

# Optimal Capacity Expansion and Contraction under Demand Uncertainty

Metin Çakanyıldırım \*

School of Management, University of Texas at Dallas

Robin O. Roundy

School of Operations Research and Industrial Engineering, Cornell University

July 2, 2002

## Abstract

This paper presents a novel approach to compute optimal machine capacity expansion/contraction times under uncertain demand. A polynomial time Expansion/Contraction (EC) algorithm is developed to jointly solve the expansion and contraction problems when the demand is first stochastically increasing and then stochastically decreasing. The paper considers multiple machine types and allows for positive lead times for each type. It uses bottleneck policies (BP); Always buy machines of the bottleneck machine type to increase capacity and always retire machines in the reverse order. This order is used to optimally sequence machines types for expansion and contraction for regular service and capacity costs. The paper uses lost sales costs as a measure of the service. Capacity costs are computed through three components of machine specific costs: purchase and retirement costs that are independent of the usage, and machine rent that is proportional to the usage. An extension of EC algorithm to several demand cycles is discussed. EC algorithm is illustrated with a real life example drawn from the semiconductor industry.

---

\*For correspondence: [metin@utdallas.edu](mailto:metin@utdallas.edu)

# 1 Introduction

This paper models and illustrates an optimal solution strategy for a capacity expansion and contraction problem. A single product family experiences stochastic demand. The product family requires various operations on different machine groups. As demand increases new machines must be purchased to increase capacity. Conversely, machines are retired when the demand falls down.

Our model is motivated by equipment intensive industries, in which capacity expansion and contraction is critical and costly. For example, a new semiconductor fab costs around one billion dollars, and a single lithography machine costs around 4-5 million dollars. It is predicted that machine prices will continue to increase as products become more complex (known as Moore's Law, Standard and Poor's Industry Surveys [26] p.7). Finding the means to finance new capacity is a major problem. After the capacity is acquired, upper management naturally asks for high utilization. However machine purchase lead times are long (for example 6-12 months is common in the semiconductor industry [27] p.116), and future needs are affected by a variety of uncertainties. Consequently it is very important to buy and retire the *correct set of machines at the correct times*, so that demand matches productive capacity. The main purpose of this paper is to provide decision makers with a decision support tool to achieve such a match.

In many equipment intensive industries demand is volatile. Especially demand forecasts beyond the purchase lead times carry substantial uncertainty. The semiconductor industry and electric utilities are good examples. In the case of the semiconductor industry, for capacity planning purposes the most relevant demand figures are the ones extending from about 6 months to about 3 years into the future. The volatility of these figures are discussed in Çakanyıldırım and Roundy [8]. Due to rapid technology shifts there is a high risk that inventoried products will become obsolete. Consequently semiconductor companies carry only minimal inventories (Çakanyıldırım and Roundy [9]), and inventory provides only minimal protection against uncertainty. In the case of electric utilities, short term capacity planning is mostly done daily against uncertain demand reaching its peak about early evenings. There can be substantial uncertainty in the hourly power demand. The product is a commodity and will not become obsolete. However inventories cannot be used to smooth out demand fluctuations because the storage is not efficient or economical. Summarizing, demand should be matched by regulating capacity because relying on inventory, except in the short term, is generally not economical.

We are assuming that all products have the same or very similar processing requirements, i.e. we have a single product family which might be obtained by aggregating several products. In many industries a single product family is often composed of many different product versions. When a new manufacturing technology

becomes available, existing products are often migrated from an older technology to a newer one. New products undergo a verification process before they are ready for production. Thus the uncertainty in the total demand for a product family arises from several major sources - uncertainty in total marketplace demand, uncertainty in the company's market share, uncertainty in the timing of migrations of a product from one technology to another, and uncertainty in the time when new products will be ready for production. We model the variability in the total demand for the product family, without differentiating between the different sources of uncertainty.

Customer demands generally cycle with the economic climate, increasing in economic booms and decreasing in economic slumps. Even high-tech industries which traditionally experience strong demand growth cannot escape from these cycles. A semiconductor professional once told: "... maybe ... in the past 20-25 years we have had a 5-year period of growth, but I do not remember it. Two to three years-yes." Therefore, capacity contractions must be studied along with expansions so we model both. Our model considers a finite time horizon, treats time as a continuous variable, and models demand as a stochastic process. It does not permit any accumulation of inventory as motivated by the semiconductor industry and electric utilities, and unfilled demand is lost. Each machine type (or generator in electric utilities) has a lead time for purchase, installation and qualification. We assume that machine capacities remain constant over the decision horizon. We will next discuss previous research and then the mathematical model will be presented. In section 4, solution procedure will be discussed and illustrated with a real life example. We provide a brief conclusion in sections 5.

## 2 Literature Survey

An extensive survey, capturing application areas and multi-location models, is in Luss [19], we discuss the models that appeared thereafter. Initially, capacity expansion work generally focused on models with deterministic demands. Neebe and Rao [20] provides a shortest path formulation and a Lagrangian relaxation scheme to sequencing and selecting expansion options. There have also been efforts to convert the stochastic expansion problem to an equivalent deterministic problem in the sense that both problems have the same optimal solutions. When product demands are transformed Brownian motion or transformed birth-death processes, Bean, Higle and Smith [4] provide an equivalent formulation by replacing stochastic demand with its deterministic counterpart and reducing the interest rate. Fong and Srinivasan [13] study a transportation problem where the capacity of facilities is expanded as the deterministic demand grows. Li and Tirupati [18] explicitly consider flexible and dedicated capacity expansions for multiple products and provide heuristics. Rajagopalan [22] presents a capacity model by assuming that a combination of machines exist whose total capacity is exactly

equal to the demand. The model captures capacity expansions, disposals and replacements.

If time is a continuous variable rather than a discrete one, the expansion problem becomes an optimal control problem whose objective is usually the integral of cost (production, expansion, inventory) over time, e.g., Khmelnitsky and Kogan [17] provide an algorithm to calculate the optimal expansion rate function for deterministic demand. Davis, Dempster, Sethi and Vermes [10] regulate the expansion rate with an investment rate function. As soon as the cumulative investment reaches the random price of a discrete capacity unit, that unit becomes available. This paper enriches classical approaches by introducing a nonconstant investment rate and random capacity prices which are more common in infrastructure industries than in manufacturing. Benavides, Duley and Johnson [5] study the optimal expansion times for semiconductor fabs without differentiating between machine groups.

In the economics community, the capacity expansion problem is recently addressed by works of Dixit [11], and Eberly and Van Mieghem [12]. The latter establishes structural properties for expansion and contraction policies for multiple factors contributing to capacity. It provides a closed form solution of the optimal policy in the case of IID stochastic processes and stationary costs. It also introduces the concept of ordering expansions of different factors of capacity (see Proposition 3), which inspires the Bottleneck Policies of the current paper. Harrison and Van Mieghem [14] revisit Eberly and Van Mieghem [12] and study manufacturing costs explicitly. Their model is for discrete-time, continuous-capacity-expansion and multi-product case whereas the current paper proposes a model for continuous-time, discrete-capacity-expansion and single-product case. Angelus, Porteus and Wood [1] consider semiconductor fab capacity expansion with fixed costs, and stochastically increasing and correlated demand and apply inventory theory. They show that the optimal expansion policy is  $(s,S)$  type where both parameters depend on the most recently observed demand. Rocklin, Kashper and Varvaloucas [24] prove the optimality of  $(s,S)$  capacity expansion/contraction policies under certain conditions on the cost function and for a specific case (when demand exceeds the capacity, capacity is installed in an amount at least as large as the capacity deficit).

Rajagopalan, Singh and Morton [23] study the replacement of old vintage machines with new ones, under both certain and uncertain technology arrival times, and with deterministic, nondecreasing demand. They show some structural properties of the optimal solution and exploit those with a dynamic program. In competitive markets, companies pay attention to each other's capacity expansions to avoid too much slack capacity. Using game-theoretic techniques, Bashyam [2] models a duopolistic market with a few stochastic demand scenarios under two cases. In the first (second) case, each company makes decisions without (with) the knowledge of the

other company's decisions. The learning effect sometimes is not negligible, bringing per-unit costs and manufacturing times down. Under the learning effect, Hiller and Shapiro [15] provide a mixed integer programming formulation of capacity expansion where concave manufacturing costs are approximated by piecewise linear functions.

Although capacity expansions can be made at any time, quite a few models discretize time. Ong and Adams [21] examine the effects of granularity of time on cost for three cases: certain demand without shortage, certain demand with shortage, and uncertain demand. It is noted that the uncertain demand case is more sensitive to time granularity. The study advises using nonuniform granularity i.e., decisions are made more frequently in the short run than in the long run. In addition to making time discrete, another modeling practice is curtailing the decision horizon. It is valuable to know the shortest decision horizon such that the first period's decisions do not change in response to events beyond that (decision) horizon. Bean and Smith [3] establish some criteria for the existence of a finite decision horizon and provide an algorithm to calculate it.

