, the total amount of goods to sell.

$$p_s = a_s + d_s X_s, \text{ where } X_s = \sum_{i \in B(s)} x^{out}_{sj}, d_s \geqslant 0 \text{ and } a_s \geqslant 0.$$

Sink node t does not represent a firm, it corresponds to an end market. The price function p_t at sink node t is given and assumed to be an affine function on the total amount of goods, X_t, sold to market t.

$$p_t = a_t - b_t X_t, \text{ where } X_t = \sum_{i \in S(t)} x^{in}_{it} = \sum_{i \in S(t)} x^{out}_{it}, a_t > 0, \text{ and } b_t > 0.$$

Note that the market must accept all the goods thus does not have a choice to reject. That is, $x^{in}_{it} = x^{out}_{it}$ for each $i \in S(t)$. Generally, for a trade $ij \in E$, the buyer j cannot obtain more than what the seller i offers, thus $x^{in}_{ij} \leqslant x^{out}_{ij}$. We assume that each intermediary firm i cannot get goods from any other source besides his sellers. Therefore the outflow of i cannot be more than the inflow of i,

$$\sum_{j \in B(i)} x^{out}_{ij} \leqslant \sum_{k \in S(i)} x^{in}_{ki}.$$

The price for intermediate goods for each node $i \in V$, denoted as p_i, is determined endogenously such that the corresponding intermediate market at i clears.

Furthermore, agents do not get any value from retaining the goods. They incur a processing cost, which we assume to be quadratic in the quantity of goods the agents sells.

The payoff of the source firm s is

$$\Pi_s = \sum_{j \in B(s)} p_j x^{in}_{sj} - p_s \sum_{j \in B(s)} x^{out}_{sj} - \frac{c_s}{2}\Big(\sum_{j \in B(s)} x^{out}_{sj}\Big)^2 \text{ where } c_s \geqslant 0. \tag{1}$$

The utility of an intermediate agent $i \in V$ is

$$\Pi_i = \sum_{j \in B(i)} p_j x_{ij}^{in} - p_i \sum_{k \in S(i)} x_{ki}^{out} - \frac{c_i}{2}\Big(\sum_{k \in S(i)} x_{ki}^{out}\Big)^2 \text{ where } c_i \geqslant 0. \tag{2}$$

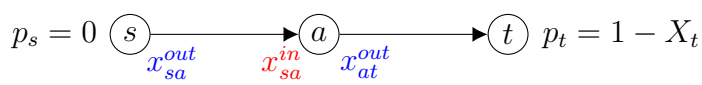
The formula decomposes the utility function into three terms: the total revenue from $j \in B(i)$, the total cost of materials from $k \in S(i)$, and the processing cost.

The timing of the game is as follows. The producer (source) makes its decision first. A firm makes his decision on the selling quantity to his downstream, once all of his sellers have made decisions.[1] When choosing their order quantities to maximize their expected profits, firm i also needs to take into account the strategies of both the competing firms and the firms downstream. When a firm makes its decision, it only knows the quantities offered by the firms upstream and makes prediction based on rational expectation of other firms' strategies.

Here is a toy example for the equilibrium at a supply chain:

Example 1

Assume no processing cost in this example.



Suppose source s makes an decision to sell $x_{sa}^{out} = x$ amount of goods to agent a. Now for agent a, since he has no benefit from unsold goods, his buying amount will be equal to the selling amount, denoted as $x_a = x_{sa}^{in} = x_{at}^{out}$. Meanwhile, the utility function is

$$\pi_a(x_a) = (1 - x_a)x_a - p_a x_a$$

where p_a is the market clearance price. And the optimal decision for a ($\frac{\partial \pi_a}{\partial x_a} = 0$) is

$$x_a = \frac{1 - p_a}{2}$$

By the definition of the market clearance price, we have $x_a = x_{sa}^{out}$. Thus, the relation between selling amount and market clearance price at agent a is

$$p_a = 1 - 2x_{sa}^{out}$$

[1] By the market clearance price, at the firm's optimal decision, it will consume all the supply from upstream.

Now consider the utility function of the source,

$$\pi_s = p_a x_{sa}^{out} = (1 - 2x_{sa}^{out})x_{sa}^{out}$$

Finally, we have the optimal supplying amount at the source $x_{sa}^{out} = 1/4$*, which will result in a market clearance price* $p_a = 1/2$*, and processing amount through* a *is also* $1/4$*. Note that this is the unique equilibrium flow in this toy supply chain.*

2.2 Series Parallel Graph

In this paper, we consider the case when G is a *Series Parallel Graph* (SPG). This class of networks is well studied and has several applications in graph theory. (See for example Duffin [1965]). For completeness, we provide a formal definition as follows.

Definition 2.1 (SPG) *A* single-source-and-sink SPG *is a graph that may be constructed by a sequence of series and parallel compositions starting from a set of copies of a single-edge graph, where:*

1. Series composition *of* X *and* Y*: given two SPGs* X *with source* s_X *and sink* t_X*, and* Y *with source* s_Y *and sink* t_Y*, form a new graph* $G = S(X, Y)$ *by identifying* $s = s_X$*,* $t_X = s_Y$*, and* $t = t_Y$*.*

2. Parallel composition *of* X *and* Y*: given two SPGs* X *with source* s_X *and sink* t_X*, and* Y *with source* s_Y *and sink* t_Y*, form a new graph* $G = P(X, Y)$ *by identifying* $s = s_X = s_Y$ *and* $t = t_X = t_Y$*.*

3 Equilibrium Characteristics and Computation

In this section, before describing how an equilibrium can be computed, we observe some properties of an equilibrium and series parallel graphs.

3.1 Properties of Equilibrium

First observe that the best strategy for agent i is to always sell as much as bought since it cannot benefit from paying more for those unsold goods. At the selling side, suppose firm i is willing to offer x_{ij}^{out} quantity of goods to firm j, but part of the goods got rejected, i.e. $x_{ij}^{in} < x_{ij}^{out}$. However, this can never happen in an equilibrium, because i will be better off by rejecting $x_{ij}^{out} - x_{ij}^{in}$ amount of goods from its upstream at the beginning.

The next proposition lists the properties of supplying quantities at an equilibrium:

Proposition 3.1 *With market clearance price, where p_s and p_t are given, each agent $i \in V$ gets to decide x_{ij}^{out} where $j \in B(i)$ and x_{ki}^{in} where $k \in S(i)$, and s gets to decide x_{sj}^{out} for $j \in B(s)$. The equilibrium satisfies:*

1. *$x_{ij}^{out} = x_{ij}^{in}$ for $ij \in E$.*
2. *$\sum_{k \in S(i)} x_{ki}^{in} = \sum_{j \in B(i)} x_{ij}^{out}$, i.e. inflow is equal to outflow for agent $i \in V$.*

For later notations, at the equilibrium, we will set x_{ij} as the flow along the edge ij, i.e. $x_{ij} = x_{ij}^{out} = x_{ij}^{in}$, and no longer use x_{ij}^{in} and x_{ij}^{out}. Meanwhile, since each firm accepts all the offers and sells everything they bought, we denote this sum of flow as processing quantity for firm i, i.e. $X_i = \sum_{k \in S(i)} x_{ki} = \sum_{j \in B(i)} x_{ij}$. For market t, the price is given as $p_t = a_t - b_t X_t$ because t always accepts everything. For example, at equilibrium, the flows of the supply chain in Fig. 1 is

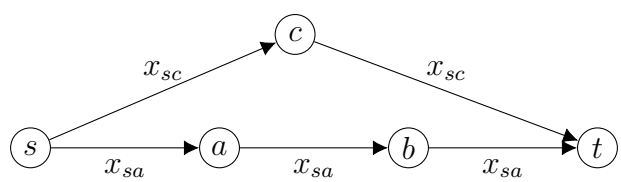


Figure 2: Flows in a Supply Chain at Equilibrium

With the above new notations, by rewriting equation 2, the utility of agent i becomes

$$\Pi_i = \sum_{j \in B(i)} p_j x_{ij} - p_i \sum_{j \in B(i)} x_{ij} - \frac{c_i}{2} (\sum_{j \in B(i)} x_{ij})^2. \tag{3}$$

and by rewriting equation 1, the utility of source firm s becomes

$$\Pi_s = \sum_{j \in B(s)} p_j x_{sj} - p_s \sum_{j \in B(s)} x_{sj} - \frac{c_s}{2} (\sum_{j \in B(s)} x_{sj})^2. \tag{4}$$

For the flow activities along each edge, we define an edge $ij \in E$ is *active* if $x_{ij} > 0$, and *inactive* if $x_{ij} = 0$. Note that for every agent, the buying price should be at most the selling price so that the agent can obtain non-negative utility, thus whenever ij is active, $p_i \leqslant p_j$. Otherwise, i could have been better off by rejecting some goods from upstream and choose not to offer any goods to j.

Proposition 3.2 *For each $ij \in E$ that is active, the market clearance price at an equilibrium satisfies $p_i \leqslant p_j$.*

3.2 Properties of Series Parallel Graphs

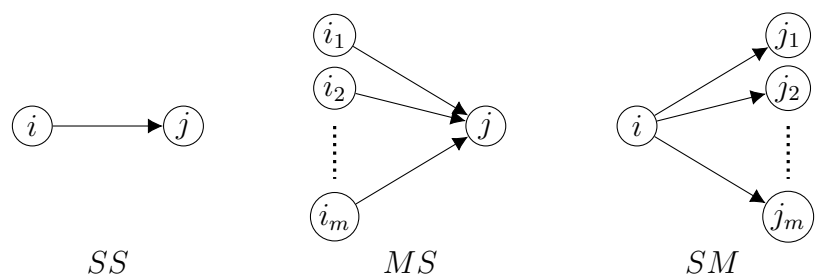
Consider a path $l_{ij} = (i, v_1, ..., v_k, j)$ from i to j. If there is an edge $ij \in E$, then we say ij is a *shortcut* of l_{ij}. The intuition is i always prefers selling to j directly than through the intermediate agents along the path l_{ij}, and we prove it in the following proposition. The proof is provided in Appendix A.1.

PROPOSITION 3.3 *At an equilibrium of a series parallel graph G, if $ij \in E$ is a shortcut of a path l_{ij}, then there is no trade on l_{ij}. Thus, all the edges on the path l_{ij} are inactive.*

By this observation, without loss of generality, we can assume that G does not have any shortcuts.

Here we introduce the node relations in SPG. Node k is called parent node of i if there is a directed path from k to i. The set of parent nodes of i is denoted as $P(i)$. By a similar idea, we can define child node and set of children $C(i)$. If consider the relation between direct parent and child $i \to j$, i.e. $ij \in E$, there are three possibilities in SPG:

- Single seller and single buyer, $|S(j)| = |B(i)| = 1$. (SS)
- Multiple sellers and single buyer, $|S(j)| \geqslant 2, |B(i)| = 1$. (MS)
- Single seller and multiple buyers, $|S(j)| = 1, |B(i)| \geqslant 2$. (SM)



Sometimes there are multiple paths from a parent node to one of its children, we call these paths *disjoint* if they do not have any common intermediary nodes, that is, all nodes except the starting and the ending ones are different. Base on this definition, we can define the merging nodes with respect to node i.

DEFINITION 3.1 (SELF-MERGING CHILD NODE) *Node $j \in C(i)$ is a* self-merging child node *of i if there are disjoint paths from i to j. The set of such nodes j is denoted as $C_S(i)$.*

Definition 3.2 (Parent-merging Child Node) *Node $j \in C(i)$ is a* parent-merging child node *of i, if there exist node $k \in P(i)$, such that there are disjoint paths from k to j. The set of such nodes j is denoted as $C_P(i)$.*

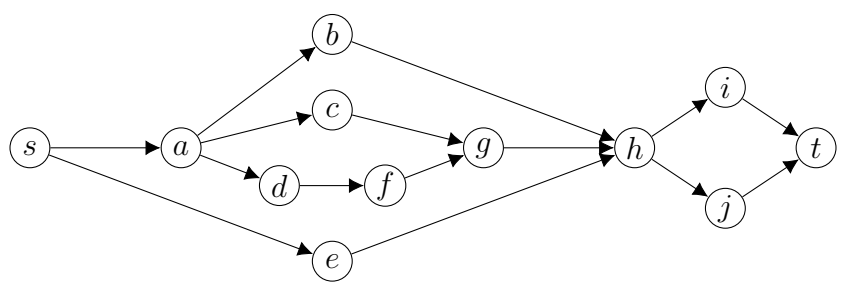
We also introduce the special self-merging child nodes of i and its child j as $C_T(i,j) = C_S(i) \cap C(j) \backslash C_P(i)$. This notation is useful because it helps us capture the "internal" merging nodes that are responsible for the price of i and flow to j later on.

Proposition 3.4 *A series parallel graph has the following properties:*

1. $C_P(s) = C_P(t) = \emptyset$.
2. *In SS case, for $ij \in E$, $C_P(j) = C_P(i)$.*
3. *In SM case, for $ij \in E$, $C_P(j) = C_P(i) \sqcup C_T(i,j)$.*
4. *In MS case, for $ij \in E$, $C_P(i) = C_P(j) \sqcup \{j\}$.*

Note that $\sqcup$ stands for disjoint set union.

Example 2



In this graph, for node a, $C_S(a) = \{g,h\}$, because $\{g,h\} \subset C(a)$ and there are multiple disjoint paths from a to g and h, while $t \notin C_S(a)$ because all the paths from a to t must go through the common node h which are not disjoint paths; $C_P(a) = \{h\}$ because $h \in C(a)$, $s \in P(a)$, and there are multiple disjoint paths from s to h; $C_T(a,b) = \emptyset$, while $C_T(a,c) = \{g\}$.

For node c, $C_P(c) = \{g,h\}$, while $C_S(c) = \emptyset$; For node g, $C_P(g) = \{h\}$, while $C_S(g) = \emptyset$.

Since $a \to c$ is the SM relation, by Proposition 3.4, $C_P(c) = \{g,h\} = C_P(a) \sqcup C_T(a,c)$. Also, $C_P(c) = \{g,h\} = C_P(g) \sqcup \{g\}$, because $c \to g$ belongs to the MS relation.

3.3 Equilibrium Computation

In this section, we present an algorithm to compute the equilibrium supplying quantities at every edge. To do that, we first derive a closed-form expression for the market clearance price at each firm through a backward algorithm in section 3.3.1. Then, the unique optimal quantities for each firm can be solved following the decision sequence from source to sink as in section 3.3.2.

3.3.1 Market Clearance Price Computation.

A key characteristic of the equilibrium is that all edges are active. The market clearance prices have closed-form expressions and quantities can be computed based on the composition of SPG.

LEMMA 3.1 *At equilibrium, if $a_s < a_t$, then all the edges are active. Also the market clearance price at agent i is an inverse linear function of X_i and flows to its parent-merging children nodes.*

$$p_i = a_t - b_i X_i - \sum_{k \in C_P(i)} b_k X_k$$

where $b_i > 0, \forall i \in V$ is a constant that only depends on the structure of G and processing cost.

The above lemma shows a concise way to present the price function at equilibrium. The last piece of work for price function computation is to find the value of b_i for $i \in V$. This is provided in the proof given in Appendix A.2. By adapting the main equations in that proof, here we introduce a backward Algorithm 1 to compute the market clearance price at equilibrium (starting from $ALG_1(j = t, G)$).

Algorithm 1 : Price Function Computation (Backward)

1: Given the downstream buyer j's clearance price function p_j, compute the upstream seller i's clearance price case by case:

- Single seller and single buyer case,

$$b_i = 2b_j + \sum_{k \in C_P(j)} b_k + c_i. \tag{SS}$$

- Multiple sellers and single buyer case, for each seller,

$$b_i = b_j + \sum_{k \in C_P(j)} b_k + c_i. \tag{MS}$$

- Single seller and multiple buyers case ($|C_S(i)| = 1$) [2],

$$b_i = \frac{2}{\sum_{j \in B(i)} \frac{1}{b_j}} + 2b_h + \sum_{k \in C_P(j) \setminus \{h\}} b_k + c_i. \tag{SM}$$

2: Set the price function at seller i: $p_i = a_t - b_i X_i - \sum_{k \in C_P(i)} b_k X_k$.

3: **if** seller i is the source **then**

4: Return.

5: **else**

6: Run $ALG_1(j = i, G)$.

