Int. J. Production Economics 133 (2011) 439–450

Contents lists available at ScienceDirect

Int. J. Production Economics

journal homepage: www.elsevier.com/locate/ijpe

An iterative approach to item-level tactical production and inventory planning

Feng Tian [a], Sean P. Willems [b],*, Karl G. Kempf [c]

[a] College of Business & Public Administration, Governors State University, University Park, IL 60484, USA
[b] School of Management, Boston University, Boston, MA 02215, USA
[c] Decision Technologies Group, Intel Corporation, Chandler, AZ 85044, USA

ARTICLE INFO

Available online 21 August 2010

Keywords:
Inventory
Production
Hierarchical planning
Supply chain management

ABSTRACT

In this paper, we propose an iterative approach to jointly solve the problems of tactical safety stock placement and tactical production planning. These problems have traditionally been solved in isolation, even though both problems operate in the same decision making space and the outputs of one naturally serve as the inputs to the other. For simple supply chain network structures, two stages and one or many products, we provide sufficient conditions to guarantee the iteration algorithm's termination. Through examples, we show how the algorithm works and prove its applicability on a realistic industrial-scale problem.

1. Introduction

The strategic-tactical-operational framework developed by Anthony (1965) is ingrained in the operations-management lexicon. In a classic manifestation of this framework, determining how much production capacity to have and where to have it are strategic decisions, determining how to allocate production capacity to product families is a tactical decision, and producing an item-level production schedule is an operational decision. Not only do these decisions operate at different frequencies (i.e., a company does not evaluate its capacity acquisition strategy on a weekly basis), they also operate with different levels of scope and granularity. For example, setting the production schedule for the next day requires a precise statement of every item at each location while a biannual capacity acquisition study aggregates items beyond the product family to product types that represent major market segments by manufacturing origin.

The literature that addresses supply chain aspects of Anthony's (1965) framework is vast. Even with attention limited in scope to production–inventory problems, researchers must make hard choices to limit the scope and granularity of their models. We will restrict our attention to the large subset of the literature that models the interaction of tactical production planning with a number of other production–inventory problems. This subset can be divided into approaches that break the problems into a hierarchy of decisions and approaches that solve a monolithic unified model.

Hax and Meal (1975) propose a hierarchical solution procedure that spans capacity planning through detailed scheduling. The hierarchical planning approach relies on aggregating data for higher-level decisions and having the optimal decisions from each higher-level model serve as a constraint for the next-lower model in the hierarchy.

Bitran et al. (1981) solves the production allocation and item-level scheduling problems for a multiple-item single-echelon system. Family and item disaggregation subsystems are both represented by means of knapsack problems. Bitran et al. (1982) expands this approach to a two-echelon system. While the application of the framework to the two-echelon setting is conceptually straightforward, problem-specific knowledge must be exercised to determine the appropriate aggregation structure. Specifically, the solution to the aggregate top-level model does not ensure the existence of a feasible disaggregation for the item-level problem. To ensure feasibility, it is necessary to either add sufficient conditions at the aggregated planning level (Gfreer and Zapfel, 1995), or apply an iterative scheme in the hierarchical structure (Jornsten and Leisten, 1995).

Billington et al. (1983) and Bradley and Arntzen (1999) are representative of monolithic approaches. Billington et al. (1983) simultaneously determine the stage lead-times and the item-level production plan. To ease the computational burden, product structure compression is employed to collapse stages that do not influence the resulting solution. Compression works well in cases where only a few resources are constrained. Bradley and Artzen develop a monolithic mathematical program to address strategic capacity acquisition, tactical production planning, and operational

* Corresponding author. Tel.: +1 617 353 2287; fax: +1 617 353 4098.
E-mail addresses: f-tian@govst.edu (F. Tian), willems@bu.edu (S.P. Willems), karl.g.kempf@intel.com (K.G. Kempf).

0925-5273/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.ijpe.2010.07.011

scheduling. The decision variables are capacity investments, raw material purchases, and the production schedule. The objective function maximizes return on assets. For this class of modeling, demand is deterministic so inventory represents time-phased imbalances between production and demand. Neither model explicitly considers setting safety stock levels, although exogenously determined safety stocks could be incorporated as constraints. Spitter et al. (2005) and Fandel and Stammen-Hegene (2006) are indicative of the advances in this line of modeling. Spitter et al. (2005) is similar in spirit to Billington et al. (1983) but allows capacity for an order to be allocated any time during the leadtime. Fandel and Stammen-Hegene (2006) improve the production plan by considering general lot sizing and scheduling across multiple machines.

Byrne and Bakir (1999) adopts a hybrid simulation-analytical approach to protect a production plan against operational sources of variability. A linear program generates an optimal production plan and then a simulation verifies the feasibility of the production levels. The solution procedure adjusts capacity between successive iterations until capacity constraints are satisfied. Kim and Kim (2001) propose an extended linear programming model and include more information during iterations for a similar hybrid approach. Numerical analysis shows that their approach can find a better solution in fewer iterations than Byrne and Bakir (1999).

De Kok and Fransoo (2003) present a problem to coordinate the release of materials and resources across a multi-echelon network. They refer to this as the supply chain operations problem (SCOP) and present two solution methods. One approach formulates a linear program (LP) that assumes starting inventories are zero. They then conduct a simulation to determine the appropriate safety stock levels to support the plan and then rerun the linear program. The second approach assumes synchronized base stock (SBS) policies and analytically computes the resulting base stock levels. For a set of test problems, the SBS approach outperforms the LP approach.

While Byrne and Bakir (1999), Kim and Kim (2001), and De Kok and Fransoo (2003) are notable exceptions, the majority of the literature does not focus on the determination of safety stock inventory. Hax and Candea (1984) is indicative of the more standard approach where tactical production planning problems and detailed operational scheduling are clearly laid out with established linkages but safety stock is determined exogenously and at best serves as a constraint to production planning and scheduling models. Maxwell et al. (1983) explicitly recognize this problem and propose a three-phase modeling framework to recognize the relationship between lead time, capacity, lot sizes, and safety stock. They propose phase one as creation of the master production plan, phase two as planning for uncertainty, and phase three as real time resource allocation. Safety stock setting is the key problem in phase two since it provides protection for the created master production plan.

Our work takes a different philosophical perspective, conceptually outlined in Kempf (2004). In effect, this research approach is iteratively solving the phase one and phase two problems outlined by Maxwell et al. (1983). We propose a procedure to iteratively solve two optimization problems: the tactical problem of production planning and the tactical problem of safety stock placement. The value of this approach is it integrates two well-developed research streams, allowing the joint solution to overcome the limitations of each individual approach while simultaneously preserving the optimality, within constraints, of each individual solution.

Each research stream has made significant advancements in isolation. The area of tactical safety stock optimization is summarized in Graves and Willems (2003). In brief, tactical safety stock optimization seeks to optimize inventory levels across the multi-echelon supply chain. To accomplish this, these approaches must make additional assumptions and settle for heuristic solutions relative to the exact solutions that can be derived when the problem scope is limited to single-stage inventory problems. On the positive side, papers including Billington et al. (2004) and Bossert and Willems (2007) document that these models have been successfully applied in practice.

The area of tactical production planning has a vast associated literature. Beyond the articles referenced earlier, summary overviews are provided by Shapiro (1993) and Fleischmann and Meyr (2003). For our purposes, we are concerned with linear-programming based approaches that minimize the sum of production cost, inventory cost, and penalty cost over a tactical horizon that often measures 12–24 weeks. A specific example of a relevant formulation is presented in Bean et al. (2005).

The rest of the paper is arranged as follows: Section 2 describes the iteration algorithm. Section 3 establishes termination criteria for a two-stage single-echelon network producing either one or N products. Section 4 shows the implementation of the algorithm for a realistic industrial-scale planning problem. Section 5 concludes and describes future research.

2. An iterative algorithm

The supply chain is modeled at the SKU-location level as a graph with node set N and arc set A. Every stage corresponds to a processing function. Examples include transportation from one location to another, manufacturing, and placement in a warehouse to satisfy demand. Arcs denote the precedence relationship between stages. We will find it useful to partition N into three disjoint sets: N_S, N_I, N_D. The set of supply stages, N_S, have no incoming arcs and demand stages, N_D, have no outgoing arcs. The set of intermediate stages, N_I, each have at least one incoming arc and one outgoing arc. Inventory will only be held at the end of stages in N_I, after their processing activity has completed. N_S and N_D can be thought of as dummy stages which are required to populate data for the model.

We model a production system with stationary demand operating under a periodic review policy. Demand must be filled in the period it arrives, otherwise it is lost. The ending inventory for any internal stage j in period t is found by the balance equation

$$I_{j,t} = I_{j,t-1} + P_{j,t-T_j} - \sum_{k:(j,k) \in A} S_{j,k,t} \tag{1}$$

where $P_{j,t}$ is the quantity started at stage j in period t, $S_{j,k,t}$ is the quantity shipped from stage j to stage k in period t. T_j is stage j's processing lead time. The ending inventory for stage j is the starting inventory plus the units that started at stage j. T_j time periods ago minus the units stage j ships out.

Production minimums are introduced to enforce a practical policy that if a stage is designed to produce a certain product, it is always required to make at least a minimum amount of this product every period. This is a common occurrence in many industries ensuring the stage maintains its capability to produce a product according to the designed tolerances. Planners usually impose a minimum production amount for each product assigned to each plant (Intel, 2005 and Intel, 2006).