### 3 Multiple Machine Capacity Expansion and Contraction Model

In this section we will provide a mathematical description of the model that considers machine expansions and contractions. We consider a single product family. The family demand at time  $t$  is  $D_t(\omega)$  ( $\omega \in \Omega$ ), a nonnegative and bounded stochastic process over  $[0, T]$  where  $T$  is the length of the decision horizon. We assume that  $D_t$  first stochastically increases and then stochastically decreases, i.e.  $P(D_t \geq y)$  is unimodal for any  $y \geq 0$ . We can set  $P(D_t \geq y)$  to increase (decrease) to model the situation where demand only expands (contracts).

Let  $[0, S]$  and  $[S, T]$  be the expansion and contraction intervals, such an  $S$  exists as long as  $P(D_t \geq y)$  is unimodal. If  $S$  is to depend on demand scenarios, we consider the conditional distribution of demand  $[D_t|S]$  for every scenario. It may be convenient to specify our demand process in two steps, first specify  $S$  and then  $[D_t|S]$ . From these, we can construct  $D_t$  via unconditioning as  $E_S[D_t|S]$ . Fortunately,  $P(D_t \geq y)$  can still be unimodal even when  $S$  depends on specific demand scenarios. For example, let  $T = 4$  and suppose  $[D_t|S = 1]$  is  $U(0, 1)$  if  $t \in [0, 1] \cup [2, 3]$ , it is  $U(0, 2)$  if  $t \in [1, 2]$  and it is zero everywhere else. Suppose that  $[D_t|S = 2] = [D_{t-1}|S = 1]$  and  $[S = 1]$  ( $[S = 2]$ ) happens with probability  $p_1$  ( $p_2$ ). Then it follows that  $P(D_t \geq y) = E_S P(D_t \geq y|S)$  is unimodal with maximum at  $S = 1$  ( $S = 2$ ) if  $p_1 \geq p_2$  ( $p_1 \leq p_2$ ). A sufficient condition for concavity (so unimodality) of  $P(D_t \geq y)$  in  $t$  is that each  $P(D_t \geq y|S)$  is concave. In summary, in an application where expansion interval depends on demand scenarios, we can still have unimodal  $P(D_t \geq y)$  and that is all we

assume: We allow for some demand sample paths to increase (decrease) after (before)  $S$ .

We consider  $M$  machine groups indexed by  $i$ , and we assume that all machines within a given group have the same capacity. If a machine of type  $i$  is purchased at time  $t$  then the machine is available at time  $t + L(i)$ .  $L(i)$  is nonnegative machine installation lead time for machines in group  $i$ , it can include purchase and process set up/qualification lead times. After  $t + L(i)$ , the machine capacity is constant at  $c_i$  units per time. When a machine is retired at time  $t$ , it immediately becomes unavailable at time  $t$ . We use machine purchases (retirements) for capacity expansion (contraction) as demand fluctuates. Let  $n_i(t)$  ( $n_i(t) \geq 0$ ) represent the number of type- $i$  machines available at  $t$ .  $n_i(t)$  is the number of machines (of type  $i$ ) existed initially plus those purchased by  $t$ , minus those retired by  $t$ . The overall capacity at time  $t$ ,  $K_t$ , can be expressed as

$$K_t = \min\{c_i \cdot n_i(t) : i = 1..M\}$$

Thus  $K_t$  is a step function. Figure 1 depicts the capacity functions  $c_i * n_i(t)$  for two machine groups, and a realization of the demand  $D_t$ . The vertical bars in Figure 1 stand for the amount produced at time  $t$ , i.e.  $\min\{D_t, K_t\}$ .

– **Figure 1** –

We model two kinds of costs, capacity costs and lost-sales costs. Capacity costs include the cost of financing the purchase and installation of machines, and maintenance costs for the machines. We call capacity costs *regular* if postponing the purchase or earlier retirement of a machine does not increase them. This is typically the case.

The lost-sales cost measures service - the company's ability to meet market demand. The measure we use in most of this paper is the expected value of the lost sales incurred during the time horizon  $[0, T)$ . Other service measures could be used, such as the expected number of weeks during which demand is fully met. We call a service measure *regular* if it depends on the machine purchase/retirement schedule only through the capacity  $K_t$ , and postponing the purchase or earlier retirement of a machine cannot decrease it. The measures described above are regular. We limit attention to regular service measures.

The installation of the  $k$ -th machine of type  $i$  at time  $t$  will raise the capacity of machine group  $i$  to  $a(i, k) := c_i n_i(t)$ , recall that  $n_i(t)$  also accounts for existing machines. If this  $k$ -th machine is purchased at time  $t(i, k) - L(i)$ , capacity goes up at the *availability time*  $t(i, k)$ . Thus  $L(i) \leq t(i, k) \leq T$ . For machines that are available initially,  $t(i, k) = 0$ . Similarly, the retirement of the  $k$ th machine of type  $i$  at time  $u(i, k)$  will

lower the capacity of machine group  $i$  from  $a(i, k)$  down to  $a(i, k - 1)$ . Naturally,  $0 \leq t(i, k) \leq u(i, k) \leq T$ . If  $t(i, k) = u(i, k)$  then the  $k$ -th machine of type  $i$  is never purchased.

Let  $\mathcal{K}$  be an upper bound on the capacity that we would consider installing before time  $S$ . The set of machines, including the existing machines,  $\{(i, k) : a(i, k) < \mathcal{K}\}$  is sorted in increasing order of  $a(i, k)$  and indexed by  $n, 1 \leq n < N$ , so that  $a(i_n, k_n) =: a_n \leq a_{n+1}$  and let  $t_n := t(i_n, k_n)$ . Ties are broken arbitrarily. Define  $L_n := L(i_n)$  and  $u_n := u(i_n, k_n)$ . We set  $t_n = 0$  for all existing machines,  $t_N = u_N$ ,  $u_0 = T$  and  $a_N = \mathcal{K}$ . A *bottleneck policy (BP)* is a policy in which machines are made available for production in increasing order of  $n$  and are retired on a decreasing order of  $n$ , i.e.,  $t_n \leq t_{n+1}$  and  $u_{n+1} \leq u_n$ .

**Lemma 1** *If the machine purchasing problem has a regular cost function, a bottleneck policy minimizes the expected cost.*

Proof: Suppose that we are given an instance of the machine purchasing problem and a set of machine availability and retirement times  $\{t_n, u_n : 1 \leq n \leq N, 0 \leq t_n \leq u_n \leq T\}$ . First, let  $n$  be the smallest integer such that  $t_n > t_{n+1}$ . We set  $t_{n+1}$  equal to  $t_n$ . The capacity  $K_t, 0 \leq t < T$  is not effected. The capacity costs are regular so they do not increase, and for every sample path of  $D(t)$  the service cost does not change. Iterating this procedure we put availability times in BP order without increasing the cost of the schedule. Now let  $n$  be the smallest integer such that  $u_{n+1} > u_n$  and set  $u_{n+1} = u_n$ . Iterating this procedure, we can obtain a BP policy that does not cost more than the initial solution, for every sample path.  $\square$

The bottleneck policy can be implemented in a manner such that the most recently purchased machine is not retired first when demand starts falling. BP merely says that the most recently purchased machine and the first machine to retire must be of the same type. This distinction is important to avoid retiring new machines which may be slightly more efficient than older ones in practical applications, although our model assumes that all characteristics of the machines of the same type are the same.

For now we assume that there are no existing machines, we will relax this assumption later on. We restrict attention to BPs. This determines the sequence in which machines are brought on line and are retired, but we still need to select the availability times  $t_n$  and retirement times  $u_n$ , subject to the constraint

$$0 \leq t_1 \leq t_2 \leq \dots \leq t_N = u_N \leq \dots \leq u_2 \leq u_1 \leq u_0 = T.$$

The capacity of the system over  $[t_n, t_{n+1}) \cup (u_{n+1}, u_n]$  is  $a_n$ , see Figure 1.