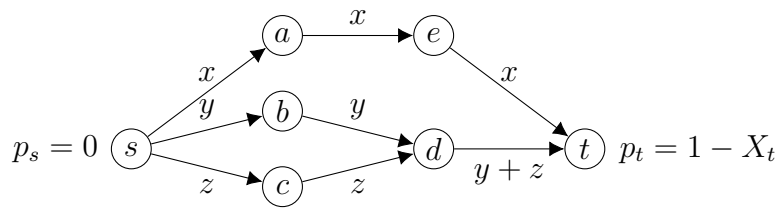
In each iteration, we just compute b_i and this can be done in $O(deg^+(i))$ time where $deg^+(i)$ is the outdegree of i. Besides, we also store the convex coefficients of each downstream node $j \in B(i)$. The number of b_i computation is bounded by $O(|V|)$. Therefore, it takes linear time to compute the price functions by Algorithm 1.

Below is an example about the price function computation, for the general form expression as in Algorithm 1, please check Example 14.

Example 3 (Price Function Computation)

Assume no processing cost in this example.

[2] If $|C_S(i)| \geqslant 2$, the computation of b_i is more complicated, the detail is provided in Appendix A.2.



From Proposition 3.1, we know that inflow must equal to outflow for each firm at equilibrium. Therefore, we can set $x_{sa} = x_{ae} = x_{et} = x$, $x_{sb} = x_{bd} = y$, $x_{sc} = x_{cd} = z$, *and* $x_{dt} = y + z$.

Consider the utility of e,

$$\Pi_e(x) = p_t x - p_e x = (1 - x - y - z)x - p_e x.$$

Market clearance price function of e *can be derived by solving the stable condition of the utility maximization problem:*

$$\frac{\partial \Pi_e(x)}{\partial x_{et}} = 1 - 2x - y - z - p_e = 0 \Rightarrow p_e = 1 - 2x - y - z.$$

Similarly, we can obtain the following price functions:

$$\begin{aligned} p_a &= 1 - 4x - y - z, \\ p_d &= 1 - x - 2y - 2z, \\ p_b &= 1 - x - 4y - 2z, \\ p_c &= 1 - x - 2y - 4z. \end{aligned}$$

Note that the above price functions can be written as the form of

$$p_i = a_t - b_i X_i - \sum_{k \in C_P(i)} b_k X_k.$$

As in Lemma 3.1, for example,

$$p_b = 1 - x - 4y - 2z = 1 - b_b X_b - b_d X_d - b_t X_t.$$

where $b_b = 2, b_d = b_t = 1, C_P(b) = \{d, t\}$.

The utility of s *is*

$$\Pi_s(x, y, z) = p_a x + p_b y + p_c z - p_s(x + y + z).$$

Let p_{s_a} be the price function that has to be satisfied if $\frac{\partial \Pi_s(x,y,z)}{\partial x} = 0$, where p_{s_b} and p_{s_c} are defined similarly. Hence, the following stable condition is obtained:

$$\begin{aligned}
\frac{\partial \Pi_s(x,y,z)}{\partial x} &= 0 \Rightarrow p_{s_a} = 1 - 8x - 2y - 2z, \\
\frac{\partial \Pi_s(x,y,z)}{\partial y} &= 0 \Rightarrow p_{s_b} = 1 - 2x - 8y - 4z, \\
\frac{\partial \Pi_s(x,y,z)}{\partial z} &= 0 \Rightarrow p_{s_c} = 1 - 2x - 4y - 8z.
\end{aligned}$$

Note that all above three equations are necessary conditions for p_s, by using the convex coefficients $\mu_1 = \frac{2}{5}, \mu_2 = \mu_3 = \frac{3}{10}$, we write p_s as function of total flow $X_s = x + y + z$,

$$\begin{aligned}
p_{s_{abc}} &= \mu_1 p_{s_a} + \mu_2 p_{s_b} + \mu_3 p_{s_c} \\
&= 1 - \frac{22}{5}(x + y + z) \\
&= 1 - \frac{22}{5} X_s
\end{aligned}$$

Till here, we have the equilibrium price function at every node. Furthermore, we can find the total flow at equilibrium X_s at source by solving

$$p_{s_{abc}} = p_s = 0 \Rightarrow X_s = \frac{5}{22}.$$

Base on the closed-form relation between seller and buyer as in SS, MS, and SM, we can prove a stronger version of Proposition 3.2. The proof can be found in Appendix A.3.

PROPOSITION 3.5 *If an edge is active in an SPG, then the price at corresponding seller is strictly less than the price at buyer.*

3.3.2 Equilibrium Quantities Computation.

After having the closed-form of the market clearance price function, we present an algorithm that finds the unique supply quantities at equilibrium. Consider the quantities decision for firm i to its downstream buyers $j \in B(i)$. Suppose there is only a single outflow for firm i, i.e., $|B(i)| = 1$, by Proposition 3.1, inflow equals outflow at firm i, and firm j will take all the supplying quantities from i, formally, $x_{ij} = X_i$. Hence, in the following analysis, we focus on the nontrivial case when firm has multiple downstream buyers, i.e., $|B(i)| \geqslant 2$. How to optimally allocate the supplying quantities to different buyers? In particular, firm

i's decision x_{ij}, where $j \in B(i)$, is to optimize its utility Π_i. Recall the utility equation 3:

$$\Pi_i = \sum_{j \in B(i)} p_j x_{ij} - p_i \sum_{j \in B(i)} x_{ij} - \frac{c_i}{2}(\sum_{j \in B(i)} x_{ij})^2.$$

Note that before firm i makes decision, p_i is determined by upstream flows, but p_j may be affected by x_{ij} where $j \in B(i)$. By Lemma 3.1, we can write the price function of seller j as

$$p_j = a_t - b_j x_{ij} - \sum_{k \in C_P(j)} b_k X_k. \tag{5}$$

Note that by the property of SPG, $|S(j)| = 1$ when $B(i) \geqslant 2$. Thus, $X_j = x_{ij}$.

To find the optimal supply quantities to downstream firm j, take the derivative of the utility function with respect to x_{ij}, and obtain

$$\frac{\partial \Pi_i}{\partial x_{ij}} = p_j - \sum_{l \in B(i)} \frac{\partial p_l}{\partial x_{ij}} x_{il} - p_i - c_i X_i. \tag{6}$$

Expand the second term of equation 6 as

$$\begin{aligned} \sum_{l \in B(i)} \frac{\partial p_l}{\partial x_{ij}} x_{il} &= b_j x_{ij} + \sum_{l \in B(i)} (\frac{\partial \sum_{k \in C_P(l)} b_k X_k}{\partial x_{ij}}) x_{il} \\ &= b_j x_{ij} + \sum_{l \in B(i)} (\sum_{k \in C_P(l) \cap C(j)} b_k) x_{il} \\ &= b_j x_{ij} + \sum_{h \in C_T(i,j)} b_h X_h + \sum_{k \in C_P(i)} b_k X_i. \end{aligned} \tag{7}$$

Plug equation 5 and equation 7 back into equation 6, we get

$$\begin{aligned} \frac{\partial \Pi_i}{\partial x_{ij}} &= a_t - 2b_j x_{ij} - \sum_{k \in C_P(j)} b_k X_k - \sum_{h \in C_T(i,j)} b_h X_h - \sum_{k \in C_P(i)} b_k X_i - c_i X_i - p_i \\ &= a_t - 2b_j x_{ij} - 2 \sum_{h \in C_T(i,j)} b_h X_h - p_i - const. \end{aligned} \tag{8}$$

Note that X_i and X_k where $k \in C_P(i)$ are given constant predetermined by upstream supply. By point 3 of Proposition 3.4, $C_P(j) = C_P(i) \sqcup C_T(i,j)$ and we have

$$const = (\sum_{k \in C_P(i)} b_k + c_i) X_i + \sum_{k \in C_P(i)} b_k X_k.$$

Observe the utility of firm i (equation 3) is concave. At the equilibrium, if $x_{ij} > 0$, then $\frac{\partial \Pi_i}{\partial x_{ij}} = 0$; if $x_{ij} = 0$, then $\frac{\partial \Pi_i}{\partial x_{ij}} \leqslant 0$. This problem is equivalent to the following linear complementary problem (LCP) with variables x_{ij} where $j \in B(i)$.

$$\begin{cases} \frac{\partial \Pi_i}{\partial x_{ij}} x_{ij} = 0, \\ \frac{\partial \Pi_i}{\partial x_{ij}} \leqslant 0, \\ x_{ij} \geqslant 0, \quad \forall j \in B(i). \end{cases} \tag{LCP}$$

To solve the above system of equations LCP, we introduce a convex quadratic program:

$$\begin{aligned} \min_{x_{ij}, X_k} \quad & \sum_{j \in B(i)} b_j x_{ij}^2 + \sum_{k \in C_S(i) \setminus C_P(i)} b_k X_k^2 \\ \text{subject to} \quad & a_t - 2b_j x_{ij} - \sum_{k \in C_T(i,j)} 2b_k X_k - const \leqslant p_s \quad \text{for } j \in B(i), \\ & x_{ij} \geqslant 0 \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{for } j \in B(i). \end{aligned} \tag{CQP}$$

By examining the KKT conditions of the quadratic program, the independent variables X_k satisfy $X_k = \sum_{j:k \in C(j)} x_{ij}$, which fits the definition of X_k. Besides, equation LCP also holds. The proof of Lemma 3.2 is provided in Appendix A.4.

Lemma 3.2 *Problem LCP is equivalent to the convex optimization problem CQP, and the solution is unique.*

After the market clearance price function is computed by Algorithm 1, by solving CQP directly, we have the optimal decision of each firm in polynomial time. In fact, the algorithm can be sped up by distributing the flow from i to $j \in B(i)$ proportionally to the convex coefficients pre-computed in Algorithm 1. Besides, all the p_j's have the same price value so that i has no preference about whom to sell to. The proof of Lemma 3.3 is provided in Appendix A.5.

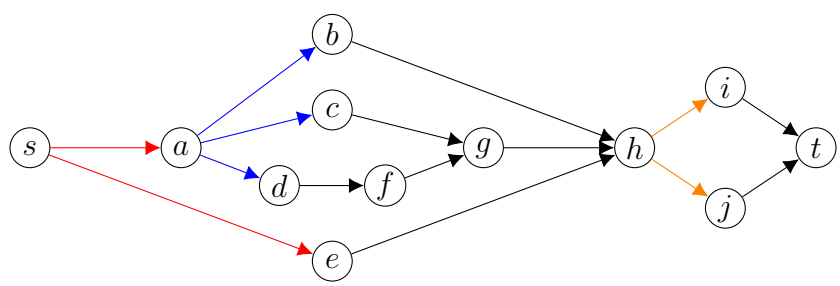
Lemma 3.3 *For the SM case, Π_i is maximized by distributing the flow to $j \in B(i)$ proportionally to the convex coefficients pre-computed in Algorithm 1. Besides, all the p_j's have the same price value.*

Algorithm 2 : SPG Flow Computation (Forward)

(Initialize $X_j = 0, \forall j \in V$. Start with $Alg_2(i = s, p_i = p_s, G)$.)
Distribute the flow x_{ij} where $j \in B(i)$ proportionally to the convex coefficients.
for $k \in C_S(i)$ **do**
    if $X_k = 0$ **then**
        $X_k = \sum_{j:k \in C_P(j)} x_{ij}$.
for $j \in B(i)$ **do**
    if $X_j = 0$ **then**
        $X_j = x_{ij}$.
    $p_j = a_t - b_i X_j - \sum_{k \in C_P(j)} b_k X_k$.
    Run $Alg_2(j, p_j, G)$.
Return.

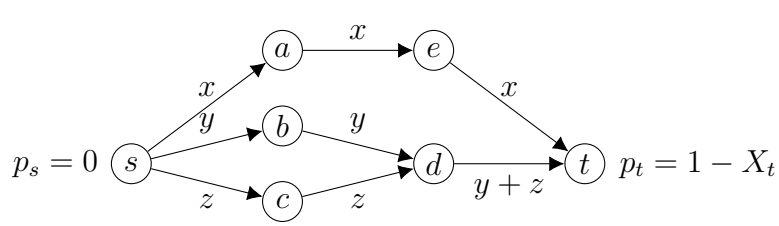
The algorithm starts with solving the equilibrium flow at source, then based on the flow decision, each $j \in B(i)$ is considered as the new source node, and their equilibrium flow decisions were solved along the path to the sink, as demonstrated in the following examples.

Example 4 (Flow Computation Order)
Consider the same instance as Example 2:



Algorithm 2 solves the flow quantities along the red edges first, then those along the blue and orange edges. Note that the flow along the black edge is equal to the total inflow to the upstream firm by the definition of market clearance price (e.g. $x_{gh} = X_g = x_{cg} + x_{fg}$).

Example 5 (Flow Computation)
Consider the same instance as Example 3:



We already have $X_s = \frac{5}{22}$. By distributing the flow proportionally to the convex coefficients $\mu_1 = \frac{2}{5}, \mu_2 = \mu_3 = \frac{3}{10}$, we have $x = \mu_1 X_s = \frac{1}{11}$ and $y = z = \mu_2 X_s = \frac{3}{44}$.

We calculate the price values p_a, p_b, and p_c from the flow values x, y, and z:

$$
\begin{aligned}
p_a &= 1 - 4x - y - z = 1 - 4 \times \frac{1}{11} - \frac{3}{44} - \frac{3}{44} = \frac{1}{2}, \\
p_b &= 1 - x - 4y - 2z = 1 - \frac{1}{11} - 4 \times \frac{3}{44} - 2 \times \frac{3}{44} = \frac{1}{2}, \\
p_c &= 1 - x - 2y - 4z = 1 - \frac{1}{11} - 2 \times \frac{3}{44} - 4 \times \frac{3}{44} = \frac{1}{2}.
\end{aligned}
$$

THEOREM 3.1 *For SPG, there exists a linear time algorithm to solve the equilibrium flow and prices, and the equilibrium is unique.*

Proof. The equilibrium flow and prices can be found by Algorithm 1 and Algorithm 2 in linear time as aforementioned. The uniqueness of equilibrium can be proved by encoding this problem into LCP and its corresponding CQP has a unique solution. ■

4 Structural Analysis of Network Pricing Equilibria

In this section, we compare the equilibria and analyze the influence of different operations on SPG, e.g. switching the order of two components in SPG, or inserting a new component to a given SPG. The criteria of the influence is the network efficiency defined as follows:

DEFINITION 4.1 (EFFICIENCY) *A supply chain network is more efficient if it has a larger total flow value at equilibrium.*

Following are some general results for SPG. The first proposition shows that the direct selling from source to sink is the most efficient supplying network,

PROPOSITION 4.1 *Singe-edge graph is the most efficient SPG supplying network.*

For single-edge graph, let p_s^0 be the source price, then $p_s^0 = a_t - (2b_t + c_s)X_s$. For general SPG, by induction, we show that the market clearance price for every firm is higher than p_s^0. The induction step is similar to the proof of Lemma 3.1, and the proof details can be found in Appendix B.1.

Interpret a_t as the *demand* of the market, the following proposition shows the relation between demand and efficiency.

PROPOSITION 4.2 *The market efficiency increases if the demand at the market increases or material cost at the source decreases.*

Proof. From Lemma 3.1:

$$p_s = a_t - b_s X_s = a_s + d_s X_s \text{ (the given source price).}$$

It follows that $X_s = \frac{a_t - a_s}{d_s + b_s}$, so the increasing demand at market (a_t) or decreasing cost at the source (a_s or d_s) will make the supply chain more efficient. ■

4.1 Components' Series Order

In this section, we examine the relation between efficiency and local structure of an SPG, i.e. the order of *components*.

Definition 4.2 (Component) *X is a component of G if X only contains one node or $X \subseteq G$ is an SPG whose head s_X and tail t_X satisfy $t_X \in C_S(s_X)$. Besides, X contains all the nodes in $P(t_X) \cap C(s_X)$.*

If component X's tail is Y's head (or the reverse), then we say X and Y are series components. Note that we can extend the definition of component by treating $S(X, Y)$ as a component too, while all the results in this section still hold.

Obviously the efficiency of a supply chain is highly related to its components, and we define component efficiency as follows.

Definition 4.3 (Component Efficiency) *Component efficiency of X is $\lambda(X, b_{t_X}) = \frac{b_{s_X}}{b_{t_X}}$.*

We can see measures the changes of slopes by component X, and it has high component efficiency if $\lambda(X, b_{t_X})$ is small. Let us first consider the simpler case that the processing cost is absent, as a result, the component efficiency is irrelevant to b_{t_X}. The proof is provided in Appendix B.3.

Lemma 4.1 *Assume no processing cost in component X, then*

$$b_{s_X} = \lambda(X, b_{t_X}) = \lambda(X) b_{t_X}$$

where $\lambda(X) \geqslant 2$ is a constant only relevant to the graph structure.

