Manufacturing & Service Operations Management

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Allocating Fibers in Cable Manufacturing

Daniel Adelman, George L. Nemhauser, Mario Padron, Robert Stubbs, Ram Pandit,

Allocating Fibers in Cable Manufacturing

Daniel Adelman • George L. Nemhauser • Mario Padron • Robert Stubbs • Ram Pandit
Graduate School of Business, The University of Chicago, Chicago, Illinois 60637
School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332
Lucent Technologies, Atlanta Works, Norcross, Georgia 30071
Lucent Technologies, Atlanta Works, Norcross, Georgia 30071
Consolidated Freightways, Inc., Portland, Oregon 97208-6696

We study the problem of allocating stocked fibers to made-to-order cables with the goals of satisfying due dates and reducing the costs of scrap, setup, and fiber circulation. These goals are achieved by generating remnant fibers either long enough to satisfy future orders or short enough to scrap with little waste. They are also achieved by manufacturing concatenations, in which multiple cable orders are satisfied by the production of a single cable that is afterwards cut into the constituent cables ordered.

We use a function that values fibers according to length, and which can be viewed as an approximation to the optimal value function of an underlying dynamic programming problem. The daily policy that arises under this approximation is an integer program with a simple linear objective function that uses changes in fiber value to take into account the multi-period consequences of decisions. We describe our successful implementation of this integer program in the factory, summarizing our computational experience as well as realized operational improvements.

(*Cable Manufacturing*; *Resource Pricing and Allocation*; *Remnant Inventory System*; *Discrete Optimization*; *Dynamic Programming Approximations*)

1. Fiber-optic Cable Manufacturing

1.1. Introduction

In most inventory systems, raw materials depart the system once they are allocated to orders. We present a different type of system in which remnant raw materials return to inventory. Eventually after many allocations, a unit of raw material is so consumed that it is no longer allocable and consequently is scrapped. We discovered this *remnant inventory system* in working with the largest fiber-optic cable manufacturer in the U.S., who asked us to develop a new methodology for managing fiber allocations. Because of an existing world-wide shortage of optical fiber (Schiesel 1996), we had the opportunity not only to save millions of dollars annually in lost production and reduce environmental waste, but also to accelerate the expansion of the information super-highway.

The fiber-optic cable manufacturing plant consumes hundreds of millions of fiber meters each week. It maintains 20,000–30,000 colored fibers in inventory, and it can have hundreds of cable orders (one cable per order) awaiting fiber allocation, with each requiring anywhere from 2 to 216 fibers. It stocks fibers in inventory having lengths from 1 up to 40 kilometers, and receives orders for cable lengths ranging from 1 to 13 kilometers. It holds fibers in 13 different colors, with color being just one of over 60 physical properties important to customers. With the magnitude of fiber flow in this factory, even modest improvements in the allocation of fibers to orders can result in significant savings.

In this paper, we present our development and successful implementation of an economic mechanism for allocating stocked fibers to made-to-order cables so as to reduce the costs of scrap, setup, and fiber circulation

1523-4614/99/0101/0021$05.00

(i.e. moving fibers in and out of inventory). These inefficiencies result from making undesirable fiber allocations to satisfy customer due dates. In addition to the immediate costs of these allocations, there are future costs arising from the remnants generated. For example, although remnants can later be allocated to new orders, they cannot be spliced and so care must be taken to generate remnants long enough to be salable, yet also likely to eventually produce low scrap. Our methodology implicitly accommodates such complex issues through a fiber and order valuation scheme. Work is still emerging on understanding and computing these values (Adelman 1997; Adelman and Nemhauser 1996), but in practice our approach has resulted in over a 30% reduction in end scrap.

The core idea of our approach is to construct a value function that values fiber as a function of length, and then to use this function to make allocation decisions. If a fiber having value V_{fiber} is allocated to a cable and returns as a remnant with value V_{remnant}, then the cost of this allocation is $V_{\text{fiber}} - V_{\text{remnant}}$, the net decrease in the value of the inventory. Furthermore, a cable whose due date is near, or one requiring substantial fiber length, should be willing to take more net value out of the fiber inventory than a less urgent or substantial cable. Thus arose the idea of giving each cable order a budget with which it "purchases" an allocation of fibers, and which increases as its due date nears. This economic approach was appealing to operations managers, who find it natural to think of each fiber held in inventory as having some "value," and who subsequently have adopted this way of thinking.

In contrast to our intuitive value function approach, the old system made allocation decisions using a greedy algorithm. Previous research on related fiber allocation problems (Johnston 1992; Northcraft 1974) focused on such algorithms, which we have also seen in the copper cable manufacturing industry and now briefly summarize. Starting with orders due in the near term, orders were sorted longest to shortest, and then concatenations (see 1.4) were partially enumerated. For each cable, a feasible allocation of fibers was sought by searching for the shortest fibers available within pre-specified length ranges. These ranges were prioritized. If a feasible allocation of fibers was found, then the cable was selected and the fibers were removed from consideration. New cables were successively considered until no more could be satisfied. To take advantage of current fiber opportunities, orders with more distant due dates were then considered.

Rather than sort through fiber and cable opportunities sequentially, as do these algorithms, we implement our value function in an integer program that considers all current allocation opportunities simultaneously. Therefore, our approach generally can match more orders with fibers. Furthermore, the value function can generate more favorable remnants because it is not inhibited by sequential logic.

When our problem is modeled as a dynamic program, and the optimal value function is approximated by a linear form using the fiber values, implementing the dynamic programming policy reduces to solving our integer program repeatedly over time. Such approximations have been investigated recently in other contexts (Bertsekas and Tsitsiklis 1996; Godfrey and Powell 1997). In §2 we state the dynamic program and present the optimal value function approximation and ensuing integer program. Then in §3 we discuss how the fiber values and budgets are computed, and describe an auxiliary linear program that can aid in these computations.

