

Application of Model Predictive Control to Robust Management of Multi-Echelon Demand Networks in Semiconductor Manufacturing

Martin W. Braun^{1*}, Daniel E. Rivera^{1†}, W. Matthew Carlyle^{2‡}, and Karl G. Kempf³

¹Department of Chemical and Materials Engineering
Control Systems Engineering Laboratory,
Arizona State University, Tempe, Arizona 85287-6006

²Department of Industrial Engineering,
Arizona State University, Tempe, AZ, USA 85287-5906

³Decision Technologies, Intel Corporation,
5000 W. Chandler Blvd., Chandler, AZ, USA 85226

May 13, 2003

*Currently with Texas Instruments Inc., 13570 N. Central Expressway, MS 3701, Dallas, TX, 75243

†To whom all correspondence should be addressed. phone: (480) 965-9476 fax: (480) 965-0037; e-mail: daniel.rivera@asu.edu

‡Currently with the Operations Research Department, Naval Postgraduate School, Monterey, CA 93943

Abstract

Model Predictive Control (MPC) is presented as a robust, flexible decision framework for dynamically managing inventories and satisfying customer demand in demand networks (a.k.a. supply chains). In this paper, a formulation and the benefits of an MPC-based, control-oriented tactical inventory management system meaningful to the semiconductor industry are presented via two significant examples. The translation of available information in the supply chain problem into MPC variables is demonstrated with a single product, two-node supply chain example. Simulations demonstrating the ability of a properly tuned MPC control system to maintain performance and robustness in spite of plant-model mismatch are shown. Insights gained from these simulations are used to formulate a partially decentralized MPC implementation for a six node, two product, three echelon demand network problem developed by Intel Corporation. These simulations show that the demand network is well managed under conditions that involve simultaneous demand forecast inaccuracies and plant-model mismatch. Ultimately, the “bullwhip” effect is mitigated and required safety stock levels in demand networks can be significantly reduced as a result of the performance demonstrated by the MPC approach.

KEYWORDS: Supply chain management, model predictive control, inventory control.

1 Introduction

A supply chain (a.k.a. demand network or value web) consists of the interconnected components required to transform ideas and raw materials into delivered products and services. The entire structure is organized and managed with the goal of maintaining a high level of customer service, while minimizing costs and maximizing revenues. Companies no longer compete against other companies. Instead, supply chains compete against other supply chains. The supply chain that gains market share does so by providing customers the right product, in the right amount, at the right time, for the right price, and at the right place (Bodington and Shobry, 1999; Kempf *et al.*, 2001).

Among the most difficult of supply chains to optimize are those in the semiconductor industry. The combination of features that make semiconductor supply chains especially challenging including long production lead times and large demand forecast errors combined with short product lifecycles. Traditional planning methods applied in these circumstances lead to safety stock levels that cover as much as a year's worth of demand (Lee *et al.*, 1997). Recent literature suggests that billions of dollars can be achieved through improved management of semiconductor supply chains (PriceWaterhouseCoopers, 2000; Kempf *et al.*, 2001).

In this paper, an approach using Model Predictive Control (MPC) is proposed as a means to manage supply chains in general and those in the semiconductor industry in particular. It is proposed that the long and variable lead times and error-prone demand forecasts encountered in semiconductor supply chains can be robustly controlled using MPC. Since the semiconductor industry generates enormous wealth, direct experimentation involves untenable financial risk and simulation is required. The MPC framework is shown to control qualitatively realistic simulations of semiconductor supply chains with safety stock levels well below levels suggested by industry heuristics. Control is demonstrated on a simple one product, two echelon, two node problem, then scaled up for a more realistic two product, three echelon, six node problem. Further up-scaling is needed for larger semiconductor problems, but the initial results are very encouraging. In addition, MPC is regularly utilized in large scale plant-wide control applications in the petroleum refining and chemical manufacturing industries (García *et al.*, 1989).

Recently, work utilizing Model Predictive Control has been found to provide an attractive alternative for inventory control (Tzafestas *et al.*, 1997), and supply chain management (Flores *et al.*, 2000; Perea-Lopez *et al.*, 2000). These approaches are conceptually

different and require less detailed knowledge in comparison with cost-optimal stochastic programming solutions which require many “what-if” cases to be run and examined by highly skilled professionals (Kafoglis, 1999). Yet MPC offers the same flexibility in terms of the information sharing, network topology, and constraints that can be handled. The appeal of MPC for dynamic inventory management in supply chains can be summarized as follows: as an optimizer, MPC can minimize or maximize an objective function that represents a suitable measure for supply chain performance. As a controller, MPC can be tuned to achieve stability, robustness, and performance in the presence of plant/model mismatch, failures and disturbances which affect the system.

The paper is organized as follows. Section 2 provides an overview of Model Predictive Control. Section 3 presents a two node demand network problem consisting of a factory and retailer to describe, simulate, and evaluate the fundamentals of the MPC-based inventory management decision tool. The translation of the available information in a supply chain setting to process control variables and the selection of proper tuning parameters is demonstrated to result in well behaved management of a two-node supply chain. These insights are used in Section 4 to manage a six node, two product, three echelon network that mimics the “back end” configuration of a semiconductor chain. The network is robustly managed using MPC under realistic information and modeling inaccuracies. Details of the simulation development and implementation environment are presented in Section 5. The paper concludes with a summary of the ideas presented and some extensions for future research (Section 6).

2 Model Predictive Control Decision Framework

Model Predictive Control has long been the preferred algorithm for robust, multivariable control in the process industries, with the number of implementations numbering in the thousands. The popularity of MPC stems from the relative ease with which it can be understood, and its ability to handle input and output constraints (García *et al.*, 1989). The objective function of an MPC controller can be written as

$$\begin{aligned}
 J = & \sum_{\ell=1}^p Q_e(\ell)(\hat{y}(k+\ell|k) - r(k+\ell))^2 \\
 & + \sum_{\ell=1}^m Q_{\Delta u}(\ell)(\Delta u(k+\ell-1|k))^2 \\
 & + \sum_{\ell=1}^m Q_u(\ell)(u(k+\ell-1|k) - u_{target}(k+\ell-1|k))^2
 \end{aligned} \tag{1}$$

where $Q_e(\ell)$, $Q_{\Delta u}(\ell)$, and $Q_u(\ell)$ represent penalty weights on the control error, control move size, and control signal, respectively. The move suppression weight $Q_{\Delta u}(\ell)$ plays a particularly important role in providing robustness to the MPC control system, as will become evident in the examples and case studies presented in Sections 3 and 4. p and m represent the controller prediction and move horizons, respectively, while k represents time. r represents the setpoint trajectory, u the control signal/manipulated variable, and \hat{y} is the estimated output; the relationship of these variables to demand network information is summarized in Table 1 and further discussed in Section 3 of the paper. The three terms in the MPC cost function penalize predicted setpoint tracking error, excess movement of the manipulated variable, and deviation of the manipulated variable from a target value, respectively.

The MPC optimization problem can be written

$$\min_{\Delta u(k|k)\dots\Delta u(k+m-1|k)} J \quad (2)$$

s.t.

$$\begin{aligned} u_{min} &\leq u(k + \ell - 1|k) \leq u_{max}, \\ \Delta u_{min} &\leq \Delta u(k + \ell - 1|k) \leq \Delta u_{min}, \\ y_{min} &\leq y(k + \ell - 1|k) \leq y_{max}, \end{aligned}$$

The optimization problem is readily solved by standard quadratic programming (QP) algorithms. Only the first control element of the solution is implemented. At the next time step the optimization problem is solved again with updated information from the system. This is referred to as the receding-horizon property of MPC as illustrated in Figure 1. Note that the MPC controller explicitly uses a model relating the inputs and measured disturbances to the outputs.

3 Single Product, Two Echelon, Two Node Problem Analysis

In this section, a single product, two node demand network representing a factory and a retailer is used to establish the linkages between the inventory management problem and the process control one. The assignment of demand network variables to process control

variables is summarized in Table 1. Figure 2 illustrates the material flows from the Factory to the Retailer and on to the Customer. The dynamics of each node is simulated with a difference equation arising from the conservation of total mass

$$I_A(k+1) = P_A(k) - S_{2A}(k) + I_A(k), \quad (3)$$

where

$$P_A(k) = S_{1A}(k - \Theta_p). \quad (4)$$

I_A is the inventory of A; S_{1A} is the incoming stream of A; S_{2A} is the outgoing stream of A; Θ_p represents the processing delay; $P_A(k)$ is the material which has completed processing at time k of species A . Figure 3 provides a graphical representation of the information transfer between the MPC controllers and nodes of the two-node system.

Current customer demand is fed directly to the retailer and the retailer can immediately fill that demand that day. It is assumed that a demand forecast is known although it may have bias or random error as dictated by the simulation conditions. The demand (measured disturbance) and demand forecasts (estimated future measured disturbances) are fed to the first echelon MPC controller. Using the current inventory (outputs) information from the retailer, the first echelon MPC controller decides what orders (inputs) for product A should be placed with the factory, and what the order forecast (estimated inputs) will look like. This order forecast is shared with the second echelon MPC controller.