The tactical production planning problem is formulated as a linear program **P1**

$$\text{P1} \quad \max \sum_{t=1}^{T} \left[\sum_{j \in N_D} r_j \sum_{i:(i,j) \in A} S_{i,j,t} - \sum_{j \in N_I} (c_j P_{j,t} + h_j I_{j,t} + e_j o_{j,t} + e_j u_{j,t}) \right] \tag{2a}$$

$$s.t.\quad I_{j,t}=I_{j,t-1}+P_{j,t-T_j}-\sum\nolimits_{k:(j,k)\in A}S_{j,k,t}\quad \forall j\in N_I;t=1,\ldots,T \tag{2b}$$

$$\sum\nolimits_{j:(j,k)\in A}S_{j,k,t}\le D_{k,t}\quad \forall k\in N_D;t=1,\ldots,T \tag{2c}$$

$$P_{j,t}\le \sum\nolimits_{i:(i,j)\in A}S_{i,j,t}\quad \forall j\in N_I;t=1,\ldots,T \tag{2d}$$

$$I_{j,t}+u_{j,t}-o_{j,t}=B_{j,t}\quad \forall j\in N_I;t=1,\ldots,T \tag{2e}$$

$$\sum\nolimits_{j\in\omega}\varphi_{\omega,j}P_{j,t}\le C_\omega\quad \forall\omega\in\Omega;t=1,\ldots,T \tag{2f}$$

$$P_{j,t}\ge P_{\min}\quad \forall j\in N_I;t=1,\ldots,T \tag{2g}$$

$$P_{j,t},S_{i,j,t},I_{j,t},o_{j,t},u_{j,t}\ge 0\quad \forall j\in N_I;(i,j)\in A;t=1,\ldots,T \tag{2h}$$

The objective function maximizes profit equal to sales revenue net manufacturing cost, inventory holding cost, and a penalty cost for deviating from each period's safety stock target. The per unit revenue from demand at stage j is r_j. The per unit production and holding costs are c_j and h_j, respectively. The cost per unit per time period for any inventory deviation from the safety stock target incurs a cost e_j. T is the planning horizon. The decision variables are the $P_{j,t}$ and $S_{i,j,t}$. $D_{j,t}$ denotes the forecast for customer demand at demand stage j in period t.

(2b) represents the inventory balance constraints, previously defined in (1). (2c) forces the shipments to a stage in a period to not exceed the stage's demand for the period. (2d) constrains the production at any intermediate stage in a period by the materials it received from upstream stages in the period. For each period and stage, (2e) measures any inventory in excess of the safety stock target $B_{j,t}$ with $o_{j,t}$ and any deficit by $u_{j,t}$. Ω denotes the set of capacity constraints defined in (2f) where each $\omega\in\Omega$ defines a collection of stages where each unit of production from stage j contributes $\varphi_{\omega,j}$ to the collection's total capacity constraint of C_ω. (2g) enforces the minimum production for each product, which is denoted as P_{min}.

To streamline the presentation of **P1**, most of the constants have been presented as simply as possible. In practice, these constants would vary by time and in the case of the penalty cost, e_j, would split the overage and underage penalties into separate constants for $o_{j,t}$ and $u_{j,t}$; inventory deficits are often penalized more than inventory excess. Furthermore, we have assumed the simplest goes-into structure in this problem formulation. Namely, each unit of an upstream stage's shipment corresponds to one unit of the downstream stage's production. A more complex goes-into structure is conceptually straightforward to incorporate, but at the cost of significant additional notation. Since our contribution is not a new formulation for **P1**, we have omitted these details.

In **P1**, the safety stock targets $B_{j,t}$ are an input. A separate tactical safety stock optimization problem, **P2**, will optimally set the $B_{j,t}$. As noted in Graves and Willems (2003), there are two general approaches to optimize tactical safety stock targets: the guaranteed service (GS) model and the stochastic service (SS) model. Both models employ heuristics. The GS model assumes that safety stock is only designed to meet demand within a certain bound, and countermeasures like expediting will be utilized when demand exceeds the bound. The SS model assumes safety stock is the only countermeasure to satisfy demand uncertainty. The implication is that the system will operate the same regardless of the demand rate and that when a stock out situation occurs, the system does not change behavior. Both the GS and SS models employ strong assumptions, and represent extremes where the reality is somewhere in the middle, but for our purposes we assume the GS model is used to solve **P2**. We adopt the GS model for two reasons. First, its assumptions are quite consistent with the tactical production planning problem **P1**. While the GS model does not explicitly address what happens when a stock out occurs, neither does **P1**. Second, the guaranteed service times from the GS model are a natural fit for the planned lead times in **P1**. Thus, the majority of the inputs between the two models are the same.

As with **P1**, **P2** models the supply chain at the SKU-location level with node set N and arc set A. Since demand is assumed stationary, stage j's demand variability per period is denoted σ_j^2. We assume a constant safety factor z is maintained at all stages. The expected safety stock at a stage is a function of the stage's net replenishment time, defined as the maximum incoming service time to stage j, SI_j, plus stage j's processing lead time, T_j, minus the outgoing service time, R_j, that stage j quotes to its downstream-adjacent stages. Mathematically, the expected safety stock at stage j is $z\sigma_j\sqrt{SI_j+T_j-R_j}$.

The tactical safety stock planning problem is formulated as a nonlinear program **P2**

$$\text{P2}\quad \min\sum_{j=1}^{N}h_j z\sigma_j\sqrt{SI_j+T_j-R_j} \tag{3a}$$

$$s.t.\quad R_j-SI_j\le T_j\quad \forall j\in N \tag{3b}$$

$$R_i-SI_j\le 0\quad \forall(i,j)\in A \tag{3c}$$

$$R_j\le E_j\quad \forall j\in N_D \tag{3d}$$

$$SI_j=0\quad \forall j\in N_S \tag{3e}$$

$$R_j,SI_j\ge 0 \text{ and integer}\quad \forall j\in N \tag{3f}$$

The objective function minimizes total safety stock cost. (3b) ensures a stage's outgoing service time does not exceed its incoming service time and processing time. (3c) enforces the incoming service time to stage j to be no less than the maximum outgoing service times quoted by nodes directly supplying j. The outgoing service time to end customers are bounded by the E_js in (3d) while (3e) assumes that supply stages have zero incoming service time. (3f) imposes integrality and nonnegativity constraints where appropriate. A more general formulation of **P2** can be found in Humair and Willems (in press). As in the case of **P1**, since the innovation is in the iteration approach and not the formulation of **P2**, we omit the most general formulation possible for **P2**.

Ensuring consistency between the solutions to **P1** and **P2** requires reconciling the inputs that differ between the two models. In particular, for **P1,** demand originates at stages in N_D, the arc set dictates which nodes in N_S and N_I can be used to satisfy that demand, and the supply plan $S_{i,j,t}$ dictates in what amount each stage satisfies these demands. Similar to the forecasted demand in **P1**, in the GS model, **P2**, the demand variability, expressed as the standard deviation of forecast error (SDFE), is specified only at the stages in N_D. However, it is not immediately obvious how to properly allocate the SDFE to stages in N_I. Safety stock inventory is used to protect against the variability in demand from downstream stages. Therefore, the SDFE is calculated at each stage in N_I based on how **P1** allocates supply across the network.

For a stage j supplying stage k such that $(j,k)\in A$, we define the split ratio α_{jk} as,

$$\alpha_{jk}=\frac{\sum_{t=1}^{T}S_{j,k,t}}{\sum_{t=1}^{T}\sum_{i:(i,k)\in A}S_{i,k,t}} \tag{4}$$

Given a split ratio, σ_j at stage $j \in N_I$ is found by

$$\sigma_j = \sqrt{\sum_{k:(j,k) \in A} \alpha_{jk}\sigma_k^2} \quad (5)$$

(5) codifies the planning heuristic, witnessed in Intel (2005) and Intel (2006), that when a stage is planned to meet a certain percentage of its customer's demand, the stage needs to provide safety stock sufficient to protect the same percentage of the customer's variability.

The derivation of (4) and (5) lie at the heart of an iterative approach to jointly solving the planning and inventory problems posed by **P1** and **P2**. Let $\mathbf{S}^n$ denote a $|N_S+N_I| \times |N_I+N_S| \times T$ matrix with element S^n_{jkt} denoting the shipment from j to $k$in period t in iteration n. Similarly, $\boldsymbol{\alpha}^n$ is a $|N_S+N_I| \times |N_I+N_D|$ matrix with element α^n_{jk} denoting the percentage of demand from k satisfied by j and $\mathbf{B}^n$ is a $|N_I| \times T$ matrix where B^n_{jt} denotes the safety stock target for j in period k for iteration n. Fig. 1 presents a flow chart of the iteration algorithm hereafter referred to as PRODINV.

The initialization of PRODINV begins with an estimate of the split ratio; a reasonable starting estimate would be the ratio of the capacities of downstream-adjacent stages. This serves as input to **P2** which produces $\mathbf{B}^n$ as output. **P1** then employs $\mathbf{B}^n$ as constraints and outputs $\mathbf{S}^n$. $\mathbf{S}'$, comprised of a linear combination of $\mathbf{S}^{n-1}$ and $\mathbf{S}^n$, is used to calculate $\boldsymbol{\alpha}^{n+1}$ which serves as the input to populate each stage's standard deviation in the next iteration of **P2**. The algorithm terminates when $\mathbf{S}^n = \mathbf{S}^{n-1}$ within a specified tolerance; a typical tolerance is 0.001 or less.