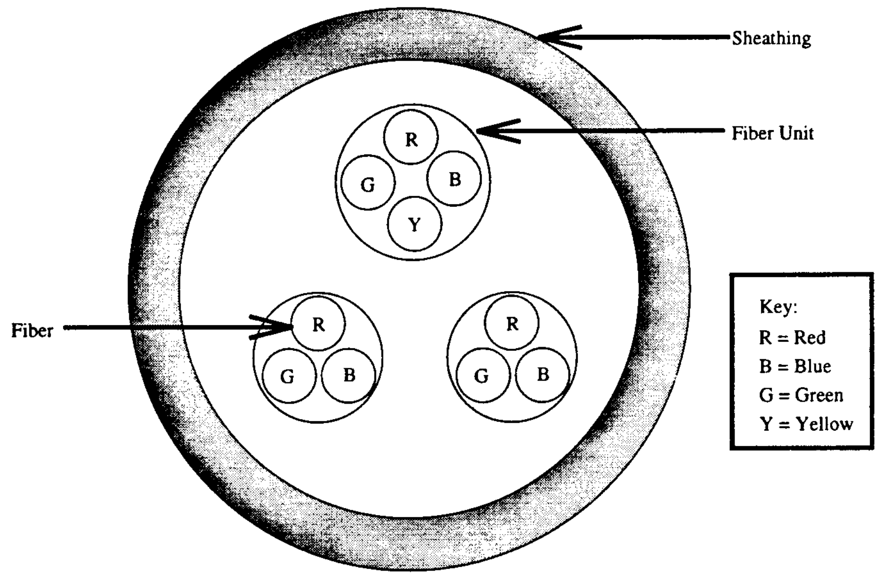
Our experiments showed that real-life factory instances of the integer programming model were readily solvable with commercial, "off-the-shelf" software such as CPLEX (CPLEX 1995). An additional advantage of this approach is that management preferred implementation in a simple, self-documenting algebraic language such as AMPL (Fourer et al. 1993) rather than a large amount of computer code which would be hard to maintain and even harder to modify. Using an algebraic language also shortened software development time, and consequently permitted us to provide a proof of concept quickly. In §4 we present computational results and an overview of our solution methodology.

In §5 we report on observed factory performance improvements, and finally in §6 we briefly discuss some future research directions. We now describe some details of fiber-optic cable manufacturing.

1.2. The Product

Within a fiber-optic cable, shown in Figure 1, colored fibers are organized into *fiber units* containing 2 to 12

Figure 1 Cross-Section of a Fiber-Optic Cable



fibers bundled together. For each fiber, in addition to color, the customer specifies a minimally acceptable set of fiber characteristics. For example, cables that a customer intends to use for a sensitive application may require exceptionally low transmission loss fibers. We categorize these characteristics into *fiber types.*

The customer specifies the *fiber configuration* of the cable, which indicates how many fibers of each fiber type are required. However, multiple fiber types may satisfy these requirements. For example, low loss fiber types can *substitute* for higher loss fiber type requirements. The allowable substitutions are specified by a *substitution matrix*, $\{S_{t,r}\}$, where $S_{t,r} = 1$ if a type t fiber can substitute for a type r fiber requirement, and 0 otherwise.

In addition to the fiber configuration, customers also specify the plastic insulation, support rods, and other materials that make up the *sheath* inside which the fibers are placed. Of course, they also specify a cable length. All together, the requirements are so customer-specific that almost all cables must be made-to-order. There are a few standardized cable specifications that would allow for some cables to be made-to-stock, but oftentimes even these are *finished-to-order* (Lee and Tang 1997).

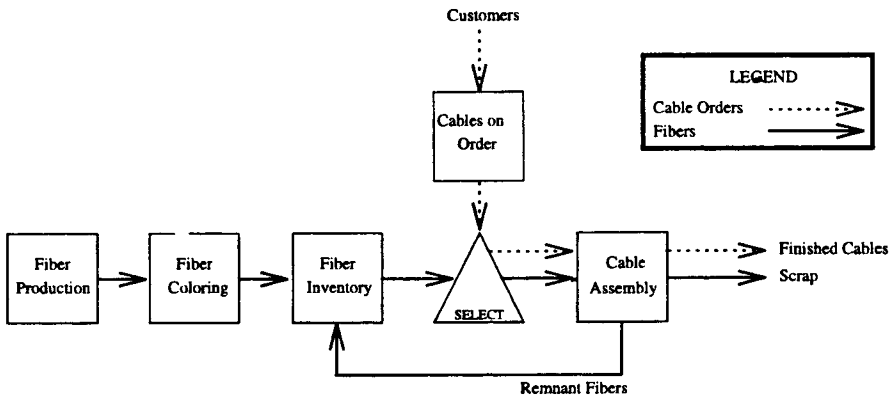
1.3. Manufacturing Process Flow

Figure 2 depicts a simplified manufacturing process flow. Gue et al. (1997) and Clements et al. (1997) discuss some aspects of this factory in more detail. First, the factory produces clear fibers of various lengths and types (Murr 1992) and stores them on spools. The factory inevitably produces a full spectrum of lengths because fibers break at random during the draw process. After drawing, the spools are transported to operators who color the fibers according to a production plan, and who then place them into inventory where they await allocation.

Fibers are not colored-to-order nor cut-to-order as they are drawn. This is because most orders have quoted lead times that are too short to wait for coloring and cutting. Furthermore, with ten thousands of spools circulating, management considers the required coordination to be too expensive and complicated to implement at present. Since there are no standard lengths in this industry, as there are in the paper industry, for example, the factory does not cut fibers manually before they break during draw, but instead tries to draw fibers as long as possible, up to the maximum length that can be stored on a spool.

The factory also draws fiber at capacity since any

Figure 2 Manufacturing Process Flow



fiber unused by local cable production, assuming it is long enough, can easily be sold raw on the open market. Since the decision of how much fiber to sell to third parties versus stock in-house, i.e., the effective input of fiber, is outside the scope of our fiber allocation problem, we do not directly consider inventory holding costs. Indirectly, however, with lower scrap rates and more salable remnants, the factory can operate with less fiber inventory and sell more fiber to third parties.