The second echelon MPC controller uses the order forecast (now an estimated future measured disturbance) from the first echelon MPC controller, and the inventory information (outputs) from the factory to decide on production starts (inputs) for the day. Both MPC controllers contain models that determine the effect that orders (measured disturbances) from downstream entities have on the future inventories (estimated outputs) in their node. This model also relates orders to the factory (inputs for the first echelon MPC controller) and production starts (inputs for the second echelon controller) to the inventory levels (outputs). The inventory targets (setpoint trajectories) are a forward time shifted version of the estimated future measured disturbances (plus safety stock), for the first echelon controller. For the factory, the inventory targets are an exact replication (plus safety stock), of the estimated measured disturbances for the factory since there is no direct feed through (i.e. orders placed today are only on backorder if not filled tomorrow).

From Figure 3 it is clear that each node entity has its own MPC controller to manage inventory levels. Both controllers are developed with a slight modification of the interpretation of inventory. Namely, the net stock (Silver *et al.*, 1998) is used as the measured output of the system. Net stock is the measured inventory at time k less the cumulative outstanding backorders at time k . Relying on net stock improves the performance of the network, since effect of backorders on the controlled variable is now recognized by the MPC controllers.

Figures 4 and 5 show the responses of the proposed MPC structure to a scenario in which the production leadtime in the factory and the shipping delay to the retailer are 2 and 0 time units, respectively. With no plant-model mismatch and in the absence of constraints on production and shipping quantities there is no need to use move suppression; as shown in Figures 4 and 5 demand is perfectly satisfied with no backorders; inventory levels are systematically built up or diminished to meet demand.

The MPC control approach is then evaluated under conditions of plant-model mismatch. The production leadtime in the Factory is actually 3 time units, while the Factory MPC controller makes use of a model that assumes a production leadtime of 2 time units. The Retailer MPC controller is implemented as if there exists no shipping delay between the Factory and the Retailer, however in the simulation there is 1 time unit of delay. There is no production delay in the Retailer for either simulation or controller. First the controllers are implemented with no move suppression. The time series and metrics for the Retailer and Factory are shown in Figures 6 and 7, respectively. Inventory levels show wild fluctuations and increase to exorbitant levels; customer service could be better. Using move suppression values of 150 for both controllers, the system is stabilized, and the 1500 units of safety stock are sufficient to eliminate backorders as shown in Figure 8 and 9.

4 Two Product, Three Echelon, Six Node Problem Analysis

The performance and robustness of the MPC control system is now demonstrated on a six node network simulated using realistic plant-model mismatch, biased forecast error, and a realistic demand profile. These conditions were recommended for evaluation by Intel Corp. on a six node, two product, three echelon simulation. The simulation mimics the packaging, distribution and retail sale of semiconductor products, in each of the three echelons.

Figure 10 illustrates the interconnections of the material flows for the six node Intel Problem (Armbruster *et al.*, 2001); As noted in the diagram, transportation delays range from 2 to 4 days. The direct shipment routes each require 2 days. The cross-shipment routes require 3 days between Assembly/Test and Warehouse echelons. The cross-shipment routes require 4 days between Warehouse and Retailer echelons. In the Warehouse and Retailer nodes, material that enters the receiving dock does not show up in the inventory until the following day. The Assembly/Test nodes require an additional 10 days for processing.

For this demand network pattern the transportation delays can be modeled in the same way as was done with the 2-node supply chain. There are three new modeling entities to consider: Factories, Geographic Warehouses, and Retailers. Each has a different configuration of input and output flows. For the single inflow, two outflow Factory as shown in Figure 12, the material balances for each species can be written

$$I_A(k+1) = P_A(k) - S_{2A}(k) - S_{3A}(k) + I_A(k) \quad (5)$$

$$I_B(k+1) = P_B(k) - S_{2B}(k) - S_{3B}(k) + I_B(k) \quad (6)$$

with the constitutive relationships

$$P_A(k) = S_{1A}(k - \Theta_p) \quad (7)$$

$$P_B(k) = S_{1B}(k - \Theta_p) \quad (8)$$

As with the two node problem, I represents inventory, S represents in and out stream values, Θ_p represents processing delays while $P_A(k)$ and $P_B(k)$ represent material which has completed processing at time k of species A and B , respectively. These material balances are subject to the following constraints

$$I_A(k+1) + I_B(k+1) \leq C_I \quad (9)$$

$$S_{1A}(k) + S_{1B}(k) \leq C_1 \quad (10)$$

$$S_{2A}(k) + S_{3A}(k) + S_{2B}(k) + S_{3B}(k) \leq C_{23} \quad (11)$$

$$S_{2A}(k) + S_{2B}(k) \leq C_2 \quad (12)$$

$$S_{3A}(k) + S_{3B}(k) \leq C_3. \quad (13)$$

These constraints represent capacity limitations on inventory, shipping for stream #1, release rate from the Factory, shipping for stream #2, and shipping for stream #3, respectively.

The next entity to consider is the two inflow, two outflow warehouse, as represented in Figure 13. The material balances of this system are a natural extension of the single inflow, two

outflow material balance equations.

$$I_A(k+1) = S_{1A}(k) + S_{2A}(k) - S_{3A}(k) - S_{4A}(k) + I_A(k) \quad (14)$$

$$I_B(k+1) = S_{1B}(k) + S_{2B}(k) - S_{3B}(k) - S_{4B}(k) + I_B(k) \quad (15)$$

These material balances are subject to the following constraints

$$I_A(k+1) + I_B(k+1) \leq C_I \quad (16)$$

$$S_{1A}(k) + S_{1B}(k) \leq C_1 \quad (17)$$

$$S_{2A}(k) + S_{2B}(k) \leq C_2 \quad (18)$$

$$S_{1A}(k) + S_{2A}(k) + S_{1B}(k) + S_{2B}(k) \leq C_{12} \quad (19)$$

$$S_{3A}(k) + S_{3B}(k) \leq C_3 \quad (20)$$

$$S_{4A}(k) + S_{4B}(k) \leq C_4 \quad (21)$$

$$S_{3A}(k) + S_{4A}(k) + S_{3B}(k) + S_{4B}(k) \leq C_{34} \quad (22)$$

These constraints represent capacity limitations on inventory, shipping for stream #1, shipping for stream #2, total inflow rate, shipping for stream #3, shipping for stream #4, and release rate from the Warehouse, respectively.

For the two inflow, single outflow Retailer (Figure 14), the material balances are written

$$I_A(k+1) = S_{1A}(k) + S_{2A}(k) - S_{3A}(k) + I_A(k) \quad (23)$$

$$I_B(k+1) = S_{1B}(k) + S_{2B}(k) - S_{3B}(k) + I_B(k). \quad (24)$$

These material balances are subject to the following constraints

$$I_A(k+1) + I_B(k+1) \leq C_I \quad (25)$$

$$S_{1A}(k) + S_{1B}(k) \leq C_1 \quad (26)$$

$$S_{2A}(k) + S_{2B}(k) \leq C_2 \quad (27)$$

$$S_{1A}(k) + S_{2A}(k) + S_{1B}(k) + S_{2B}(k) \leq C_{12} \quad (28)$$

$$S_{3A}(k) + S_{3B}(k) \leq C_3 \quad (29)$$

These constraints represent capacity limitations on inventory, shipping for stream #1, shipping for stream #2, total inflow rate, and shipping/release for stream #3 from the Retailer, respectively.

Having investigated different facets of the demand network management problem with the two-node network, this insight can be applied to the six node problem. Figure 11 illustrates the information flow between the three MPC controllers and the six nodes of the demand network. For this problem, one monolithic MPC controller would become computationally prohibitive. A fully decentralized structure (one controller for each node), would require intervention by an outer layer to manage the constraints due to the interconnections of the nodes, the release rate capacities, and the shipping capacities. Due to the nature of the constraints, a compromise can be made between the two strategies. Three controllers are used. Each controller handles the flow and composition decisions for each echelon, passing forecasted information to the upstream node and receiving forecasted information from the downstream node. By resolving the information flows in this manner the solution is scalable, yet still feasible for managing the constraints.

The MPC controllers for the Warehouse and Retailer echelons of the network have to be modified slightly to account for the cross-shipment routes that can occur between echelons. The problem inputs now number eight, since there are two shipping lanes from each node and each lane can transport either or both products. To handle the extra degrees of freedom this brings to the problem, the penalties Q_u for the cross-shipment routes are purposefully set high, so these routes are not favored in the cost function. This penalty also allows for a unique solution to the MPC problem.

Experience with the actual performance of Assembly/Test nodes and the estimated lead times by facility personnel, suggest that the facility personnel traditionally provide themselves a lead time buffer of one day. So for example, if in reality the process takes 9 days to complete, a 10 day lead time estimate will be quoted to others in the organization. Thus, a 9 day actual/10 day estimate plant-model mismatch is adopted for both Assembly/Test facilities.

The Sales and Marketing personnel have been generally known to determine forecasts which are biased in an optimistic manner. As an example, the sales forecast for the next time period might be 11,000 units, when in fact the actual sales will be 10,000 units. To mimic this type of forecast bias, all demand forecasts passed to the Retailer level MPC controller are biased by +1000 units.

Lastly, products have been observed to follow demand patterns which may be correlated at times, and uncorrelated at other times. To mimic this type of behavior, the demand

patterns for Product A and Product B follow correlated, deterministic steps up until time 110. The remainder of the time, the demand patterns remain uncorrelated. This behavior can be observed in Figure 15.