We use  $S(t_1, \dots, t_{N-1}, u_{N-1}, \dots, u_1)$ , the expected value of the total unmet demand in  $[0, T)$  as our service measure. Let  $n_{D_t}(a) := E[(D_t - a)^+]$ , the expected amount by which the demand at time  $t$  exceeds  $a$ . Then

$$\begin{aligned}
S(t_1, \dots, t_{N-1}, t_N = u_N, u_{N-1}, \dots, u_1) &= \\
&= \sum_{n=1}^N \left( \int_{t=t_{n-1}}^{t_n} n_{D_t}(a_{n-1}) dt + \int_{t=u_n}^{u_{n-1}} n_{D_t}(a_{n-1}) dt \right) \\
&= \sum_{n=1}^N \left( \int_{t=t_{n-1}}^{t_n} \sum_{k=n}^N \{n_{D_t}(a_{k-1}) - n_{D_t}(a_k)\} + n_{D_t}(a_N) dt \right) \\
&\quad + \sum_{n=1}^N \left( \int_{t=u_n}^{u_{n-1}} \sum_{k=n}^N \{n_{D_t}(a_{k-1}) - n_{D_t}(a_k)\} + n_{D_t}(a_N) dt \right) \\
&= \sum_{k=1}^N \sum_{n=1}^k \left( \int_{t=t_{n-1}}^{t_n} \{n_{D_t}(a_{k-1}) - n_{D_t}(a_k)\} dt + \int_{t=u_n}^{u_{n-1}} \{n_{D_t}(a_{k-1}) - n_{D_t}(a_k)\} dt \right) \\
&\quad + \sum_{n=1}^N \int_{t=t_{n-1}}^{t_n} n_{D_t}(\mathcal{K}) dt + \sum_{n=1}^N \int_{t=u_n}^{u_{n-1}} n_{D_t}(\mathcal{K}) dt \\
&= \sum_{k=1}^N \left( \int_{t=0}^{t_k} \{n_{D_t}(a_{k-1}) - n_{D_t}(a_k)\} dt + \int_{t=u_k}^T \{n_{D_t}(a_{k-1}) - n_{D_t}(a_k)\} dt \right) + \int_{t=0}^T n_{D_t}(\mathcal{K}) dt \\
&= \sum_{k=1}^{N-1} \eta_k(t_k) + \zeta_k(u_k) + SC
\end{aligned}$$

where

$$\eta_k(t_k) := \int_{t=0}^{t_k} \{n_{D_t}(a_{k-1}) - n_{D_t}(a_k)\} dt, \quad \zeta_k(u_k) := \int_{t=u_k}^T \{n_{D_t}(a_{k-1}) - n_{D_t}(a_k)\} dt$$

and

$$SC := \int_{t=0}^T n_{D_t}(\mathcal{K}) + n_{D_t}(a_{N-1}) - n_{D_t}(a_N) dt = \int_{t=0}^T n_{D_t}(a_{N-1}) dt.$$

Since  $SC$  is a sunk cost, independent of the timing of machine purchases, we will not include it in our objective function. Actually by choosing  $\mathcal{K}$  sufficiently large,  $SC$  can be brought down to zero. Note that the service measure is a separable and additive function of  $\{t_n, u_n : 1 \leq n < N\}$ .

**Example:** Suppose that  $T = 1$  and the product demand is Uniformly distributed i.e.,  $D_t \sim U(0, t)$  for  $t \in [0, 1]$ . Since the demand grows, the capacity is not contracted. First note that,

$$E[(D_t - a)^+] = \begin{cases} t/2 - a + 0.5a^2/t & \text{if } t \geq a \\ 0 & \text{otherwise} \end{cases} \quad \text{for } 0 \leq a, t \leq 1.$$

Since  $\eta_n(t_n) = 0$  for  $t_n < a_{n-1}$ , we study  $\eta_n(t_n)$  only over  $t_n \geq a_{n-1}$ :

$$\eta_n(t_n) = \int_0^{t_n} n_{D_t}(a_{n-1}) - n_{D_t}(a_n) dt = \begin{cases} (a_n - a_{n-1})t_n - 0.5(a_n^2 - a_{n-1}^2) \ln t_n + Constant & \text{if } a_n \leq t_n \\ t_n^2/4 - a_{n-1}t_n + 0.5a_{n-1}^2 \ln t_n + Constant & \text{if } a_{n-1} \leq t_n < a_n \end{cases}$$

where  $Constant$ 's do not change with the availability time  $t_n$  so they can be ignored. We will return to this example in the next section.

We express the capacity costs as

$$\sum_{n=1}^{N-1} G_n \cdot 1_{(0 < t_n < u_n)} + h_n(u_n - t_n) + H_n \cdot 1_{(t_n < u_n < T)}$$

where  $G_n$  is the time-independent fixed cost (perhaps a portion) of buying and installing machine  $n$ .  $G_n$  is incurred if the machine is purchased, i.e.,  $0 < t_n < u_n$ .  $H_n$  is the salvage cost. It can take positive or negative values but typically  $-H_n \leq G_n$ .  $H_n$  is incurred if the machine is bought and retired before  $T$ , i.e.,  $t_n < u_n < T$ .  $h_n$  is an arbitrary constant. It captures usage  $(u_n - t_n)$  dependent costs: such as the amortized cost of the capital (perhaps a portion of it) required to purchase and install the  $n$ th machine, plus the periodic maintenance cost. We coin the term machine rent for  $h_n$ . Another interpretation of this cost structure is via subcontracting; Machine  $n$  is subcontracted at  $t_n$  by paying the fixed transaction cost  $G_n$ . Machine is used until  $u_n$  by paying a rent of  $h_n$  per unit time. Subcontracting is terminated at time  $u_n$  by incurring the fixed transaction cost  $H_n$ .

The problem of expanding / contracting facility capacities in a supply chain can be attacked with this cost structure. Then machine types correspond to production or transportation facilities and each machine corresponds to a chunk of capacity. BP order can be based on the minimum capacity facility in the chain. Moreover, this cost structure can be used to study a simplified version of Unit Commitment Problem ([25]) of power generators. Then machine purchases (retirements) will correspond to turning the generators on (off). In order to apply this model to Unit Commitment Problem, a good sequence for the activation and the retirement of generators must be determined in advance. If that is not possible, this model can be used to evaluate optimal costs for given sequences, perhaps as a part of a (sequence) search heuristic. Note that decision horizons for unit commitment problem is significantly shorter than machine or facility capacity expansion / contraction problems. Also  $S$  and  $T$  are easier to find out in the unit commitment problem as demand for power peaks early in the evening and dwindles early in the morning.

For now, we assume that  $G_n = H_n = 0$  for all  $n$ . In the next section, we will discuss how to treat nonzero purchase costs and salvage values. Recall that  $L_n, 0 \leq L_n < T$  is the lead time required for purchase, installation and qualification of the  $n$ th new machine. Let  $B(t \geq L) := \infty$  for  $t < L$  and  $B(t \geq L) := 0$  otherwise. We model the total cost associated with the purchase and retirement of  $n$ th machine as

$$f_n(t_n) := \eta_n(t_n) - h_n t_n + B(t_n \geq L_n), \quad 1 \leq n \leq N, \quad 0 \leq t_n \leq T$$

$$g_n(u_n) := \zeta_n(u_n) + h_n u_n, \quad 1 \leq n \leq N, \quad 0 \leq u_n \leq T \tag{1}$$

Without loss of generality, we assume that capacity costs  $G_n, h_n$  and  $H_n$  are already scaled by the unit lost sales cost so that we can add capacity and lost sales costs without further manipulation. We use (1) in our computational study, but our algorithm does not require  $f_n(\cdot)$  or  $g_n(\cdot)$  to have any particular algebraic form. The machine capacity problem ( $\mathcal{P}$ ) then becomes

$$(\mathcal{P}) \quad \min \left\{ \sum_{n=1}^{N-1} f_n(t_n) + g_n(u_n) : 0 \leq t_1 \leq t_2 \leq \dots \leq t_{N-1} \leq t_N = u_N \leq u_{N-1} \leq \dots \leq u_1 \leq u_0 = T \right\}.$$

We break ties by favoring larger values of  $t_n$ . It is evident from ( $\mathcal{P}$ ) that all  $t_n$  and  $u_n$  are determined at  $t = 0$  against the demand distribution available then. When the demand distribution is updated, we resolve ( $\mathcal{P}$ ). In the execution mode, at any time we implement only those expansions / contractions that have to be done before the next demand update. In summary, we apply a rolling horizon approach.

The next section describes an algorithm for the computation of optimal machine purchasing and retirement times. The algorithm for computing optimal times requires that cost functions are convex and concave only once in  $[0, T]$ .

**Lemma 2** *Cost  $f_n(t)$  ( $g_n(t)$ ) is convex over expansion (contraction) interval and concave in the remaining.*

Proof: Since  $P(D_t \geq y)$  is unimodal, let  $S$  be the first time this probability is maximized, i.e. demand stochastically increases over the expansion interval  $[0, S]$  and decreases in the remaining. Since our result deals with concavity and convexity, we can ignore linear parts of  $f_n$  and  $g_n$  and focus on  $\eta_n(t)$  and  $\zeta_n(t)$ . We provide the proof for only  $f_n$ , similar arguments apply to  $g_n$  as well. It suffices to study the derivative of  $\eta_n(t)$  in  $t$ .

$$\frac{d\eta_n(t)}{dt} = n_{D_t}(a_{n-1}) - n_{D_t}(a_n) = \int_{a_{n-1}}^{a_n} P(D_t \geq y) dy. \quad (2)$$

The integrand is nondecreasing over  $0 \leq t \leq S$  and nonincreasing afterwards by the definition of  $S$ . Thus  $\eta_n(t)$  is convex until  $t = S$  and is concave afterwards.  $\square$

Unimodality plays a central role in Lemma 2, however it may not hold over the entire horizon  $[0, T]$ . We will later treat the case where  $P(D_t \geq y)$  is unimodal within demand cycles which span the entire horizon when concatenated together. We use Lemma 2 to restrict machine expansions (contractions) to demand expansion (contraction) intervals. This observation is formalized with the next Lemma.