There are three noteworthy facets of PRODINV. First, ρ dictates how S' adjusts from one iteration to the next. In iteration n, with $\rho=1$, the split ratio depends solely on S^n, the current iteration's shipment between stages. This implies the safety stock target in iteration $n+1$ is entirely based on how stages in iteration n satisfy demands. Choosing $\rho<1$ reserves some capacity for the corresponding changes in safety stock inventory caused by the new split ratio and $\rho>1$ prompts more aggressive changes.

Second, it should not be obvious that PRODINV will terminate. In fact, we will show for specific system parameters that an upper limit of ρ will be a sufficient condition for PRODINV to terminate.

Third, satisfying the termination criterion at iteration n only guarantees that the safety stock targets $\mathbf{B}^n$ are consistent with the shipment plan $\mathbf{S}^n$. While this is a theoretically interesting result, it does not accomplish what a business user wants to understand. In particular, a business user needs to understand whether $\mathrm{I}^n = \mathbf{B}^n$; i.e., whether the shipment plan from **P1** produces inventory levels, $\mathbf{I}^n$, that can achieve the safety stock targets, $\mathbf{B}^n$, from **P2**. We say PRODINV converges when at termination $\mathbf{I}^n = \mathbf{B}^n$, within a specified tolerance. While we can prove that termination is guaranteed if ρ satisfies certain conditions, convergence will only occur if the system has sufficient capacity to meet the system's demand and inventory requirements.

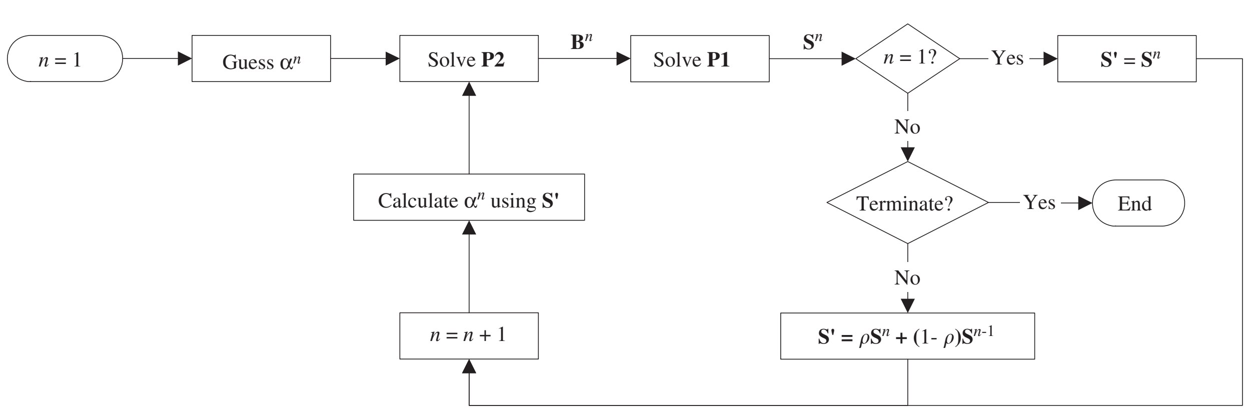

Fig. 1. Iteration algorithm flow chart.

3. Analytical results for two-stage systems

By restricting the supply chain to two stages in a single echelon supplying one or N products, we can obtain analytical results that provide insight into the mechanics of PRODINV.

3.1. Two-stage single-product problem

Fig. 2 considers a simple two-stage single-product system. Raw material is procured from a single supplier and production can occur at one of two manufacturing facilities which both satisfy demand. Demand for the product is i.i.d. Normal.

We assume A and B have identical processing times, and both have a limited capacity. If there is no cost difference between them, then the problem simplifies to a single facility problem. Without loss of generality, we assume that both the production cost and inventory penalty cost at A are lower than B. This is often realized in practice when similar factories are located in different countries.

Since there are only two stages in N_I, we can simplify the notation for the split ratio and let α denote the split ratio from A to demand and denote B's split ratio as $1-\alpha$. It is straightforward to see that if the total safety stock required is SS then A's allocation will be αSS.

Assume the demand per period is normally distributed with mean 4000 and standard deviation 800. The selling price is \$10 per unit and service level is 95%. Stage A has capacity of 3000 units, production time of 2 days, production cost of \$1 per unit, holding cost rate of \$0.35 per unit per year, and a missed safety stock target of \$0.385 per unit per period. Stage B has capacity of 3500 units, production time of 2 days, production cost of \$2 per unit, holding cost rate of \$0.70 per unit per year, and a missed safety stock target of \$0.77 per unit per period. We assume that $\varphi_{w,j}=1$ for all w and j. Table 1 shows the iteration detail when $\rho=1$.

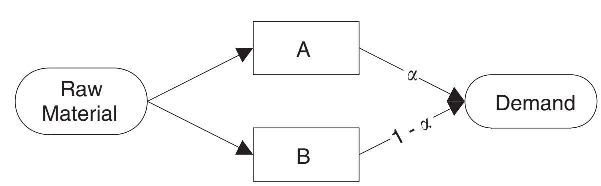

Fig. 2. System structure.

For the first iteration, the capacity ratio of A and B serve as the split ratio estimate. With $\alpha^1_{AD}=0.4615$ and $\alpha^1_{BD}=0.5385$, $B^1_{At}=859$, and $B^1_{Bt}=1002$. Taking these safety stock targets as input, **P1** produces 3000 units at A with $S^1_{ADt}=2141$ and B produces 2861 units and $S^1_{BDt}=1859$. With $\rho=1$, $\alpha^2_{AD}=53.53\%$ so the safety stock targets in the second iteration differ from the first iteration. PRODINV terminates at iteration 11. Since there is ample capacity in this scenario, PRODINV also converges with $B^{11}_{At}=I^{11}_{At}$ and $B^{11}_{Bt}=I^{11}_{Bt}$. Fig. 3 plots each iteration's split ratio.

The initial split ratio does not fully use A's capacity. In iteration 2, PRODINV increases the split ratio but overloads A. The figure shows that the adjustments overcompensate between iterations. The reason is when $\rho=1$ the adjustment is only based on how the stages supply demand, which does not take into account the safety stock required to support that supply.

Fig. 4 shows how the iteration process changes with ρ. When ρ is less than 1, PRODINV reserves some capacity for the safety stock. If ρ is small enough, the over compensation between iterations disappears. However, if ρ is too small, termination occurs slowly.

For the two-stage single-product problem, Proposition 1 is proven in Appendix.

Proposition 1. *If $\rho<(2D_t/(D_t+SS_t))$, PRODINV terminates.*

For our example, Proposition 1 guarantees that PRODINV will terminate when $\rho<1.365$. Furthermore, while the optimal split ratio at termination cannot be derived analytically for arbitrary multi-echelon networks, for this two-stage single-product network the proof demonstrates that the optimal split ratio at termination can be analytically determined.

Table 1 also reports the sum of the inventory holding and penalty cost for each iteration. The sum of these costs is reduced by 3.4% comparing iteration 1 to iteration 11.

Table 1
Iteration detail of single product example.

Iteration	Stage	α^n_{jD} (%)	B^n_{jt}	P^n_{jt}	S^n_{jDt}	I^n_{jt}	Total cost
1	A	46.15	859	3000	2141	859	1002
	B	53.85	1002	2861	1859	1002	
2	A	53.53	996	3000	2004	996	969
	B	46.47	865	2861	1996	865	
3	A	50.10	932	3000	2068	932	976
	B	49.90	929	2861	1932	929	
4	A	51.69	962	3000	2038	962	969
	B	48.31	899	2861	1962	899	
5	A	50.95	948	3000	2052	948	976
	B	49.05	913	2861	1948	913	
6	A	51.30	955	3000	2045	955	969
	B	48.70	906	2861	1955	906	
7	A	51.14	952	3000	2048	952	972
	B	48.86	909	2861	1952	909	
8	A	51.21	953	3000	2047	953	968
	B	48.79	908	2861	1953	908	
9	A	51.18	952	3000	2048	952	968
	B	48.82	909	2861	1952	909	
10	A	51.19	953	3000	2047	953	968
	B	48.81	908	2861	1953	908	
11	A	51.18	953	3000	2047	953	968
	B	48.82	908	2861	1953	908	

3.2. Case 2: two-stage N-product problem

Fig. 5 depicts a network where two stages satisfy demand for n products:

As before, stage A is assumed to be the lower-cost stage for all products and PRODINV operates as described in Section 2. If a product is produced by only one stage, we can see that setting safety stock levels will be straightforward and the product's required production capacity will effectively be netted out of the stage's capacity. Therefore, we ignore these sole-sourced products, and only consider the products that can be produced from both stage A and stage B.

