Investigation of Inventory Dynamics in General Network Supply Chain Systems Using Agent-Based Modeling Approach

Dajun Yue[[1]](#footnote-1)\*

Department of Chemical and Biological Engineering, Northwestern University, Evanston, IL 60208

# Abstract

Inventory is a critical component of the supply chain system. It can be highly dynamic since the external demands from customers are uncertain. Understanding its dynamics would help the companies to manage their inventory more efficiently and to achieve higher profit. Due to its significance to the economic viability of modern industries, extensive studies on the inventory dynamics in supply chains can be found in the existing literatures, mainly from the operations research community. The traditional approach was via theoretical analysis using statistical methods and probability tools. However, this approach is prohibitive if we want to address the inventory dynamics in the supply chain systems with complex network configurations and abundant features. To overcome this limitation, we propose novel agent-based models coded in NetLogo which view every player in the supply chain as self-autonomous agents. The most significant advantage of using agent-based modeling approach is that it is straightforward to model the complex interaction mechanisms and inventory operation features by specifying the agent rules in a bottom-up manner, while it enables us to observe inventory dynamic patterns from a macroscopic perspective. In this work, we first propose an agent-based supply chain model consisting of three echelons: supplier, distributor, and retailer. Complex interaction mechanisms and inventory operation features are explored and analyzed afterwards. Furthermore, we also propose a HubNet activity version of the network supply chain system, which allows human clients to play as an inventory manager who is directly involved in the inventory dynamics. It would allow human clients to experience the inventory dynamics in simulation and to develop better understanding of the inventory management strategies.

***Key words:*** *network supply chain system, inventory dynamics, agent-based modeling, NetLogo, HubNet.*

# Introduction

Inventory is ubiquitous in our daily life. The product on the store shelf is inventory. The gasoline storage at the gas station is also inventory. In fact, any stocked goods can be considered as inventory. The location and facility where the goods are kept is called stock keeping unit (SKU). Generally, a supply chain system is consisting of multiple SKUs. These SKUs can be categorized into several echelons: typically supplier, distributor, and retailer. In order for the customers to receive the product, the product would flow from suppliers via distributors to retailers and eventually be delivered to the customers. The customers go to the retailers to purchase the product. The purchasing amount at the retailer is considered external demand from the marketplace. Within the supply chain system, the retailers also order from the distributors, and the distributors order from the suppliers. These orders are considered internal demand. As can be seen, the product flows from upstream to downstream. But the order information flow from downstream to upstream.

If the external demands are constant, the inventory in the supply chain would be steady as well. However, in the real-world marketplace, the external demands are highly uncertain and unpredictable. Correspondingly, the inventory within the supply chain system would also exhibit dynamic behaviors. It is critical to understand the inventory dynamics in the supply chain system for companies that want to manage their inventory more efficiently and to achieve higher profit or lower cost. Higher inventory level would guarantee a higher customer service level, which is the frequency that the customers’ demand are satisfied. Failure to satisfy the demands would hurt the customers’ willing of purchase and cause back-order penalties. But meanwhile, higher inventory level also incurs a significant inventory holding cost, which usually corresponds to maintenance and utility fees. Thus, how to adjust the ordering strategy and inventory level is a significant question over every inventory manager’s mind.

Traditionally, researchers employ statistical methods and probability tools to study the inventory dynamics in the supply chain system. However, this approach has its critical limitation that it is prohibitive to address supply chain systems with complex network structure and to encompass complicated interaction mechanisms between SKUs and individual inventory operations [[1](#_ENREF_1)]. This is also the reason why thorough analysis of inventory dynamics is merely restricted to single stage SKU and serial supply chains. Most studies on network supply chains are subject to a fair amount of ideal assumptions and simplifications. Fortunately, agent-based modeling can provide alternatives for resolving this issue. It enables us to model the supply chain system in a bottom-up manner [[2](#_ENREF_2)]. It is straight forward to design complicated interaction mechanisms between SKUs and to include abundant inventory operation features at individual SKUs using the agent-based modeling approach by specifying rules for agents. More importantly, it allows us to observe the inventory dynamics from a macroscopic perspective through simulation.