Now consider the efficiency of series components $S(X, Y)$ and assume no processing cost

in X and Y, by Lemma 4.1:

$$\begin{aligned}\lambda(S(X,Y),b_t) &= \lambda(X,\lambda(Y,b_t))\\ &= \lambda(X)\lambda(Y)b_t\\ &= \lambda(Y)\lambda(X)b_t\\ &= \lambda(S(Y,X),b_t).\end{aligned}$$

which means the order of series components does not matter, and we obtain the following theorem (proof detail is provided in Appendix B.4.)

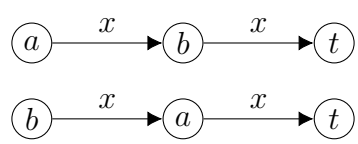
THEOREM 4.1 *Assume no processing cost, switching the order of series components does not change the efficiency.*

Now consider the case with processing cost:

$$\Pi_i = \sum_{j\in B(i)} p_j x_{ij} - p_i X_i - \frac{c_i}{2}X_i^2, \text{ where } c_i > 0.$$

If we change the order of series components, the total flow and the slope efficiency may change as shown in this following example.

EXAMPLE 6 *Consider the price functions of source for the following two graphs, where $c_a > 0, c_b = 0$, $p_t = a - bX_t$, and $p_s = 0$:*



Every edge is active in both graphs, price functions for the first graph are

$$\begin{aligned}p_b &= a - 2bx,\\ p_a &= a - (4b + c_a)x = 0.\end{aligned}$$

As a result, the total flow is $x_1 = \frac{a-p_s}{4b+c_a}$. While the price functions for the second graph are:

$$\begin{aligned}p_a &= a - (2b + c_a)x,\\ p_b &= a - (4b + 2c_a)x = 0.\end{aligned}$$

As a result, the total flow is $x_2 = \frac{a-p_s}{4b+2c_a} < x_1$. It follows that the first graph is more efficient than the second one, and the series order of a and b does influence the efficiency.

In the general case with processing cost, each components has complex influence on the ratio of b_s to b_t, and it is unclear to us what is the efficient algorithm to find the optimal series order of the components. Nevertheless, for some simple cases, we can find the pattern of optimal order.

PROPOSITION 4.3 *For series composition of components X and Y, suppose there is processing cost in X, but no processing cost in Y, then the composition with X close to the source is more efficient than the composition with X close to the sink.*

The proof is provided in Appendix B.5.

One natural interpretation of the above result is the later the processing cost occurs, the worse the efficiency. At equilibrium, upstream firms will consider the cost from downstream. Therefore, the later cost hinders the incentive of upstream firms to supply more goods.

Suppose the supply chain is a straight line, the pattern is clearer, the processing cost c_i is the only criteria to decide the optimal order. Without loss of generality, denote the optimal order as firm 0, 1, ..., $n-1$, n from source 0 to sink n.

PROPOSITION 4.4 *In the most efficient order arrangement of a straight line model, firm i has higher order than firm j if and only if $c_i \leqslant c_j$, and this relation always holds:*

$$a_0 = a_n,$$
$$b_0 = 2^n b_n + \sum_{i=1}^{n} 2^i c_i.$$

The proof is provided in Appendix B.6.

This indicates that it is always better to put the node with higher cost closer to the source, and the fact is the processing cost will be amplified (exponentially) along the path from sink to source.

4.2 Series Insertion, Parallel Insertion

This section focuses on in which way and at what location, adding a component to a given supply chain network will change the efficiency. The two operations we are most interested in are series insertion and parallel insertion.

DEFINITION 4.4 (SERIES INSERTION) *An SPG X is series-inserted into an SPG G at node i by setting $s_X = i$, $t_X = i$.*

DEFINITION 4.5 (PARALLEL INSERTION) *An SPG Y is parallel-inserted into an SPG G at component X by setting $s_Y = s_X$ and $t_Y = t_X$.*

To gain a deeper intuition, parallel insertion provides another path for the flow in the supply chain, while series insertion just makes the supply chain redundant, and we have the following theorem illustrating our intuition.

THEOREM 4.2 *Series insertion always decreases the total flow, while parallel insertion always increases the total flow.*

The proof is provided in Appendix B.7.

Base on the fact that series insertion is always bad, while parallel insertion is always good, the next question is, given a components, where is the most efficient location to insert?

To analyze the changes of efficiency from different parallel insertion location, we can start with a special case, where G can be written as a series composition of two components.

LEMMA 4.2 *Suppose $G = S(X_1, X_2)$, then $P(G, Y)$ is more efficient than parallelly inserting Y at X_1 and also more efficient than parallelly inserting Y at X_2.*

The proof is provided in Appendix B.8 and it can be extended to general SPG as mentioned in the following theorem.

THEOREM 4.3 *Parallel insertion into the entire SPG is more efficient than parallel insertion into a component of the SPG.*

Proof. Proof by induction, starting from the smallest series of components, it is always better off by parallel insertion at the head and tail nodes by Lemma 4.2, and we can repeat this until stopping at the global parallel insertion. ■

This can be interpreted as global parallel insertion will bring more competition to the supply chain network than local parallel insertion. As a result, the network is more efficient after global insertion.

4.3 Firm Location and Individual Utility

This section focuses on firm's utility at equilibrium. Specifically, how does the position of a firm in the network influence its utility at equilibrium? To address this question, we first check the result of a simple example.

EXAMPLE 7 (FIRM UTILITY IN STRAIGHT LINE)

$$p_s = 0 \; (s) \xrightarrow{x} (a) \xrightarrow{x} (t) \; p_t = 1 - X_t$$

Assume processing cost is 0. Price at firm a and s are $p_a = 1 - 2x$ and $p_s = 1 - 4x$. Therefore, the utilities are $\Pi_a = (p_t - p_a)x = x^2$ and $\Pi_s = 2x^2 = 2\Pi_a$.

The above example shows an intuition of the location advantage, that the firm closer to the source may have higher utility than its downstream buyers. However, this is not always true in SPG, especially when there are strong competition among upstream buyers (i.e. MS case). To gain a deeper intuition, we would say the upstream firm who controls all the flow of its downstream firm has a relatively better utility at equilibrium. Therefore, we introduce the following new definition.

Definition 4.6 (Dominating Parent) *i is a dominating parent of j if all the flow from source to j must go through i.*

As in Example 4, a is a dominating parent of b and g, but neither a dominating parent of h nor a dominating parent of i.

For firm i, the utility is

$$\begin{aligned}\Pi_i &= \sum_{j \in B(i)} (p_j - p_i)x_{ij} - \frac{c_i}{2}X_i^2 \\ &= \sum_{j \in B(i)} (b_i X_i + \sum_{k \in C_P(i)} b_k X_k - b_j X_j - \sum_{k \in C_P(j)} b_k X_k)x_{ij} - \frac{c_i}{2}X_i^2.\end{aligned}$$

By using the coefficient relation between buyer and seller as in equation SS, MS, and SM, we can find the closed-form of the utility. The proof is provided in Appendix B.9.

Lemma 4.3 *The utility at equilibrium can be written as*

$$\Pi_i = \frac{1}{2}(b_i + \sum_{k \in C_P(i)} b_k)X_i^2. \tag{9}$$

Based on the utility function, we can prove the following key theorem which shows the location advantage of a dominating parent. Namely, if a firm controls the other firm's flow in the supply chain, then its utility is at least twice as much as its child. The proof is provided in Appendix B.10.

Theorem 4.4 *If firm i is a dominating parent of firm j, then firm i has at least twice as much utility as firm j.*

The following corollary shows that the seller benefits a lot from the competition among the buyer side, and the proof is provided in Appendix B.11.

Corollary 4.1 *In the SM case, the utility of the seller is larger than the utility sum of all the buyers.*

To sum up, we proved a dominating parent always has better utility and the *double utility rule* will hold, which demonstrates the great value of controlling the upstream flows in the real world.

5 Equilibrium in Generalized Series Parallel Graph

In this section, we discuss the equilibria properties in the extension cases when the series parallel graph has multiple sources or sinks. In particular, we will show:

- Multiple-sources-and-single-sink SPG: There exists a unique equilibrium and it can be found in polynomial time.
- Single-source-and-multiple-sinks SPG: Price function of a firm may be piecewise linear under simple settings. Besides, there may exist multiple equilibria.
- Multiple-sources-and-multiple-sinks SPG: There may exist multiple equilibria or there is no equilibrium.

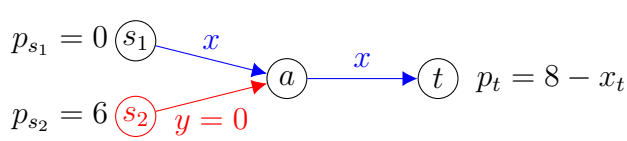
5.1 Multiple Sources and Single Sink

A series parallel graph with multiple sources and single sink (MSPG) is defined as follows.

Definition 5.1 (MSPG) *G is* multiple-source-and-single-sink SPG *if it can be constructed by deleting the source node of an SPG and setting the adjacent nodes of source as the new source nodes.*

Assume every source producer makes decision simultaneously. In contrast to SPG that all edges are active, there may exist inactive edges in MSPG.

Example 8 (Inactive Edges)



By Algorithm 1, Price functions at firm a is $p_a = 8 - 2X_a$. By solving the LCP as in section 3.3.2, the equilibrium flow is $x = 2, y = 0$, where firm s_2 and edge x_{s_2a} are inactive.

By the proof A.2 of Lemma 3.1 (SM case), if a firm is active, all the sub-flows are active too. Therefore, it is sufficient to identify all the inactive edges by check the seller's activity status, and here is an algorithm to identify all the inactive edges in MSPG:

Algorithm 3 : Determinate Inactive Edges

1: Similar to Algorithm 1, compute the price function of all nodes.
2: Solve the convex optimization problem CQP at the source nodes, get the equilibrium flow x_{sj} where $j \in B(s)$ for each source node s.
3: For any firm k, if all of its inflow edges are red, also mark k and its outflow edges as red. Repeat that until no new red firm or edge appears.
4: Firms and edges are inactive if and only if it is marked as red.

Similar to the SPG procedure, we can apply Algorithm 1 and Algorithm 2 to compute the price and quantities at equilibrium.

THEOREM 5.1 *For MSPG, there exists a polynomial time algorithm to solve the equilibrium flow and price, and the equilibrium is unique.*

The proof is quite similar to Theorem 3.1 and is omitted here. Note that uniqueness is because flow quantity is a solution of CQP (Lemma 3.2).

5.2 Single Source and Multiple Sinks

In this section, we focus on the extension of multiple sinks, and the definition is similar to Definition 5.1.

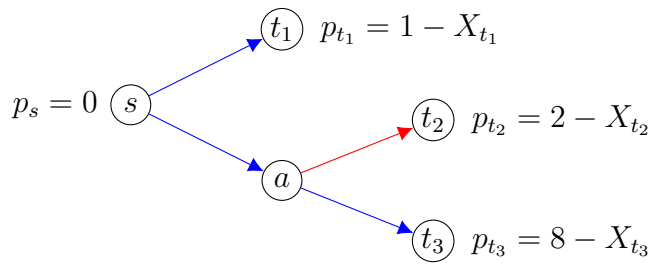
DEFINITION 5.2 *G is* single-source-and-multiple-sinks SPG *if it can be constructed by deleting the sink node of an SPG and setting the adjacent nodes of sink as the new sink nodes.*

First we consider a special case that all markets have the same demand a_t, then all markets are *active*, i.e. every market has positive incoming flow. The proof is provided in Appendix C.1.

THEOREM 5.2 *If all markets have the same demand, then all markets are active and there exists a unique equilibrium.*

However, one major difference multiple sinks cast to SPG is that depending on the selling price from upstream, the ending markets may be *inactive*, that is, the incoming quantity is zero, while the single ending market is always active in SPG. For example,

Example 9 (Markets Activities)



Since $a_{t_1} > p_s$, it is clear that market t_1 is active. Suppose market t_2 is active, market clearance price function at a is $p_a = 5 - X_a$. When source s makes decision, note that flow x_{st_1} and x_{sa} can be handled independently, it is easy to see the optimal decision that maximizes the utility $(5 - X_a)X_a$ of s from a is $X_a = 2.5$ and $p_a = 2.5 > a_{t_2}$, contradicting to market t_2 is active. Therefore, market t_2 is inactive, even though it has higher demand than market t_1.

Note that, the above example is against the intuition that the market with higher demand is more likely to be active (t_2 is inactive while t_1 is). While the truth is not only market demand, but also the competitors and network structure have influence on market activity. Namely, market t_2 is inactive because it has a longer supply chain than t_1 and a strong competition between t_3. As a result, it is less favorable than t_1 and t_3.

Based on the activity status of the ending markets, we introduce two types of processing strategies of upstream firms.

Definition 5.3 (Low Price Strategy) *Firm processes relatively large quantity of goods at a relatively low price, such that all the markets are active.*

Definition 5.4 (High Price Strategy) *Firm processes relatively small quantity of goods at a relatively high price, such that some markets are inactive.*

Note that firm's decision of strategies only depends on individual utility maximization. Because of various choice of strategies, we will see the price functions are piecewise linear in this case. Furthermore, some counterintuitive results will occur, i.e. the increase of demand may result in the decrease of total flow and social welfare (comparing to Proposition 4.2).

To understand these differences, it is helpful to consider an example as in Figure 3, where the two supply chain networks have identical structure but different market demands.

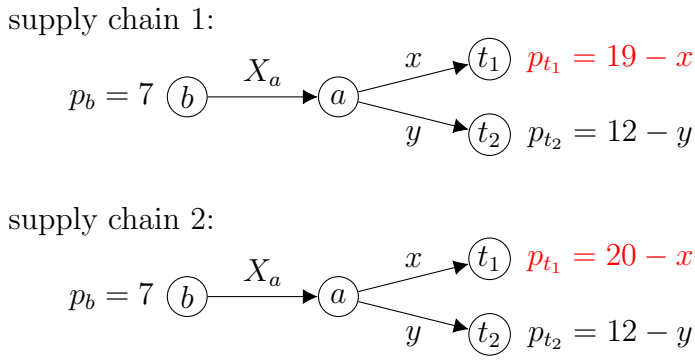


Figure 3: Multiple Sinks Supply Network

It seems that supply chain 2 with higher market demand should have larger flow and social welfare. However, the truth is supply chain 1 is more efficient. To explain this, let us check the market clearance price at b and a first as in Figure 4. Note that the source firm b has two strategies when $p_b = 7$, and both low and high price strategies are feasible. Interestingly, when $a_{t_1} = 20$, the utility of b is maximized by choosing high price strategy and only market t_1 is active. However, when demand at market t_1 drops, low price strategy is preferred by b.

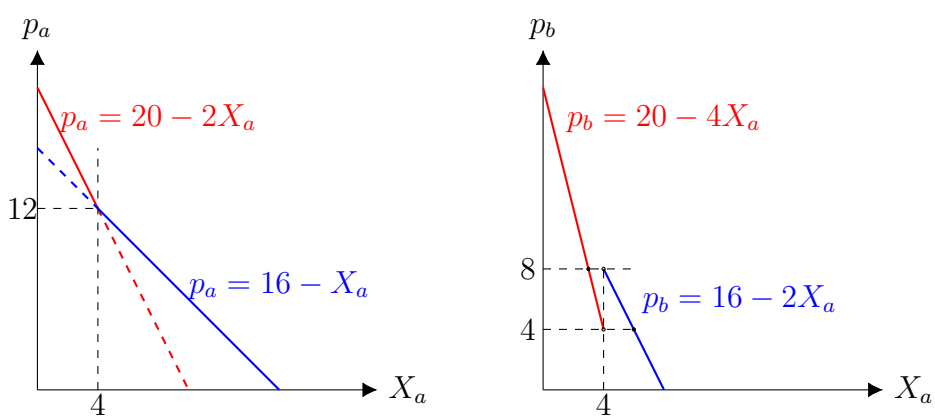


Figure 4: Piecewise Linear Price Functions of Supply Chain 2

By fixing demand at market 2 and adjusting the demand at market 1 (a_{t_1}), Figure 5 shows the numerical results of firm b's corresponding surplus, consumer surplus, total flow and social welfare. Note that the intersecting point at $a_1 \approx 19.5$ shows that increasing demand at market hurts the supply chain efficiency.