Given these assumptions, Proposition 2 is a multi-product generalization of Proposition 1:

Proposition 2. *If $\rho<\min(1,(2D_j/(D_j+SS_j)),j\in[1,n])$, PRODINV terminates for the two-stage N-product case.*

Notice this generalization is not a simple extension of Proposition 1. The major difference is that $\rho\leq 1$ is part of this sufficient condition. Proposition 1 guarantees termination in the single-product case. The direct extension of Proposition 1 is the requirement of $\rho<\min(2D_j/(D_j+SS_j)),j\in[1,n])$. As we discuss in the proof of Proposition 2, the multi-product system is more complicated. Between iterations, the production plan of more than one product may be adjusted. The condition $\rho\leq 1$ will

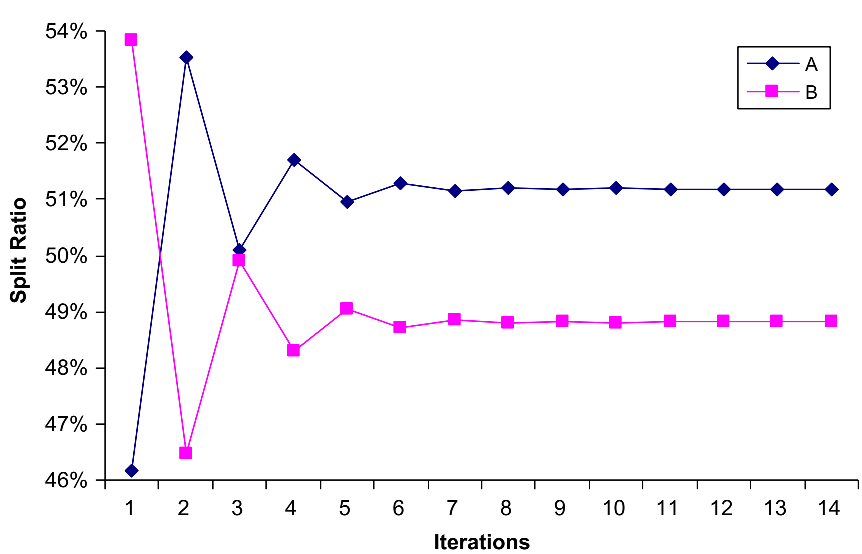

Fig. 3. Split ratio convergence process.

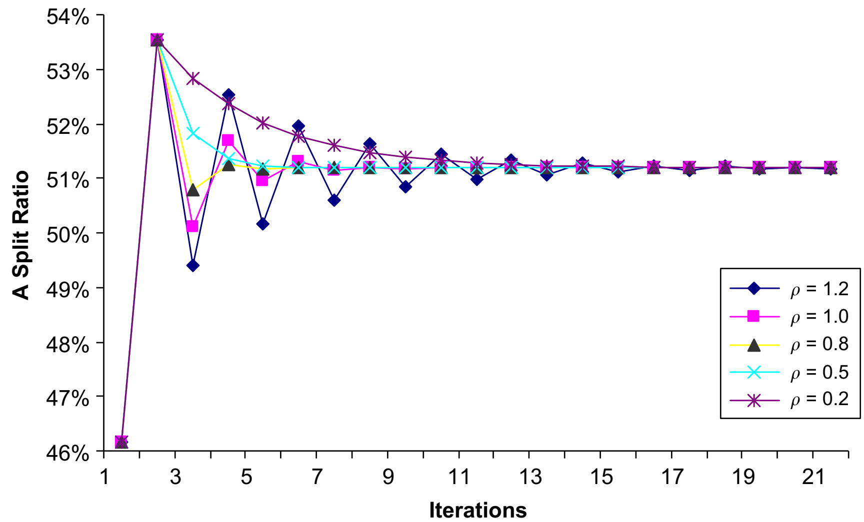

Fig. 4. Iteration process with different ρ values.

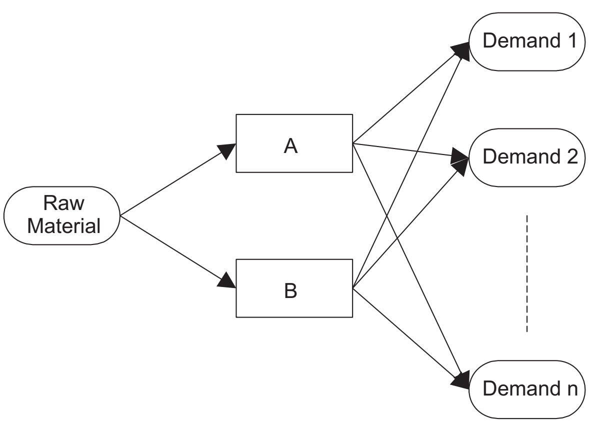

Fig. 5. System structure for two-stage *N*-product problem.

guarantee that the adjustment between iterations will be less and less. Eventually, it will reduce to the case where only one product's production plan needs to be adjusted, at which point termination is guaranteed by the rest of the condition.

4. Applying PRODINV to real-world data

Section 3 establishes conditions where PRODINV will terminate for a single-echelon problem. Thus far we have been unable to prove that PRODINV will terminate for general acyclic networks. However, our intuition is that PRODINV will terminate for real-world problems since there is more flexibility regarding inventory deployment in general networks and there are more production paths to satisfy demand. To test our hypothesis, we have started to test PRODINV at a leading semiconductor company. It will be useful to frame the overall problem in terms of the industry that motivated this research.

The semiconductor manufacturing process is shown in Fig. 6. At a high level, the process consists of three major sets of operations: fabrication-sort, assembly-test, and finish-pack. In fabrication, transistors are built on silicon wafers and interconnected to form circuits. Fabrication consists of more than 300 production steps and takes roughly eight weeks to complete. Each wafer is then sorted by identifying die that do not work and classifying working die into broad categories based on their physical characteristics.

Sorted wafers are then passed to assembly, where die are cut from the wafers and mounted in packages to protect them and enable connection with other devices such as printed circuit boards. A variety of packages are available depending on the target application (i.e. servers, desktops, or laptops). The assembly process includes about 30 production steps, and can take up to two weeks. Once packaged, devices are thermally stressed to induce infant mortality and tested again for final classification into performance categories according to operational speed.

In the finish process, devices are permanently configured for speed depending on their intended application with the possibility of using higher performance products to fill demand for lower performance products (but not vice versa). In pack, devices are individually labeled and packed for shipment. The whole finish and packing process has roughly 10 production steps, and takes only a few days to complete.

The core repetitive decisions of the supply-demand network are (1) how much of what material to release into fabrication, assembly, and finish facilities in every time period, (2) how much material to put into which package in assembly and how much of what semi-finished material to configure into which products in finish, and (3) how much inventory to hold of raw materials before fabrication, die and packages before assembly, semi-finished goods before finish, and packing materials before pack.

Fig. 7 presents a SKU-location view of a typical semiconductor supply chain. A stage represents a location that can hold inventory after the stage's processing function is complete. From the figure, we can see that virtually all portions of the supply chain are dual sourced and dispersed geographically. The mapping between intermediate product categories (functional categories, performance categories, etc.) and between intermediate product categories and finished goods is not one to one. Providing more than one route to connect different facilities can improve the system's robustness. Problems at one facility can be accommodated by diverting production to an identical facility located elsewhere in the world. However, this makes the system more complicated, and can expand the scale of the planning problem very quickly. In actual practice, there could be as many as 2500 end products with the associated number of semi-finished goods, package types, and wafers. Across the globe, there would be as many as 20 factories and 200 inventory holding positions with the equivalent number of transportation links.

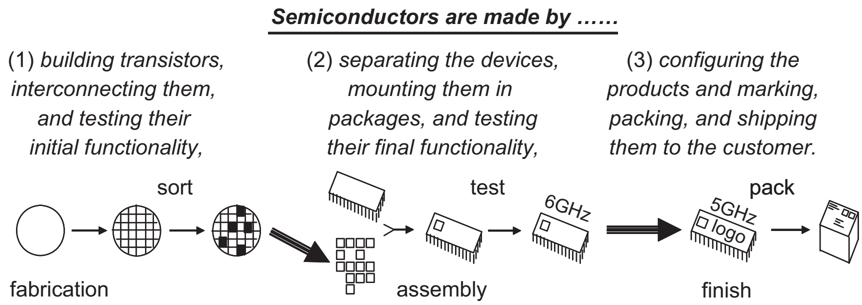

Fig. 6. The basic flow in seminconductor manufacturing.