**Lemma 3** *There exists an optimal solution where all machines are purchased (retired) during demand expansion (contraction) interval.*

Proof: Formally, there exists an optimal solution  $\{t_n^*, u_n^*\}$  such that  $0 \leq t_n^* \leq S \leq u_n^* \leq T$ . Suppose we have an optimal solution with  $n_0$  as the smallest index such that  $S < t_{n_0}^* \leq T$ . Let  $C = \{n_0, n_0 + 1, \dots, N\}$  be the set of all expansions happening at or after  $t_{n_0}^*$ . We argue that  $\sum_{n \in C} f_n(t_n^*) + g_n(u_n^*)$  cannot be increased by pulling both expansion and contraction times towards  $S$  i.e. by setting  $t_n = t_n^* - (t_{n_0}^* - S)$  and  $u_n = u_n^* - (t_{n_0}^* - S)$  for all  $n \in C$ . The costs associated with other expansions and contractions outside  $C$  are not affected with this modification. Since  $u_n - t_n$  does not change with this modification, let  $x_n = u_n - t_n = u_n^* - t_n^* \geq 0$ , then

$$\sum_{n \in C} f_n(t_n) + g_n(t_n + x_n) = \sum_{n \in C} \int_0^{t_n} n_{D_t}(a_{n-1}) - n_{D_t}(a_n) dt + \int_{t_n+x_n}^T n_{D_t}(a_{n-1}) - n_{D_t}(a_n) dt + h_n x_n$$

This cost cannot increase as we decrease  $t_n^*$  because

$$\frac{d}{dt} \{f_n(t) + g_n(t+x)\} = n_{D_t}(a_{n-1}) - n_{D_t}(a_n) - \{n_{D_{t+x}}(a_{n-1}) - n_{D_{t+x}}(a_n)\} = \int_{a_{n-1}}^{a_n} P(D_t \geq y) - P(D_{t+x} \geq y) dy \geq 0.$$

The last inequality follows from stochastically decreasing demand for  $t \geq S$ . Since our modification does not increase costs, it yields a better or equivalent solution with  $0 \leq t_n^* \leq S$ . The argument for  $S \leq u_n^* \leq T$  is symmetric and is omitted.  $\square$

Lemma 3 implies that capacity expansions (contractions) happen during demand growth (fall), thus without loosing any generality we can set

$$t_N = S = u_N. \tag{3}$$

Using this, we can split  $(\mathcal{P})$  into expansion  $(\mathcal{P}^E)$  and contraction  $(\mathcal{P}^C)$  problems as

$$\begin{aligned} (\mathcal{P}^E) \quad & \min \left\{ \sum_{n=1}^{N-1} f_n(t_n) : 0 \leq t_1 \leq t_2 \leq \dots \leq t_{N-1} \leq t_N = S \right\}, \\ (\mathcal{P}^C) \quad & \min \left\{ \sum_{n=1}^{N-1} g_n(u_n) : S = u_N \leq u_{N-1} \leq \dots \leq u_1 \leq u_0 = T \right\}. \end{aligned}$$

In the next section we will illustrate how to obtain the optimal solution by solving  $(\mathcal{P}^E)$  and  $(\mathcal{P}^C)$  separately. In the remaining of the paper, we count machine retirements starting from  $N + 1$  so that we can use the same index  $n$  for machine purchases and retirements;  $n$  indicates a purchase (retirement) if  $n \leq N$  ( $n > N$ ). Then we can use  $f_n$  instead of  $g_n$  when  $n > N$ . We will also use  $t$  to denote both availability and retirement times when the meaning is clear from the context.

## 4 Calculation of Optimal Availability and Retirement Times

The capacity increment that results from installing machine  $n$  is  $a_n - a_{n-1}$ . Since machines are usually of different types, the ratio of  $a_n - a_{n-1}$  to the cost of machine  $n$  can be very small or very large. In isolation, machines with small (large) capacity increment to cost ratio would be made available late (early). However, because of BP order, availability times of machines collide and certain sets of machines share the same availability time. Then the total capacity increment that results from making a set of machines simultaneously available becomes commensurate with the total cost of these machines. Similar comments can be made for machine retirements. Consequently, it is reasonable to expect that some machines are made available simultaneously while some others are retired simultaneously. We will call these group of machines clusters.

A *cluster* is a set of consecutive machines  $C := \{p, p + 1, \dots, q\}$ , let  $[p, q] = \{p, p + 1, \dots, q\}$  for notational convenience. If  $C = [p, q]$  is an expansion (contraction) cluster then  $1 \leq p \leq q \leq N$  ( $N < p \leq q \leq 2N$ ). We justify not considering machine clusters that span from the expansion problem to the contraction problem by Lemma 3. We define  $\min(C) := \min\{n : n \in C\}$  and  $\max(C) := \max\{n : n \in C\}$ . The *root* of cluster  $C$  is  $\min(C)$ . Let

$$f_C(t) := \sum_{n \in C} f_n(t).$$

The availability or retirement time associated with a given cluster  $C$  is computed by solving the following problem, called  $(\mathcal{P}_C)$ .

$$(\mathcal{P}_C) \quad \left\{ \begin{array}{ll} \min\{f_C(t_C) : 0 \leq t_C \leq S\} & \text{if } \max(C) \leq N \\ \min\{f_C(t_C) : S \leq t_C \leq T\} & \text{if } \min(C) > N \end{array} \right\}.$$

Restriction of  $f_C$  to domain  $[0, S]$  or  $[S, T]$  make it convex. Thus, solving  $(\mathcal{P}_C)$  is simply minimization of a convex function over a closed domain. Problems  $(\mathcal{P}^E)$  and  $(\mathcal{P}^C)$  are exactly the same as the ones solved to optimality with the PAV algorithm in Best, Chakravarti and Ubhaya [7]. We add some steps to the PAV algorithm and name it as the Cluster Algorithm shown in Table I.

- Table I -

If  $J$  is a set of clusters that constitute a partition of  $\{1, \dots, N\}$ , then  $C(n)$  is the cluster in  $J$  containing the  $n$ th machine. Thus  $\min(C(n)) \leq n \leq \max(C(n))$  for all  $n$ . Remember that we are using the same index  $n$  for machine purchases and retirements. Let  $(\mathcal{P}_{r,s})$  be a smaller version of  $(\mathcal{P})$  where purchases or retirements only in  $[r, s]$  are considered. Since we consider expansion and contraction problems separately, there is no

need to consider a subproblem  $(\mathcal{P}_{r,s})$  where  $r \leq N < s$ . Then, we choose  $r$  and  $s$  such that either  $r, s \leq N$  or  $r, s > N$  so that  $(\mathcal{P}_{r,s})$  is a smaller version of type  $(\mathcal{P}^E)$  or  $(\mathcal{P}^C)$ . Consequently, we have families of subproblems  $\{(\mathcal{P}_{r,s}) : 1 \leq r \leq s \leq N\}$  for expansion and  $\{(\mathcal{P}_{r,s}) : N+1 \leq r \leq s \leq 2N\}$  for contraction. In subproblems,  $\{(\mathcal{P}_{r,s}) : 1 \leq r \leq s \leq N\}$ ,  $t_n = 0$  for  $n < r$  and  $t_n = S$  for  $n > s$ . In subproblems,  $\{(\mathcal{P}_{r,s}) : N+1 \leq r \leq s \leq 2N\}$ ,  $t_n = S$  for  $n < r$  and  $t_n = T$  for  $n > s$ . Fixing these times makes purchases and retirements before  $r$  and after  $s$  irrelevant for cost computations, therefore cost functions associated with these purchases and retirements can be ignored. Let  $R(r, s)$  be the roots of a set of clusters that give rise to an optimal solution to  $(\mathcal{P}_{r,s})$ . Let the set of clusters deduced by  $R(r, s)$  be

$$\{C : \min(C) \in R(r, s) \text{ and } \max(C) + 1 \in R(r, s) \cup \{s + 1\}\}.$$

This set is a partition of the machine set  $\{r, \dots, s\}$  and all clusters in this set start with a root in  $R(r, s)$  and end before the next root. Clusters deduced by  $R(r, s)$  of the Cluster Algorithm are optimal for  $(\mathcal{P})$  by [7]. Best and Chakravarti [6] also study  $(\mathcal{P})$  under the name *isotonic regression*.

An important property of  $(\mathcal{P})$  is that if we know where the breaks between clusters are, we can optimize each cluster separately by solving a problem of type  $(\mathcal{P}_C)$ . From the definition of  $(\mathcal{P})$ ,  $(\mathcal{P}^E)$ ,  $(\mathcal{P}^C)$  and (3), we can solve  $(\mathcal{P})$  by solving  $(\mathcal{P}^E)$  and  $(\mathcal{P}^C)$  separately. Optimal clusters for  $(\mathcal{P}^E)$  and  $(\mathcal{P}^C)$  can then be put all together to construct a solution to  $(\mathcal{P})$ :

**Corollary 1** *Optimal solution to  $(\mathcal{P})$  can be deduced by patching the optimal clusters of  $(\mathcal{P}^E)$  and  $(\mathcal{P}^C)$ .*

We now examine the running time of the Cluster Algorithm. First assume that (1) holds, then

$$\sum_{n \in C} \eta_n(t) = \int_{\tau=0}^{\tau=t} \{n_{D_\tau}(a_{\min(C)-1}) - n_{D_\tau}(a_{\max(C)})\} d\tau \text{ if } \max(C) \leq N,$$

$$\sum_{n \in C} \zeta_n(t) = \int_{\tau=t}^{\tau=T} \{n_{D_\tau}(a_{\min(C)-1}) - n_{D_\tau}(a_{\max(C)})\} d\tau \text{ if } \min(C) > N.$$

Thus  $f_C(t)$  can be evaluated in time that is constant in  $|C|$ . Note that  $f_C(t)$  is convex and we are minimizing it over a bounded interval. Consequently, each minimization takes time  $T^c$  which is constant in the number of machines in  $C$  but increases with the desired accuracy of the minimizer. As discussed in [7], at most  $O(N)$  minimizations are performed with the Cluster algorithm so it runs in  $O(N T^c)$ .