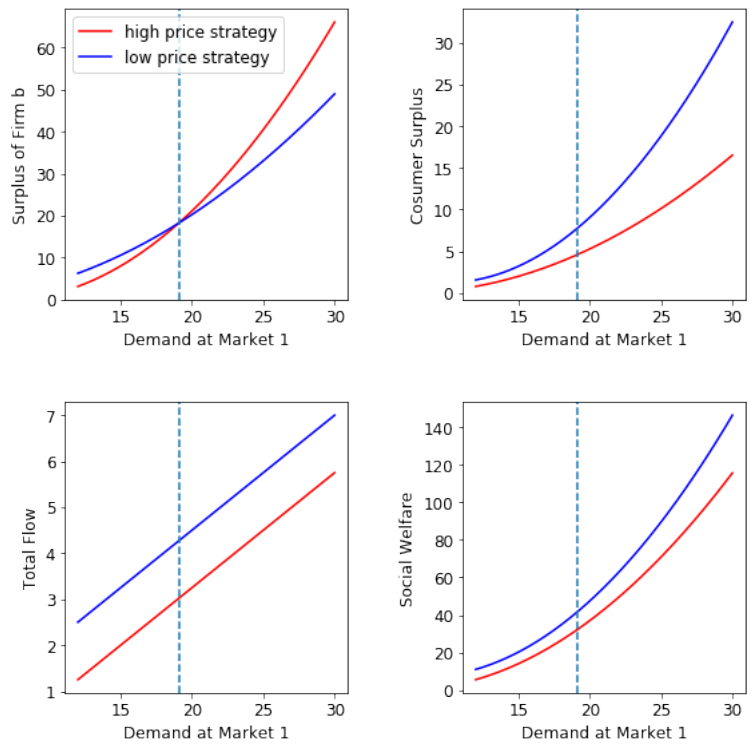


Figure 5: Price Strategies Simulation

Remark. For the supply chain networks in Figure 3, we have the following results:

- Supply chain under low price strategy is always more efficient than under high price strategy.
- When the demand difference between two markets is small enough, low price strategy gives better payoff for source firm b. If the difference is large enough, high price strategy gives better payoff for source firm b.
- Low price strategy always produces higher total surplus of firms and consumers. Hence, the social welfare is also higher.

In short, low price strategy is preferred by b if the demand difference is not large. Besides, with low price strategy, everyone is usually better off. For more interpretation of these results, please check Appendix C.2.

When upstream chooses the optimal strategy and flow, there may exist multiple equilibria for downstream firms. Details are in Example D.7.2.

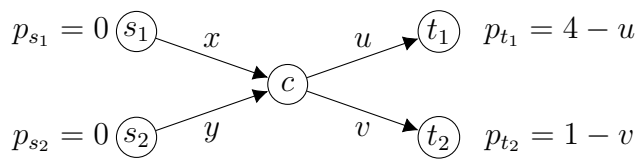
5.3 Multiple Sources and Multiple Sinks

In the multiple sources and multiple sinks case, the problem may become intractable as showed in the following examples:

- Multiple pure strategy equilibria exist (Example 10).
- No pure strategy equilibrium exists (Example 11).

Therefore, it is difficult to analysis the behavior of the firms in the supply chain without any further assumption in this case.

Example 10 (Multiple pure strategy equilibria)



Assume no processing cost. Π_1^h *is the utility of* s_1 *with high price strategy and* Π_1^l *is the utility of* s_1 *with low price strategy. The notations for* s_2 *are similar.*

- *Suppose restricted to high price strategy, the optimal quantities are* $x = y = \frac{2}{3}$*, then*

$$\Pi_1^h = \Pi_2^h = \frac{8}{9}.$$

  If s_2 *increases supply to low price strategy level* ($y' = \frac{11}{12}$)*, his optimal payoff at the new low price strategy is*

$$\Pi_2^{l'} = \frac{121}{144} < \Pi_2^h.$$

  Thus, exists equilibrium at high price strategy.

- *Suppose restricted to low price strategy, the optimal quantities are* $x = y = \frac{5}{6}$*, then*

$$\Pi_1^l = \Pi_2^l = \frac{25}{36}.$$

  If s_2 *decreases supply to high price strategy level* ($y' = \frac{7}{12}$)*, his optimal payoff at the new high price strategy is*

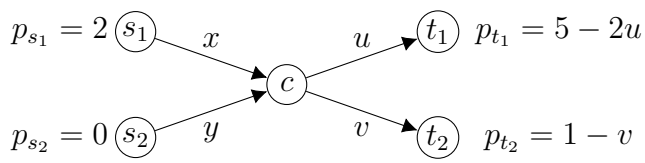
$$\Pi_2^{h'} = \frac{49}{72} < \Pi_2^l.$$

Thus, exists equilibrium at low price strategy.

In summary, both high and low price strategies are equilibria. Computation details can be found in Appendix D.7.1

The following example shows that it is possible that no equilibrium exists in the multiple sources and multiple sinks case.

Example 11 (No pure strategy equilibrium)



Assume no processing cost. Π_1^h *is the utility of* s_1 *with high price strategy and* Π_1^l *is the utility of* s_1 *with low price strategy. The notations for* s_2 *are similar.*

- *Firm* s_1 *never accepts low price strategy, because when market* t_2 *is active* p_c *has to be smaller than* 1*, but* $p_{s_1} > 1 > p_c$*.*

- *If firm* s_1 *is not active* ($x = 0$)*, firm* s_2 *will prefer high price strategy which gives a higher utility,*

$$\Pi_2^l = \frac{49}{32} < \frac{50}{32} = \Pi_2^h.$$

  while the price function at c *is greater than the material cost of firm* s_1*,*

$$p_c = 2.5 > p_{s_1}.$$

  Therefore, this is not an equilibrium because firm s_1 *will prefer participating the supply network and* $x > 0$*.*

- *If firm* s_1 *is active* ($x > 0$)*, then assume they agrees on a local optimal at high price strategy. However, firm* s_2 *will prefer increasing production and switching to low price strategy because*

$$\Pi_2^h = \frac{49}{36} < \frac{50}{36} = \Pi_2^l.$$

  Thus, it is not an equilibrium either.

In summary, neither high nor low price strategy exists equilibrium. Computation details can be found in Appendix D.8.

6 Conclusion

We considered a network model of sequential competition in supply chain networks. Our main contribution is that when the network is series parallel, the model is tractable and allows for a rich set of comparative analysis. In particular, we provide a polynomial time algorithm to compute the equilibrium and the algorithm helps us to study the influence of the network to the total flow of the equilibrium. Furthermore, we show that slightly extending the network structure beyond series parallel graphs makes the model intractable. Several questions are left for future research such as extending the model to capture uncertainty, risks and asymmetric information.

References

K. Bimpikis, S. Ehsani, and R. Ilkilic. Cournot competition in networked markets. In *EC*, page 733, 2014.

K. Bimpikis, O. Candogan, and S. Ehsani. Supply disruptions and optimal network structures. 2017.

S. M. Carr and U. S. Karmarkar. Competition in multiechelon assembly supply chains. *Management Science*, 51(1):45–59, 2005.

C. J. Corbett and U. S. Karmarkar. Competition and structure in serial supply chains with deterministic demand. *Management Science*, 47(7):966–978, 2001.

R. J. Duffin. Topology of series-parallel networks. *Journal of Mathematical Analysis and Applications*, 10(2):303–318, 1965.

A. Federgruen and M. Hu. Sequential multiproduct price competition in supply chain networks. *Operations Research*, 64(1):135–149, 2016.

K. Kannan and T. Nguyen. Welfare implications in intermediary networks. *Working paper*, 2018.

R. E. Kranton and D. F. Minehart. A theory of buyer-seller networks. *American economic review*, 91(3):485–508, 2001.

F. Nava. Efficiency in decentralized oligopolistic markets. *Journal of Economic Theory*, 157: 315–348, 2015.

T. Nguyen. Local bargaining and supply chain instability. *Operations Research*, 65(6), 2017.

T. Nguyen, V. Subramanian, and R. Berry. Delay in trade networks. *Operations Research*, 64(3):646–661, 2016.

J. Z. Pang, H. Fu, W. I. Lee, and A. Wierman. The efficiency of open access in platforms for networked cournot markets. In *INFOCOM 2017-IEEE Conference on Computer Communications, IEEE*, pages 1–9. IEEE, 2017.

R. Wright and Y.-Y. Wong. Buyers, sellers, and middlemen: Variations on search-theoretic themes. *International Economic Review*, 55(2):375–397, 2014.

Appendix

A Proofs in Section 3

A.1 Proof of Proposition 3.3

Proof. Suppose ij is a shortcut of path $l_{ij} = (i, v_1, ..., v_k, j)$, and assume the path l_{ij} is active, i.e., every edge has postive flow.

Since firms never loss money in the supply chain (otherwise just choose to buy and sell nothing), we know

$$p_{v_1} \leq \cdots \leq p_{v_n} \leq p_j.$$

Considering the case that $p_{v_1} < p_j$ at the equilibrium, by the property of series parallel graph and market clearance price, all the flow from i to v_1 will go through firm j. If firm i moves all the flow x_{iv_1} to x_{ij}, the total flow through j will keep the same, and p_j will remain the same price, too. Therefore, firm i is better off by

$$\pi_i = p_j(x_{ij} + x_{iv_i}) > p_j x_{ij} + p_{v_1} x_{iv_1} = \pi_i^*,$$

which cannot happened at the equilibrium. Thus, $p_{v_1} = p_j$ must hold, and

$$p_{v_1} = \cdots = p_{v_n} = p_j.$$

Now consider the optimal decision for v_n, given the market clearance price p_{v_n}, if he buys all the goods supplied to him and sell them to j, his profit is 0, because $p_{v_n} = p_j$. However, he would make a positive profit if processed less amount of goods. Because this would decrease the flow to j and raise the market price at j,

$$p_j' > p_j = p_{v_n},$$

which contradicts to the fact that p_{v_n} is the market clearance price of firm v_n. Hence, the path l_{ij} is inactive.

■

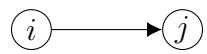
A.2 Proof of Lemma 3.1

Proof. Suppose $j \in B(i)$ and by induction, assume

$$p_j = a_t - b_j X_j - \sum_{k \in C_P(j)} b_k X_k.$$

Obviously it is true when $j = t$, where $c_P(t) = \emptyset$.

Case 1 (SS): $|B(i)| = 1$ and $|S(j)| = 1$:



Utility function of i is

$$\Pi_i = p_j x_{ij} - p_i x_{ij} - \frac{c_i}{2} x_{ij}^2.$$

To compute the price function at i, when $X_i > 0$, which means $x_{ij} > 0$, we have $\frac{\partial \Pi_i}{\partial x_{ij}} = 0$ so that i can maximize its utility. Thus:

$$\begin{aligned} p_i &= p_j + \frac{\partial p_j}{\partial x_{ij}} x_{ij} - c_i x_{ij} \\ &= a_t - b_j x_{ij} - \sum_{k \in C_P(j)} b_k X_k - (b_j + \sum_{k \in C_P(j)} b_k) x_{ij} - c_i x_{ij} \\ &= a_t - (2b_j + \sum_{k \in C_P(j)} b_k + c_i) X_i - \sum_{k \in C_P(i)} b_k X_k \end{aligned}$$

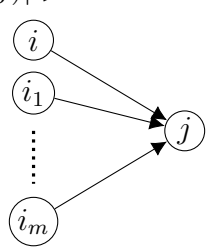
where $X_i = x_{ij}$, $C_P(i) = C_P(j)$ in this case.

p_i is the market clearing price since from above equation, given p_i, we can solve the optimal X_i too.

Summary SS:

$$\begin{aligned} b_i &= 2b_j + \sum_{k \in C_P(j)} b_k + c_i, \\ C_P(i) &= C_P(j). \end{aligned}$$

Case 2 (MS): $|B(i)| = 1$ and $|S(j)| \geqslant 1$:



Utility function of i is

$$\Pi_i = p_j x_{ij} - p_i x_{ij} - \frac{c_i}{2} x_{ij}^2.$$

To compute the price function at i, when $X_i > 0$, which means $x_{ij} > 0$, we have $\frac{\partial \Pi_i}{\partial x_{ij}} = 0$ so that i can maximize its utility. Thus

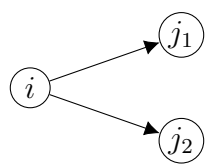
$$\begin{aligned}
p_i &= p_j + \frac{\partial p_j}{\partial x_{ij}} x_{ij} - c_i x_{ij} \\
&= a_t - b_j X_j - \sum_{k \in C_P(j)} b_k X_k - (b_j + \sum_{k \in C_P(j)} b_k) x_{ij} - c_i x_{ij} \\
&= a_t - (b_j + \sum_{k \in C_P(j)} b_k + c_i) x_{ij} - b_j X_j - \sum_{k \in C_P(j)} b_k X_k \\
&= a_t - (b_j + \sum_{k \in C_P(j)} b_k + c_i) X_i - \sum_{k \in C_P(i)} b_k X_k
\end{aligned}$$

where $X_i = x_{ij}$, $C_P(i) = C_P(j) \sqcup \{j\}$ in this case.

Summary MS:

$$\begin{aligned}
b_i &= b_j + \sum_{k \in C_P(j)} b_k + c_i, \\
C_P(i) &= C_P(j) \sqcup \{j\}.
\end{aligned}$$

Case 3 (Simple SM): $|B(i)| \geqslant 2$, $|S(j)| = 1$, and $C_S(i) = \{h\}$:



Remark. X_k where $k \in C_S(i)$ is a function of x_{ij}. This is because market clearance price function ensures downstream firms will buy all the supply from upstream firms. Therefore, x_{ij} is part of X_k.

Notice in this simple SM case, $C_P(j_1) = C_P(j_2)$ (by induction based on the compositions of SPG). Thereby, we just denote them as $C_P(j)$ in the following proof, and price functions are

$$\begin{aligned}
p_{j_1} &= a_t - b_{j_1} X_{j_1} - \sum_{k \in C_P(j)} b_k X_k, \\
p_{j_2} &= a_t - b_{j_2} X_{j_2} - \sum_{k \in C_P(j)} b_k X_k.
\end{aligned}$$

and the corresponding derivatives with respect to x_{ij_1} are

$$\frac{\partial p_{j_1}}{\partial x_{ij_1}} = b_{j_1} + \sum_{k \in C_P(j_1)} b_k, \tag{10}$$

$$\frac{\partial p_{j_2}}{\partial x_{ij_1}} = \sum_{k \in C_P(j_1)} b_k. \tag{11}$$

Utility function of i is

$$\Pi_i = p_{j_1} x_{ij_1} + p_{j_2} x_{ij_2} - p_i X_i - \frac{c_i}{2} X_i^2.$$

Because i has multiple sub-flows and it is possible that some sub-flows are inactive, we will first prove the following claim.

Claim: For any firm i in SPG, its sub-flows are all active.

At equilibrium, by $\frac{\partial \Pi_i}{\partial x_{ij_1}} \leqslant 0$, combined with price derivative equations 10:

$$\begin{aligned}
p_i &\geqslant p_{j_1} + \frac{\partial p_{j_1}}{\partial x_{ij_1}} x_{ij_1} + \frac{\partial p_{j_2}}{\partial x_{ij_1}} x_{ij_2} - c_i X_i \\
&= a_t - b_{j_1} x_{j_1} - \sum_{k \in C_P(j_1)} b_k X_k - (b_{j_1} + \sum_{k \in C_P(j_1)} b_k) x_{ij_1} - \sum_{k \in C_P(j_1)} b_k x_{ij_2} - c_i X_i \\
&= a_t - 2 b_{j_1} x_{ij_1} - (\sum_{k \in C_P(j_1)} b_k + c_i) X_i - \sum_{k \in C_P(j_1)} b_k X_k \\
&= p_{i_1}.
\end{aligned}$$

Similarly by $\frac{\partial \Pi_i}{\partial x_{ij_2}} \leqslant 0$:

$$\begin{aligned}
p_i &\geqslant a_t - 2 b_{j_2} x_{ij_2} - (\sum_{k \in C_P(j_2)} b_k + c_i) X_i - \sum_{k \in C_P(j_2)} b_k X_k \\
&= p_{i_2}
\end{aligned}$$

where $X_{j_1} = x_{ij_1}$, $X_{j_2} = x_{ij_2}$, and $C_P(j) = C_P(j_1) = C_P(j_2)$ in this case.

To prove both branches are active, first assume $x_{ij_1} > 0$ and $x_{ij_2} = 0$, then $p_i = p_{i_1}$ and

$$p_{i_2} - p_{i_1} = 2 b_{j_1} x_{ij_1} > 0 \Rightarrow p_{i_2} > p_{i_1} = p_i,$$

a contradiction. Same argument leads to a contradiction if we assume $x_{ij_1} = 0$ and $x_{ij_2} > 0$.