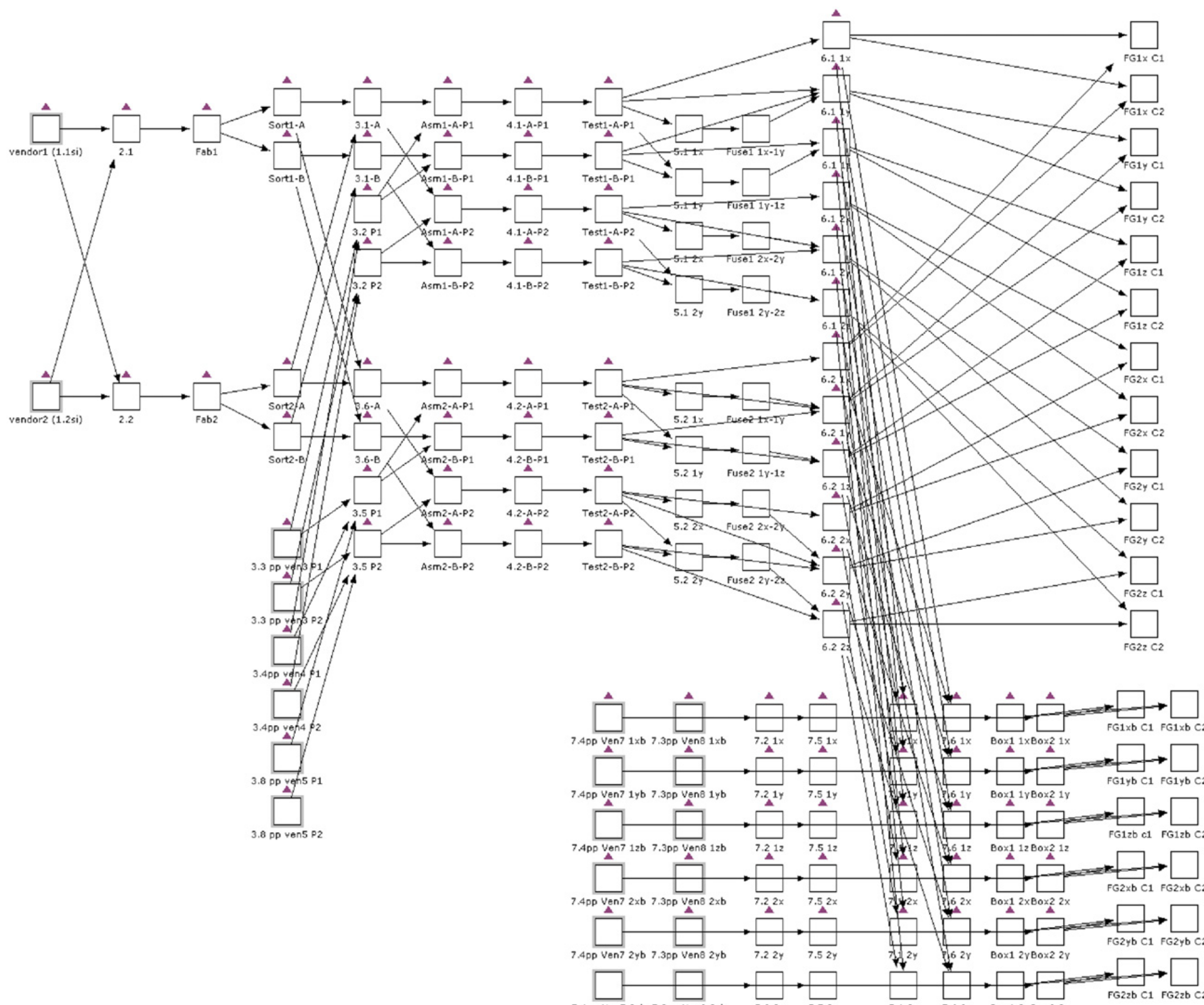

Fig. 7. SKU-location diagram of semiconductor supply chain.

When dealing with manufactured goods that have long lead times and high costs, supply chains often operate under a build-to-forecast strategy. Planners in such companies are faced with the problem of determining the amount of material to release into production on a regular basis. As witnessed in Intel (2005) and Intel (2006), this decision process includes at least a capacity statement with lead times, the amount of work in process (WIP) and finished goods in inventory, and a demand forecast over time. The art of the planners is to devise a material release plan that allocates supply (as capacity, WIP, and inventory) to satisfy the demand forecast while accounting for the demand forecast's inherent uncertainty. In practice, the planners' success is measured by both demand satisfaction (and misses) and inventory levels. Intuition developed over time indicates that inventory above a certain level will probably be marked down or written off while inventory below another level risks stock outs and lost revenue.

We tested our algorithm on a subsystem of the supply chain shown in Fig. 8. Semi-Finished Goods Inventory (SFGI) is the output of SFGI stages that represent the assembly/test process. Finished goods are output of FG stages that corresponds to the finish process. Both echelons can hold inventory. All SFGIs can be

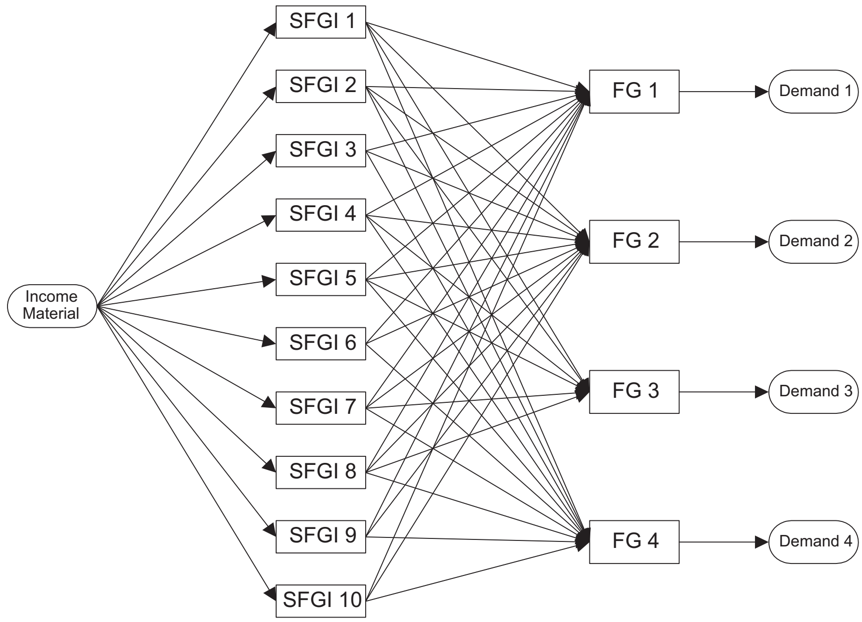

Fig. 8. SFGI start problem diagram.

Table 2
Iteration process of safety stock target.

Iteration	SFGI 1	SFGI 2	SFGI 3	SFGI 4	SFGI 5	SFGI 6	SFGI 7	SFGI 8	SFGI 9	SFGI 10
1	2873	5632	2546	4038	7738	2176	2254	3787	3682	5831
2	3382	6706	2849	6397	4586	2146	3050	3527	2814	5100
3	3480	6924	2740	5268	5239	2346	3183	3303	2913	5159
4	3631	6654	2453	4924	5661	2283	3321	3342	3271	5017
5	3875	6912	2578	4968	5075	2279	3349	3355	3165	5001
6	3972	7009	2626	4888	4935	2276	3354	3358	3145	4994
7	4011	7045	2645	4833	4902	2275	3355	3358	3142	4991
8	4026	7060	2653	4805	4894	2275	3355	3358	3141	4990
9	4033	7065	2656	4792	4892	2275	3355	3358	3141	4990
10	4035	7067	2657	4787	4892	2275	3355	3358	3141	4990
11	4036	7068	2657	4785	4891	2275	3355	3358	3141	4990
12	4036	7069	2657	4784	4891	2275	3355	3358	3141	4990
13	4037	7069	2657	4784	4891	2275	3355	3358	3141	4990
14	4037	7069	2657	4784	4891	2275	3355	3358	3141	4990

processed to any kind of FG, except that SFGI 3, 6, 9, and 10 cannot be used to make FG3. The value of an SFGI is represented by what kinds of FG it can make. A SFGI that can be transformed to high end FGs is more valuable than a SFGI that can only be converted to low end FG. The production of each kind of SFGI is constrained by assembly and test capacity.

The numerical exercise determines the weekly production plan for the next four weeks. The demand for each FG is assumed to be stationary over this planning horizon. Given the complexity of the network in Fig. 8, it is necessary to adopt industrial strength software to satisfy the requirements of **P1** and **P2**. In particular, **P1**, is a mathematical programming formulation developed by Intel's Decision Technologies Group using ILOG CPLEX as described in Bean et al. (2005) and **P2** uses a software tool from Optiant called PowerChain Inventory as described in Billington et al. (2004). Decision variables in **P1** are how many units of each SFGI to produce, and how many units of each SFGI will be released to make each FG. To maintain business confidentiality, the specific problem parameters for the network are not shown here. While the data included in this section has been disguised, the essence of the problem is not changed. Decision variables in **P2** are the safety stock targets for each SFGI and FG location.

The PRODINV iteration process is the same as articulated in Section 2. Due to the properties of the finish process, there are no production minimums in SFGI or FG. Table 2 shows the iteration process of the safety stock target, $\mathbf{B}^n$, at SFGI. The safety stock targets at FGs are decided by the end-item forecast error, so they will not change between iterations.

We can see that the termination criterion is satisfied at iteration 14. Table 3 shows how the split ratio, $\boldsymbol{\alpha}^n$, between SFGIs and FG1 evolves with the iteration process.

5. Conclusions

In this paper, we propose an iterative approach to jointly solve the problems of tactical production planning and tactical safety stock placement. For simple network structures, two stages and one or n products, we provide sufficient conditions to guarantee the algorithm's termination. Through examples, we show how the algorithm works and prove its applicability on a realistic industrial-scale problem.

There are several opportunities to extend this work. The first is to establish sufficient conditions for termination in more complicated

Table 3
Iteration of split ratio of FG1.