Each morning production control solves the allocation problem, depicted as "SELECT" in Figure 2:

INPUT: Given the cables on order and the fibers in inventory.
PROBLEM: Select a collection of cables to assemble and allocate fibers to each chosen cable.

Afterwards, the cable assembly department moves the allocated fibers out of inventory and stages them in front of the cable assembly operation, where they are collated and eventually encased in a sheath to form a cable. The remnants generated either return to the fiber inventory or are scrapped, depending upon whether their individual lengths are above or below, respectively, a threshold length specified by management. Remnants that return to inventory must pass through a remeasurement operation (not depicted) where their lengths and other physical properties are verified.

From the time a fiber is staged for cable assembly until its remnant returns to inventory, it is unavailable for allocation because the information system does not track its status and location once it leaves inventory. Thus, it is impossible to coordinate the movement of spools within the cable assembly area, for example between multiple assembly lines. Nor can the factory precisely predict when a remnant will return to inventory, especially because the transportation and remeasurement operations are not scheduled. Consequently, in solving the allocation problem, a spool is assigned to at most one cable at a time, and only spools currently in inventory are available for allocation. A spool's next use is not explicitly planned.

1.4. Concatenations

Because of the way setups need to be done, cables generate *process scrap* when they are assembled. Further length is wasted because cables are made slightly longer than the cable length ordered. This is done to hedge against statistical uncertainty in measuring fiber lengths, and to account for extra cutting allowances at downstream operations. Nandakumar and Rummel (1998) discuss related issues as they arise in the fiber draw operation.

To reduce process scrap and setups, multiple cable orders sometimes can be satisfied with the production of a single long cable, called a *concatenation,* that is afterwards cut into the individual cables ordered. All suborders must require compatible sheaths and fiber configurations, since a concatenation is made as one continuous cable with the same physical makeup throughout. For example, orders with different fiber loss requirements can be concatenated if the fibers in the concatenated cable have sufficiently low loss to satisfy all suborder requirements.

In solving the daily allocation problem, we must decide not only what cable orders to satisfy, but also what concatenations to make. We must "select" cables. From now on, we adopt the following definitions:

- *Concatenation:* a collection of orders to be manufactured together as a single cable
- *Cable:* refers to the cable actually to be manufactured, which may be a single order or a concatenation
- *Suborder:* an order contained within a concatenation.

1.5. Factory Objectives

The primary objectives are to reduce scrap while improving the satisfaction of orders by their due dates, i.e., lateness. We discussed above how to reduce process scrap through concatenation. To reduce end scrap, which occurs as spools are emptied, the future use of remnants must be taken into account as fibers are allocated. However, there is a tradeoff between satisfying customer due dates and producing desirable remnants. Remnants are undesirable not only because of the scrap they may produce, but also because they may be too short to satisfy orders for long cables. Thus, the salability of the fiber inventory must also be considered.

In addition to these considerations, concatenating more orders reduces setup and fiber circulation costs since fewer cables are produced and more fiber is consumed at once. Circulation costs include the labor cost to move spools in and out of inventory, including remeasurement, and are at least an order of magnitude smaller than scrap costs.

2. Approximation of a Multi-Period Problem

In this section, we describe the multi-period problem and discuss how we approximate it using a sequence of one-period problems.

2.1. Multi-Period Objective and Dynamics

To impose structure on our multi-period setting, we present a generic description of a dynamic programming model with a long rolling time horizon of N periods. Let the state of the system at the beginning of period n be given by

$$s_n = (\mathit{Orders}_n,\ \mathit{Fibers}_n,\ \mathit{Allocated\ Fibers}_{(n-\tau+1)},\ \ldots,\ \mathit{Allocated\ Fibers}_{(n-1)}),$$

where Orders_n is the set of orders available, Fibers_n is the set of fibers available in inventory, and $\mathit{Allocated\ Fibers}_t$ is the set of fibers allocated in period t. For simplicity we assume it takes a spool of fiber τ time periods to return with a remnant to inventory, and so the state keeps track of all returning remnants. The decisions made at the beginning of period n are then given by

$$d_n = (\mathit{Satisfied\ Orders}_n,\ \mathit{Allocated\ Fibers}_n,\ \mathit{Cables\ Selected}_n),$$

with a cost of $g_n(s_n, d_n)$. The actual assignments of fibers to cables are implied, so that the remnant lengths are known from d_n. We do not control new fiber production, i.e., *New Fibers*$_n$. The form of the optimization problem is then

$$\text{minimize Expected}\left[\sum_{n=0}^{N} g_n(s_n, d_n) - \text{Value of } s_{N+1}\right], \tag{1}$$

$$\mathit{Orders}_{n+1} = \textbf{order balance}\ (\mathit{Orders}_n,\ \mathit{New\ Orders}_{n+1},\ \mathit{Satisfied\ Orders}_n) \quad \forall n = 0, \ldots, N, \tag{2}$$

$$\mathit{Fibers}_{n+1} = \textbf{fiber balance}\ (\mathit{Fibers}_n,\ \mathit{New\ Fibers}_{n+1},\ \mathit{Allocated\ Fibers}_{n+1-\tau},\ \mathit{Allocated\ Fibers}_n,\ \mathit{Scrapped\ Fibers}_n) \quad \forall n = 0, \ldots, N, \tag{3}$$

$$\mathit{Allocated\ Fibers}_n \text{ satisfy all requirements of } \mathit{Cables\ Selected}_n \quad \forall n = 0, \ldots, N, \tag{4}$$

$$\mathit{Orders}_n \text{ are satisfied at most once by } \mathit{Cables\ Selected}_n \quad \forall n = 0, \ldots, N. \tag{5}$$

We are given as initial conditions the orders and fibers available, i.e., Orders_0, Fibers_0, and $\mathit{Allocated\ Fibers}_{(1-\tau)}, \ldots, \mathit{Allocated\ Fibers}_{-1}$. We are also given probability distributions on lengths and quantities of arriving fibers and orders for periods $1, \ldots, N+1$.