The experiment is run with a value of 300 for all move suppression parameters $Q_{\Delta u}$, 0 for all penalties for values other than zero for direct shipment Q_u , and a penalty of 100 for all penalties for values other than zero for cross-shipments Q_u . Table 2 holds the prediction horizons N_p and control horizons N_u for all three MPC controllers. Safety stock is set at 5,000 units per product, per node. All inventory control error weights Q_e are left at 1. All entities use a “pecking order” dispatch rule. The orders of Warehouse 1, Retailer 1, and Customer 1 take precedence over the orders of the corresponding counterparts. Secondly, orders for Product A take precedence over orders for Product B. Figures 16 and 17 demonstrate the performance of this approach with plots of inventories, demands, shipments, and factory starts.

At time 1, the Retailer MPC controller adjusts orders and order forecasts to the upstream nodes to start bringing in more product, since the demand forecast is now 11,000 even though the actual amounts being demanded are 10,000. Because of the move suppression, the increases in order amounts are less than 1,000. The increase in orders is evident in the increase in direct and cross shipments from the Warehouse echelon. Soon the Retailer MPC echelon realizes the actual amount supplied to customers is *not* increasing as suggested by the forecast and the Retailer MPC adjusts to account for the forecast error and reduce inventory.

In the first few time units, inventories in the Factory echelon and Warehouse echelon are drained below their target levels. The Factory echelon MPC controller now observes the effects of the plant-model mismatch, since the effects of changes in the starts show up sooner than expected. The inventory levels of the Assembly/Test nodes fluctuate, but the fluctuation remains at reasonable levels. No backorders take place throughout the entire experiment. This is rather impressive, since the general rule of thumb practiced for this network requires a safety stock level equivalent to between two and four times the expected demand for the next time period (e.g. if tomorrow’s demand is expected to be 10,000 units, safety stock held today may range from 20,000 to 40,000 units).

Note that the cross-shipments in this example are used whenever there are rapid changes in the order/demand forecasts. The MPC controllers make use of the cross-shipments, since the costs associated with the move suppression weightings of the direct shipments be-

come comparatively large at these times. This may make sense not only from an optimization standpoint, but in a realistic setting it may also allow nodes to hedge against uncertainties or disturbances in transportation links or nodes connected by the direct shipment lanes.

5 Simulation Details

All simulation runs reported were performed on a Pentium IV 1.33 GHz processor with 128 MB of memory. The runs were on the order of minutes for the 2 node network, and 14 hours for the 6-node network for typical runs.

The simulations were constructed and run in a MATLAB 5.3/Simulink 3.0 environment. The work began by creating a library of Simulink blocks and accompanying S-functions for each of the representative types of nodes that might be used in the construction of a supply-demand network: factory, warehouse, retailer. Each library block was designed so that multiple blocks could be placed in the same Simulink model file with the material and information flows readily connected to form the network. Each node was designed with a Graphical User Interface (GUI) mask to allow for custom configuration of each node. Demand source blocks are constructed to read in deterministic or create stochastic demands. Blocks to determine forecasts from the stochastic demand source blocks and deterministic demand source blocks were included in the library. Finally, the MPC management blocks were constructed to allow for scalar and vector forecast inputs to accommodate real-time orders or real-time and future order forecasts. These blocks make use of the Optimization Toolbox for MATLAB and the accompanying GUI masks allow the user to readily modify all relevant optimization, constraint, and horizon length parameters. Every block constructed includes options for generating high quality plots of the final results, as well as diagnostic plots at intermediate time steps in the simulation.

The Matlab/Simulink simulation environment and the model library constructed allow for rapid redesign and configuration in a visual, object-oriented manner. The GUI masks not only serve for quick configuration, but variables can be directly put in place of static values to allow for scripting. Since MATLAB readily allows for a master script document to control the Simulink model file, a set of runs for a given Design of Experiments (DOE) can be readily scripted and executed with little need for user intervention. The myriad of subroutine toolboxes available for MATLAB allow for quick implementation of ideas to generate proof-of-concept results. Moreover, the powerful graphical tools of MATLAB can readily be accessed for plotting results from intermediate time steps in a simulation run, the summary

results from a given run, or full response surfaces from a given DOE.

Some of the drawback to the approach include the fact that MATLAB is commercial software and additional toolboxes are added at additional cost. The MATLAB language is interpretive which slows down the performance compared to compiled language environments, however the Compiler toolbox can be used with modifications to the custom code. The Optimization toolbox routines are not streamlined and robust for MPC application; additional work to improve constraint handling and indeterminate solutions would be beneficial. Lastly, the simulation time could benefit substantially from a parallel processing approach, which could solve the MPC blocks in parallel, rather than in a sequential manner. This is not an existing feature of the MATLAB/Simulink environment.

6 Conclusions

As demonstrated throughout this paper, MPC offers the ability to develop decision policies that stabilize inventory in supply chains despite imperfect information regarding the system dynamics and inaccurate demand forecasts. The move suppression term in the MPC cost function represents a penalty on order changes; incorporating this term in the MPC optimization problem leads to increased robustness which may not result from optimization schemes based on strictly cost-optimal approaches. This was demonstrated through simulations performed with the two-node supply chain. An MPC approach was shown to be flexible enough to manage a system approaching industrial size and configuration - the six node demand network proposed by Intel Corp.

A natural extension of this work is to incorporate the *demand management* side of the organization. Sales and marketing functions serve to manipulate and predict demand forecasts which may also be used to further maximize profits or minimize costs for the organization. By incorporating advertising and marketing models to predict sales volume as a function of advertising dollars spent, the supply chain management function and sales and marketing function may be better integrated.

7 Acknowledgment

Financial support from the National Science Foundation (NSF-DMI Award 0075682) and the Intel Research Council is gratefully acknowledged.

References

- Armbruster, D., R. Chidambaram, G. Godding, K. G. Kempf and I. Katzorke (2001). Modeling and analysis of decision flows in complex supply networks. In: *Proceedings of IV SIMPOI/POMS 2001*. Guaruja, Brazil. pp. 1106–1114.
- Bodington, C. E. and D. E. Shobrys (1999). Optimize the supply chain. In: *Advanced Process Control and Information Systems for the Process Industries* (L. A. Kane, Ed.). pp. 236–240. Gulf Publishing Company. Houston, TX.
- Flores, M. E., D. E. Rivera and V. Smith-Daniels (2000). Managing supply chains using model predictive control. In: *AIChE 2000 Annual Meeting, Nov. 12–17*. Los Angeles, CA. paper 262f.
- García, C. E., D. M. Prett and M. Morari (1989). Model predictive control: Theory and practice—a survey. *Automatica* **25**(3), 335–348.
- Kafoglis, C. C. (1999). Maximize competitive advantage with a supply chain vision. *Hydrocarbon Processing* pp. 47–50.
- Kempf, K., K. Knutson, B. Armbruster, P. Babu and B. Duarte (2001). Fast accurate simulation of physical flows in demand networks. In: *Advanced Simulation Technologies Conference*. The Society for Computer Simulation International. Seattle, WA. pp. 111–116.
- Lee, H. L., V. Padmanabhan and S. Whang (1997). The bullwhip effect in supply chains. *Sloan Management Review* pp. 93–102.
- Perea-Lopez, E., I. E. Grossman and B. E. Ydstie (2000). Application of mpc to decentralized dynamic models for supply chain management. In: *AIChE 2000 Annual Meeting, Nov. 12–17*. Los Angeles, CA. paper 262e.
- PriceWaterhouseCoopers (2000). Silicon 2000: Millennium semiconductor survey on supply chain management. <http://www.ebusiness.pwcglobal.com/external/gen-reg.nsf/semi?OpenForm&Seq=1>.
- Silver, E. A., D. F. Pyke and R. Peterson (1998). *Inventory management and production planning and scheduling*. Wiley. New York.
- Tzafestas, S., G. Kapsiotis and E. Kyriannakis (1997). Model-based predictive control for generalized production planning problems. *Computers in Industry* **34**(2), 201–210.

Table 1: Variable mapping for MPC controllers.

Process	Demand
Control Variable	Network Information
setpoints r	inventory targets of species A
outputs y	inventories of species A minus cumulative outstanding backorders
estimated outputs \hat{y}	forecasted inventories of species A
measured disturbances u_d	demand or orders for species A being placed at the node
estimated future measured disturbances \hat{u}_d	forecasted demand or orders for species A being placed at the node
estimated inputs \hat{u}	forecasted orders for species A being placed at the upstream node
inputs u	orders for species A being placed at the upstream node

Table 2: Prediction and control horizons used in six node controllers.