**Example continued:** We revisit the example of uniform product demand presented before. Suppose that machines A and B are both needed to manufacture the product and a single A (B) machine has 0.3 (0.4) units

of capacity. Further suppose that currently 1 A and 1 B are available so  $a_0 = 0.3$ . The bottleneck machine is A. After installing a single A at  $t(A, 1)$ , B becomes the bottleneck with capacity 0.4 so  $a_1 = 0.4$ . After the first installation of B at  $t(B, 1)$ , A becomes the bottleneck with capacity 0.6 so  $a_2 = 0.6$ . Installing machines in BP order  $a_3 = 0.8$  with 3 A's and 2 B's, and  $a_4 = 0.9$  with 3 A's and 3 B's. Suppose that  $\mathcal{K} = 1$  so we do not consider installing the third A machine and  $N = 5$ . We use bottleneck policy to get:

$$0 \leq [t_1 := t(A, 1)] \leq [t_2 := t(B, 1)] \leq [t_3 := t(A, 2)] \leq [t_4 := t(B, 2)] \leq [t_5 := t(A, 3)] = [S = 1]$$

Since the demand is stochastically increasing,  $S = T$  and no machines are retired. Suppose that all machine capacity costs are zero except for the per-time operation costs and rents for machines A and B, namely  $h_n = 0.05$  for all machines.

Using the  $\eta_n(t)$  obtained before in the context of uniform demand, for any  $C \subseteq \{1 \dots 5\}$ ,

$$f_C(t) = \left\{ \begin{array}{ll} (a_{\vee C} - a_{\wedge C} - h(C))t - 0.5(a_{\vee C}^2 - a_{\wedge C}^2) \ln t + Constant & \text{if } a_{\vee C} \leq t \\ t^2/4 - (a_{\wedge C} + h(C))t + 0.5a_{\wedge C}^2 \ln t + Constant & \text{if } a_{\wedge C} \leq t < a_{\vee C} \\ -h(C)t & \text{if } t < a_{\wedge C} \end{array} \right\}$$

where  $\vee_C = \max(C)$ ,  $\wedge_C = \min(C) - 1$  and  $h(C) = \sum_{n \in C} h_n$ . In this example,  $h_n$  values are chosen large enough so that  $t_C \geq a_{\vee C}$  for all the clusters  $C$  considered below. Then the optimal availability time for cluster  $C$  is

$$t_C = \min\left\{\frac{0.5(a_{\vee C}^2 - a_{\wedge C}^2)}{a_{\vee C} - a_{\wedge C} - h(C)}, 1\right\}.$$

Now we will use the Cluster algorithm to solve this 5 machine problem,  $(\mathcal{P}) = (\mathcal{P}_{1,5})$  so  $r = 1$  and  $s = 5$ . At the start  $R = \{1, 2, 3, 4, 5\}$  and  $n = 2$ . When we are solving subproblem  $(\mathcal{P}_{1,2})$ ,  $t_1 = 0.7 = t_{C(k)}$  and  $t_2 = 0.66 = t_{C'}$  so we unite machine expansions into  $C = \{1, 2\}$  and set  $R = \{1, 3, 4, 5\}$ . In the next two subproblems  $(\mathcal{P}_{1,3})$  and  $(\mathcal{P}_{1,4})$ ,  $t_{1,2} = 0.675 = t_{C(k)}$  and  $t_3 = 0.93 = t_{C'}$ , and  $t_3 = 0.93 = t_{C(k)}$  and  $t_4 = 1 = t_{C'}$  so no other unite operations are executed. Thus root set  $R = \{1, 3, 4, 5\}$  remains unaltered and it leads to the optimal machine clusters  $\{1, 2\}$ ,  $\{3\}$ ,  $\{4\}$  and  $\{5\}$ . In terms of expansion times,  $t(A, 1) = t(B, 1) = 0.675$ ,  $t(A, 2) = 0.93$ ,  $t(B, 2) = t(A, 3) = 1$ , the second B machine is not purchased within the planning horizon.

Examining the Cluster Algorithm, we see that we are solving a series of problems of type  $(\mathcal{P}_{r,s})$ ,  $1 \leq r \leq s$ , while solving a single  $(\mathcal{P}_{1,s})$ . Thus, all  $(\mathcal{P}_{r,s})$  for  $1 \leq r \leq s$  can be solved in  $O(N T^e)$ . Then, the family of problems  $\{(\mathcal{P}_{r,s}) : 1 \leq r \leq s \leq N\}$  can be solved in  $O(N^2 T^e)$ . A similar argument yields that all problems of type  $\{(\mathcal{P}_{r,s}) : N < r \leq s \leq 2N\}$  can be solved in  $O(N^2 T^e)$ . These observations show that steps A and B of the Expansion/Contraction (EC) algorithm in Table II can be completed in  $O(N^2 T^e)$ .

- Table II -

We discuss the validity of EC algorithm. Let  $s$  be the index of the last machine installed and  $r$  be the index of the last machine retired (counting retirement indices starting from 1, i.e.  $r > N$ ). In order to retire a machine, it must have been installed earlier, i.e.,  $2N + 1 - s \leq r$ . Suppose that there are no fixed purchase or salvage costs ( $G_n = H_n = 0$  for all  $n$ ). Let  $e_{i,s}$  be the cost of the expansion problem if first  $i$  machines are already existing at time 0 and  $s$  is the last machine to be installed. Let  $c_{2N+1-s,r}$  be the cost of the contraction problem where  $2N + 1 - s$  is the index of the first machine retired (by BP order  $s$  is the last machine installed) and  $r$  is the index of the last machine retired.

As promised earlier, we generalize the EC algorithm for  $G_n$  and  $H_n$  are nonzero, and for the case of initially existing machines. Let  $I_0$  be the existing machine with the largest BP index and assume that the set of existing machines respect the BP, i.e., if  $n$  is an existing machines so are  $[1, n - 1]$ . It may be profitable to retire some of the existing machines at time  $t = 0$  and make them available at a later time. This operation will save some machine rent and can bring in some salvage value but will require to pay for the fixed purchase cost. Formally, let  $[I + 1, I_0]$  be the set of machines to retire initially, then the cost of expansion problem if  $[I + 1, I_0]$  are retired at  $t = 0$  is:

$$\bar{e}_{I,s} = e_{I,s} + \sum_{n=I+1}^{I_0} H_n + \sum_{n=I+1}^s G_n \text{ for } I \leq I_0, I \leq s$$

Note that we incur salvage costs when retiring existing machines. We minimize  $\bar{e}_{I,s}$  over  $1 \leq I \leq I_0$  and let

$$e_s = \min_{1 \leq I \leq I_0} \bar{e}_{I,s} \text{ and } I^*(s) = \arg \min_{1 \leq I \leq I_0} \bar{e}_{I,s}.$$

$e_s$  is the optimal expansion cost with the option of retiring some of the existing machines at  $t = 0$  if  $s$  is the last machine to be installed. To achieve the cost  $e_s$ , we retire all the machines in  $[I^*(s), I_0]$  at  $t = 0$ .

In an optimal solution to  $(\mathcal{P})$ , naturally there is a last machine installed and a last machine retired. If we know the indices of these machines,  $s$  and  $r$ , the optimal cost of  $(\mathcal{P})$  is

$$e_s + c_{2N+1-s,r} + \sum_{n=2N+1-s}^r H_n \text{ where } 1 \leq s < N < 2N + 1 - s \leq r \leq 2N. \quad (4)$$

If  $e_s$  and  $c_{2N+1-s,r}$  are available, this cost calculation can be done in  $O(N^2)$ . Note that this is step C of EC Algorithm. Clearly step D takes only  $O(N T^c)$ . In summary, EC Algorithm takes  $O(N^2 T^c)$ . It also gives the optimal solution to  $(\mathcal{P})$  because it searches over all possible  $s$ ,  $r$  and  $I^*(s)$  indices. Thus we arrive at our main theorem.