The functions **order balance**$(\cdot)$ and **fiber balance**$(\cdot)$ are standard linear flow balance relations. The constraint (2) maintains flow balance for orders, which arrive and are satisfied in each period. The constraint (3) maintains flow balance for fibers, accounting for new fiber production and remnant fibers. ($\mathit{Allocated\ Fibers}_{n+1-\tau}$ represents the remnants returning from fibers allocated in period $n + 1 - \tau$.) Spools depart the inventory when they are allocated and when they are scrapped. The constraint (4) ensures that the fiber requirements of all cables selected are satisfied. Since the cables selected may include concatenations, constraint (5) ensures that each order is satisfied at most once, i.e., an order cannot appear more than once in a concatenation or across all *Cables Selected*$_n$.

The objective function (1) minimizes the expected total N-period costs, minus the expected value of the ending state s_{N+1}. The increase in total lateness cost arises from due orders that are not selected. All fibers below the threshold length are scrapped at a specified cost per unit length. Each cable selected incurs a constant process scrap charge for each fiber contained in it, as well as a sheathing setup charge for the cable itself. Also, each fiber allocated incurs a constant cost for circulating it.

In theory, this multi-period problem can be solved using a dynamic programming algorithm (Bellman 1957). Let $-W_n(s_n)$ be the optimal expected cost over periods $n, \ldots, N+1$ (including the ending value) given the state s_n, so that $W_n(s_n)$ is the optimal value function. Then

$$W_n(s_n) = \max_{d_n} \mathsf{E}[-g_n(s_n, d_n) + W_{n+1}(s_{n+1}) \mid (s_n, d_n)] \quad \forall n = N, \ldots, 0, \quad (6)$$

where the next state s_{n+1} depends on the current state s_n and decision d_n as governed by (2)–(5).

2.2. Linear Approximation

Since the state space of the dynamic program is extremely large, it is impractical to solve (6) exactly. We approximate the optimal value function $W_n(\cdot)$ with a linear function of the system state, $\tilde{W}_n(\cdot)$. Bellman and Dreyfus (1959) may have been the first to use this approach, which has recently been investigated by Godfrey and Powell (1997) in the context of logistics problems, and also studied by Bertsekas and Tsitsiklis (1996). We take the approximating value function in period n to be equal to the value of the fibers in inventory, minus the fiber value associated with the cables on order, minus the expected incremental lateness penalty into the future for not satisfying the orders in this period.

Let $V_{L_f,t_f}(n)$ be the value of a fiber $f \in$ *All Fibers*$_n$ having length L_f and type $t_f \in$ *Types* in period n. *All Fibers*$_n$ is the set of all fibers available in inventory as well as in circulation. We take the value of the inventory state to be the simple function

$$\sum_{f \in All\ Fibers_n} V_{L_f,t_f}(n).$$

Since the supply and demand for fibers may change over time, the values are a function of n as well. We discuss the structure and computation of the value function in §3.

Suppose a fiber with index f having length L_f is allocated to cable c having length L_c (including process scrap). Then the value of the inventory decreases by $V_{L_f,t_f}(n) - V_{L_f-L_c,t_f}(n)$. In allocating fibers, therefore, we wish to minimize this quantity. We choose the value $V_{L_f-L_c,t_f}(n)$ so that the net change in value accounts for the immediate fiber scrap cost in g_n from (6), including process scrap and the length $L_f - L_c$ if it is end scrap. However, $V_{L_f,t_f}(n) - V_{L_f-L_c,t_f}(n)$ also includes the change in the expected future scrap cost, which could be reduced by transforming unfavorable fiber lengths into favorable remnants, or increased by the opposite transformation.

Since an order with a very distant due date has essentially zero expected future lateness cost, it should be selected only if a "perfectly fitting" (i.e., zero remnant length) allocation of fibers is available, or at least one equally inexpensive. As its due date nears, the order is less insistent on a "perfectly fitting" fiber allocation and is willing to incur greater cost. This increasing willingness is mediated by an increasing budget for the cable, although the cheapest allocation of fiber available is always sought. Thus, the budget of each cable order $c \in$ *Orders*$_n$, $B_c(n)$, is the total value of a "perfectly fitting" set of fibers plus the (approximated) expected future lateness cost if it is not selected in period n. As an order's due date approaches or recedes further, we assume that the expected future lateness cost is nondecreasing, and so $B_c(n)$ is nondecreasing as n increases.

In conclusion, for state s_n we approximate $W_n(s_n)$ in (6) with

$$\tilde{W}_n(s_n) = \sum_{f \in All\ Fibers_n} V_{L_f,t_f}(n) - \sum_{c \in Orders_n} B_c(n) \quad (7)$$

When cable order c is satisfied, the value of the state increases by $B_c(n)$. Other functional forms are possible but we chose (7) because it was easy for management to understand, and because it simplifies the decision problem expressed by the right-hand side of (6) for each period n.

2.3. The Daily Allocation Problem

Given (7), our problem becomes an integer program that considers future periods only through a simple linear objective function. Thus, the model variables and parameters involve only period n decisions and data. To simplify the remainder of the paper, with the exception of sets we suppress the dependence on n in the notation.

Let the set $Cables_n$ represent all single-order cables and concatenations possible given $Orders_n$. For decision variables, let

$$Y_c = \begin{cases} 1 & \text{if cable } c \text{ is selected,} \\ 0 & \text{otherwise,} \end{cases} \qquad \forall c \in Cables_n$$

$$X_{f,[c,r]} = \begin{cases} 1 & \text{if fiber } f \text{ is allocated to satisfy a type } r \\ & \text{requirement of cable } c, \\ 0 & \text{otherwise,} \end{cases} \qquad \forall (f, [c, r]) \in \Omega,$$

where

$$\Omega = \{(f, [c, r]) : f \in Fibers_n, c \in Cables_n, \\ r \in Types\ L_f \geq L_c, S_{t_f,r} = 1\},$$

that is, the set of all feasible fiber allocations. Only fibers currently in inventory, i.e., not circulating, are available for allocation in Ω.