Parameter	Echelon #1	Echelon #2	Echelon #3
N_p	49	41	33
N_u	42	34	12

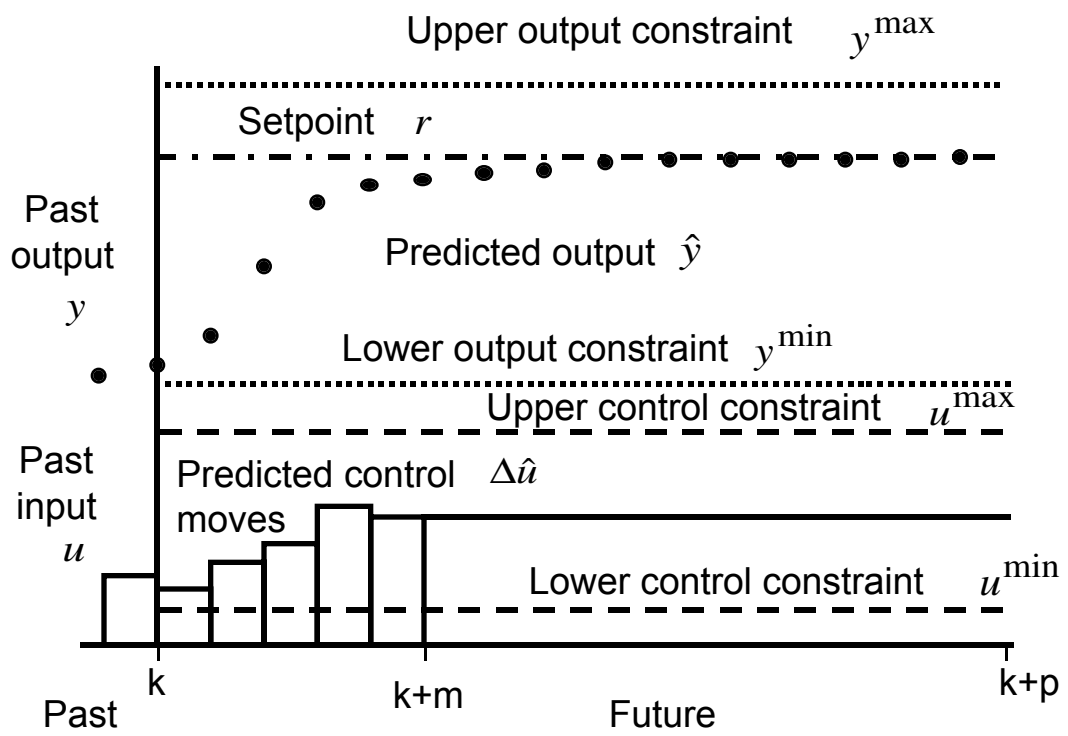


Figure 1: MPC receding horizon philosophy.

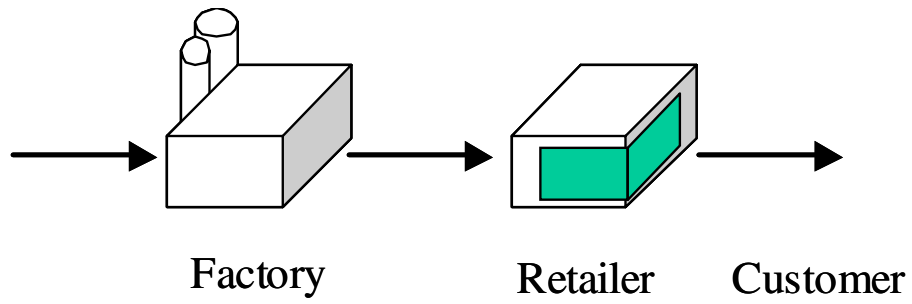


Figure 2: Two-node network material flows.

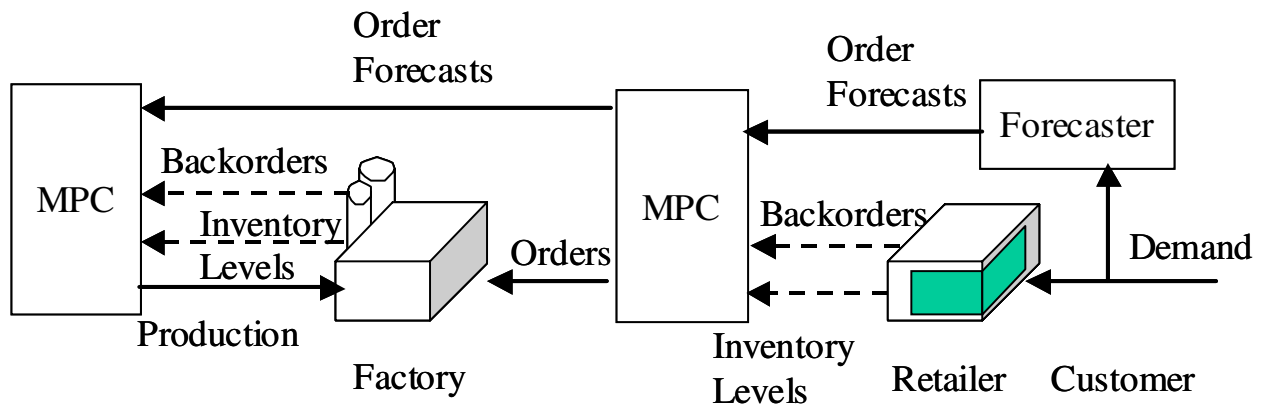


Figure 3: Two-node network MPC information flows.

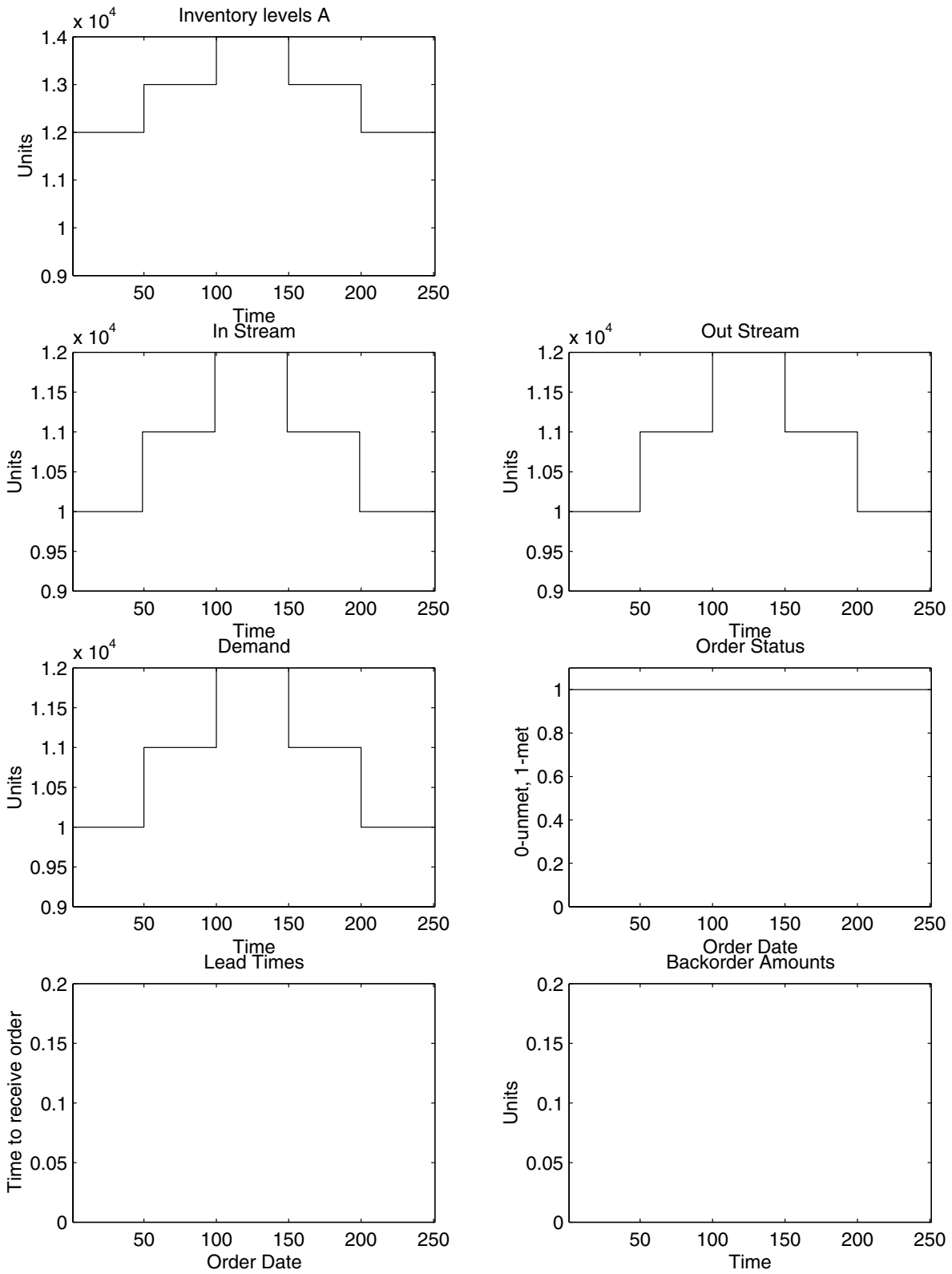


Figure 4: Retailer Responses and Metrics, Two-Node Example, No Move Suppression, No Plant-Model Mismatch