Suppose $x_{ij_1} = x_{ij_2} = 0$, then $X_i = x_{ij_1} + x_{ij_2} = 0$. By repeating that, we can prove all the parent nodes including source s have zero flow, a contradiction. Thus, both sub-flows are active, and $p_i = p_{i_1} = p_{i_2}$. So far, the claim above is proved.

We know a convex combination of p_{i_1} and p_{i_2} is a necessary condition of p_i. By using the

following convex combination coefficients:

$$\alpha_1 = \frac{\frac{1}{b_{j_1}}}{\frac{1}{b_{j_1}} + \frac{1}{b_{j_2}}}; \quad \alpha_2 = \frac{\frac{1}{b_{j_2}}}{\frac{1}{b_{j_1}} + \frac{1}{b_{j_2}}},$$

and p_i can be written as function of $X_i = x_{ij_1} + x_{ij_2}$:

$$\begin{aligned} p_i &= \alpha_1 p_{i_1} + \alpha_2 p_{i_2} \\ &= a_t - b'_i x_{ij_1} - b'_i x_{ij_2} - \sum_{k \in C_P(j)} b_k X_k \\ &= a_t - b'_i X_i - \sum_{k \in C_P(j)} b_k X_k \end{aligned} \tag{12}$$

where

$$b'_i = \frac{2}{\frac{1}{b_{j_1}} + \frac{1}{b_{j_2}}} + \sum_{k \in C_P(j)} b_k + c_i.$$

Since h is the only merging node ($C_S(i) = \{h\}$), the flow from i will come through h again, i.e. $X_h = X_i$. Also $C_P(j) = C_P(i) \cup \{h\}$ holds. Hence, coefficient b_i is obtained from $b'_i + b_h$:

$$b_i = \frac{2}{\frac{1}{b_{j_1}} + \frac{1}{b_{j_2}}} + 2b_h + \sum_{k \in C_P(j) \setminus \{h\}} b_k + c_i.$$

Meanwhile, equation 12 can be written as the expected format:

$$p_i = a_t - b_i X_i - \sum_{k \in C_P(i)} b_k X_k \tag{13}$$

Note that the above argument can be generalized to $B(i) \geqslant 2$ easily. Suppose $B(i) = \{j_1, ..., j_m\}$, $m \geqslant 3$ and $|C_S(i)| = 1$ ($C_P(j_l)$ are all the same for $l = 1, ..., m$). By similar argument as in the previous claim, $ij, j \in B(i)$ must be active. The convex combination coefficient from price p_{j_l} is

$$\alpha_l = \frac{\frac{1}{b_{j_l}}}{\sum_{j \in B(i)} \frac{1}{b_j}}.$$

Eventually, by similar reasoning:

$$b_i = \frac{2}{\sum_{j \in B(i)} \frac{1}{b_j}} + 2b_h + \sum_{k \in C_P(j) \setminus \{h\}} b_k + c_i.$$

Summary Simple SM:

$$b_i = \frac{2}{\sum_{j\in B(i)} \frac{1}{b_j}} + 2b_h + \sum_{k\in C_P(j)\setminus\{h\}} b_k + c_i \text{ where } h \text{ is the merging node.}$$

Case 4 (General SM): $|B(i)| \geqslant 3$, $|S(j)| = 1$, and $|C_S(i)| \geqslant 2$ (there are multiple self merging child nodes):

At equilibrium, by $\frac{\partial \Pi_i}{\partial x_{ij_1}} \leqslant 0$,

$$\begin{aligned}
p_i &\geqslant p_j + \sum_{l\in B(i)} \frac{\partial p_l}{\partial x_{il}} x_{il} - c_i X_i \\
&= a_t - b_j x_j - \sum_{k\in C_P(j)} b_k X_k - b_j x_{ij} - \sum_{l\in B(i)} \sum_{k\in C_P(l)} b_k x_{il} - c_i X_i \\
&= a_t - 2b_j x_{ij} - \sum_{h\in C_T(i,j)} b_h X_h - (\sum_{k\in C_P(i)} b_k + c_i) X_i - \sum_{k\in C_P(j)} b_k X_k \\
&= a_t - 2b_j x_{ij} - 2\sum_{h\in C_T(i,j)} b_h X_h - (\sum_{k\in C_P(i)} b_k + c_i) X_i - \sum_{k\in C_P(i)} b_k X_k \qquad (14) \\
&= p_{i_j}.
\end{aligned}$$

Similarly we can prove every sub-flow is active, and

$$p_i = a_t - 2b_j x_{ij} - 2\sum_{h\in C_T(i,j)} b_h X_h - (\sum_{k\in C_P(i)} b_k + c_i) X_i - \sum_{k\in C_P(i)} b_k X_k. \qquad \text{(GSM-p)}$$

At the same time, we have

$$b_i X_i = 2b_j x_{ij} + 2\sum_{h\in C_T(i,j)} b_h X_h + (\sum_{k\in C_P(i)} b_k + c_i) X_i. \qquad \text{(GSM-b)}$$

To write p_i as in the form of

$$p_i = a_t - b_i X_i - \sum_{k\in C_P(i)} b_k X_k,$$

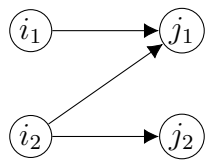
first note that for different $j \in B(i)$, $C_T(i,j)$ in equation 14 may be different. Therefore, we cannot merge these flows all together directly as in the previous case. Meanwhile, we can rank the nodes in $C_T(i,j)$ by the parent-child order as $h_1, \cdots, h_n$ where h_t is the parent of h_{t+1}. By the property of merging nodes, we know:

- For every j, set $C_T(i,j)$ has the common last node h^*, and $X_i = X_{h^*}$.
- For every j, there exists a set $B_k(i) \subseteq B(i)$ whose nodes share the same $C_T(i,j)$. Denote $B_k(i) = \{h_1^k, h_2^k, \cdots, h^*\}$.

Instead of merging all the flows together, general SM case starts merging flows among each set $B_k(i)$. By similar reasoning to the simple SM case, merging among $B_k(i)$ can be

done by using the convex coefficients $\alpha_l = \frac{\frac{1}{b_{j_l}}}{\sum_{j \in B_k(i)} \frac{1}{b_j}}$ for $j_l \in B_k(i)$. We create an *aggregate variable* $b_{B_k(i)} = \frac{1}{\sum_{j \in B_k(i)} \frac{1}{b_j}} + b_{h_1^k}$ to represent the coefficient for flow $X_{h_1^k} = \sum_{j \in B_k(i)} x_{ij}$. Afterwards, we group the new aggregated flows $X_{h_1^k}$ by the same $C_T(i, h_1^k)$, and repeat the above merging operation again for h_2, h_3, and so on. Once h^* is reached, by applying equation SM, we have the final coefficient b_i for node i. Example 14 in the appendix shows the general SM computation.

Case 5 (MM): $|B(i)| \geqslant 2$, $|S(j)| \geqslant 2$:



This is impossible in an SPG, proved by induction since any SPG can be constructed by series and parallel insertion:

- Series insertion: it is easy to see MM will not appear after this.
- Parallel insertion: check the merging head and tail, and it is easy to see MM will not appear either.

Therefore, MM never happens in an SPG. ■

A.3 Proof of proposition 3.5

Proof. Suppose i sells to j, we finish the proof by discussion over case by case. For the SS case, by equation SS:

$$p_j - p_i = (b_j + \sum_{k \in C_P(i)} b_k + c_i) X_i.$$

If $X_i = x_{ij} > 0$, then $p_j > p_i$.
For the SM case, by equation GSM-p:

$$p_j - p_i = b_j x_{ij} + \sum_{h \in C_T(i,j)} b_h X_h + \sum_{k \in C_P(i)} b_k X_i + c_i X_i.$$

If $x_{ij} > 0$, then we prove $p_j > p_i$.
For the MS case, if $X_i = x_{ij} > 0$, by equation MS:

$$p_j - p_i = b_i X_i > 0.$$

■

A.4 Proof of Lemma 3.2

Proof. Consider the Lagrangian function:

$$L(x_{ij}, X_s, X_k, \lambda_{ij}) = \sum_{j \in B(i)} b_j x_{ij}^2 + \sum_{k \in C_S(i) \backslash C_P(i)} b_k X_k^2 - \sum_{j \in B(s)} \lambda_{ij}(a_t - 2b_j x_{ij} - \sum_{k \in C_T(i,j)} 2b_k X_k - const - p_s).$$

Stationarity condition:

- Take the derivative with respect to x_{ij}:

$$\frac{\partial L(x_{ij}, X_j, X_k, \lambda_{ij})}{\partial x_{ij}} = 2b_j x_{ij} - 2b_j \lambda_{ij} = 0$$

  infers $x_{ij} = \lambda_{ij}$.

- Take derivative with respect to X_k where $k \in C_S(i) \backslash C_P(i)$:

$$\frac{\partial L(x_{ij}, X_j, X_k, \lambda_{ij})}{\partial X_k} = 2b_k X_k - \sum_{j: k \in C_P(j)} 2b_k \lambda_{ij} = 0$$

  infers $X_k = \sum_{j:k \in C(j)} \lambda_{ij} = \sum_{j:k \in C(j)} x_{ij}$, which is exactly the definition of X_k (the total flow through k).

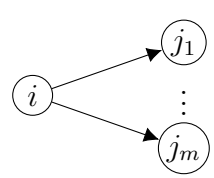
Complimentary condition:
$\forall j \in B(s)$ (recall $x_{ij} = \lambda_{ij}$):

$$\lambda_{ij}(a_t - 2b_j x_{ij} - \sum_{k \in C_T(i,j)} 2b_k X_k - const - p_s) = x_{ij} \frac{\partial \Pi_i}{\partial x_{ij}} = 0.$$

Combined with the primal feasibility conditions $\frac{\partial \Pi_i}{\partial x_{ij}} \leqslant 0$ and $x_{ij} \geqslant 0$, we can see the KKT condition of this convex programming is equivalent to the LCP. Meanwhile, this problem is strictly convex, so the solution is unique. ■

A.5 Proof of Lemma 3.3

Proof. We consider the SM case: $B(i) = \{j_1, ..., j_m\}$ where $m \geqslant 2$.



The decision variables of i are x_{ij}'s where $j \in B(i)$. Recall equation 8:

$$\frac{\partial \Pi_i}{\partial x_{ij}} = a_t - 2b_j x_{ij} - 2 \sum_{k \in C_T(i,j)} b_k X_k - p_i - const.$$

Notice that $X_k = \sum_{j:j \in B(i) \text{ and } k \in C_P(j)} x_{ij}$ and $\frac{\partial \Pi_i}{\partial x_{ij}} = 0$ for all $j \in B$ because ij's are all active, we can rewrite equation 8 as a linear system in the following form:

$$A\vec{x} = (a_t - p_i - const)\vec{1} \tag{15}$$

where $\vec{1}$ is a vector of m ones, $\vec{x} = [x_{ij_1}, ..., x_{ij_m}]^T$, and $A \in \mathbb{R}^{m \times m}$.

First we prove that A is symmetric. Consider $A_{l_1 l_2}$ and $A_{l_2 l_1}$ where $l_1 \neq l_2$, we have

$$A_{l_1 l_2} = 2 \sum_{k \in C_S(i) \cup (C(j_{l_1}) \cap C(j_{l_2})) \setminus C_P(i)} b_k = A_{l_2 l_1},$$

so A is symmetric.

Recall that in Algorithm 1, before computing p_i, we had $p_j = a_t - b_j X_j - \sum_{k \in C_P(j)} b_k X_k$ for $j \in B(i)$. The utility of i is

$$\Pi_i = \sum_{j \in B(i)} p_j x_{ij} - p_i X_i - \frac{c_i}{2} X_i^2.$$

By Lemma 3.1 and equation 8, since ij's are all active, we have $\frac{\partial \Pi_i}{\partial x_{ij}} = 0$. Therefore

$$p_i = a_t - 2b_j x_{ij} - 2 \sum_{k \in C_T(i,j)} b_k X_k - [(\sum_{k \in C_P(i)} b_k + c_i) X_i + \sum_{k \in C_P(i)} b_k X_k].$$

Denote the later part, $(\sum_{k \in C_P(i)} b_k + c_i) X_i + \sum_{k \in C_P(i)} b_k X_k$, as L. Note that in Algorithm 1, L is some unknown value different from the constant pre-computed in Algorithm 2. However, L will not be effected by the convex coefficients, since we only care about the nodes between i and the last self merging node of i.

Let p_{i_l} be the price equation after taking derivative with respect to x_{ij_l}. Then in Algorithm 1, we had the convex coefficients $\alpha_1, ..., \alpha_m$ such that $\sum_{l=1}^m \alpha_l = 1$ and

$$p_i = \sum_{l=1}^m \alpha_l p_{i_l} = a_t - \sum_{l=1}^m \alpha_l A_l \vec{x} - L = a_t - b_i X_i - L$$

where A_l is the l-th row of A and $X_i = \sum_{j \in B(i)} x_{ij}$.

Note that for any $j \in B(i)$, the coefficient of x_{ij} is $\sum_{l=1}^m \alpha_l A_{lj} = b_i$. Since A is symmetric, this can be presented as the following:

$$A^T \vec{\alpha} = A\vec{\alpha} = b_i \vec{1} \tag{16}$$

where $\vec{\alpha} = [\alpha_1, ..., \alpha_m]^T$.

By comparing equation 15 and equation 16, we know $\vec{x}$ is proportional to $\vec{\alpha}$.

To prove that all the price value p_j for $j \in B(i)$ are the same, we can also rewrite equation 8 to obtain a relation between $\frac{\partial \Pi_i}{\partial x_{ij}}$ and p_j:

$$\begin{aligned}\frac{\partial \Pi_i}{\partial x_{ij}} &= a_t - 2b_j x_{ij} - 2\sum_{h \in C_T(i,j)} b_h X_h - p_i - (\sum_{k \in C_P(i)} b_k + c_i)X_i - \sum_{k \in C_P(i)} b_k X_k \\ &= 2(a_t - b_j x_{ij} - \sum_{k \in C_P(j)} b_k X_k) - a_t - (\sum_{k \in C_P(i)} b_k + c_i)X_i + \sum_{k \in C_P(i)} b_k X_k - p_i \\ &= 2p_j - const' - p_i\end{aligned} \tag{17}$$

where $const' = a_t + (\sum_{k \in C_P(i)} b_k + c_i)X_i - \sum_{k \in C_P(i)} b_k X_k$. From equation 17 and the fact that all edges are active, we know that

$$0 = 2p_j - const' - p_i.$$

Therefore, $p_j = \frac{p_i + const'}{2}$ for any $j \in B(i)$. ■

B Proofs in Section 4

B.1 Proof of Proposition 4.1

Proof. For simplicity, we just consider the case without processing cost, and the proof can be extended to the case with processing cost easily. Suppose the market price function is $p_t = a_t - b_t X_t$, for single-edge graph, the utility is $\Pi_s = p_t x - p_s x$. At equilibrium, $\frac{\partial \Pi_s}{\partial x} = 0$ infers $p_s = a_t - 2b_t X_s$.

For general SPG, proof by induction. From $ij \in E$, it is easy to see for the SS case, $b_i \geqslant 2b_j$, and for the MS case $b_i \geqslant b_j$ by the proof in Appendix A.2. For the simple SM case:

$$b_i = \frac{2}{\sum_{j \in B(i)} \frac{1}{b_j}} + 2b_h + \sum_{k \in C_P(j) \setminus \{h\}} b_k > 2b_h \geqslant 2b_t$$

where h is the merging node.

Meanwhile, it is easy to show it also holds for general SM case. Therefore, it always holds that $b_i \geqslant 2b_t$ if $ij \in E$ is the SS case or SM case. Note that s is the only source so $b_s \geqslant 2b_t$ for general SPG. The total flow satisfies

$$p_s = a_s + d_s X_s = a_t - b_s X_s \Rightarrow X_S = \frac{a_t - a_s}{d_s + b_s}.$$

$b_s = 2b_t$ only holds in the single-edge graph and $b_s \geqslant 2b_t$ in any other SPG. Therefore, the single-edge graph is the most efficient SPG supply chain network. ■

B.2 Proof of Proposition 4.2

Proof. From Lemma 3.1:

$$p_s = a_t - b_s X_s = a_s + d_s X_s \text{ (the given source price).}$$

It follows that $X_s = \frac{a_t - a_s}{d_s + b_s}$, so the increasing demand at market (a_t) or decreasing cost at the source (a_s or d_s) will make the supply chain more efficient. ■

B.3 Proof of Lemma 4.1

Proof. By Lemma 3.1, we know that $p_s = a_t - b_s X_s$. While calculating the price function from sink, b_i where $i \in V$ changes proportionally to b_t since there is no "offset" c_i.