Iteration	SFGI 1 (%)	SFGI 2 (%)	SFGI 3 (%)	SFGI 4 (%)	SFGI 5 (%)	SFGI 6 (%)	SFGI 7 (%)	SFGI 8 (%)	SFGI 9 (%)	SFGI 10 (%)
1	0.00	0.00	0.00	7.61	17.19	22.82	18.86	0.00	0.00	33.52
2	0.00	0.00	7.29	21.90	6.65	13.61	7.29	6.03	10.94	26.30
3	0.00	10.17	17.35	17.03	2.64	13.83	2.89	2.39	4.34	29.36
4	0.00	13.71	14.20	20.29	2.85	14.00	1.16	0.96	1.74	31.09
5	0.00	15.25	12.92	20.56	3.94	14.05	0.46	0.38	0.70	31.74
6	0.00	15.90	12.40	20.42	4.62	14.06	0.19	0.15	0.28	31.99
7	0.00	16.16	12.19	20.30	4.94	14.06	0.07	0.06	0.11	32.09
8	0.00	16.27	12.11	20.24	5.09	14.06	0.03	0.02	0.04	32.12
9	0.00	16.32	12.08	20.21	5.15	14.06	0.01	0.01	0.02	32.14
10	0.00	16.33	12.06	20.20	5.17	14.06	0.00	0.00	0.01	32.14
11	0.00	16.34	12.06	20.20	5.18	14.06	0.00	0.00	0.00	32.15
12	0.00	16.34	12.06	20.20	5.19	14.06	0.00	0.00	0.00	32.15
13	0.00	16.35	12.06	20.19	5.19	14.06	0.00	0.00	0.00	32.15
14	0.00	16.35	12.06	20.19	5.19	14.06	0.00	0.00	0.00	32.15

networks. The second is to determine whether other variants of PRODINV perform better on practical problems. For example, with slight modifications, PRODINV could begin with an initial estimate for $\mathbf{B}^1$ and then proceed to solve **P1** first and **P2** second. Third, there could be value in relaxing the assumption regarding how the split ratio is determined. For example, in more complex networks it might be desirable to have one stage handle only a stable portion of demand while allowing another stage to handle the safety stock requirements.

Another research direction is to exploit properties of particular solution tools for **P1** and **P2** to make the iteration process more efficient. This include using **P2**'s result to guide **P1** to find a more desirable production plan within the given capacity, and using **P1** information to help **P2** set feasible safety stock targets. This is extremely useful when **P1** and **P2** do not converge, i.e. capacity is insufficient to meet both demand and safety stock requirement, under default parameter settings.

Appendix: Proofs

Proof of Proposition 1

As proposed in the algorithm, the iteration process starts with a guess of the split ratio.

(1) Sufficient capacity: Since the capacity of the system is enough to meet demand plus inventory, at A, LP will satisfy its inventory requirement, and then fill the demand with the remaining capacity. Which means we will have the following iteration results, where S^i_{ADt} is the amount of demand that is satisfied from A.

$$\alpha^1_{A,t}: \text{initial guess} \quad S^1_{ADt} = C_A - \alpha^1_{A,t} SS_t + I_{A,t-1}$$

$$\alpha^2_{A,t} = \frac{S^1_{ADt}}{D_t} \quad S^2_{ADt} = \rho(C_A - \alpha^2_{A,t} SS_t + I_{A,t-1}) + (1-\rho) S^1_{ADt}$$

$$\alpha^3_{A,t} = \frac{S^2_{ADt}}{D_t} = (1-\rho)\frac{S^1_{ADt}}{D_t} + \rho\frac{(C_A - \alpha^2_{A,t} SS_t + I_{A,t-1})}{D_t}$$

$$S^3_{ADt} = \rho(C_A - \alpha^3_{A,t} SS_t + I_{A,t-1}) + (1-\rho) S^2_{ADt}$$

$$= \left(1-\rho-\rho\frac{SS_t}{D_t}\right)\alpha^2_{A,t} + \rho\frac{C_A + I_{A,t-1}}{D_t}$$

$$\vdots$$

$$\alpha^n_{A,t} = \frac{S^{n-1}_{ADt}}{D_t} = (1-\rho)\frac{S^{n-2}_{ADt}}{D_t} + \rho\frac{(C_A - \alpha^{n-1}_{A,t} SS_t + I_{A,t-1})}{D_t}$$

$$= \left(1-\rho-\rho\frac{SS_t}{D_t}\right)\alpha^{n-1}_{A,t} + \rho\frac{C_A + I_{A,t-1}}{D_t}$$

$$= \left(1-\rho-\rho\frac{SS_t}{D_t}\right)^{n-2}\alpha^2_{A,t} + \rho\frac{C_A + I_{A,t-1}}{D_t}$$

$$\times\left[1 + \left(1-\rho-\rho\frac{SS_t}{D_t}\right) + \cdots + \left(1-\rho-\rho\frac{SS_t}{D_t}\right)^{n-3}\right]$$

$$= \left(1-\rho-\rho\frac{SS_t}{D_t}\right)^{n-2}\alpha^2_{A,t} + \rho\frac{C_A + I_{A,t-1}}{D_t}\frac{1-\left(1-\rho-\rho\frac{SS_t}{D_t}\right)^{n-2}}{1-(1-\rho-\rho\frac{SS}{D_t})}$$

$$= \left(1-\rho-\rho\frac{SS_t}{D_t}\right)^{n-2}\alpha^2_{A,t} + \frac{C_A + I_{A,t-1}}{D_t + SS_t}\left[1-\left(1-\rho-\rho\frac{SS_t}{D_t}\right)^{n-2}\right]$$

$$= \frac{C_A + I_{A,t-1}}{D_t + SS_t} + \left(1-\rho-\rho\frac{SS_t}{D_t}\right)^{n-2}\left(\alpha^2_{A,t} - \frac{C_A + I_{A,t-1}}{D_t + SS_t}\right)$$

So as long as $1-\rho-\rho\frac{SS}{D_t} > -1 \Rightarrow \rho < \frac{2D_t}{D_t + SS_t}$,

$$\alpha_{A,t} = \lim_{n\to\infty}\alpha^n_{A,t} = \frac{C_A + I_{A,t-1}}{D_t + SS_t}$$

(2) Insufficient capacity, which means that $C_A + C_B + I_{A,t-1} + I_{B,t-1} < D_t + SS_t$: When capacity is not enough to meet demand plus inventory, the production output of A and B will equal to their capacities. Also, LP will start to reduce the inventory at A first because of the lower penalty cost. When we have capacity to keep some inventory at A, the LP inventory will be less than the inventory level required by the inventory policy, however, the inventory at B will still be met. Hence the demand satisfied from A depends on how much demand is filled from B, i.e. $S_{A,t} = D_t - [C_B - (1-\alpha_{A,t})SS_t + I_{B,t-1}]$. Now we have the iteration process as

$$\alpha^1_{A,t}: \text{initial guess} \quad S^1_{ADt} = D_t - [C_B - (1-\alpha^1_{A,t})SS_t + I_{B,t-1}]$$

$$\alpha^2_{A,t} = \frac{S^1_{ADt}}{D_t} \quad S^2_{ADt} = \rho[D_t - [C_B - (1-\alpha^2_{A,t})SS_t + I_{B,t-1}]] + (1-\rho)S^1_{ADt}$$

$$\alpha^3_{A,t} = \frac{S^2_{ADt}}{D_t} = (1-\rho)\frac{S^1_{ADt}}{D_t} + \rho\frac{D_t - [C_B - (1-\alpha^2_A)SS_t + I_{B,t-1}]}{D_t}$$

$$S^3_{ADt} = (1-\rho)S^2_{ADt} + \rho[D_t - [C_B - (1-\alpha^3_{A,t})SS_t + I_{B,t-1}]]$$

$$= \left(1-\rho-\rho\frac{SS_t}{D_t}\right)\alpha^2_{A,t} + \rho\frac{D_t - C_B + SS_t - I_{B,t-1}}{D_t}$$

$$\vdots$$

$$\alpha^n_{A,t} = \frac{S^{n-1}_{ADt}}{D_t} = (1-\rho)\frac{S^{n-2}_{ADt}}{D_t} + \rho\frac{D_t - [C_B - (1-\alpha^{n-1}_A)SS_t + I_{B,t-1}]}{D_t}$$

$$= \left(1-\rho-\rho\frac{SS_t}{D_t}\right)\alpha^{n-1}_{A,t} + \rho\frac{D_t - C_B + SS_t - I_{B,t-1}}{D_t}$$

$$= (1-\rho-\rho\frac{SS_t}{D_t})^{n-2}\alpha^2_{A,t} + \rho\frac{D_t - C_B + SS_t - I_{B,t-1}}{D_t}$$

$$\times\left[1 + \left(1-\rho-\rho\frac{SS_t}{D_t}\right) + \cdots + \left(1-\rho-\rho\frac{SS_t}{D_t}\right)^{n-3}\right]$$

$$= \left(1-\rho-\rho\frac{SS_t}{D_t}\right)^{n-2}\alpha^2_{A,t} + \rho\frac{D_t - C_B + SS_t - I_{B,t-1}}{D_t}$$

$$\times \frac{1-(1-\rho-\rho(SS_t/D_t))^{n-2}}{1-(1-\rho-\rho(SS_t/D_t))}$$
$$= \left(1-\rho-\rho\frac{SS_t}{D_t}\right)^{n-2}\alpha_{A,t}^2+\frac{D_t-C_B+SS_t-I_{B,t-1}}{D_t+SS_t}$$
$$\times\left[1-\left(1-\rho-\rho\frac{SS_t}{D_t}\right)^{n-2}\right]$$
$$=1-\frac{C_B+I_{B,t-1}}{D_t+SS_t}+\left(1-\rho-\rho\frac{SS_t}{D_t}\right)^{n-2}\left(\alpha_{A,t}^2-1+\frac{C_B+I_{B,t-1}}{D_t+SS_t}\right)$$