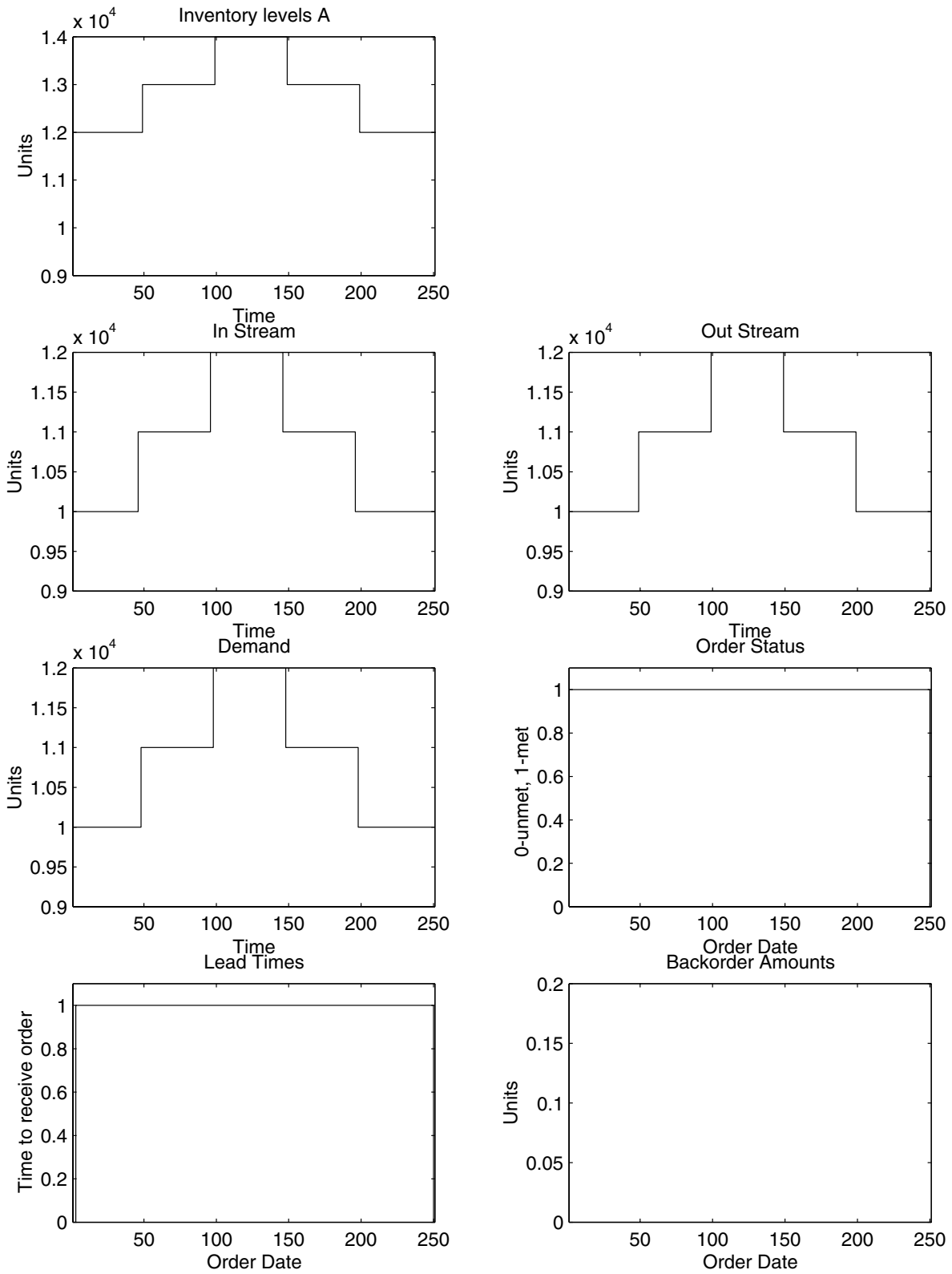


Figure 5: Factory Responses and Metrics, Two-Node Example, No Move Suppression, No Plant-Model Mismatch

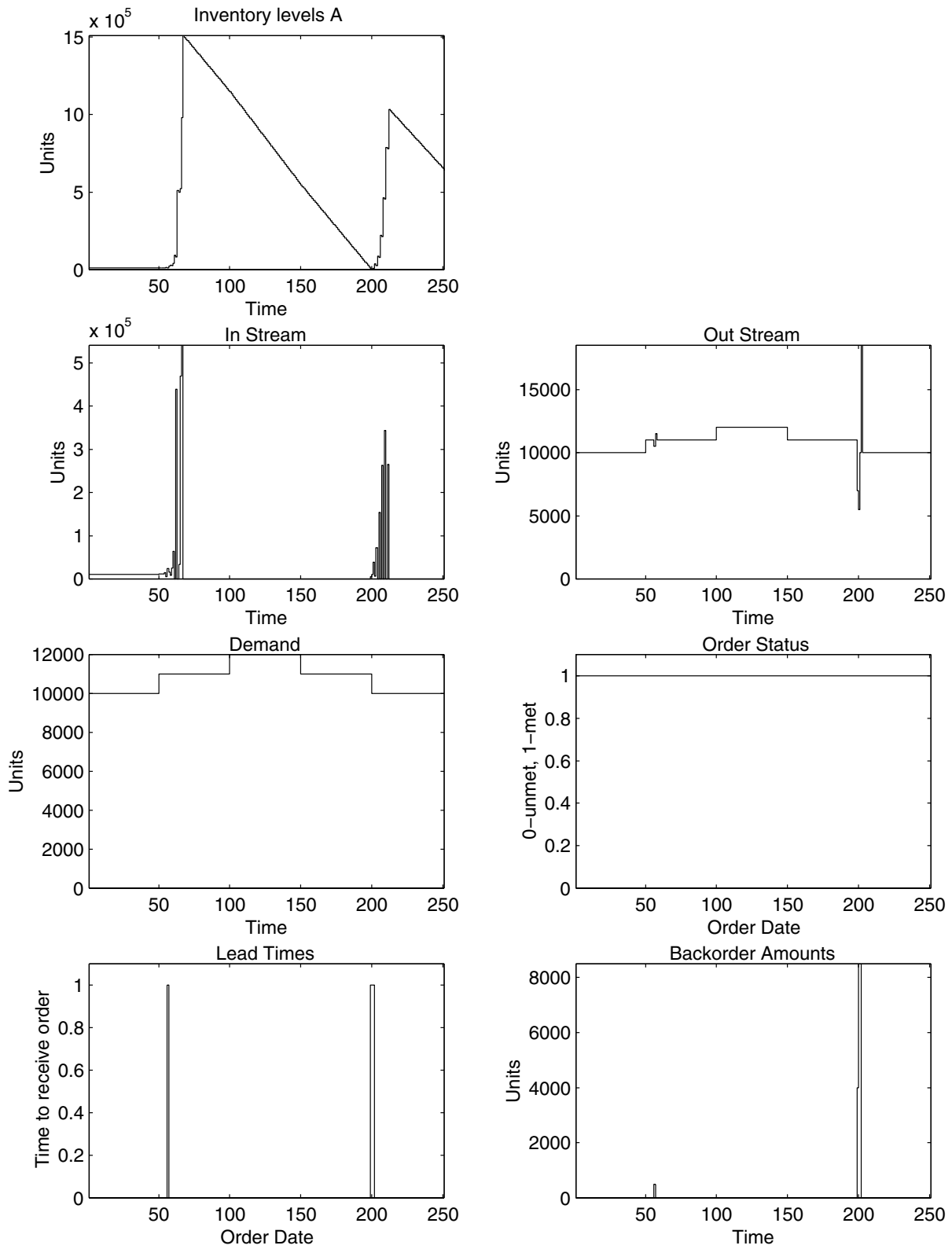


Figure 6: Retailer Responses and Metrics, Two-Node Example, No Move Suppression, With Plant-Model Mismatch.