By Proposition 4.1, the most efficient network is the single-edge graph and $b_s = 2b_t$. For general SPG, $b_s \geqslant 2b_t$ since it is less efficient and the source price is a given value. ■

B.4 Proof of Theorem 4.1

Proof. Consider series components X and Y, and the larger component $G' = P(X, Y)$, where $t_x = s_y$, $s' = s_X$, and $t' = t_y$.

By lemma 4.1:

$$\begin{aligned} b_{s'} &= \frac{b_{s_X}}{b_{t_x}} \frac{b_{t_x}}{b_{t_y}} b_{t_y} \\ &= \lambda(X)\lambda(Y) b_{t'}. \end{aligned}$$

Now if we change the order of this components, and let $s_X = t_y$, $s' = s_y$, $t' = t_x$, then

$$\begin{aligned} b_{s'} &= \frac{b_{s_y}}{b_{t_y}} \frac{b_{t_y}}{b_{t_x}} b_{t_x} \\ &= \lambda(Y)\lambda(X) b_{t'}. \end{aligned}$$

Thus, we can consider X and Y as one components and switching the inner order does not change the slope

$$b_{s'} = \lambda(X)\lambda(Y) b_{t'} = \lambda(G') b_{t'}$$

and does not change the price function of the other components. Thus, the total flow remains the same. ■

B.5 Proof of Proposition 4.3

For the case with processing cost, $\lambda(\cdot)$ is a function of b_t, and we first prove the following lemma.

Lemma B.1 *With processing cost, for any $\alpha \leqslant 1$,*

$$\lambda(X, \alpha b_t) \geqslant \alpha\lambda(X, b_t).$$

For any $\alpha \geqslant 1$,

$$\lambda(X, \alpha b_t) \leqslant \alpha\lambda(X, b_t).$$

Proof. For any $\alpha \leqslant 1$, we proved it by induction, starts from t, and consider its buyer, which must be SS or MS cases.

For the SS case, by equation SS:

$$b'_i = 2\alpha b_t + \sum_{k \in C_P(t)} b_k + c_t \geqslant \alpha b_i.$$

For the MS case, by equation MS:

$$b'_i = \alpha b_t + \sum_{k \in C_P(t)} b_k + c_t \geqslant \alpha b_i.$$

For the SM case, by induction, $b'_j \geqslant \alpha b_j, j \in B(i)$, by equation SM:

$$\begin{aligned} b'_i &= \frac{2}{\sum_{j \in B(i)} \frac{1}{b'_j}} + 2b'_h + \sum_{k \in C_P(j) \setminus \{h\}} b_k + c_i \\ &\geqslant \frac{2\alpha}{\sum_{j \in B(i)} \frac{1}{b_j}} + 2\alpha b_h + \sum_{k \in C_P(j) \setminus \{h\}} b_k + c_i \\ &\geqslant \alpha b_i. \end{aligned}$$

Similar result applies to the general SM case. Therefore, $\lambda(X, \alpha b_t) = b'_s \geqslant \alpha b_s = \lambda(X, b_t)$. The proof when $\alpha \geqslant 1$ is very similar thus it is omitted here. ■

Now we begin to prove the proposition.

Proof. Denote $S(X, Y)$ and $S(Y, X)$ as SPG 1 and SPG 2. By Lemma 3.1, let $a_t - b^1_s X_s$ be the source price of SPG 1 and $a_t - b^2_s X_s$ be the source price of SPG 2.

We prove $b^1_s \leqslant b^2_s$ as follows:

$$\begin{aligned} b^1_s &= \lambda(X, \lambda(Y, b_t)) \\ &= \lambda(X, \lambda(Y) b_t) \\ &\leqslant \lambda(Y)\lambda(X, b_t) \\ &= \lambda(Y, \lambda(X, b_t)) \\ &= b^2_s \end{aligned}$$

where the second and second last inequality used Lemma 4.1, the third inequality used Lemma B.1 with $\lambda(Y) \geqslant 1$.

Then the flow of SPG 1 is $X_s^1 = \frac{a_t - p_s}{b_s^1 + d_s}$, which is larger than the flow of SPG 2 $X_s^2 = \frac{a_t - p_s}{b_s^2 + d_s}$. Hence, SPG 1 is more efficient. ■

B.6 Proof of Proposition 4.4

Proof. Consider n agents in the straight line model, suppose the firms are labeled by the order as $0, 1, \ldots, n$, where 0 is the source and n is the sink.

Under market clearance price, every node has the same inflow and outflow, denoted as x. The utility function for agent i is

$$\Pi_i = (a_{i+1} - b_{i+1}x)x - p_i x - \frac{c_i}{2}x^2,$$

and its derivative is

$$\frac{\partial \Pi_i}{\partial x} = a_{i+1} - (2b_{i+1} + c_i)x - p_i.$$

Since $x > 0$, $\frac{\partial \Pi_i}{\partial x}$ and we have

$$p_i = a_{i+1} - (2b_{i+1} + c_i)x,$$

the following update rule holds:

$$\begin{aligned} a_i &= a_{i+1}, \\ b_i &= 2b_{i+1} + c_i, \end{aligned}$$

and we can use this to compute the source price function:

$$\begin{aligned} a_0 &= a_n, \\ b_0 &= 2^n b_n + \sum_{i=1}^{n} 2^i c_i. \end{aligned}$$

The coefficient of c_i is 2^i with i (closer to the sink). Consequently, putting the node with higher processing cost c_i closer to source results in a better efficiency. ■

B.7 Proof of Theorem 4.2

We need to prove the following two lemmas first, based on the b_i computation from A.2. Let b_i' be the slope coefficient of i after the insertion.

Lemma B.2 *Series insertion on node i always increases the price function slope b_k where $k \in S(i) \cup i$.* ■

Proof. After a series insertion on node i, we know $b_i' > b_i$ since by Lemma 3.1, $b_i > b_j$ if $ij \in E$ for the SS case and the SM case. By induction and the proof of Lemma 3.1, we know

$b'_k > b_k, \forall k \in S(i)$. Finally, $b'_s > b_s$ infers the total flow decreases. ■

LEMMA B.3 *Parallel insertion on $path_{ij}$ always decreases the price function slope b_k where $k \in S(i) \cup i$.*

Proof. After a parallel insertion on $path_{ij}$, by case SM in Lemma 3.1, the new slope b'_i satisfies $b'_i < b_i$. By induction and the proof of Lemma 3.1, we know $b'_k < b_k, \forall k \in S(i)$. Finally, $b'_s < b_s$ infers the total flow increases. ■

Proof. Suppose the original price function at source is $p_s = a_t - b_s X_s$. If the raw material is sold at price p_s, then at equilibrium:

$$X_s = \frac{a_t - a_s}{d_s + b_s}.$$

By Lemma B.2, after series insertion, $b'_s > b_s$, then we know the total inflow at equilibrium is decreased:

$$X'_s = \frac{a_t - a_s}{d_s + b'_s} < \frac{a_t - a_s}{d_s + b_s} = X_s.$$

While after parallel insertion, $b'_s < b_s$ by Lemma B.3, thus the total inflow at equilibrium is increased:

$$X'_s = \frac{a_t - a_s}{d_s + b'_s} > \frac{a_t - a_s}{d_s + b_s} = X_s.$$

■

B.8 Proof of Lemma 4.2

Proof. For global parallel insertion, the only common child of two branches X, Y is $\{t\}$, denote the new coefficient at s as b_s^G:

$$\begin{aligned} b_s^G &= \frac{1}{\frac{1}{b_s - c_s - 2b_t} + \frac{1}{b}} + c_s + 2b_t \\ &= f(b_s^0, b) b_s^0 + c_s + 2b_t \end{aligned}$$

where $b_s^0 = b_s - c_s - 2b_t$ and define $f(x, y) = \frac{y}{x+y}$.

- Local insertion on component X_2, denote the new coefficient at s as b_s^{L2}:

  $$b_2^{L2} = f(b_2^0, b) b_2^0 + c_2 + 2b_t$$

  where $b_2^0 = b_2 - c_2 - 2b_t$.

Since $b_s - c_s > b_2$, we know $b_s^0 > b_2^0$. Thus, $f(b_s^0, b) < f(b_2^0, b)$, and by induction (similar to the proof of proposition B.5), we can prove

$$\begin{aligned} b_s^{L2} &\geq f(b_2^0, b) b_s^0 + c_s + 2b_t \\ &\geqslant f(b_s^0, b) b_s^0 + c_s + 2b_t \\ &= b_s^G. \end{aligned}$$

Therefore, global parallel insertion is more efficient than local parallel insertion $P(X_2, Y)$.

- Local insertion on component X_1, denote the new coefficient at s as b_s^{L1}:

$$b_s^{L1} = \frac{1}{\frac{1}{b_s - c_s - 2b_2} + \frac{1}{b'}} + c_s + 2b_2.$$

Because $b_2 > b_t$, we have $b' > b$, thus

$$b_s^{L1} \geqslant \frac{1}{\frac{1}{b_s - c_s - 2b_2} + \frac{1}{b}} + c_s + 2b_2.$$

Furthermore, by the fact that $(t < x)$,

$$\frac{1}{\frac{1}{x-t} + \frac{1}{y}} + t \geq \frac{1}{\frac{1}{x} + \frac{1}{y}}.$$

Again, since $b_t < b_2$,

$$\begin{aligned} b_s^{L1} &\geq \frac{1}{\frac{1}{b_s - c_s - 2b_2} + \frac{1}{b}} + c_s + 2b_2 \\ &\geqslant \frac{1}{\frac{1}{b_s - c_s - 2b_t} + \frac{1}{b}} + c_s + 2b_t \\ &= b_s^G. \end{aligned}$$

Therefore, global parallel insertion is more efficient than local parallel insertion $P(X_1, Y)$.

■

B.9 Proof of Lemma 4.3

Proof.

- For the SS case, $X_i = X_j = x_{ij}, C_P(i) = C_P(j)$. Consider the utility of i, by equation SS:

$$\begin{aligned}\Pi_i &= (p_j - p_i)x_{ij} - \frac{c_i}{2}X_i^2 \\ &= (b_iX_i - b_jX_j)x_{ij} - \frac{c_i}{2}X_i^2 \\ &= (b_i - \frac{b_i - \sum_{k\in C_P(i)} b_k - c_i}{2})X_i^2 - \frac{c_i}{2}X_i^2 \\ &= \frac{1}{2}(b_i + \sum_{k\in C_P(i)} b_k)X_i^2.\end{aligned}$$

- For the SM case, $X_j = x_{ij}$. Consider the utility of i, by equation GSM-p:

$$\begin{aligned}\Pi_i &= \sum_{j\in B(i)} (p_j - p_i)x_{ij} - \frac{c_i}{2}X_i^2 \\ &= \sum_{j\in B(i)} (b_jx_{ij} + \sum_{h\in C_T(i,j)} b_hX_h + \sum_{k\in C_P(i)} b_kX_i + c_iX_i)x_{ij} - \frac{c_i}{2}X_i^2.\end{aligned}$$

By equation GSM-b:

$$\begin{aligned}\Pi_i &= \frac{1}{2}\sum_j (b_iX_i + \sum_{k\in C_P(i)} b_kX_i + c_iX_i)x_{ij} - \frac{c_i}{2}X_i^2 \\ &= \frac{1}{2}(b_i + \sum_{k\in C_P(i)} b_k)X_i^2.\end{aligned}$$

- For the MS case, $C_P(i) = C_P(j) \cup j$.

$$\begin{aligned}\Pi_i &= (p_j - p_i)X_i - \frac{c_i}{2}X_i^2 \\ &= (a_t - b_jX_j - \sum_{k\in C_P(j)} b_kX_k - a_t + b_iX_i + \sum_{k\in C_P(i)} b_kX_k)X_i - \frac{c_i}{2}X_i^2 \\ &= b_iX_i^2 - \frac{c_i}{2}X_i^2. \qquad (18)\end{aligned}$$

By equation MS:

$$\begin{aligned}b_i &= b_j + \sum_{j\in C_P(j)} b_k + c_i \\ &= \sum_{j\in C_P(i)} b_k + c_i. \qquad (19)\end{aligned}$$

Plug equation 19 into equation 18:

$$\begin{aligned}\Pi_i &= \frac{1}{2}(b_i + \sum_{k \in C_P(i)} b_k + c_i)X_i^2 - \frac{c_i}{2}X_i^2 \\ &= \frac{1}{2}(b_i + \sum_{k \in C_P(i)} b_k)X_i^2.\end{aligned}$$

■

B.10 Proof of Theorem 4.4

Proof. For the SS or SM case, since $C_P(i) \subset C_P(j)$, $X_i \geqslant X_j = x_{ij}$. Plug equation GSM-b into the utility function of i as in equation 9:

$$\begin{aligned}\Pi_i &= \frac{1}{2}(b_i X_i + \sum_{k \in C_P(i)} b_k X_i)X_i \\ &\geqslant \frac{1}{2}(2b_j x_{ij} + 2\sum_{h \in C_T(i,j)} b_h X_h + \sum_{k \in C_P(i)} b_k X_i + \sum_{k \in C_P(i)} b_k X_i)X_i \\ &\geqslant (b_j X_j + \sum_{k \in C_P(j)} b_k X_j)X_j \\ &= 2\Pi_j\end{aligned}$$

where the second inequality holds because $X_i \geqslant X_j$ and $C_P(j) = C_P(i) \sqcup C_T(i,j)$.

Now suppose there is MS relation to j, consider the the closest dominate parent i of j. Let $l \in B(i)$, and $j \in C(l)$. Then

$$C_P(l) = C_T(i,l) \sqcup C_P(i) = C_P(j) \sqcup \{j\}.$$

Combine this with equation 9:

$$\begin{aligned}\Pi_i &= \frac{1}{2}(b_i X_i + \sum_{k \in C_P(i)} b_k X_i)X_i \\ &\geqslant \frac{1}{2}(2\sum_{h \in C_T(i,l)} b_h X_h + \sum_{k \in C_P(i)} b_k X_i + \sum_{k \in C_P(i)} b_k X_i)X_i \\ &\geqslant (b_j X_j + \sum_{k \in C_P(j)} b_k X_j)X_j \\ &= 2\Pi_j.\end{aligned}$$

■

B.11 Proof of Corollary 4.1

Proof. By equation GSM-p:

$$\begin{aligned}
\Pi_i &= \sum_{j\in B(i)} (p_j - p_i)x_{ij} - \frac{c_i}{2}X_i^2 \\
&= \sum_{j\in B(i)} (b_j x_{ij} + \sum_{h\in C_T(i,j)} b_h X_h + \sum_{k\in C_P(i)} b_k X_i + c_i X_i)x_{ij} - \frac{c_i}{2}X_i^2 \\
&\geqslant \sum_{j\in B(i)} (b_j X_j + \sum_{h\in C_T(i,j)} b_h X_j + \sum_{k\in C_P(i)} b_k X_j)X_j \\
&= \sum_{j\in B(i)} (b_j + \sum_{k\in C_P(j)} b_k)X_j^2 \\
&= \sum_{j\in B(i)} \Pi_j
\end{aligned}$$

where the second last equality is because $C_P(j) = C_P(i) \sqcup C_T(i,j)$, and the last equality is by equation 9. ■

C Proofs in Section 5

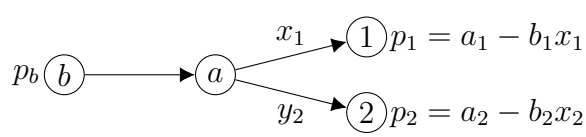
C.1 Proof of Theorem 5.2

Proof. Proof by contradiction to show all edges are active. Suppose there is an inactive market t, then there exists an active firm i such that for any path from i to t, the edges in the path are all inactive. Similar to the proof of Lemma 3.1, the price of every firm j can be presented as a function like a_t minus the sum of some constants time X_j and X_k where $k \in C_P(j)$. If $ij \in E$ is on the path from i to t, then $p_j = a_t$ since $X_j = x_{ij} = 0$ and by the structure of SPG, $X_k = 0$ for any $k \in C_P(j)$. i as an active firm must have sold some goods to another firm k with price less than a_t. However, i could have just sold the goods to j with a higher price to increase its utility. A contradiction.