If cable $c \in Orders_n$, then it is not a concatenation and the budget B_c is as described in §2.2. Otherwise, c is a concatenation, and so its budget B_c also accounts for immediate process scrap, setup, and fiber circulation cost savings. We return to this in §3.

Letting $a_{[c,r]}$ be the number of type r fibers required by cable c, where $r \in Types$, $c \in Cables_n$, we can now state the problem solved in period n:

(IP$_n$) **maximize** Net Budget

$= $ Total Budget of Selected Cables

$-$ Total Cost of Allocated Fibers

$$= \sum_{c \in Cables_n} B_c Y_c - \sum_{(f,[c,r]) \in \Omega} (V_{L_f,t_f} - V_{L_f - L_c,t_f}) \cdot X_{f,[c,r]} \quad (8)$$

subject to

Fiber supply: Fiber f can be allocated at most once.

$$\sum_{c \in Cables_n, r \in Types} X_{f,[c,r]} \leq 1 \qquad \forall f \in Fibers_n \quad (9)$$

Cable demand for fibers:
If cable c is selected, then all of its fiber requirements must be satisfied.

$$\sum_{f \in Fibers_n} X_{f,[c,r]} = a_{[c,r]} Y_c \quad \forall c \in Cables_n,\ r \in Types \quad (10)$$

Satisfy each customer order at most once.

$$\sum_{\bar{c} \in Cables_n \text{containing } c} Y_{\bar{c}} \leq 1 \qquad \forall c \in Orders_n \quad (11)$$

Integrality

$$X_{f,[c,r]} \geq 0, \text{ integer} \qquad \forall (f, [c, r]) \in \Omega, \quad (12)$$

$$Y_c \in \{0, 1\} \qquad \forall c \in Cables_n. \quad (13)$$

Although not explicit, the summations in (9) and (10) include only variables $X_{f,[c,r]}$ with $(f, [c, r]) \in \Omega$. Each type requirement r for a cable c must be satisfied with exactly $a_{[c,r]}$ fibers, and each allocated fiber can satisfy at most one type requirement r for the cable. Therefore, it is necessary to subscript X with r. Otherwise, with substitutions, a single fiber could simultaneously satisfy requirements for multiple fibers of different types.

The objective function (8) maximizes the net budget remaining, but only for the cables selected. For any selected cable c in an optimal solution,

$$B_c \geq \sum_{(f,[c,r]) \in \Omega} (V_{L_f,t_f} - V_{L_f - L_c,t_f}) X_{f,[c,r]}. \quad (14)$$

Cables with the greatest slack in this inequality are the most profitable to the objective function (8), and thus have advantage in the competition for scarce fibers. Since cables c containing more urgent orders are given larger budgets B_c, the orders are implicitly prioritized according to urgency.

There is a correspondence between the constraints (9)–(13) and (2)–(5). The constraint (9) limiting the fiber supply in period n is a subcomponent of the global system constraint (3). The same is true between (13) and (2). The constraint (10) is a formal statement of (4), and (11) is a formal statement of (5).

Once the Y_c variables are fixed, this integer program reduces to a transportation problem. The supply nodes represent fibers and the demand nodes are cable/type requirements. The binary variables are needed in (10) because the demand nodes for cables not selected, i.e., $Y_c = 0$, must be removed from the transportation problem. On the other hand, cables that are selected, i.e., $Y_c = 1$, must have each fiber type requirement met. This is handled by constraints (10). However, since each order may be satisfied at most once, constraints (11) dictate which combinations of demand nodes may be active at any time. Because of the network structure, the $X_{f,[c,r]}$ will be integral once the binary variables are fixed, and so we can relax integrality in (12).

3. Fiber Values and Cable Budgets

3.1. Computing the Fiber Values

We now present a linear program that can be used to compute a starting point for the fiber value function, and to give us some insight into various properties the value function should satisfy. Adelman and Nemhauser (1996) demonstrate empirically that when the fiber value function from this linear program is used in a special case of ($\mathbf{IP}_n$), the scrap rate is minimized. The model considers a long-run horizon and captures the flow rates of remnants through the factory that result from making daily allocations over time.

For convenience, we consider only a single fiber type, although the model easily extends to handle multiple fiber types. We also use the notation i to represent a fiber length, so that $i \in F = \{0, 1, \ldots, L\}$, the set of all possible fiber lengths. We discretize F with an appropriate level of granularity up to the maximum fiber length L. Let P_i be the given fraction of raw fibers drawn at length i $\forall i \in F$.

Let C be the set of cut-lengths taken from fiber, and let λ_j $\forall j \in C$ be the average rate cuts of length j are taken. This cut-length includes not only singly made orders and their associated process scrap, but also concatenations and their associated process scrap. The λ's can be estimated using historical data and demand forecasts. Also let $A = \{(i, k): i, k \in F, (i - k) \in C\}$ be the set of all fiber transformations from length i to a remnant of length k, by satisfying a cut-length of $i - k$. Let $A_j \subseteq A$ be the set of fiber transformations for cut-length $j \in C$, i.e., $A_j = \{(i, k) \in A : i - k = j\}$.

Now define the following decision variables. Let S_i $\forall i \in F$ be the rate at which we scrap fibers having length i, and let μ be the consumption or effective input rate of raw spools of fiber. Finally, let $Y_{i,k}$ $\forall (i, k) \in A$ be the rate at which we cut fibers having length i to produce remnants of length k, by satisfying cut-length $i - k$.