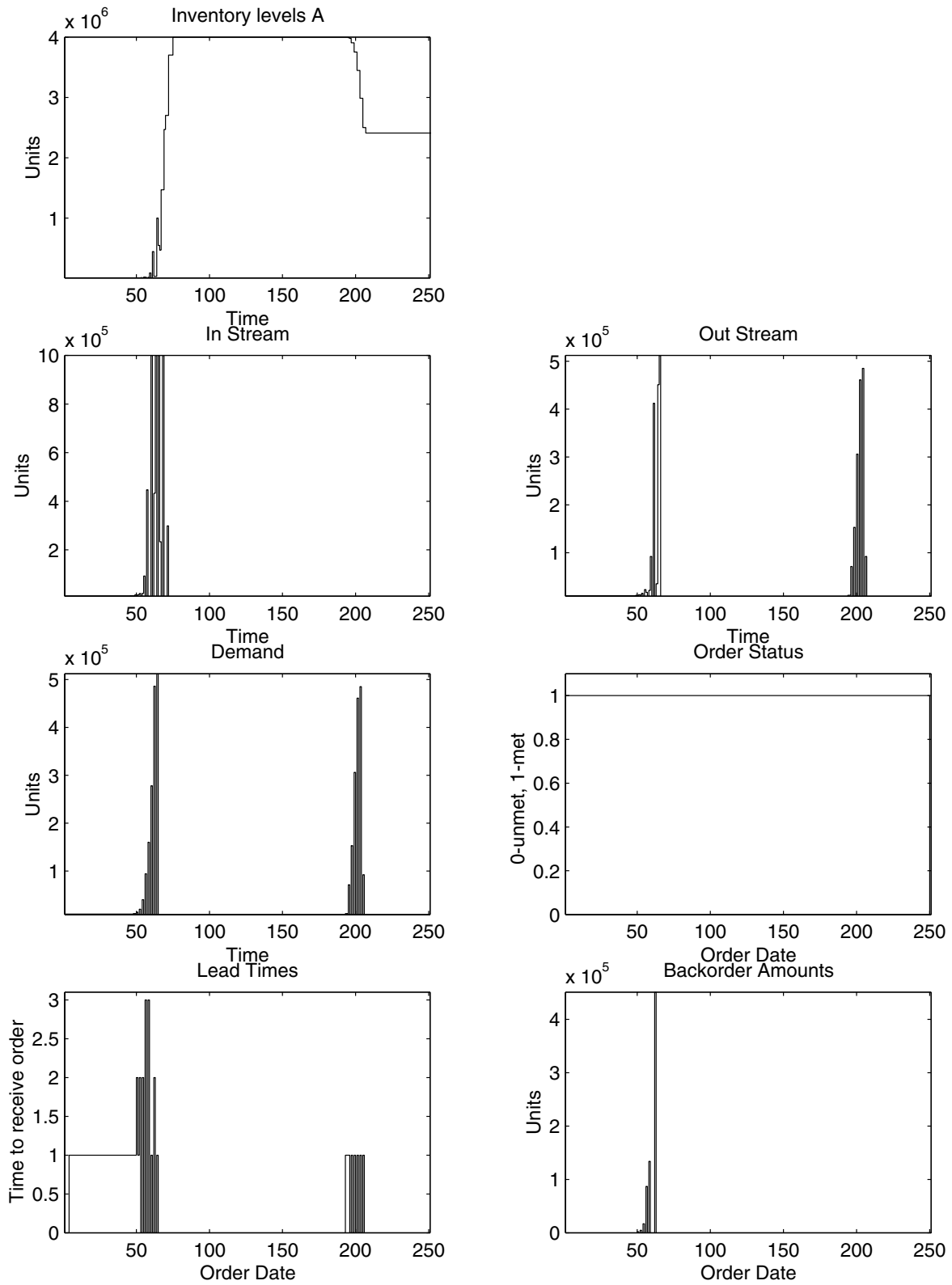


Figure 7: Factory Responses and Metrics, Two-Node Example, No Move Suppression. With Plant-Model Mismatch.

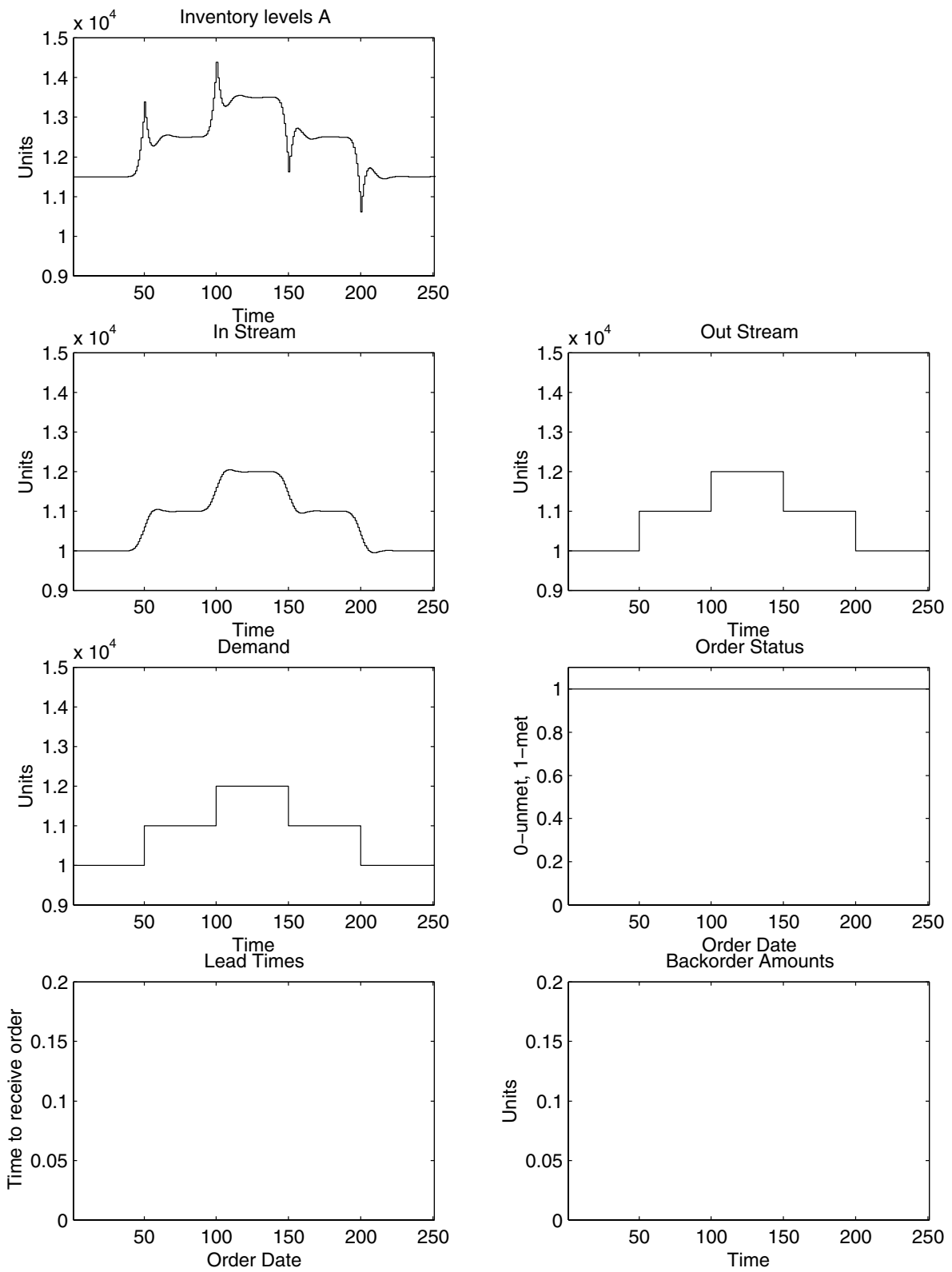


Figure 8: Retailer Responses and Metrics, Two-Node Example, With Move Suppression and Plant-Model Mismatch

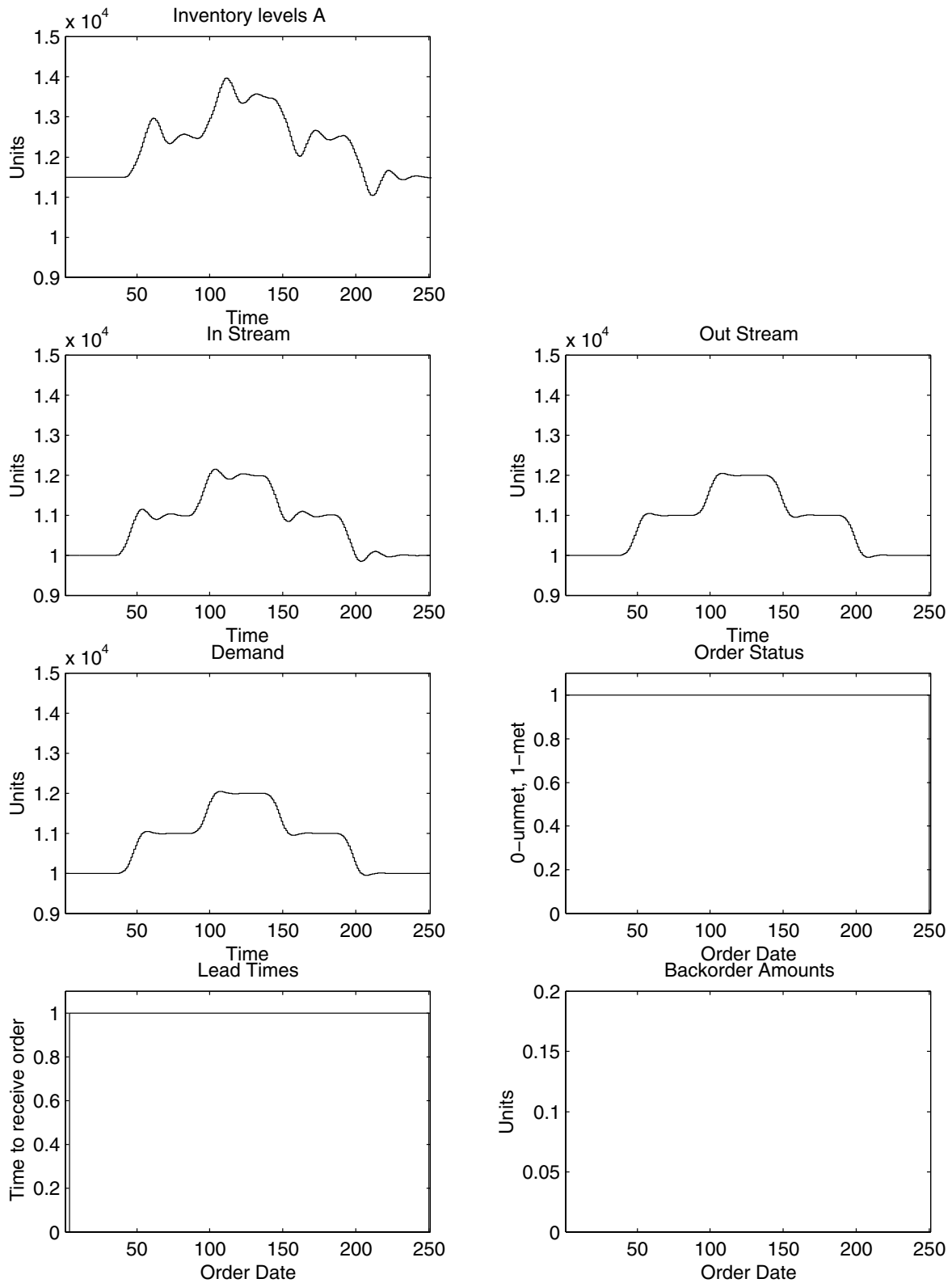


Figure 9: Factory Responses and Metrics, Two-Node Example, With Move Suppression and Plant-Model Mismatch