From the fact that every edge is active, we have a unique price function for each firm, and similar to Theorem 3.1, we can prove the supply quantities at equilibrium is also unique. ■

C.2 Proof of Remark 5.2

We consider the following supply chain network:



For convenience, we denote the first market price as p_1 and the second market price as p_2. The production cost is a constant p_b. Suppose the two price functions at the markets are:

$$p_1 = a_1 - b_1 x_1,$$
$$p_2 = a_2 - b_2 x_2,$$

where $a_1 \geqslant a_2 \geqslant p_b$.

- Supply chain under low price strategy is always more efficient than under high price strategy.

  Proof. Optimal flow X_h at high price strategy is

$$p_b = a_1 - 4b_1 X_h,$$
$$X_h = \frac{a_1 - p_b}{4b_1}.$$

  Optimal flow X_l at low price strategy is

$$p_a = (a_1/b_1 + a_2/b_2)B - 2BX_l,$$
$$p_b = (a_1/b_1 + a_2/b_2)B - 4BX_l,$$
$$X_l = \frac{(a_1/b_1 + a_2/b_2)B - p_b}{4B},$$

  where $B = \frac{1}{\frac{1}{b_1} + \frac{1}{b_2}}$.

  Then we have the difference of total flow between these two strategies:

$$\begin{aligned} X_l - X_h &= \frac{(a_1/b_1 + a_2/b_2)B - p_s}{4B} - \frac{a_1 - p_b}{4b_1} \\ &= \frac{a_2}{4b_2} - \frac{p_b}{4B} + \frac{p_b}{4b_1} \\ &= \frac{a_2}{4b_2} - \frac{p_b}{4b_1} - \frac{p_b}{4b_2} + \frac{p_b}{4b_1} \\ &= \frac{a_2}{4b_2} - \frac{p_b}{4b_2} \\ &\geqslant 0. \end{aligned}$$

■

- When the demand difference between two markets is small enough, low price strategy gives better payoff for source firm. If the difference is large enough, high price strategy gives better payoff for source firm.

Proof. Let CS be the consumer surplus, PS_a be the surplus of firm a, PS_b be the surplus of firm b, SW be the social welfare. At the high price strategy:

$$\begin{aligned}
&CS = \frac{1}{2}b_1X_h^2,\\
&PS_a = b_1X_h^2,\\
&PS_b = 2b_1X_h^2 = \frac{(a_1-p_b)^2}{8b_1},\\
&SW = CS + PS_a + PS_s = \frac{7}{2}b_1X_h^2 = \frac{7}{2}b_1(\frac{a_1-p_b}{4b_1})^2 = \frac{7(a_1-p_b)^2}{32b_1}.
\end{aligned}$$

For social welfare at the low price strategy, let x_1 be the inflow of the first market and x_2 be the inflow of the second market. From the flow relation:

$$\begin{aligned}
&a_1 - 2b_1x_1 = a_2 - 2b_2x_2,\\
&x_1 + x_2 = X_h,
\end{aligned}$$

infers

$$\begin{aligned}
x_1 &= \frac{a_1 - a_2 + 2b_2X_h}{2b_1+2b_2},\\
x_2 &= \frac{2b_1X_h - a_1 + a_2}{2b_1+2b_2},\\
X_h &= \frac{(a_1/b_1 + a_2/b_2)B - p_b}{4B}.
\end{aligned}$$

Therefore

$$\begin{aligned}
&CS = \frac{1}{2}b_1x_1^2 + \frac{1}{2}b_2x_2^2,\\
&PS_a = b_1x_1^2 + b_2x_2^2,\\
&PS_b = 2BX_l^2 = 2B[\frac{(a_1/b_1+a_2/b_2)B - p_b}{4B}]^2 = \frac{[(a_1/b_1+a_2/b_2)B-p_b]^2}{8B},\\
&SW = CS + PS_a + PS_b = \frac{7}{2}b_1X_h^2.
\end{aligned}$$

Notice $PS_b \leqslant PS_a$ in this case.

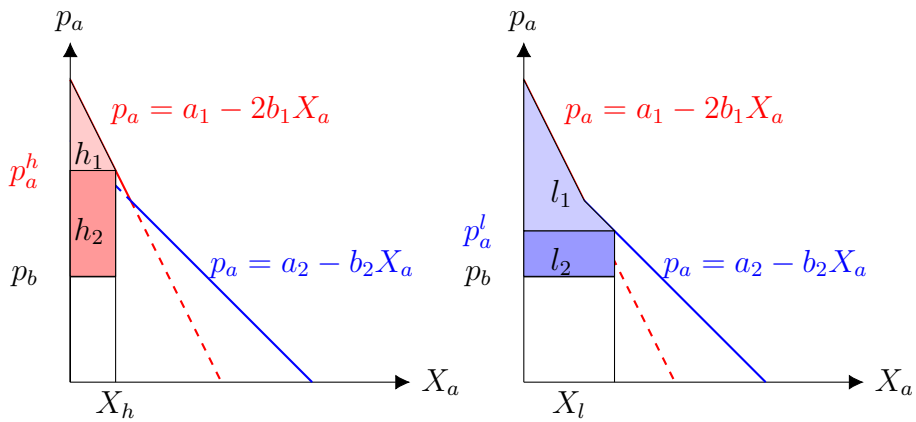
However, to prove this statement, we only need:

$$\frac{PS_b^h}{PS_b^l} = \frac{b_1X_h^2}{BX_l^2} = \frac{b_1+b_2}{b_2}\frac{X_h}{X_h+\Delta}$$

where PS_b^h is the surplus of b at high price and PS_b^l is the surplus of b at low price, and $\Delta = \frac{a_2}{4b_2} - \frac{p_b}{4b_2}$ is irrelevant to a_1. Therefore, as a_1 increases, value $\frac{PS_s^h}{PS_s^l}$ increases from less than 1 to greater than 1. ■

- Low price strategy always produces higher total surplus of firms and consumers. Hence, the social welfare is also higher.

  Proof. We will use a proof by picture. Consider the following figure:



  - For high price strategy, the area of the upper triangle h_1 is PS_a^h, while the area of the lower rectangle h_2 is PS_b^h.
  - For low price strategy, we can compute x_1, x_2 from the intersecting point first. The area of $\frac{1}{2}b_1x_1^2$ and $\frac{1}{2}b_2x_2^2$ is larger than l_1, while the area of the lower rectangle l_2 is PS_b^l.

  Comparing these areas, we can easily see

$$PS_l = PS_a^l + PS_b^l > l_1 + l_2 > h_1 + h_2 = PS_a^h + PS_b^h = PS_h.$$

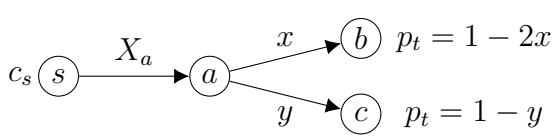
  For consumer surplus, from the fact that the flow to the first and second market is higher with low price strategy and the market prices are inverse linear, the total market surplus is higher.

  Social welfare is the sum of total firm surplus and total consumer surplus. This is a direct result by the fact that low price strategy always produces higher total surplus of firms and consumers. ■

D Examples

D.1 Convex combination

Example 12 (Convex combination in price function computation)



Aim: show the unique market clearing price function is

$$p_a = 1 - \frac{4}{3}(x+y) = 1 - \frac{4}{3}X_a$$
$$p_s = 1 - \frac{8}{3}X_a = 1 - \frac{8}{3}X_s$$

and $X_s = X_a$.

Proof: given $p_a < 1$*, first we know both* $x, y > 0$*, payoff of* a*,*

$$\Pi_a = (1-2x)x + (1-y)y - p_a(x+y)$$

Taking the derivative,

$$\frac{\partial \Pi_a}{\partial x} = 1 - 4x - p_a = 0$$
$$\frac{\partial \Pi_a}{\partial y} = 1 - 2y - p_a = 0$$

By convex combination, we know $x+y$ *must satisfy,*

$$p_a = \frac{1}{3}(1-4x) + \frac{2}{3}(1-2y)$$
$$= 1 - \frac{4}{3}(x+y)$$

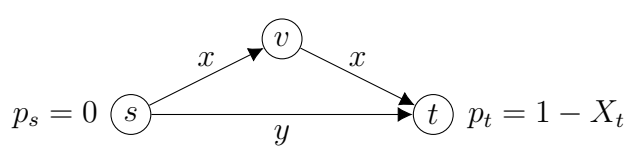
Also by the assumption, $p_a = f(X_a) = f(x+y)$*. Thus,*

$$p_a = 1 - \frac{4}{3}X_a$$

Similarly,

$$\frac{\partial \Pi_s}{\partial X_a} = 0 \Rightarrow p_s = 1 - \frac{8}{3}X_a$$
$$\Rightarrow p_s = 1 - \frac{8}{3}X_s$$

Example 13 (SPG with Shortcut) *Consider the following network where path* (s,t) *is a shortcut of path* (s,v,t)*. Assume no processing and producing cost.*



The utility of v *is:*

$$\Pi_v = (1-x-y)x - p_v x$$

Taking the derivative:

$$\frac{\partial \Pi_v}{\partial x} = 1 - 2x - y - p_v = 0 \Rightarrow p_v = 1 - 2x - y$$

The utility of s is:

$$\begin{aligned}\Pi_s &= p_v x + p_t y - p_s(x+y) \\ &= (1-2x-y)x + (1-x-y)y\end{aligned}$$

Taking the derivative:

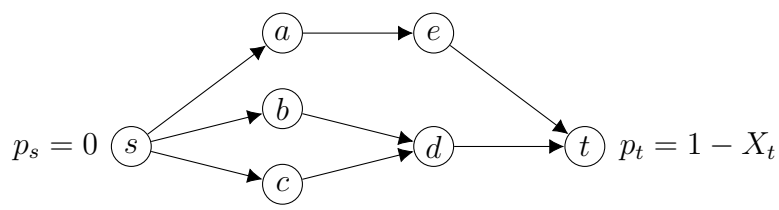
$$\begin{aligned}\frac{\partial \Pi_s}{\partial x} &= 1 - 4x - 2y = 0 \\ \frac{\partial \Pi_s}{\partial y} &= 1 - x - 2y = 0\end{aligned}$$

The solution is $x = 0$ and $y = \frac{1}{2}$. sv and vt are inactive.

D.2 Price Function Computation

Example 14 (Price Function Computation General Form)

Assume that each node in the following has no processing or producing cost.



Recall the equations in ALG_1:

SS Case:

$$b_i = 2b_j + \sum_{k \in C_P(j)} b_k + c_i$$

MS Case:

$$b_i = b_j + \sum_{k \in C_P(j)} b_k + c_i$$

SM Case:

$$b_i = \frac{2}{\sum_{j \in B(i)} \frac{1}{b_j}} + 2b_h + \sum_{k \in C_P(j) \setminus \{h\}} b_k + c_i \text{ where } h \text{ is the merging node.}$$

Backward algorithm, MS case (t to its seller f, d , and e):

$$p_d = 1 - X_d - X_t$$
$$p_e = 1 - X_e - X_t$$

MS case (d to its seller b and c):

$$p_b = 1 - 2X_b - X_d - X_t$$
$$p_c = 1 - 2X_c - X_d - X_t$$

SS case (f to its seller a):

$$p_a = 1 - 3X_a - X_t$$

To compute the price function at s (SM case), utility function at s

$$\Pi_s = p_a X_a + p_b X_b + p_c X_c - p_s(X_a + X_b + X_c)$$

Take the derivative with respect to X_a, X_b, and X_c:

$$\frac{\partial \Pi_s}{\partial X_a} = 0 \Rightarrow p_{s_a} = 1 - 6X_a - (X_a + X_b + X_c) - X_t = 1 - 6X_a - 2X_t$$
$$\frac{\partial \Pi_s}{\partial X_b} = 0 \Rightarrow p_{s_b} = 1 - 4X_b - 2X_d - (X_a + X_b + X_c) - X_t = 1 - 4X_b - 2X_d - 2X_t$$
$$\frac{\partial \Pi_s}{\partial X_c} = 0 \Rightarrow p_{s_c} = 1 - 4X_c - 2X_d - (X_a + X_b + X_c) - X_t = 1 - 4X_c - 2X_d - 2X_t$$

Note that $X_t = X_a + X_b + X_c$, so $\frac{\partial X_t}{\partial X_a} = \frac{\partial X_t}{\partial X_b} = \frac{\partial X_t}{\partial X_c} = 1$.

For the merging order, note that $C_S(s) = \{d, t\}$, by case 4 in the proof A.2 of Lemma 3.1. We start from merging flows with d:

$$\begin{aligned} p_{s_{bc}} &= \frac{1}{2} p_{s_b} + \frac{1}{2} p_{s_c} \\ &= 1 - 2X_{bc} - 2X_d - 2X_t \\ &= 1 - 4X_{bc} - 2X_t \end{aligned}$$

where X_{bc} is a flow variable considering b and c together.

After this, merge flows with t:

$$\begin{aligned} p_{s_{abc}} &= \frac{2}{5} p_{s_a} + \frac{3}{5} p_{s_{bc}} \\ &= 1 - \frac{12}{5} X_{abc} - 2X_t \\ &= 1 - \frac{22}{5} X_s \end{aligned}$$

Note that since $t \notin C_P(s)$, we substitute X_t by X_s.

The aforementioned method is based on computing the convex coefficient. The following method applies aggregate variables. First, $C_S(s) = \{d, t\}$ and d is the merging node. Therefore, $b_{bc} = \frac{1}{\frac{1}{b_b}+\frac{1}{b_c}} + b_d = \frac{1}{\frac{1}{4}+\frac{1}{4}} + 2 = 4$. b_{bc} represents the coefficient of nodes b, c, and d. Next move on to the merging node t, this is the simple SM case, so:

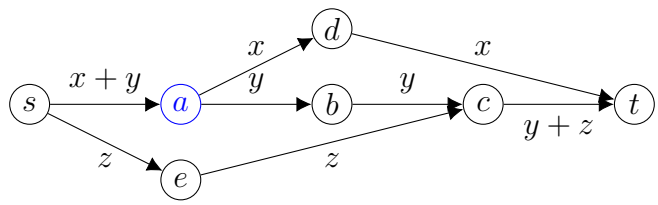
$$b_s = \frac{2}{\frac{1}{b_a} + \frac{1}{b_{bc}}} + 2b_t = \frac{2}{\frac{1}{3} + \frac{1}{4}} + 2 = \frac{22}{5}$$

Please compare it with the Example 3.

D.3 Non-SPG SM

For the parent-child relation, only SS, SM, MS three cases are possible. This example shows the graph restricted to these three relations is not necessary an SPG though.

Example 15 (Non-SPG SM)

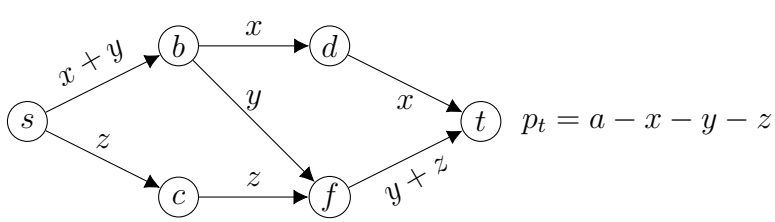


Note that when computing p_a, $C_P(d) = \{t\}$ while $C_P(b) = \{c, t\}$.

D.4 Non-SPG MM

The graph is non-SPG, since MM happens at $\{b, c\} \to \{d, f\}$. However, the equilibrium still exists and unique.

Example 16 (Non-SPG MM)



About parent-merging child nodes, $C_P(b) = C_P(c) = \{f, t\}$.

$$p_d = a - 2x - y - z$$
$$p_f = a - x - 2y - 2z$$
$$p_c = a - x - 2y - 4z$$

To compute price function at b, utility at b is

$$\begin{aligned}\Pi_b &= p_d x + p_f y - p_b(x+y)\\ &= (a-2x-y-z)x + (a-x-2y-2z)y - p_b(x+y)\end{aligned}$$

Take its derivative with respect to x and y:

$$\begin{aligned}\frac{\partial \Pi_b}{\partial x} &= a-4x-2y-z-p_b\\ \frac{\partial \Pi_b}{\partial y} &= a-2x-4y-2z-p_b\end{aligned}$$

To write it as a function of inflow $X_b = x+y$,

$$\begin{aligned}p_b &= 0.5(a-4x-2y-z) + 0.5(a-2x-4y-2z)\\ &= a-3x-3y-1.5z\end{aligned}$$

while $C_P(b) = \{f,t\}$, the above result can't be written as the form unless $b_f = 0$,

$$\begin{aligned}p_b &= a - b_b X_b - b_f X_f - b_t X_t\\ &= a - b_b(x+y) - b_f(y+z) - b_t(x+y+z)\end{aligned}$$

Some relations about market clearing price:

$$\begin{gathered}p_b = a-4x-2y-z = a-2x-4y-2z\\ \Rightarrow\\ 2x = 2y+z\end{gathered}$$

We can rewrite p_c as

$$\begin{aligned}4p_c &= 4a-4x-8y-16z\\ 4p_c + 2x &= 4a-4x-8y-16z+2y+z\\ 4p_c &= 4a-6x-6y-15z\\ p_c &= a - \frac{3}{2}(x+y) - \frac{15}{4}z\end{aligned}$$

So far, the utility of s can be written as,

$$\begin{aligned}\Pi_s &= p_b(x+y) + p_c z - p_s(x+y+z)\\ &= (a - \frac{3}{2}X_b - \frac{3}{2}X_s)X_b + (a - \frac{9}{4}X_c - \frac{3}{2}X_s)X_c - p_s X_s\end{aligned}$$

Similar to the analysis in Section 3.3.2, to maximize the utility, the optimal decision flow of source s is the solution of a system of LCP, and it's equivalent to a convex problem 4, which has unique solution.