So as long as $1-\rho-\rho\frac{SS_t}{D_t}>-1 \Rightarrow \rho<\frac{2D_t}{D_t+SS_t}$,

$$\alpha_{A,t}=\lim_{n\to\infty}\alpha_{A,t}^n=1-\frac{C_B+I_{B,t-1}}{D_t+SS_t}$$

When the inventory at A becomes zero, which means all its production is used to meet demand, this output will not change. Hence the demand split ratio will not change. So the iteration will converge in a single iteration, and the split ratio will be$\alpha_{A,t}=\frac{C_A+I_{A,t-1}}{D_t}$. □

Proof of Proposition 2

Proof. Since there are only two stages involved, the production plan of one can be derived from the production plan of the other one directly. So we will show the iteration process through the analysis of stage A. Assume the iteration starts with an initial guess of the split ratio, α_{Aj}^1, then $\alpha_{Bj}^1=1-\alpha_{Aj}^1$. For any given product j, the shipment amount from stage A can be the maximum possible amount, the minimum possible amount, or an amount in between. We denote G_1 as the set of products that are produced at the maximum amount at stage A, and G_2 as the set of products that are produced at the minimum amount at stage A.

Iteration 1 Initial guess of α_{Aj}^1

Shipment plan $\tilde{S}_{ADj}^1=D_j-P_{\min}$ for $j\in G_1$ and $S_{ADj}^1=\tilde{S}_{ADj}^1$

$\tilde{S}_{ADj}^1=P_{\min}$ for $j\in G_2$

$D_j-P_{\min}>\tilde{S}_{ADj}^1>P_{\min}$ for others

We define remaining capacity$R_i^k=C_i-\sum_{j=1}^n\alpha_{ij}^kSS_j$. The planning process can be treated as allocate remaining capacity to fill demand. If the remaining capacity is less than nP_{min}, the planning will meet all demands at the minimum amount, and safety stock shortage will happen.

Notice that the Simplex Method always searches the solution along the edge of the feasible area. With the change of remaining capacity between iterations, it will change the planning output one product at a time. If the production plans of more than one product are adjusted, it must happens in the way that adjust one to the extreme amount (minimum or maximum) first, and then change the next one to the extreme amount, and so on. So if the change in remaining capacity is small enough, then there will be only one product's production plan adjusted between iterations. The change of more than one product's production plan between iterations only happens when the change of remaining capacity is big enough. When the first scenario happens, eventually the production plans between iterations will fall into the same class, i.e. they have the same set of products that are produced at the maximum amount and minimum amount. For all other cases, the production plans from different iterations can belong to different classes. We will proof the proposition in two steps. First we will prove that all iterations will lead to iterations within a class, and then we will show that the iteration within a class will converge to solution that meets the termination criterion.

(I) We start our analysis with the sufficient capacity case. Sufficient capacity means that the total capacity is enough to meet both demand and safety stock target.

(a) If the outputs from iterations belong to different classes, there are two possible scenarios. If $\rho\ge 1$, production plan may oscillate between the case that all products are produced at the minimum amount and the case that all products are produced at the maximum amount. However, this oscillation problem will be solved by setting $\rho<1$. For all other scenarios, the safety stock targets will always be met; hence the planning problem will answer how to allocate the remaining capacity as we defined earlier. We will show that the difference of remaining capacities between iterations is monotonically decreasing if $\rho<1$. We assume the initial split ratio underestimates the capacity of stage A, and the production plan increase the production of products in group G_2. Without the loss of generality, we can sort the production plan of the first iteration as: G_2={1, …, k}, G_1={m, …, n}. The iteration will be:

Iteration 2 Split Ratio $\alpha_{Aj}^2=\frac{S_{ADj}^1}{D_j}$

Remaining Capacity $R_A^2=C_A-\sum_{j=1}^n\alpha_{Aj}^2SS_j$

Shipment plan $\tilde{S}_{ADj}^2=D_j-P_{\min}$ for $j\in\{m,\ldots,n\}$ and

$S_{ADj}^2=\rho\tilde{S}_{ADj}^2+(1-\rho)\tilde{S}_{ADj}^1$

$\tilde{S}_{ADj}^2=P_{\min}$ for $j\in\{1,\ldots,k'\}$ $k'<k$

$D_j-P_{\min}>\tilde{S}_{ADj}^2>P_{\min}$ for others

$\tilde{S}_{ADj}^2=S_{ADj}^1$ $j\in\{k+1,\ldots,m-1\}$

Iteration 3 Split Ratio $\alpha_{Aj}^3=\frac{S_{ADj}^2}{D_j}$

$$\text{Remaining Capacity}\quad R_A^3=C_A-\sum_{j=1}^n\alpha_{Aj}^3SS_j=C_A-\sum_{j=1}^{k'}\alpha_{Aj}^2SS_j-\sum_{j=k'+1}^{k}\alpha_{Aj}^3SS_j-\sum_{j=k'}^{n}\alpha_{Aj}^2SS_j$$

Shipment plan $\tilde{S}_{ADj}^3=D_j-P_{\min}$ for $j\in\{m,\ldots,n\}$ and

$S_{ADj}^3=\rho\tilde{S}_{ADj}^3+(1-\rho)\tilde{S}_{ADj}^2$

$\tilde{S}_{ADj}^3=P_{\min}$ for $j\in\{1,\ldots,k''\}k'<k''<k$

$D_j-P_{\min}>\tilde{S}_{ADj}^3>P_{\min}$ for others

$P_{\min}<\tilde{S}_{ADk''+1}^3<\tilde{S}_{ADk''+1}^2$

$\tilde{S}_{ADj}^3=\tilde{S}_{ADj}^2$ $j\in\{k''+2,\ldots,m-1\}$

The change from iteration 2 to iteration 3 is straightforward. As the production amount for some products increases from P_{min} in iteration 2, the remaining capacity in iteration 3 is smaller. Hence the production plan will reduce the production of some products. However, the change will not make the production plan return to iteration 1 if $\rho<1$. So the production plan will reduce the production of some products to P_{min}. The amount of product k'' may be reduced, but will not to P_{min}. Now we have the difference of remaining capacity between iterations

$$|R_A^3-R_A^2|=\sum_{j=k'+1}^{k}(\alpha_{Aj}^3-\alpha_{Aj}^2)SS_j=\sum_{j=k'+1}^{k}\left(\alpha_{Aj}^3-\frac{P_{\min}}{D_j}\right)SS_j$$

$$|R_A^4-R_A^3| = \sum_{j=k'+1}^{k}(\alpha_{Aj}^3-\alpha_{Aj}^4)SS_j = \sum_{j=k'+1}^{k''}\left(\alpha_{Aj}^3-\frac{\rho P_{\min}+(1-\rho)S_{ADj}^2}{D_j}\right)SS_j$$
$$+\sum_{j=k''+1}^{k}\left(\alpha_{Aj}^3-\frac{\rho\tilde{S}_{ADj}^2+(1-\rho)S_{ADj}^2}{D_j}\right)SS_j$$
$$=\sum_{j=k'+1}^{k''}\left(\alpha_{Aj}^3-\frac{\rho P_{\min}+(1-\rho)(\rho\tilde{S}_{ADj}^2+(1-\rho)S_{ADj}^1)}{D_j}\right)$$
$$\times SS_j+\sum_{j=k''+1}^{k}\left(\alpha_{Aj}^3-\frac{\rho\tilde{S}_{ADj}^2+(1-\rho)(\rho\tilde{S}_{ADj}^2+(1-\rho)S_{ADj}^1)}{D_j}\right)SS_j$$
$$<\sum_{j=k'+1}^{k}\left(\alpha_{Aj}^3-\frac{P_{\min}}{D_j}\right)SS_j = |R_A^3-R_A^2|$$

Similar process will show that above relationship holds for iteration n, $n+1$, and $n+2$. Thus the difference of remaining capacity is monotonically decreasing, eventually, the difference will be small enough that planning outputs between iterations will be in the same class.

The above analysis applies to other scenarios where the production plans between iterations belong to different classes. Hence we show that if $\rho<1$, the iteration will end into the case that planning outputs between iterations will be in the same class. i.e., only one product's production plan needs to be adjusted.