**Theorem 1** *The EC Algorithm solves the capacity expansion / contraction problem to optimality in  $O(N^2 T^c)$ .*

In the special case of no initial machine retirements and all zero salvage costs, the cost of  $(\mathcal{P})$  becomes

$$e_{I_0,s} + c_{2N+1-s,r} + \sum_{n=I_0+1}^s G_n \text{ where } I_0 \leq s < N < 2N + 1 - s \leq r \leq 2N.$$

Without machine retirements,  $e_{I_0,s}$  can be computed by solving a series of  $(\mathcal{P}_{I_0,s})$ ,  $I_0 \leq s \leq N$  in  $O(N T^c)$ . For the case of zero salvage values, let  $c_{2N+1-s,2N}$  be the optimal cost of the Contraction problem where  $2N + 1 - s$  is the index of the first machine to be retired or  $s$  is the last machine made available. In the case of zero salvage values, since the cost  $c_{2N+1-s,r^*}$  is considered correctly while finding  $c_{2N+1-s,2N}$ ,  $c_{2N+1-s,r^*} \geq c_{2N+1-s,2N}$ . We can easily argue for  $c_{2N+1-s,r^*} \leq c_{2N+1-s,2N}$  to obtain  $c_{2N+1-s,r^*} = c_{2N+1-s,2N}$ . Namely, we can get the costs correctly without solving a different family of problems for each possible last machine retired. Then the cost of  $(\mathcal{P})$  reduces to

$$e_{I_0,s} + c_{2N+1-s,2N} + \sum_{n=I_0+1}^s G_n \text{ where } I_0 \leq s < N < 2N + 1 - s \leq 2N.$$

Thus, we can simplify Step B of EC Algorithm to solve the family of problems  $\{(\mathcal{P}_{2N+1-s,2N}) : N < 2N + 1 - s \leq 2N\}$ . The Cluster Algorithm solves this family of problems in  $O(N T^c)$ . Step C involves search over only  $s$  and can be completed in  $O(N)$ . We summarize these observations with a corollary below.

**Corollary 2** *If no machines are retired initially and salvage values are all zero, EC Algorithm can be streamlined to run in  $O(N T^c)$ .*

Up to now, we assumed that existing machines respect the BP order. We now discuss how to analyze the problem if that is not the case. Suppose that machines  $[1, I_0]$  exist initially as well as machine  $j$  where  $j > I_0 + 1$  and there initially are no other machines. There are two possible actions: Either machine  $j$  is retired at time 0 or kept at least until  $S$ ; It cannot be optimal to retire machine  $j$  at  $t$  where  $0 < t < S$ . Suppose to the contrary that  $0 < t < S$ , retiring  $j$  later we pay rent  $h_j$  and save lost sales costs. Since  $0 < t$ , savings in lost sales must have balanced or exceeded the rent during  $[0, t]$ . However, because of stochastically increasing demand, lost sales savings will continue to exceed the rent as we delay  $t$  until  $S$  so  $t \geq S$ .

In order to decide whether machine  $j$  should be retired immediately or kept until  $S$ , it suffices to compare the costs of these options. If  $j$  is retired immediately, a cost of  $H_j$  is to be paid then the existing machines respect the BP order. The cost of this option is computed by adding  $H_j$  to costs in (4). When  $j$  is kept, we need to set  $a_{j-1} = a_j$  because as soon as  $j - 1$ st machine is installed the capacity becomes  $a_j$ . We remove the

$j$ th machine from Expansion problem. Note that machine  $j$ 's rent is  $h_j(u_j - t_j)$ , removing machine  $j$  from the Expansion problem causes  $t_j$  to drop from the cost expression. The remaining term  $h_j u_j$  reflects the true cost because  $u_j \geq S$  and machine  $j$ 's rent is paid during  $[0, S]$  with this option. These operations effectively make the existing machines respect the BP order and we once more compute the costs by (4).

When there are more than one existing machine (say  $j$  and  $k$ ) that destroy the BP order, the analysis above becomes more involved. It is not possible to evaluate retire and keep options for these machines separately machine by machine. In other words evaluation of whether to keep or retire machine  $j$  depends on whether machine  $k$  is kept or retired because these machines are coupled by  $a_{j-1} = a_j$  and  $a_{k-1} = a_k$  modifications suggested above. This forces us to consider all possible options together: retire  $j$ , retire  $k$ ; retire  $j$ , keep  $k$ ; keep  $j$ , retire  $k$ ; keep  $j$ , keep  $k$ . Thus, evaluation of keep and retire options is not polynomial in the number of machines violating the BP order. We close this discussion noting that the number of machines violating the BP order is limited by the number of machine types and in most practical applications there will probably be at most 8-10 machines violating the BP order.

We briefly comment on the size of the problem inputs i.e., how large  $N$  and  $T$  should be. With each machine we consider, the achievable capacity of the system increases. Since there is no point of installing capacity beyond the largest value  $D_S$  can take, an upper bound on  $N$  can be set as

$$\bar{N} = \min\{n : a_{n-1} > \sup_{\omega \in \Omega} D_S(\omega)\}.$$

Capacity planners do not need to choose  $T$ , which is generally taken as long as there are forecasts for the demand distribution. When there are multiple demand expansion and contraction cycles in  $[0, T]$ , planners must choose how many cycles to consider. In the case of zero  $G_n$  and  $H_n$ , cycles become independent. Then by the structure of  $(\mathcal{P})$  availability and retirement times in  $[0, \tau]$  ( $0 \leq \tau < T$ ) are unaffected by considering additional cycles beyond  $T$ ; as far as the immediate decisions are concerned, it is sufficient to solve a single cycle problem. For nonzero  $G_n$  and  $H_n$  cycles affect each other and planners must make a reasonable effort to consider all the demand cycles.

If several demand cycles are considered and  $G_n$  or  $H_n$  are nonzero, capacity expansions and contractions over these cycles must be optimized together. We now provide a dynamic programming formulation for this purpose, first starting with the demand characterization: Suppose there are  $K$  demand cycles and within each cycle demand stochastically increases and decreases only once. Then  $[0, T]$  can be partitioned into  $K$  expansion and contraction intervals following each other; Expansion during  $[T_{k-1}, S_k]$  and contraction during  $[S_k, T_k]$  for  $k = 1 \dots K$  where  $T_0 = 0$  and  $T_K = T$ . Many stochastic demand processes can be approximated fairly well by

processes that stochastically increase and decrease several times over  $[0, T]$ .

Our dynamic program has a stage at the end of each demand cycle. Adding these stages to the starting stage at  $t = 0$ , we index stages from 0 to  $K$ . At each stage we have an integer  $I_{k-1}$  as the state variable indicating the existing machines at the beginning of demand cycle  $k$ . For example,  $[1, I_0]$  and  $[1, I_1]$  are the set of existing machines at  $t = 0$  and  $t = T_1$ . We cast the dynamic program as a shortest path problem on a network. This network has nodes that correspond to states; we use  $(k, n)$  to denote state  $I_k = n$  at stage  $k$ ,  $0 \leq k \leq K$ , see Figure 2.

– **Figure 2** –

The network has horizontal arcs among stages and vertical arcs inside stages. The length of each horizontal arc denotes capacity expansion and contraction costs over a demand cycle. We let  $E_{i,j}^k$  be the cost of starting demand cycle  $k$  with machines  $[1, i]$  and ending it with machines  $[1, j]$ ,  $E_{i,j}^k$  does not include machine retirement costs at the beginning of cycle  $k$ . Then  $j + 1$  is the BP index of the last machine retired or  $2N - j$  is the index of the last machine retired if we start counting from  $N + 1$ . Using the the expansion and contraction costs computed with EC algorithm,

$$E_{i,j}^k = \min_{i \leq s < N} e_{i,s}^k + \sum_{n=i+1}^s G_n + c_{2N+1-s, 2N-j}^k + \sum_{n=2N+1-s}^{2N-j} H_n \text{ for } k = 1 \dots K.$$

Superscript  $k$  indicates that lost sales costs are computed against the demand during cycle  $k$ . Cost  $E_{i,j}^k$  is assigned to arc from node  $(k - 1, i)$  to  $(k, j)$ . In order to model retiring machines at the beginning of each demand cycle we add vertical arcs from node  $(k, i)$  to node  $(k, j)$  where  $i > j$  and set their length to  $\sum_{n=j+1}^i H_n$ . If the demand has continuous sample paths at the end of a cycle, there will be no need to decrease capacity at the beginning of the next cycle. Then, we do not need to include vertical arcs at any stage but stage 0. With or without vertical arcs, there will be  $O(N^2 K)$  arcs in the network.

To construct the network, we need to compute capacity expansion and contraction costs during demand cycles. In view of the running time for the EC algorithm, all arc lengths in a demand cycle can be computed in  $O(N^2 T^c)$  and in  $O(K N^2 T^c)$  for the entire network. Because of the stage structure of the network, we can compute the shortest paths to nodes primarily in the order of increasing  $k$  and secondarily in decreasing  $n$ . Each node has  $O(N)$  incoming arcs, so finding the shortest path to each node takes  $O(N)$  comparisons. In total shortest path can be found in  $O(K N^2)$  which is dominated by  $O(K N^2 T^c)$  time required to construct the network. Consequently, expansion / contraction problem over  $K$  demand cycles can be solved in  $O(K N^2 T^c)$ .

In machine and facility capacity expansion / contraction problems, a single demand cycle covers several years. Thus, demand information for several cycles will be very limited and unreliable at best. However, in unit commitment problem (where a demand cycle is a day) demand for several cycles can be captured without much difficulty; it is easier to set up the  $K$  demand cycle problem in this case.