In this work, we first propose an agent-based supply chain model with flexible network structures which is coded in NetLogo [[3](#_ENREF_3)]. Rules will be explained in detail, followed by the exploration and analysis of the results from the simulation. Furthermore, we propose a HubNet activity version of the network supply chain model, aiming to let human clients experience the inventory dynamics and develop better order strategies in the simulation.

# General network supply chain model



Figure 1. Interface of the network supply chain model

Figure 1 shows the interface of the agent-based model. There are three echelons in the supply chain system: supplier, distributor, and retailer. We assume there is only one supplier in the supply chain system, which always has sufficient inventory to satisfy the orders from the distributors. The number of distributors and retailers can be adjusted via the NUM-OF-DISTRIBUTORS and NUM-OF-RETAILERS sliders. Note that, when the NUM-OF-DISTRIBUTORS and NUM-OF-RETAILERS are both equal to one, the network supply chain system downgrade to a serial supply chain. There are two types of directed links in the model: demand links and supply links. The demand links go from retailers to distributors or from distributors to the supplier. The order information is conveyed from downstream to upstream via the demand links. In contrast, the supply links go from the supplier to distributors or from distributors to retailers. These supply links represent the so-called shipping pipelines in the supply chain system. Once the product is shipped, it takes some time to arrive at the destination. This delay in shipment is called the lead time of the pipeline. In our model, users can adjust the lead time between the supplier and distributors or between distributors and retailers, by adjusting the LEAD-TIME-SUPPLIER-DISTRIBUTOR and LEAD-TIME-DISTRIBUTOR-RETAILER sliders, respectively. For simplification, we assume the lead times are uniform for all distributors and retailers. However, it is trivial to account for player-specific lead times in the agent-based models. Below we will start explaining the rules and communication mechanisms in the agent-based model. However, due to the large amount of rules involved in this model, we only address the primary rules that related to the inventory dynamics. More detailed explanations can be found in the comments in the corresponding NetLogo codes.

## External demand

In the simulation, every tick is considered as one day. Every day, a certain amount of demand occurs at the retailers. We consider three types of demand generating mechanisms in this model: deterministic, Poisson, and truncated normal distribution, which can be chosen from the chooser on the Interface. The deterministic demand means that the daily demands are constant, which equal to the DETERMINISTIC-DEMAND. In this case, there is no uncertainty in the external demands. The Poisson demand follows the Poisson distribution, which is often encountered in the real-world market place. The mean of the Poisson distribution equals to MEAN-FOR-POISSON. The truncated normal distribution is different from the normal distribution. As we know, the normal distribution span from negative infinity to positive infinity, while the real-world demand usually has an upper bound and a lower bound. The generating mechanism of the truncated normal distribution is as follows: we first generate a random normally-distributed number with mean equal to MEAN-FOR-NORMAL and standard deviation equal to STD-FOR-NORMAL. If the number is between the LOWER-BOUND-FOR-NORMAL and UPPER-BOUND-FOR-NORMAL, we report this number as today’s demand. If the number is beyond this range, we randomly select a number between the upper bound and lower bound. In other words, we cut off the range that is beyond the bounds and equally increase the probability of the numbers that are between the bounds. In addition, to avoid fractional units of goods, we round up the number to the nearest integer.

## Ordering rules

In this model, we assume that all players in the supply chain system are rational players, which means that they are following a set of rules when ordering. In practice, there are several ordering rules (or inventory policies). The one chosen in this model is called base-stock policy. The base-stock policy suggests that one checks the inventory position every day and compares it with the base-stock level. If the inventory position is below the base-stock level, one orders the amount exactly equal to the difference between the base-stock level and the inventory position. If the inventory position is higher than the base-stock level, there is no need to order. Note that the term, inventory position, is defined as the on-hand inventory level plus the pipeline inventory minus the back-orders. The on-hand inventory is the physical goods stocked in the SKU. The pipeline inventory refers to the goods that have already been shipped from upstream nodes but not arrived at the SKU yet. Thus the pipeline inventory is also called inventory in-transit. The back-orders occur when the on-hand inventory is not sufficient to meet the order from downstream nodes or external demands. The unsatisfied demands are counted as back-orders, which are to be fulfilled once sufficient on-hand stocks are available.