An interesting point about the coefficients that

$$A\begin{pmatrix} X_b \\ X_c \end{pmatrix} = \begin{pmatrix} a - p_b \\ a - p_c \end{pmatrix}$$

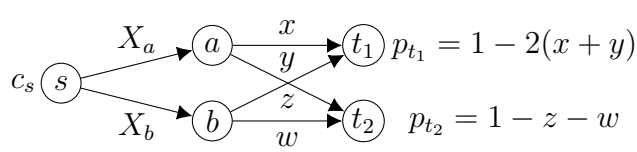
where

$$A = \begin{pmatrix} 3 & 3/2 \\ 3/2 & 15/4 \end{pmatrix}$$

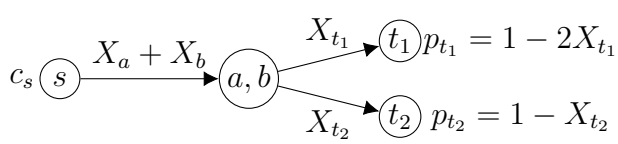
For the solvable problem, the coefficient matrix A always satisfies

- *A is positive.*
- *A is invertible.*
- *unique common coefficient (symmetric, eligible to write as a convex problem)*

Example 17 (Non-invertible A)
An example that A is not invertible,

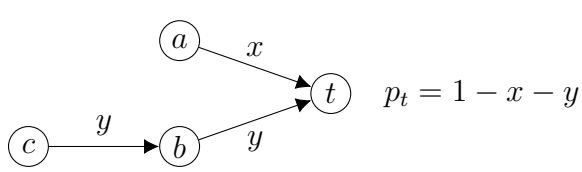


However, we can imaging this equivalent to



D.5 Decision Sequence

Example 18



Assume raw material cost is 0 at both end. Price functions:

$$p_a = 1 - 2x - y; p_b = 1 - x - 2y,$$

and the relation holds at the equilibrium: $x = \frac{1-y}{2}$. The total flow is

$$x + y = \frac{1 + y}{2}.$$

The utility of a is

$$\Pi_a = p_t x = \frac{(1-y)^2}{4}$$

1. Suppose c makes decision p_b, y first, then a, b make decision x, p_t based on the belief over each other.

$$p_b = \frac{1}{2} - \frac{3}{2}y; \quad p_c = \frac{1}{2} - 3y$$

and the optimal $y = 1/6$.

2. Suppose c, a makes decision p_b, p_t, x, y based on the belief over each other, then a make decision (given p_b and x, take y, accept p_t).

$$\begin{aligned} p_b &= 1 - x - 2y \\ p_c &= 1 - x - 4y \\ p_c &= \frac{1}{2} - \frac{7}{2}y \end{aligned}$$

and the optimal $y = 1/7$.

Note that the total flow is higher in the first case, and in the second case, utility of a is higher.

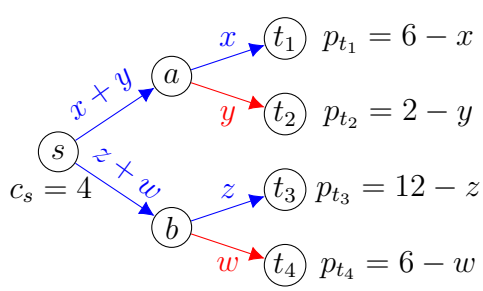
Summery: *in our model*

- *parallel decision is case 2, decide simultaneously based on the* ***belief*** *over each other (assume both act as the unique equilibrium).*
- *multiple branches is case 1, decide each sub-flows by himself, after decision any combination holds, thus any combination can be treated as the TRUE price function.*

D.6 Inactive Edges

One of the main difference between MSPG and SPG is the existence of inactive flow, if the inactive edge is mistakenly assumed active, wrong price function will be used for solving the equilibrium.

Example 19



When computing the price function, the challenge is we do not know which edges are active at equilibrium (while in single source and sink case, we proved every edge is active).

Suppose we assume all of them are active,

$$p_a = 4 - x - y$$

$$p_b = 9 - z - w$$

Thus, at "equilibrium", s makes decision not selling to a ($x + y = 0$), and decision to b is $p_b = 6.5, x_{sb} = 2.5$, and edge bt_4 is inactive ($w = 0$).

However, this is not the equilibrium. For a, since a will make profit by buying items at higher price than p_b, and sell them to t_1, and s will be better off too. For b, by solving the optimal solution at b, we know p_b is too low and x_{sb} is under demand, thus s can be better off by raising the price.

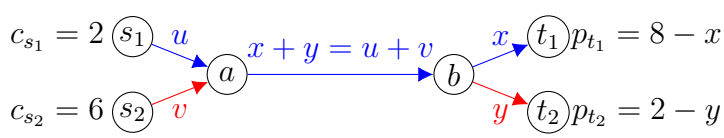
Actually, at equilibrium edge at_2, bt_4 are inactive, while at_1, bt_3 are active. So we should delete edge at_2, bt_4 before the price computation, and the true income price function at node a, b is:

$$p_a = 6 - 2x$$

$$p_b = 12 - 2z$$

Meanwhile, MSPG may also have inactive flow starts from the source.

Example 20



Due to the low profit at market t_2, $c_{s_1}, c_{s_2} \geq p_{t_2}$, it's obvious that edge bt_2 is inactive at equilibrium, so the market clearing price,

$$p_b = 8 - 2x$$

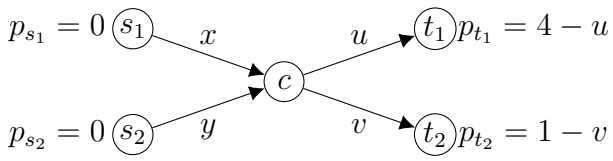
$$p_a = 8 - 4x$$

Treat a as the market of s_1, s_2, by solving a standard bipartite Cournot game, we know edge s_2a is inactive at equilibrium, while s_1a is active.

However, we do not need to worry about the inactive edges s_2a since it does not influence the price function computation of other branches. In other words, the equilibrium can be solved even though we keep this type of inactive edges in the graph. Notice this is always true by the property of series parallel graph.

D.7 Multiple Equilibria

D.7.1 Multiple Sources and Multiple Sinks (Computation of Example 10)



Assume the processing cost is 0. For convenience, denote $p_1 = p_{s_1}$ and $p_2 = p_{s_2}$.

1. High price strategy, since market 2 is inactive, $p_c = 4 - 2X_c$, and prices function at sources are

$$p_1 = 4 - 4x - 2y = 0,$$
$$p_2 = 4 - 2x - 4y = 0.$$

By solving the above equations, the optimal flows are $x = y = a_1/6 = \frac{2}{3}$, double check the price under the optimal flow:

$$p_c = 4 - \frac{8}{3} = \frac{4}{3} \geq a_2.$$

It is a high price strategy and the payoffs are

$$\Pi_1^h = \Pi_2^h = 2x^2 = \frac{8}{9}.$$

2. Low price strategy, since both markets are inactive, $p_c = \frac{5}{2} - X_c$, and prices function at sources are

$$p_1 = \frac{5}{2} - 2x - y = 0,$$
$$p_2 = \frac{5}{2} - x - 2y = 0.$$

By solving the above equations, the optimal flows are $x = y = \frac{a+1}{6} = \frac{5}{6}$, double check the price under the optimal flow:

$$p_c = \frac{5}{2} - \frac{5}{3} = \frac{5}{6} \leq a_2.$$

It is a low price strategy and the payoffs are

$$\Pi_1^l = \Pi_2^l = x^2 = \frac{25}{36}.$$

Note that high price strategy gives a higher payoff.

3. High price strategy is an equilibrium.

Recall the optimal flow $x = \frac{2}{3}$ in part 1, let's fix it for firm 1, while consider firm 2

increases y and try low price strategy:

$$p_c = \frac{5}{2} - X_c,$$
$$p_2 = \frac{5}{2} - x - 2y = 0.$$

The new flow is $y = \frac{11}{12}$, double check the price under these flows:

$$p_c = \frac{5}{2} - \frac{3}{2} - \frac{11}{12} = \frac{11}{12} \leqslant a_2$$

It is a low price strategy and the new payoffs for firm 2 is

$$\Pi_2' = y^2 = \frac{121}{144} < \Pi_2^h = \frac{8}{9}$$

Thus, high price strategy is an equilibrium.

4. Low price strategy is an equilibrium.

Recall the optimal flow $x = \frac{5}{6}$ in part 2, let's fix it for firm 1, while consider firm 2 decreases y and try high price strategy,

$$p_c = 4 - 2X_c,$$
$$p_2 = 4 - 2x - 4y = 0.$$

The new flow is $y = \frac{7}{12}$, double check the price under these flows:

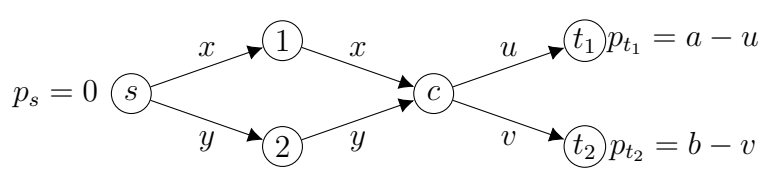
$$p_c = 4 - 2(\frac{5}{6} - \frac{7}{12}) = \frac{7}{6} \geqslant a_2.$$

It is a high price strategy and the new payoffs for firm 2 is

$$\Pi_2' = 2y^2 = \frac{49}{72} < \Pi_2^l = \frac{25}{36}.$$

Thus, low price strategy is an equilibrium. In summary, both high and low price strategy are equilibria.

D.7.2 Single Source and Multiple Sources



1. High price strategy:

$$
\begin{aligned}
p_c &= a - 2X_c \\
p_1 &= a - 4x - 2y = 0 \\
p_2 &= a - 2x - 4y = 0 \\
p_s &= a - 6x - 6y \Rightarrow X_s = \frac{a}{6}
\end{aligned}
$$

Utility of b is

$$\Pi_b^h = p_1 X_s = \frac{a}{2}\frac{a}{6} = \frac{a^2}{12}$$

2. Low price strategy:

$$
\begin{aligned}
p_c &= \frac{a+b}{2} - X_c \\
p_1 &= \frac{a+b}{2} - 2x - y = 0 \\
p_2 &= \frac{a+b}{2} - x - 2y = 0 \\
p_s &= \frac{a+b}{2} - 3x - 3y \Rightarrow X_s = \frac{a+b}{6}
\end{aligned}
$$

Utility of b is

$$\Pi_b^l = p_1 X_s = \frac{a+b}{4}\frac{a+b}{6} = \frac{(a+b)^2}{24}$$

To make low price strategy more preferable:

$$\Pi_b^l > \Pi_b^h \Rightarrow b > (1 - \sqrt{2})a$$

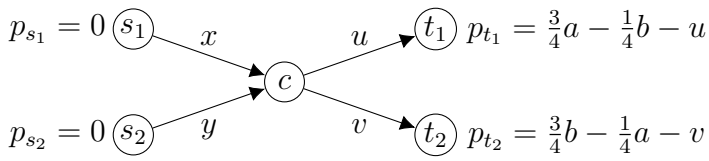
Suppose b chooses low price strategy;

$$
\begin{aligned}
p_1 = p_2 &= \frac{a+b}{4} \\
p_c &= \frac{a+b}{3}
\end{aligned}
$$

To ensure it is a low price strategy:

$$p_c < a_2 \Rightarrow b > 0.5a$$

Given p_1 and p_2, for firm 1 and 2's decision, it is equivalent to



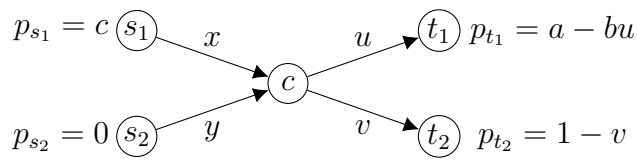
Suppose

$$\frac{3}{4}a - \frac{1}{4}b = 4(\frac{3}{4}b - \frac{1}{4}a) \Rightarrow b = \frac{7}{13}a$$

which also satisfies the above requirements for a, b, and we can apply the previous example's result to show that the equilibrium for $1, 2$ decision is not unique!

In summary, firm s will prefer low price strategy. However, the decision of downstream firms $1, 2$ will be unpredictable (multiple equilibria) if s choose the "optimal" price for low price strategy.

D.8 Non-Equilibrium (Computation of Example 11)



Assume:

$$p_{s_1} = c = 2; p_{s_2} = 0; a = 5; b = 2$$

1. High price strategy where market 2 is inactive, and price function at firm c is $p_c = a - 2bX_c$, and price functions at sources are

$$p_1 = a - 4bx - 2by = 0,$$
$$p_2 = a - 2bx - 4by = 0.$$

Solve the above equations and flows at equilibrium

$$x = \frac{a - 2c}{6b}; y = \frac{a + c}{6b}$$

($a \geqslant 2c$ so that $x_h \geqslant 0$)

Double check the price at c

$$p_c = \frac{a + c}{3} \geq 1$$

It is a high price strategy and the payoffs are

$$\Pi_2^h = 2y^2 = \frac{(a + c)^2}{18b}$$

2. Low price strategy where both markets are inactive, and price function at firm c is $p_c = \frac{a+b}{b+1} - \frac{2b}{b+1}X_c$, and price functions at sources are

$$p_1 = \frac{a+b}{b+1} - \frac{4b}{b+1}x - \frac{2b}{b+1}y = 0,$$
$$p_2 = \frac{a+b}{b+1} - \frac{2b}{b+1}x - \frac{4b}{b+1}y = 0.$$

By solving the above equations, we got the flows as

$$x = \frac{a+b-2c(b+1)}{6b}; y = \frac{a+c(b+1)}{6b} + \frac{1}{6}$$

$(a+b \geqslant 2c(b+1)$ so that $x_l \geqslant 0)$

Double check the price c

$$p_c = \frac{2}{3}\frac{a+b}{b+1} + \frac{c}{3}$$

It is a low price strategy and the payoffs are

$$\Pi_2^l = y^2$$

Note that high price strategy gives a higher payoff.

3. High price strategy is NOT an equilibrium.

Suppose firm 2 increases y and try low price strategy:

$$p_2 = \frac{a+b}{b+1} - \frac{2b}{b+1}x_h - \frac{4b}{b+1}y = 0$$

The new flow is

$$y' = \frac{a+c}{6b} + \frac{1}{4}$$

Double check the price c

$$p_c = \frac{a+b}{b+1} - \frac{2b}{b+1}X_c \leqslant 1$$

To prove high price strategy is not an equilibrium,

$$\Pi_2^h = 2by^2 = 2b(\frac{a+c}{6b})^2 < b(\frac{a+c}{6b} + \frac{1}{4})^2 = by'^2 = \Pi_2^{h\to l}$$

4. Low price strategy is NOT an equilibrium.

Suppose firm 2 decreases y and try high price strategy:

$$p_2 = \frac{a+b}{b+1} - \frac{2b}{b+1}x - \frac{4b}{b+1}y = 0$$

The new flow is

$$y' = \frac{a + c(b+1)}{6b} - \frac{1}{12}$$

Double check the price at c,

$$p_c = a - 2bX_c \geq 1$$

To prove low price strategy is not an equilibrium,

$$\Pi_2^l = by^2 = b(\frac{a+c}{6b} + \frac{c}{6} + \frac{1}{6})^2 < 2b(\frac{a+c}{6b} + \frac{c}{6} - \frac{1}{12})^2 = 2by'^2 = \Pi_2^{l \to h}$$

In summary, neither high nor low price strategy is equilibria.