Lastly, we illustrate the EC algorithm with real life data from the semiconductor industry. We obtained machine data from SEMATECH (SEmiconductor MAnufacturing TECHnology: [www.sematech.org](http://www.sematech.org)) databases. Machine data includes purchase prices, capacities (in number of wafers per month) and delivery lead times (in months). There are about 40 machine types, each of which is necessary to manufacture a single wafer. Machine delivery lead times range from 12 months to 24 months. Purchase prices are between \$0.4 M and \$11 M. We assumed that fixed machine purchase costs and salvage values are zero, and set the lost sales costs at \$1.2 K per wafer. Initial fab capacity is about 5900 wafers per month. We plan for capacity starting 12 months from now and ending 67 months from now, so lead times shorter than or equal to 12 months are irrelevant. In each month, demand is modelled by a trapezoid density with lower and upper bounds shown in Figure 3. The trapezoid density has three intervals over which it linearly increases, stays constant and linearly decreases. We choose these interval lengths equal so that a lower bound and an upper bound suffice to define the entire demand process. Demand is stochastically increasing for the first 30 months ( $S = 30$ ) and stochastically decreasing in the remaining 25 months ( $T = 55$ ). We plan to buy at most 82 machines in the 55 month planning horizon ( $N = 82$ ). According to the optimal solution shown in Figure 3, only 34 machines are installed in 5 different months (5 clusters) in the first 30 months of demand growth. In the remaining 25 months, 31 machines are retired in 5 different months (5 clusters) leaving a capacity of about 6600 wafers per month.

– **Figure 3** –

## 5 Conclusion

We provided a polynomial time algorithm to optimally plan for capacity expansions / contractions and illustrated its use with a real life example drawn from the semiconductor industry. We have assumed that a single product family experiences stochastic demand that first stochastically increases and then decreases. Machines are purchased while demand is growing and are retired later on. Unlike a good portion of the existing capacity expansion literature that deals with single machine types, we model a multiple machine type problem by introducing BP order for the sequence of purchases and retirements. BP order remains optimal for regu-

lar cost functions, in addition to our specific lost sales and capacity costs. We have discussed that capacity cost structure is general enough to accommodate several interpretations: machine purchases (with or without salvage values), subcontracting, simplified versions of unit commitment problem, facility capacity expansion / contraction in a supply chain. We also provide a dynamic programming formulation for optimal capacity expansion / contraction over several demand cycles.

Our model is motivated by equipment intensive industries such as the semiconductor industry and electric utilities where capacity is costly and have considerable installation lead times. Capacity expansions and contractions are the primary irreversible decisions that have long term effects on competitiveness and profitability. Capacity plans must consider risks of undercapacity and overcapacity that arise from uncertainty, which can be built in via a general stochastic demand model. Managers always strive to match capacity with demand, often by trading off service costs against capacity expansion and contraction costs. Since almost all the capacity planning work deals with expansions and ignores capacity contractions, our model fills an important gap in the literature by yielding an algorithm to automatically make this trade off. Moreover, it allows managers to study the trade off under different capacity budgets, and different demand scenarios. Then managers can use the algorithm to substantiate capacity budget requests from the upper management. They can even offer several pairs of a capacity budget and the corresponding achievable customer service to the upper management and let the upper management choose one of those pairs. Studying the trade off under different demand scenarios will show the significance of (accurate) forecasting. Then managers can determine if more effort should be spent for forecasting.

Several managerial insights can be drawn. The optimality of BP order for machine expansions and contractions is probably the most intuitive. The BP order can be used not only for capacity planning but also for execution; for example when a machine delivery is late, managers may ask for the postponement of the deliveries of all the machines coming after the late machine in the BP order. Another managerial insight is the optimality of machine clusters. Sometimes managers use capacity expansion heuristics, for example they target to meet a high percentage of the demand throughout the planning horizon. When the demand is growing continuously, heuristics dictate installing machines one by one, i.e. by spreading installations. Spreading installations can break down optimal clusters and can lead to suboptimal solutions. Besides, it actually is desirable to install and retire machines in clusters to minimize disruptions to operations.

**Acknowledgment** : This work was supported by a grants from Semiconductor Research Cooperation under the task “Modeling Random Processes” and University of Texas at Dallas.

## References

- [1] A. Angelus, E.L. Porteus and S.C. Wood (1997). *Optimal sizing and timing of capacity expansions with implications for modular semiconductor wafer fabs*. Research Paper No.1479. Graduate School of Business, Stanford University.
- [2] T.C.A. Bashyam (1996). *Competitive capacity expansion under demand uncertainty*. European Journal of Operational Research, Vol. 95: 89-114.
- [3] J.C. Bean and R.L. Smith (1985). *Optimal capacity expansion over an infinite horizon*. Management Science, Vol.31, No.12 : 1523-1532.
- [4] J.C. Bean, J.L. Higle and R.L. Smith (1992). *Capacity expansion under stochastic demands*. Operations Research Vol.40 Supp. No.2: S210-S216.
- [5] D.L. Benavides, J.R. Duley and B.E. Johnson (1999). *As good as it gets: optimal fab design and deployment*. IEEE Transactions on Semiconductor Manufacturing, Vol.12, No.3 : 281-287.
- [6] M.J. Best and N. Chakravarti (1990). *Active set algorithms for isotonic regression: a unifying framework*. Mathematical Programming, Vol 47: 425-439
- [7] M.J. Best, N. Chakravarti and V. Ubhaya (2000). *Minimizing separable convex functions subject to simple chain constraints*. SIAM Journal on Optimization, Vol.10, No.3: 658-672.
- [8] M. Çakanyıldırım and R. Roundy (2002). *SeDFAM: Semiconductor demand forecast accuracy model*. IIE Transactions, Vol.34, No.5: 449-465.
- [9] M. Çakanyıldırım and R. Roundy (1999). *Demand forecasting and capacity planning in the semiconductor industry*. Technical paper no: 1229, SORIE, Cornell University, NY. To appear in IEEE Transactions on Semiconductor Manufacturing.
- [10] M.H.A. Davis, M.A.H. Dempster, S.P. Sethi and D. Vermes (1987). *Optimal capacity expansion under uncertainty*. Advances in Applied Probability Vol.19: 156-176.
- [11] A. Dixit (1997). *Investment and employment dynamics in the short run and the long run*. Oxford Economic Papers, Vol.49 No.1: 1-20.

- [12] J.C. Eberly and J.A. Van Mieghem (1997). *Multi-factor dynamic investment under uncertainty*. Journal of Economic Theory Vol.75: 345-387.
- [13] C.O. Fong and V. Srinivasan (1986). *The multiregion dynamic capacity expansion problem: an improved heuristic*. Management Science Vol.32 No.9: 1140-1152.
- [14] J.M. Harrison and J.A. Van Mieghem (1999). *Multi-resource investment strategies: operational hedging under demand uncertainty*. European Journal of Operational Research, Vol.113, No.1: 17-29.
- [15] R.S. Hiller and J.J. Shapiro (1986). *Optimal capacity expansion planning when there are learning effects*. Management Science, Vol.32, No.9 : 1153-1163.
- [16] P.L. Jackson and R.O. Roundy (1991). *Minimizing Separable Convex Objectives on Arbitrarily Directed Trees of Variable Upper Bound Constraints*. Mathematics of Operations Research, Vol.16, Iss.3: 504-533.
- [17] E. Khmelnitsky and K. Kogan (1996). *Optimal policies for aggregate production and capacity planning under rapidly changing demand conditions*. International Journal of Production Research, Vol.34, No.7 : 1929-1941.
- [18] S. Li and D. Tirupati (1994). *Dynamic capacity expansion problem with multiple products: Technology selection and timing of capacity additions*. Operations Research, Vol.42, No 5: 958-976.
- [19] H. Luss (1982). *Operations research and capacity expansion problems: a survey*. Operations Research, Vol.30, No.5: 907-947.
- [20] A.W. Neebe and M.R. Rao (1986). *Sequencing capacity expansion projects in continuous time*. Management Science, Vol.32, No.11: 1467-1479.
- [21] S.L. Ong and B.J. Adams (1989). *The effect of discretizing the planning period on the optimal cost of capacity expansion systems*. European Journal of Operational Research, Vol.43, No.3: 292-304.
- [22] S. Rajagopalan (1998). *Capacity expansion and equipment replacement: a unified approach*. Operations Research Vol.46, No.6: 846-857.
- [23] S. Rajagopalan, M.R. Singh and E.M. Morton (1998). *Capacity expansion and replacement in growing markets with uncertain technological breakthroughs*. Management Science, Vol.44, No.1 : 12-30.

- [24] S.M. Rocklin, A. Kashper and G.C. Varvaloucas (1984). *Capacity expansion/contraction of a facility with demand augmentation dynamics*. Operations Research, Vol.32, No.1 : 133-147.
- [25] G.B. Sheble and G.M. Fahd (1994). *Unit Commitment Literature Synopsis*. IEEE Transactions on Power Systems, Vol.9, No.1 : 128-135.
- [26] Standard & Poor's Industry Surveys (1997). *Semiconductors*. Standard & Poor's, New York, NY.
- [27] The National Technology Roadmap for Semiconductors: Technology Needs (1997). Semiconductor Industry Association, San Jose, California.