As outlined above, the base-stock policy involves only one parameter: base-stock level. Once the base-stock level is specified, the order quantity can be calculated. The next step is to determine from which upstream nodes to order. In this model, distributors can only order from the supplier, thus there is no need for selection. However, retailers may face multiple distributors, which is distinct to that in serial supply chain. We assume that the retailer will choose one distributor to order from in each day. Once the distributor is chosen, the retailer tends to be loyal to that distributor unless that distributor turns the retailer down. This happens when the distributor fail to satisfy the retailer’s order, thus creating back-orders. In this case, the retailer may seek for other distributors for better services. We model this mechanism by first finding out the corresponding demand link(s) with the minimum back-orders. If the current distributor is among these candidates, we report the current distributor, showing the retailer’s loyalty. If the current distributor is not among these candidates, the retailer will alter his/her distributor to one of these candidates assuming they may provide better services.

## Replenishment

This section talks about how the goods are received into the SKU’s inventory. Every day, a previous order may arrive and get out of the pipeline. This is modeled by taking out the first item in the pipeline list and removing it from the list. The received goods are added to the on-hand inventory. To make sure that the supplier always has sufficient inventory to satisfy the orders from distributors, every day we reset the on-hand inventory at the supplier to a sufficiently big number (say, 10000).

## Order processing

In a network supply chain, a SKU may need to provide services to multiple downstream nodes. The total amount of new orders is equal to the sum of orders placed from all the downstream nodes. However, besides the new orders, the back-orders also need to be fulfilled. Therefore, the requested quantity is equal to the total new orders plus the total back-orders. If the on-hand inventory is sufficient to satisfy the requested quantity, one will deliver exactly the requested quantity to all downstream nodes from the current inventory. But if the on-hand inventory is less than the requested quantity, the SKU will deliver whatever in on-hand inventory and count the unsatisfied amount as back-orders. Moreover, an allocation problem needs to be answered in this situation as well. That is, how many goods to ship to each downstream node. In this model, we assume that the SKU would allocate the available on-hand inventory to downstream nodes according to their quota, which is proportional to their requested quantity. For example, Retailer 1 orders 10 units without previous back-orders and Retailer 2 places no new orders but with 5 units previous back-orders. If the distributor only has 3 units on-hand, he/she will ship 2 units to Retailer 1 and 1 unit to Retailer 2. Note that, to avoid fractional units, we also round up the shipping quantity via a special allocation algorithm. The shipping process is modeled by putting a number equal to the shipping quantity at the end of the corresponding ORDERS-FILLED pipeline list. If nothing is requested from the SKU, we simply put zero into the list. And certainly, the shipped quantity of goods is deducted from the on-hand inventory.

## Summary of the day

At the end of the day, the cost and revenue for all players will be summarized. The cost consists of two parts: inventory holding cost and back-order penalty. In this model, we assume that the inventory holding cost is $0.5/unit and the back-order penalty is $2/unit for both distributors and retailers. The revenue is $3/unit for distributors and $4/unit for retailers. These cost data were casually made up by the author, so users of this model are free to change the cost parameters. The profit is calculated as the revenue minus the cost. The on-hand inventory level and total profit of all distributors and retailers will we plotted on the Interface as well.

## Inventory policy updates

Before a new day starts, every SKU will adjust their inventory policies, specifically the base-stock levels. This is because the initial base-stock level is given by the user: INITIAL-STOCK-DISTRIBUTOR for distributors and INITIAL-STOCK-RETAILER for retailers. The initial stock level may not correctly fit into the dynamics of the supply chain system. If the base-stock level is too high, it may incur a significant inventory holding cost. But if the base-stock level is too low, it may often encounter supply shortage thus losing customers. Therefore, in this model we design an algorithm for players to adjust their base-stock levels according to their own demand history record. The algorithm will be explained below.