Consider the linear program

$$\textbf{(LP)} \qquad \text{Min} \quad \mu \tag{15}$$

$$P_i\mu + \sum_{\{k:(k,i)\in A\}} Y_{k,i} = \sum_{\{k:(i,k)\in A\}} Y_{i,k} + S_i \quad \forall i \in F, \tag{16}$$

$$\begin{aligned} \sum_{(i,k)\in A_j} Y_{i,k} &= \lambda_j && \forall j \in C, \\ Y_{i,k} &\geq 0 && \forall (i, k) \in A, \qquad (17) \\ S_i &\geq 0 && \forall i \in F, \\ \mu &\geq 0. \end{aligned}$$

The objective (15) minimizes the consumption rate of spools, which by a simple balance equation is equivalent to minimizing the end scrap rate, $\Sigma_{i\in F}\, iS_i$. Constraints (17) ensure that all cut-lengths are satisfied, stating that the rate at which units are allocated to cut-length j must equal the rate λ_j at which it is required. Constraints (16) maintain flow balance for fibers, saying that the rate fibers of length i are input into the system plus the rate at which they are generated as remnants must equal the rate at which they are allocated and scrapped. Hence, in the LP there is no inventory holding, i.e., all fibers must be used in the long-run.

The fiber value function V_i is taken from the optimal dual prices of constraints (16) and satisfies several properties such as monotonicity, which says that a fiber is more valuable than any fiber shorter than it. The fiber value function is also superadditive, since any set of allocations that two fibers having lengths i and k can perform, also can be performed with a single fiber having length $i + k$. Proofs of these and other properties, which also involve dual prices from (17), can be found in Adelman and Nemhauser (1996).

To promote salability, local adjustments to the value function are made by the user to encourage remnants at lengths salable to known future orders, using these properties as a guideline. This is done by increasing the value V_i for such lengths i. By adding an adjustment term ϵi^2 to each V_i for some $\epsilon > 0$, the user encourages the consumption of short fibers and discourages the consumption of long, salable fibers. The value of ϵ used in practice is an order of magnitude smaller than the base fiber values so that it is not large enough to override scrap preferences, but it does break ties between allocations that otherwise would have indistinguishable costs.

Although (**LP**) minimizes only μ, in practice because of the way we compute budgets in (19), our integer program ($\mathbf{IP}_n$) strives for concatenations containing many suborders so as to reduce process scrap, fiber circulation, and setups. This influences the λ's in our linear program (**LP**) so that fibers are priced based on cutting these favorable concatenation lengths. As the λ's change over time, (**LP**) can be re-solved and the fiber value function updated.

3.2. Computing Budgets

Each order $c \in Orders_n$ is given a bonus α_c, which reflects the expected lateness cost savings if it is satisfied now. Formally, its budget is

$$B_c = \sum_{r \in Types} a_{[c,r]} V_{L_c,r} + \alpha_c, \qquad \forall c \in Orders_n. \quad (18)$$

The first term is needed for the order to have a large enough budget to purchase a "perfectly fitting" set of fibers (including the requisite process scrap).

Now suppose cable c is a concatenation. (Note that the set of concatenations is $Cables_n \backslash Orders_n$.) To give an incentive to select concatenations, we give each concatenated cable $c \in Cables_n$ an extra bonus β_c that reflects the setup and fiber circulation cost savings resulting from the concatenation. Compared to producing w separate single-order cables, a concatenation of w orders each requiring m fibers saves $w - 1$ setups and $(w - 1) * m$ fiber circulations. We then add up the budgets (18) for all suborders in the concatenation, together with the bonus β_c, to arrive at a budget for the concatenation:

$$B_c = \begin{cases} \left(\displaystyle\sum_{\text{suborders } \bar{c} \text{ of } c} \left(\sum_{r \in Types} a_{[\bar{c},r]} V_{L_{\bar{c}},r} + \alpha_{\bar{c}} \right) \right) + \beta_c \\ \text{if } c \in Cables_n \backslash Orders_n \\ \displaystyle\sum_{r \in Types} a_{[c,r]} V_{L_c,r} + \alpha_c \text{ if } c \in Orders_n \end{cases}$$

$$\forall c \in Cables_n. \quad (19)$$

Since $L_{\bar{c}}$ includes the process scrap length for each suborder $\bar{c}$, the first term of the concatenated case in (19) accounts for the process scrap savings.

In principle one could use historical data to estimate the expected lateness cost of each order if not selected, but this is in general difficult to do accurately because orders are so customized. Instead, we compute how high α_c must be for each order c to purchase the most expensive allocation currently available. Sometimes the α_c's are set above this level so that the model tries to maximize the number of orders satisfied, with different echelons for priority classes based on due date urgency. However, each α_c can be set below the baseline if the user wishes to restrict certain allocations. Also, if high α_c's cause a cable to be selected with an undesirable fiber allocation, then it may be disregarded immediately, or on the next day the suborders could be reconsidered if they have not been sheathed yet.

Orders with distant due dates, and hence low α_c values, are unlikely to be satisfied early. However, if too many such orders are contained within concatenations, other orders with urgent due dates could be delayed behind them in cable assembly (Dobson et al. 1987). When sheathing capacity is tight, the user may limit the set of orders available to the model to exclude those with distant due dates. Also, by setting certain software parameters, the user can forbid concatenations of orders with disparate due dates.

4. Solution Methodologies

Because the integer programming objective is an approximation, it is not important to find an optimal solution. Management is satisfied with solutions that are within a few percent of the theoretical optimum. So we speed up the branch-and-bound solution process by imposing tolerances, under which optimal or, at the very least, provably high quality solutions are still obtained after only a few minutes of computation. We attribute this tractability to the underlying network structure of the model, as well as to the many recent algorithmic advances built into modern mathematical programming systems.

In this section we briefly describe preprocessing routines for aggregating fibers into groups and sampling concatenations for the optimization model. The fiber aggregation routine reduces the size of problem instances of ($\mathbf{IP}_n$), trying not to degrade solution quality.

(Optimality in the last paragraph is with respect to the aggregated problem.) Concatenation sampling is used rarely, but its availability is necessary when the whole set is too large to enumerate or to include in the model. The algorithms themselves are presented in detail in Adelman (1997).