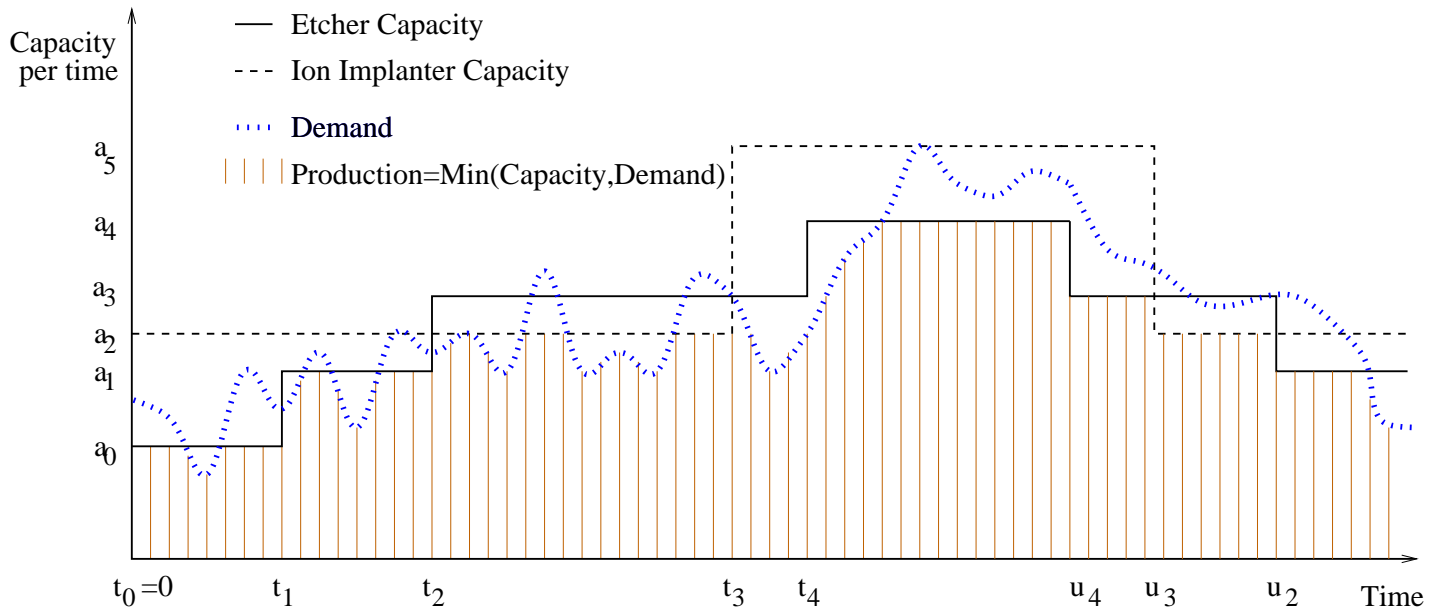


Figure 1: Production capacity versus demand for a semiconductor fab.

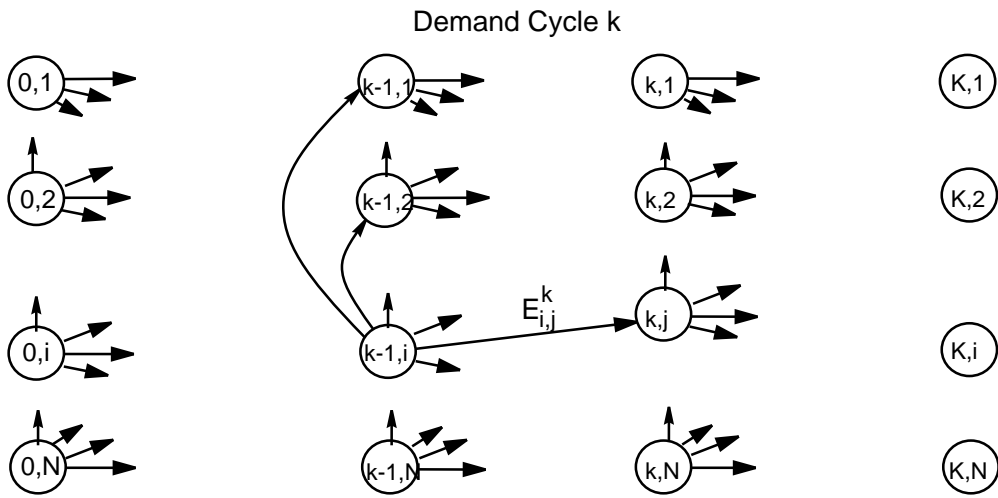


Figure 2: Demand cycle network with  $N$  machines and  $K$  cycles. Node  $(k-1, i)$  indicates starting cycle  $k$  with  $i$  existing machines. Vertical arcs are to retire machines at the beginning of cycles.

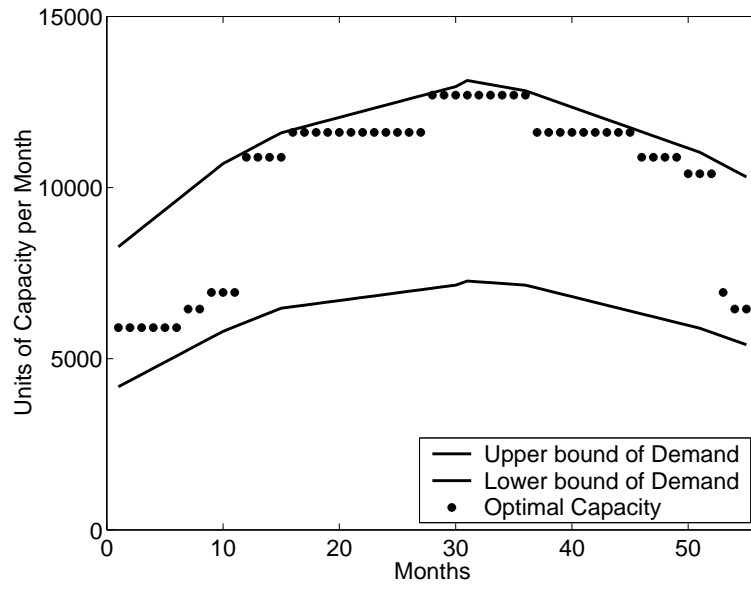


Figure 3: Optimal capacity as demand grows and falls.

- INITIALIZE:  $R := \{r, r + 1, \dots, s\}$ ,  $n := r + 1$
- While  $n < s + 1$  do
- SOLVE  $(\mathcal{P}_{r,n})$  :
  - $C' := \{n\}$ ,  $UniteComplete := false$
  - While  $\min(C') > r$  and  $UniteComplete = false$  do
    - $k := \min(C') - 1$
    - Compute  $t_{C'}$  and  $t_{C(k)}$  by minimizing convex functions  $f_{C'}(t)$  and  $f_{C(k)}(t)$
    - If  $t_{C'} < t_{C(k)}$  then
      - Unite:  $R := R \setminus \min(C')$ ,  $C' := C' \cup C(k)$
    - else
      - $UniteComplete := true$
  - endwhile
  - $n := n + 1$
- endwhile
- $R(r, s) := R$
- Break the machine set down into clusters such that every cluster  $C$  starts with a root in  $R$  and continues until the next root
- For  $1 \leq n \leq N$ , set  $t_n = t_C$  for the unique cluster  $C$  including  $n$

Table I: Cluster Algorithm to solve  $(\mathcal{P}_{r,s})$ .

A: Solve Expansion Subproblems:

for  $s = 1$  to  $N$  do

Use Cluster Algorithm to Solve  $(\mathcal{P}_{r,s})$  for  $1 \leq r \leq s$ .

Record the cost  $e_{r,s}$ .

endfor

$I^*(s) = \arg \min\{e_{I,s} + \sum_{n=I+1}^{I_0} H_n + \sum_{n=I+1}^s G_n : 1 \leq I \leq I_0\}$  and  $e_s = e_{I^*(s),s}$ .

Record  $e_s$  and  $I^*(s)$ .

B: Solve Contraction Subproblems:

for  $r = N + 1$  to  $2N$  do

Use Cluster Algorithm to Solve  $(\mathcal{P}_{r,s})$  for  $r \leq s \leq 2N$ .

Record the cost  $c_{r,s}$ .

endfor

C:  $(r^*, s^*) = \arg \min\{e_s + c_{2N+1-s,r} + \sum_{n=2N+1-s}^r H_n : 1 \leq s < N < 2N + 1 - s \leq r \leq 2N\}$

D: Retire  $[I^*(s^*) + 1, I_0]$  at  $t = 0$  and find optimal availability and retirement times using the Cluster Algorithm on  $(\mathcal{P}_{I^*(s^*),s^*})$  and  $(\mathcal{P}_{2N+1-s^*,r^*})$

Table II: EC Algorithm to solve  $(\mathcal{P})$