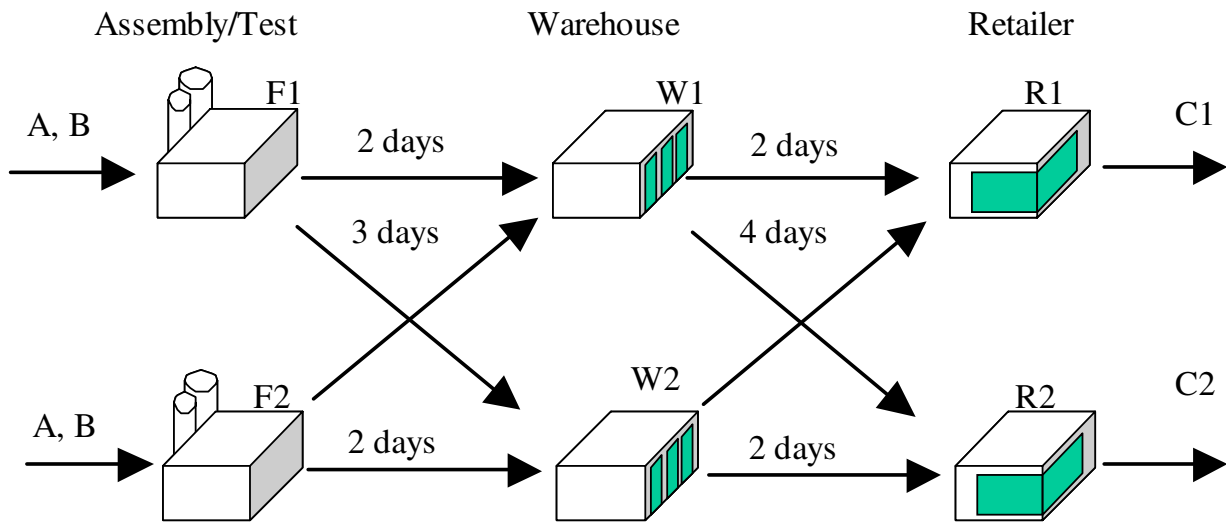


Figure 10: Six node Intel network material flow.

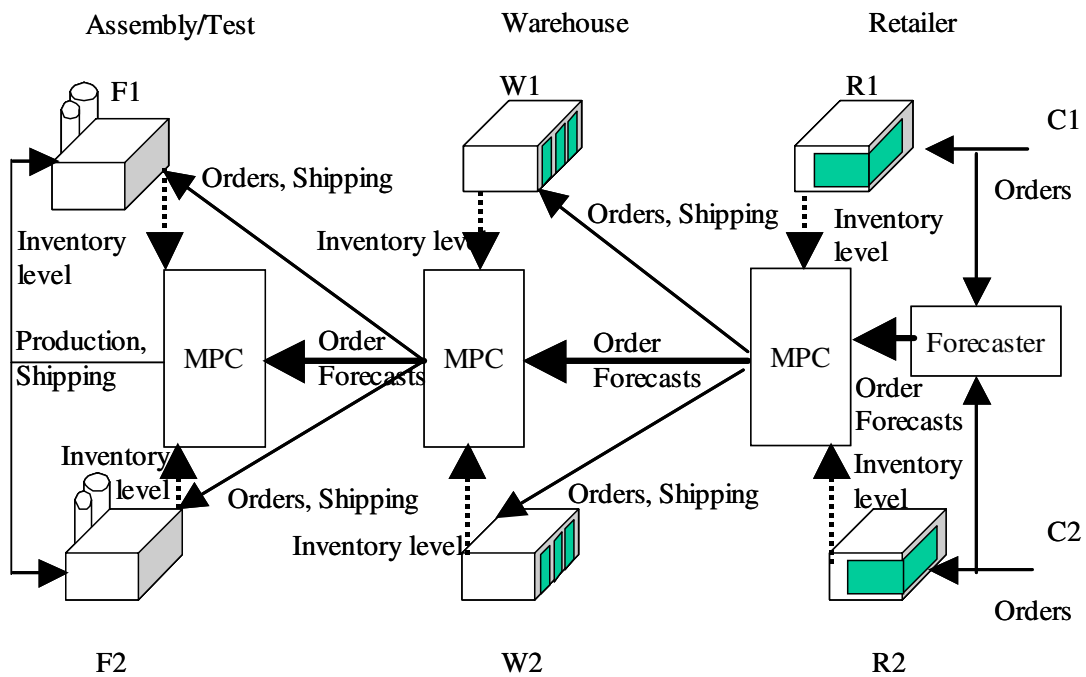


Figure 11: Six node Intel network information flow.

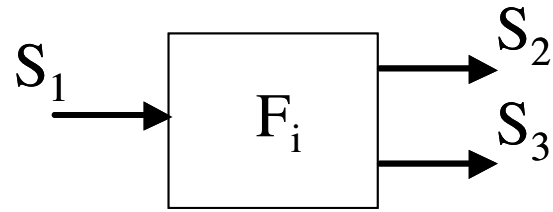


Figure 12: Single inflow, two outflow Factory material streams.

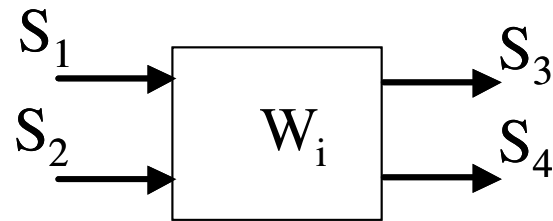


Figure 13: Two inflow, two outflow warehouse material streams.

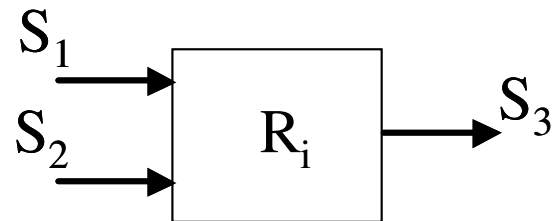


Figure 14: Two inflow, single outflow retailer material streams.

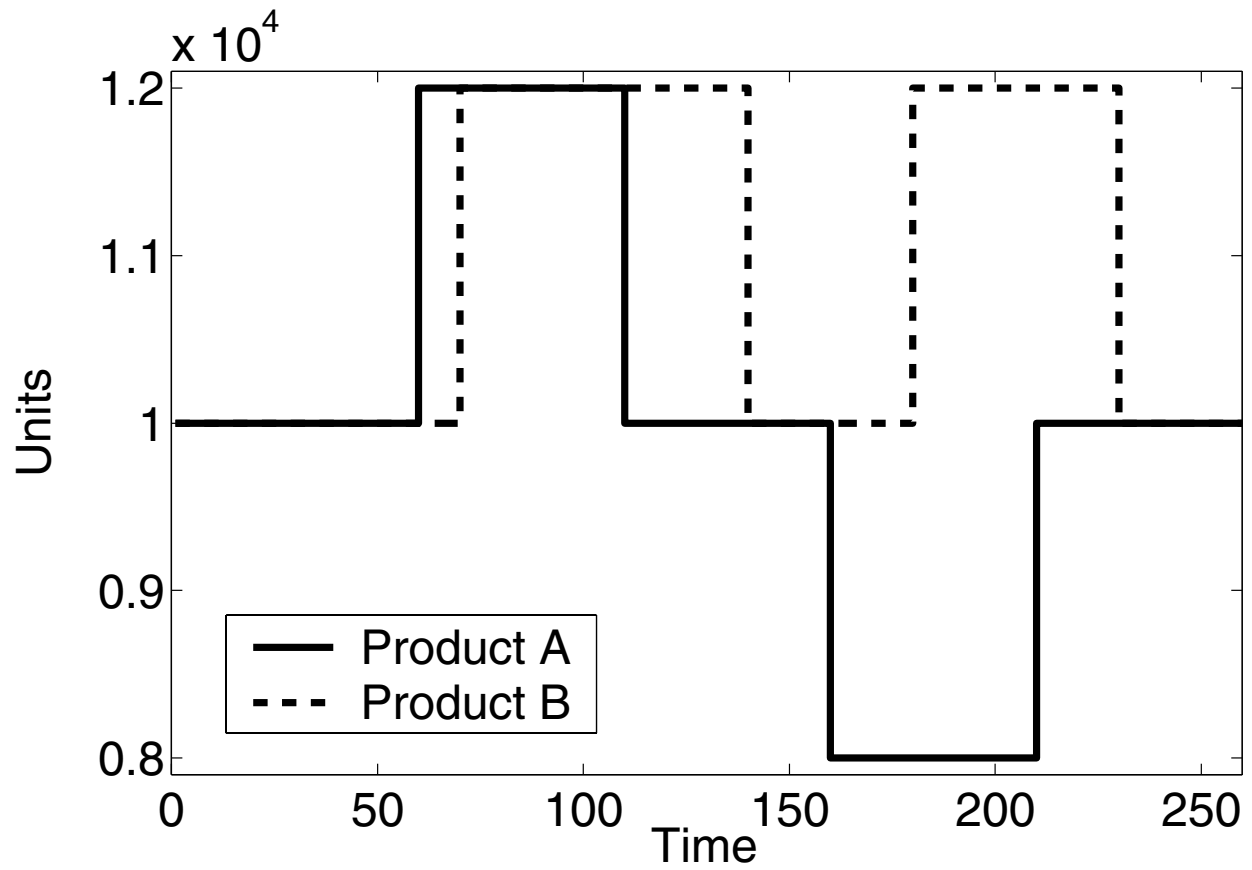


Figure 15: Demand profiles for products A and B, six node Intel network.