On the Interface, there is a slider named RECORD-LENGTH. This global variable specifies the number of recent demands that a SKU is going to keep on record. These recent demands will serve to update the base-stock level. We first calculate the mean of these recent demands and then the standard deviation. The most common practice in the inventory management is to set the base-stock level in the form of working stock plus safety stock:

 

where  is the base-stock level;  and  stand for the mean and standard deviation of the recent demands;  is the order processing delay, which equals to lead time + 1, where 1 corresponds to the 1 day review period;  stands for the safety factor. The safety factor reflects the player’s attitude towards risks. It is straightforward that the high the safety factor, the higher the safety stock level, consequently the higher the service level but higher inventory holding cost. In this model, we randomly assign different safety factors to different players, attempting to see how their different attitudes towards risks would affect their profit and services.

# Results and analysis

## Inventory fluctuation under uncertain demand

![C:\Users\Haha\AppData\Roaming\Tencent\Users\148391063\QQ\WinTemp\RichOle\I0ZX]GDMOU`YT[1@3R05][M.jpg]()

 (a) (b)

Figure 2. Inventory profiles of all players: (a) under deterministic demand, (b) under Poisson demand

Figure 2a corresponds to inventory profile under the deterministic demand and Figure 2b corresponds to the inventory profile under the uncertain daily demand following the Poisson distribution. As can be seen in Figure 2a, at the beginning of the simulation, players were adjusting their respective inventory policies according to the demands. After the optimal base-stock level was found, the network supply chain system achieves equilibrium, where the on-hand inventory at the end of the day always equal to zero. This indicates that the arrived orders from upstream nodes are just enough to satisfy the demand from downstream nodes, so that unnecessary inventory holding costs are avoided. However, when the external demands are uncertain, the inventory profiles tend to be highly fluctuating as illustrated by Figure 2b. One thing worth mentioning is that the amplitude of fluctuation at the distributors is much larger than that at the retailers. This is easy to understand because in the network supply chain, a distributor may face multiple retailers. The flow of goods at a distributor thus may be several times larger than that at a retailer. Consequently, the internal demand uncertainty felt by the distributors is also larger than the external demand uncertainty felt by the retailers.

## Peer competition

One appealing feature in this network supply chain model is that we explicitly model the competition between peers. Specifically, since the lead times and costs are uniform for all distributors, the only difference between distributors is their attitude towards risks, which is modeled by assigning different safety factors to different distributors. As mentioned in the previous section, the retailers tend to be loyal to distributors. But they may transfer to other distributors if their orders are turned down. It would be interesting to see how the distributors will gain and lose retailers during the simulation and to ask how the distributor’s attitude towards risks would affect the players profit in the long run.



Figure 3. Total profit of all players in the network supply chain system

Figure 3 is obtained from the simulation of network supply chain with two distributors and four retailers. The green and sky curves correspond to the profit of the two distributors. It is interesting to see that the profit of Distributor 1 goes up first then drops far negative afterwards, while Distributor 2 was beaten by Distributor 1 at the beginning but become far better off in the long term. In this simulation, the safety factor of Distributor 1 is 1.75 and the safety factor of Distributor 2 is 1.91. This indicates that although the lower safety inventory may help to reduce the inventory holding cost, a higher inventory factor, thus higher service level, is critical to seize the retailers. As shown in our simulation, the ability to keep the customers is very significant in order for companies to sustainably increase their profit.

# HubNet activity version of the network supply chain model

Instead of specifying the ordering policies and allocation mechanism for the players in the network supply chain, the HubNet activity allows human clients to play the role of inventory manager during the simulation. Since the strategies of human clients may not be necessarily the same as the specified rules, it would be interesting to compare the human clients’ strategies with the aforementioned agent-based model. More importantly, in the network supply chain system, it would be interesting to see how the human clients will compete with their peers with the same role.