4.1. Concatenation Preprocessing

The concatenation preprocessing routine determines which concatenations become candidates in the IP instance. Only concatenations that satisfy the following conditions are eligible:

(1). *Compatible construction:* all suborders have compatible sheath and fiber configurations,

(2). *Fibers available:* there is at least one set of fibers that meets the requirements for the cable, and

(3). *Profitability:* the budget of the concatenation minus a least cost feasible set of fibers that meets the cable's requirements is positive.

We call the quantity computed in condition (3) the *net value*, and the cheapest feasible set of fibers the *net value plan*. Comparison of the net values of two cables approximately indicates their relative competitiveness. For each possible concatenation, the preprocessing routine tries to find a net-value plan, and if it succeeds then condition (2) is satisfied. It does not include concatenations with negative net-values, or for which there does not exist a feasible allocation of fibers.

The routine adds candidates to the model according to a search scheme that gives preference to cables with higher net-values. After a user-specified number of candidates is chosen, the routine stops. Usually, there are few enough concatenation possibilities (< 100) to include all of them into the instance.

4.2. Fiber Aggregation Algorithm

Instead of having a constraint (9) and a collection of X variables for each fiber, we have them for each fiber group. We form these groups to have the property that no two fibers contained in one group have values or lengths differing by more than user-specified thresholds.

In the objective function of ($\mathbf{IP}_n$), we use the average value of fibers in each group to compute the objective coefficients, but when determining feasible allocations we use the minimum fiber length in each group. The former makes the objective function more reflective of the "true" costs, and the latter ensures that there exists a feasible disaggregation/assignment of fibers to cables. We set the tolerances to be less than 1% of the maximum fiber length or fiber value. We find that this is large enough to achieve computational efficiencies but small enough so that the scrap and lost allocation opportunities, caused by variances in the actual fiber lengths, is insignificant.

4.3. Computational Experience

The problem instances given in Table 1 are typical instances from among the thousand or more instances that have been solved and implemented successfully in the factory. The computational effort reported here is typical.

The number of cables reported is the number of candidates, i.e., variables Y_c, in the instance. In all instances, we ran the fiber aggregation and concatenation preprocessing routines. Except select3, select9, and select10, there were few enough eligible concatenations to include them all into each instance. Preprocessing CPU times are typically much smaller than CPLEX solution times, with the exception of select3. All computations for select1 through select10 were done on an IBM RS/6000 Model 590 machine. Computations for select11 through select29 were done on an NCR Model 3455 running a Pentium 133 MHz CPU. The CPLEX 3.0 MIP solver with standard defaults set to prove optimality handles these instances relatively easily, with the exception of select10 in which the node limit of 20000 was reached in about one CPU hour. This instance was more difficult in proving optimality than the others because there were many concatenations that were nearly identical to each other.

We also tested the MINTO (Nemhauser et al. 1994) primal heuristic, which is a sequential rounding procedure similar to CPLEX's. On these instances it was able to find (but not prove) the optimal solution quite early in the branch-and-bound tree (Nemhauser and Wolsey 1988), often at the root node. Therefore, in practice we set CPLEX parameters so that the CPLEX primal heuristic is always run. We also sometimes employ pruning tolerances, which deliberately prune branches in the tree unless the LP relaxation suggests at least a 10% improvement over the best integer feasible solution. Finally, after a specified time limit (one

Table 1 Actual Factory Problem Instances with Default CPLEX Results

Problem Name	# Cables	# Fiber Groups	# Variables	# Constraints	B&B Nodes	Simplex Itns	CPLEX CPU secs	Preproc. CPU secs
select1	3	582	1500	1497	1	89	1.57	0.88
select2	6	250	957	338	11	417	1.61	0.65
select3	300	10	1428	336	32	1242	2.96	33.60
select4	10	538	5250	788	37	3742	19.74	0.71
select5	73	554	18013	1455	44	6460	33.23	2.42
select6	58	2454	22108	2978	16	6471	81.07	3.91
select7	81	771	9464	1742	96	15682	121.00	3.025
select8	26	288	6159	769	1005	1004	137.75	0.78
select9	91	1257	51410	2366	174	45389	559.66	292.81
select10	200	91	2256	313	20000	725110	3359.45	2.12
select11	48	1170	12968	1303	7	154	5.8	2.87
select12	21	881	3152	596	14	368	1.5	1.16
select13	28	1014	6556	874	14	974	5.42	2.20
select14	49	1705	18220	1819	8	231	10.74	483.00
select15	33	433	3451	490	0	143	0.79	1.57
select16	6	999	1304	333	1	68	0.34	1.04
select17	13	1122	3114	667	7	606	1.59	1.22
select18	87	985	20165	1518	84	1835	56.33	1.85
select19	35	954	10403	1074	49	811	11.51	1.63
select20	20	1326	9925	1289	0	93	3.12	2.20
select21	13	1033	3368	778	11	889	2.07	1.17
select22	15	1450	3628	775	0	88	0.70	2.06
select23	40	645	31384	2483	108	15780	346.43	0.79
select24	6	784	1351	404	2	491	0.86	0.77
select25	19	763	1282	201	8	263	0.77	0.93
select26	36	981	4241	803	7	553	2.16	3.72
select27	5	463	948	350	2	212	0.48	0.45
select28	13	473	1585	340	0	342	0.52	0.50
select29	23	933	16009	1121	5	451	9.42	1.26

hour), we have CPLEX quit and return the best integer feasible solution it found so far.

5. Factory Performance

Our system has dramatically changed the distribution of remnant lengths generated, making the inventory more salable. This is attributable to the fiber value function encouraging consumption of short fibers and discouraging consumption of long, salable fibers. Figure 3 shows the distribution before implementing our system, with a high incidence of short remnants. In contrast, Figure 4 demonstrates that not only are a greater percentage of spools scrapped being scrapped with less than 250 meters upon allocation, but the generation of short remnants, i.e., in the 1000 to 3000 meter range, has been considerably reduced, and replaced with more remnants being produced at 4500 meters or longer. This is in contrast to the previous peak at 3500 meters.