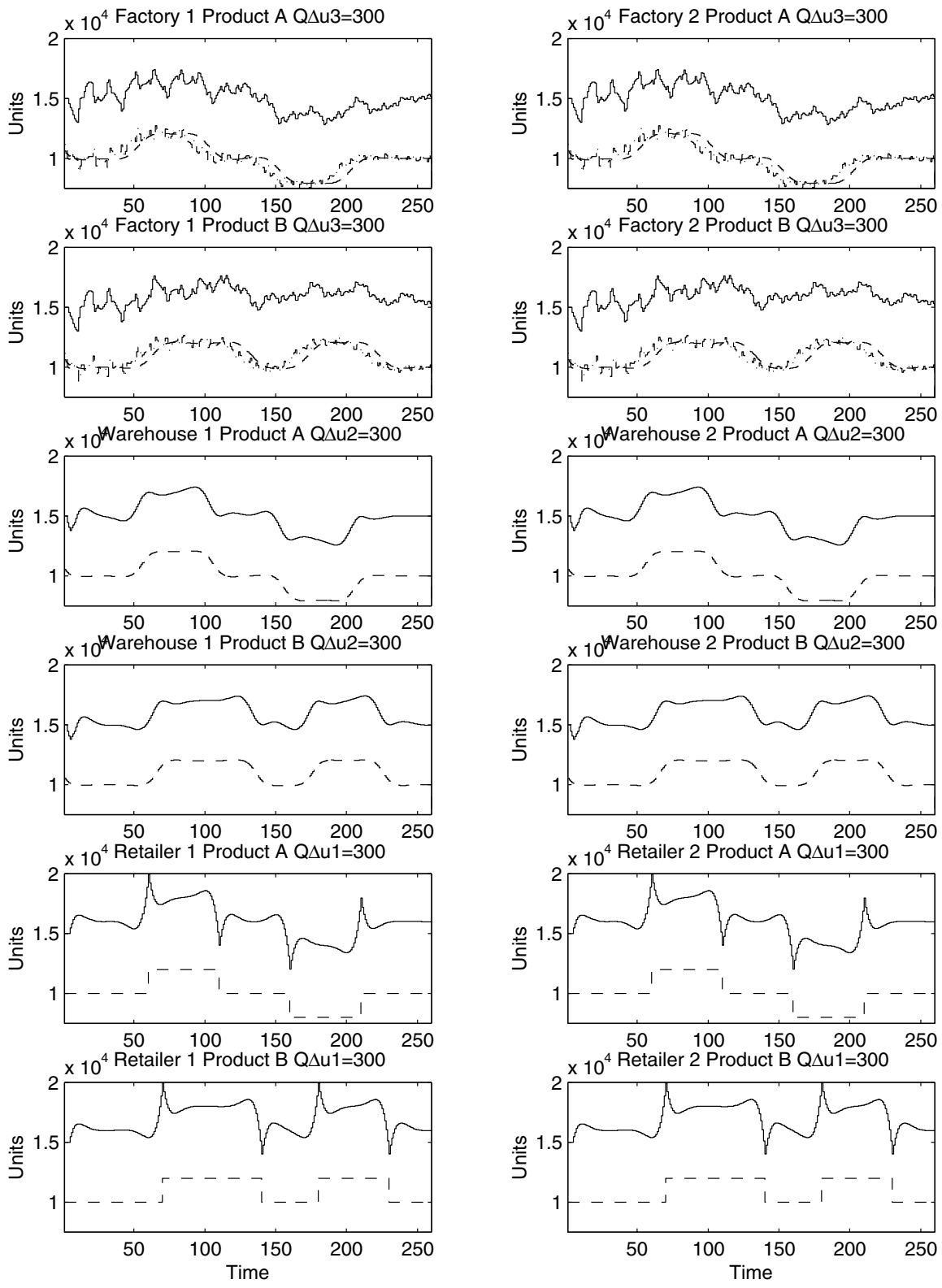


Figure 16: Inventories (solid) and demand (dashed) by facility and product; Assembly/Test plots include starts (dash-dot).

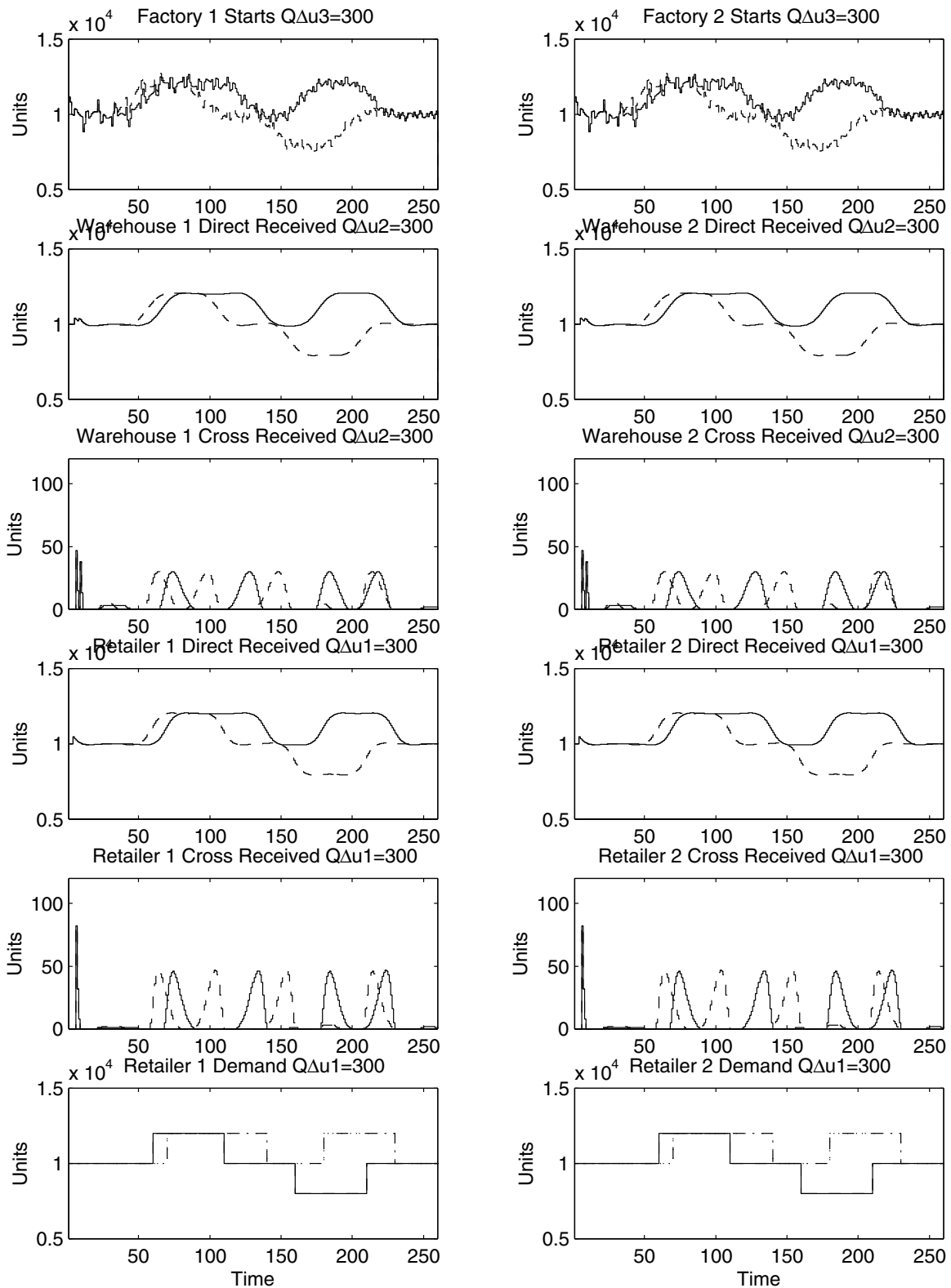


Figure 17: Material flows Product A (dashed) and Product B (solid); Demand: Product A (dotted) and Product B (dash-dot) and corresponding customer receipts Product A (dashed) and Product B (solid).