(b) Now we look at the case that all production plans from iterations are within the same class. First, we assume that only one product's production plan is changed in the following iterations, and denote the product whose shipment is changed in the second iteration as product k. Let $C_R = C_A-\sum_{j\neq k}\alpha_{Aj}^n SS_j-\sum_{j\neq k}S_{ADj}^n$, and we have $S_{ADj}^n = S_{ADj}^1$, and $\alpha_{Aj}^t=\alpha_{Aj}^1$, for $j\neq k$. The iteration with change only on product k will be like

Iteration 2 Split Ratio $\alpha_{Ak}^2 = \dfrac{S_{ADk}^1}{D_k}$

Shipment plan $\tilde{S}_{ADk}^2 = C_R-\alpha_{Ak}^2 SS_k$

$$S_{ADk}^2 = \rho\tilde{S}_{ADk}^2+(1-\rho)S_{ADk}^1$$
$$=\rho(C_R-\alpha_{Ak}^2SS_k)+(1-\rho)S_{ADk}^1$$
$$=\rho\left(C_R-\frac{SS_k}{D_k}S_{ADk}^1\right)+(1-\rho)S_{ADk}^1$$
$$=\rho C_R+\left(1-\rho-\rho\frac{SS_k}{D_k}\right)S_{ADk}^1$$

Iteration 3 Split Ratio $\alpha_{Ak}^3 = \dfrac{S_{ADk}^2}{D_k}$

Production plan $\tilde{S}_{ADk}^3 = C_R-\alpha_{Ak}^3 SS_k$

$$S_{ADk}^3 = \rho\tilde{S}_{ADk}^3+(1-\rho)P_{Ak}^2$$
$$=\rho(C_R-\alpha_{Ak}^3SS_k)+(1-\rho)S_{ADk}^2$$
$$=\rho\left(C_R-\frac{SS_k}{D_k}S_{ADk}^2\right)+(1-\rho)S_{ADk}^2$$
$$=\rho C_R+\left(1-\rho-\rho\frac{SS_k}{D_k}\right)S_{ADk}^2$$
$$=\rho\left(1+\left(1-\rho-\rho\frac{SS_k}{D_k}\right)\right)C_R$$
$$+\left(1-\rho-\rho\frac{SS_k}{D_k}\right)^2S_{ADk}^1$$

...

Iteration n Split Ratio $\alpha_{Ak}^n = \dfrac{S_{ADk}^{n-1}}{D_k}$

Shipment plan $\tilde{S}_{ADk}^n = C_R-\alpha_{Ak}^n SS_k$

$$S_{ADk}^n = \rho\tilde{S}_{ADk}^n+(1-\rho)S_{ADk}^{n-1}$$
$$=\rho(C_R-\alpha_{Ak}^nSS_k)+(1-\rho)S_{ADk}^{n-1}$$
$$=\rho\left(C_R-\frac{SS_k}{D_k}S_{ADk}^{n-1}\right)+(1-\rho)S_{ADk}^{n-1}$$
$$=\rho C_R+\left(1-\rho-\rho\frac{SS_k}{D_k}\right)S_{ADk}^{n-1}$$
$$=\rho(1+\left(1-\rho-\rho\frac{SS_k}{D_k}\right)+\ldots+\left(1-\rho-\rho\frac{SS_k}{D_k}\right)^{n-2})C_R$$
$$+\left(1-\rho-\rho\frac{SS_k}{D_k}\right)^{n-1}S_{ADk}^1$$
$$=\rho\frac{1-(1-\rho-\rho\frac{SS_k}{D_k})^{n-2}}{1-(1-\rho-\rho\frac{SS_k}{D_k})}C_R+\left(1-\rho-\rho\frac{SS_k}{D_k}\right)^{n-1}S_{ADk}^1$$
$$=\frac{1-(1-\rho-\rho\frac{SS_k}{D_k})^{n-2}}{1+\frac{SS_k}{D_k}}C_R+\left(1-\rho-\rho\frac{SS_k}{D_k}\right)^{n-1}S_{ADk}^1$$

From the above iteration process, we can see that if $\rho<(2D_k/(D_k+SS_k))$, the iteration converges. If we set that $\rho<\min(1,(2D_j/(D_j+SS_j)),j\in[1,n])$, then the iteration process will converge no matter which product is the one that changes production plan between iterations. More important, since the simplex method solves LP problem along the edges, the convergence condition will hold even if the adjustments on one product do not happen on consecutive iterations.

If the convergence point of the iteration process meets the termination criterion, it will be the solution of the problem. Otherwise, there will be capacity shortage or excess at the convergence point. The planning process will drive some products to the extreme production amount and put the solutions to another class. However, the convergence condition is the same for all solution classes.

(II) When the capacity is not enough, safety stock deficit will occur. The planning process will allocate the shortage in the sequence from the lowest penalty one to the highest penalty one. The capacity constraint only exit in the planning process and the inventory optimization has no capacity issue. Since the termination criterion equals to the condition that the split ratio of demand filling, which is the output of planning process, is the same as the split ratio of the safety stock level, which is output of the inventory optimization problem, and we know that capacity shortage will not affect the demand filling and the safety stock setting, the iteration condition of sufficient capacity scenario holds for the insufficient capacity scenario. In fact, assume the capacity deficit is C_d, we can treat the iteration process as adding C_d to the existing capacity and do the iteration as sufficient capacity case. When the process terminates, deduct the inventory volume by C_d in the sequence from the one has lowest penalty to the one has highest penalty.

References

Anthony, R.N., 1965. Planning and Control Systems: A Framework for Analysis. 1, Cambridge, Massachusetts.

Bean, J.W., Devpura, A., O'Brien, M., Shirodkar, S., 2005. Optimizing supply chain planning. Intel Technology Journal 9 (3), 223–231.

Billington, P.J., McClain, J.O., Thomas, L.J., 1983. Mathematical programming approaches to capacity-constrained MRP systems: review, formulation and problem reduction. Management Science 29, 1126–1141.

Billington, C., Callioni, G., Crane, B., Ruark, J.U., Rapp, J., White, T., Willems, S.P., 2004. Accelerating the profitability of Hewlett-Packard's supply chains. Interfaces 34, 59–72.

Bitran, G.R., Haas, E.A., Hax, A.C., 1981. Hierarchical production planning: a single stage system. Operations Research 29, 717–743.

Bitran, G.R., Haas, E.A., Hax, A.C., 1982. Hierarchical production planning: a two-stage system. Operations Research 30, 232–251.

Bossert, J.M., Willems, S.P., 2007. A periodic-review modeling approach for guaranteed service supply chains. Interfaces 37, 420–435.
Bradley, J.R., Arntzen, B.C., 1999. The simultaneous planning of production, capacity, and inventory in seasonal demand environments. Operations Research 47, 795–806.
Byrne, M.D., Bakir, M.A., 1999. Production planning using a hybrid simulation-analytical approach. International Journal of Production Economics 59, 305–311.
De Kok, A.G., Fransoo, J.C., 2003. Planning supply chain operations: definition and comparison of planning concepts. In: De Kok, A.G., Graves, S.C. (Eds.), Handbook in Operations Research and Management Science. Volume 11: Design and Analysis of Supply Chains. North-Holland, Amsterdam, pp. 597–695.
Fandel, G., Stammen-Hegene, C., 2006. Simultaneous lot sizing and scheduling for multi-product multi-level production. International Journal of Production Economics 104, 308–316.
Fleischmann, B., Meyr, H., 2003. Planning hierarchy, modeling and advanced planning systems. In: De Kok, A.G., Graves, S.C. (Eds.), Handbook in Operations Research and Management Science, Volume 11: Design and Analysis of Supply Chains. North-Holland, Amsterdam, pp. 457–523.
Gfrerer, H., Zapfel, G., 1995. Hierarchical model for production planning in the case of uncertain demand. European Journal of Operational Research 86, 142–161.
Graves, S.C., Willems, S.P., 2003. Supply chain design: safety stock placement and supply chain configuration. In: de Kok, A.G., Graves, S.C. (Eds.), Handbooks in Operations Research and Management Science. Vol. 11, Supply Chain Management: Design, Coordination and Operation. North-Holland Publishing Company, Amsterdam, The Netherlands, pp. 95–132.
Hax, A.C., Candea, D., 1984. Production and Inventory Management. Prentice-Hall, Inc., Englewoods Cliffs, New Jersey.
Hax, A.C., Meal, H.C., 1975. Hierarchical integration of production planning and scheduling. In: Geisler, M. (Ed.), TIMS Studies in Management Science. North-Holland, Amsterdam, pp. 53–69.
Humair S., Willems, S.P. Optimizing strategic safety stock placement in general acyclic networks. Operations Research, in press.
Intel. Direct observation during Feng Tian's internship at Intel Corporation, May–July 2005.
Intel. Direct observation during Feng Tian's internship at Intel Corporation, May–July 2006.
Jornsten, K., Leisten, R., 1995. Decomposition and iterative aggregation in hierarchical and decentralized planning structures. European Journal of Operational Research 86, 120–141.
Kempf, K.G., Control-oriented approaches to supply chain management in semiconductor manufacturing. In: Proceedings of the American Control Conference, 2004, pp. 4563–4576.
Kim, B., Kim, S., 2001. Extended model for a hybrid production planning approach. International Journal of Production Economics 73, 165–173.
Maxwell, M., Muckstadt, J.A., Thomas, L.J., VanderEecken, J., 1983. A modeling framework for planning and control of production in discrete parts manufacturing and assembly systems. Interfaces 13, 92–104.
Shapiro, J.F., 1993. Mathematical programming models and methods for production planning and scheduling. In: Graves, S.C, Rinnooy Kan, A.H.G, Zipkin, P.H (Eds.), Handbooks in Operations Research and Management Science. Vol. 4, Logistics of Production and Inventory. North-Holland Publishing Company, Amsterdam, The Netherlands, pp. 371–443.
Spitter, J.M., Hurkens, C.A.J., de Kok, A.G., Lenstra, J.K., Negenman, E.G., 2005. Linear programming models with planned lead times for supply chain operations planning. European Journal of Operational Research 163, 706–720.