The main interface is shown in Figure 4 and the client interface is shown in Figure 5.



Figure 4. Main interface of the HubNet model



Figure 5. Client interface of the HubNet model

This HubNet model involves a small network supply chain, consisting of one supplier with sufficient supply and two distributors and two retailers. The distributors and retailers are played by human clients, thus in total four human clients are needed to play the game. As a distributor, one needs to determine how many to order from the supplier each day. Since the supplier always has sufficient supply, the orders placed by distributors are guaranteed to be fulfilled. As a retailer, one needs to determine how many to order from each of the two distributors, which is different from that in the aforementioned agent-based model. Note that the orders from the retailer are not guaranteed to be fulfilled, because the upstream distributor may not always have sufficient on-hand inventory. The players’ objective is to minimize their costs, including the inventory holding cost and back-order penalty.

The HubNet model works as follows. First, the observer should recruit four human clients to play the game. Once the four human clients are ready, the observer sets up the network supply chain. By pressing the GO button, the simulation starts. There is a clock indicating the days of simulation, which is also shown in on the client interface. At the beginning of everyday, the demands are posted to each human client’s interface. They are required to determine how many and from whom to order by adjusting the ORDER-TO1 and ORDER-TO2 sliders. The players press FINISH button to finalize their ordering decision. Once all four players have finished their orders, the day is summarized and a new day begins.

One thing worth mentioning is the view on the client’s interface. Since the size of the circle can reflect the on-hand inventory, enabling the view on the client’s interface would provide more information to the human clients. For instance, if a retailer observes that one distributor has more on-hand inventory than the other, he/she may tend to order more from the one with higher on-hand inventory. Therefore, it would be also interesting to see whether enabling the view or not will affect the human clients’ ordering behaviors.

Unfortunately, we haven’t invited human clients to play the game yet. Thus no results can be presented at this time for the HubNet model.

# Conclusion

We have proposed an agent-based model to study the inventory dynamics of general network supply chain systems. It encompasses rich features that were difficult to model using traditional approaches, including base-stock policy, back-orders, demand allocation, supplier selection, etc. Emergent phenomena such as inventory fluctuation and peer competition were observed and discussed in this work. Furthermore, a HubNet activity version of the network supply chain model was also proposed, to allow human clients to experience in the supply chain dynamics.

# Acknowledgement

The authors are grateful to the valuable comments and help from Bryan Head, who is the teaching assistant for EECS 372/472 Spring 2013 at Northwestern University. The authors also would like to give credit to the HubNet Root Beer Game model [[4](#_ENREF_4)] proposed by Professor Uri Wilensky at Northwestern Institute on Complex Systems at Northwestern University, since part of the codes in our model are adapted from this work.

# References

[1] L. V. Snyder and Z. M. Shen, *Fundamentals of Supply Chain Theory*. Hoboken, New Jersey: John Wiley & Sons, Inc., 2011.

[2] U. Wilensky, *Modeling Nature's Emergent Patterns with Multi-agent Languages*. <http://ccl.northwestern.edu/netlogo/>. Center for Connected Learning and Computer-Based Modeling: Northwestern Institute on Complex Systems, Northwestern University, Evanston, IL, 2013.

[3] U. Wilensky, *NetLogo*. <http://ccl.northwestern.edu/netlogo/>. Center for Connected Learning and Computer-Based Modeling: Northwestern Institute on Complex Systems, Northwestern University, Evanston, IL, 1999.

[4] U. Wilensky and W. Stroup, *NetLogo HubNet Root Beer Game model*. <http://ccl.northwestern.edu/netlogo/models/HubNetRootBeerGame>. Center for Connected Learning and Computer-Based Modeling: Northwestern Institute on Complex Systems, Northwestern University, Evanston, IL, 2003.

1. \* Corresponding author. Email: dajunyue2016@u.northwestern.edu [↑](#footnote-ref-1)