Management believes that this shifting of the remnant length distribution has been a critical factor in their recent ability to achieve on-time deliveries. Producing longer remnants has made selection of long orders easier. Also, the ability of the integer programming model to satisfy more orders because of better matching of fibers to cables has improved on-time delivery. Since the business has continued to experience rapid capacity and demand expansion, it was

Figure 3 Distribution of Remnant Lengths (Including Scrap Lengths) Before Implementation

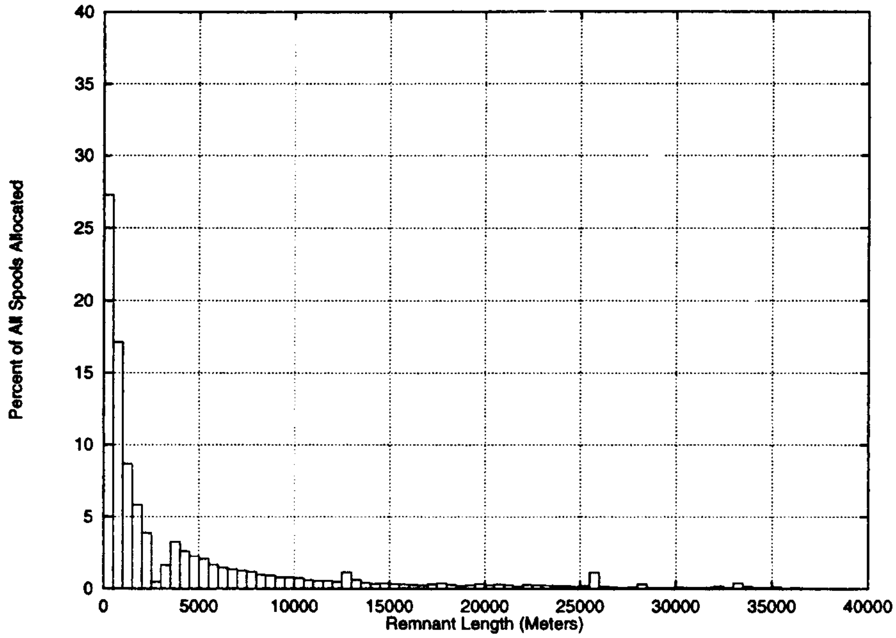
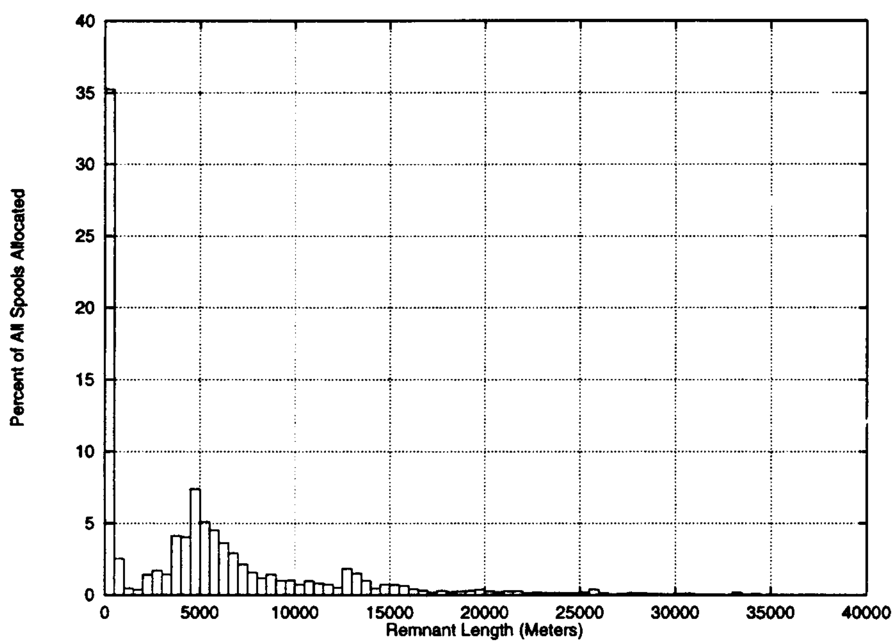


Figure 4 Distribution of Remnant Lengths (Including Scrap Lengths) After Implementation



impossible to isolate statistically our direct impact on improving due date performance.

Figure 5 shows the fraction of total fiber consumed that was scrapped over a period of 63 weeks. In week 18, our system was partially implemented, with full-scale implementation starting in week 34. In week 58, a large number of urgent cable orders that could not be concatenated had to be satisfied, regardless of the relative lack of low scrap fiber opportunities at the time. The figure reveals that our system has reduced end scrap from over 3% of total length consumed per week to under 2%, a 33% reduction. This represents millions of dollars in annual savings, which we attribute to the ability of the value function to reward low scrap.

Our system has substantially increased the incidence of concatenations, thereby reducing setup time, circulation costs, and process scrap. Figure 6 shows that the average length of fiber consumed per spool allocated has increased by over 40%. (The average length ordered over this period of time has remained relatively constant.) The data has been normalized so that 1 represents the average length before implementation. This improvement parallels an increase in the average number of suborders per cable made from 1.2 to 1.8. These increases are attributable to the β_c's in the budgets (19) increasing with number of suborders, as well as the ability of the IP to efficiently fit fibers to many more of these longer cables.

6. Future Research

As construction of the world's information superhighway continues, the fiber-optic cable industry evolves and opens up new avenues for research. The technology for manufacturing optical fibers and cables is changing rapidly. For example, new equipment may soon allow the concatenation of cables with different fiber configurations. Customer demand for fiber-optic cables is also changing rapidly. For example, rather than placing orders for individual cables having specified lengths, customers are now beginning to order a total "aggregate" length of cable, giving the manufacturer some limited flexibility in the lengths and number of subcables. Customers are also increasingly demanding cables in which sets of 8 or 12 fibers are fused

Figure 5 Fraction of Total Length Consumed That Is Scrap

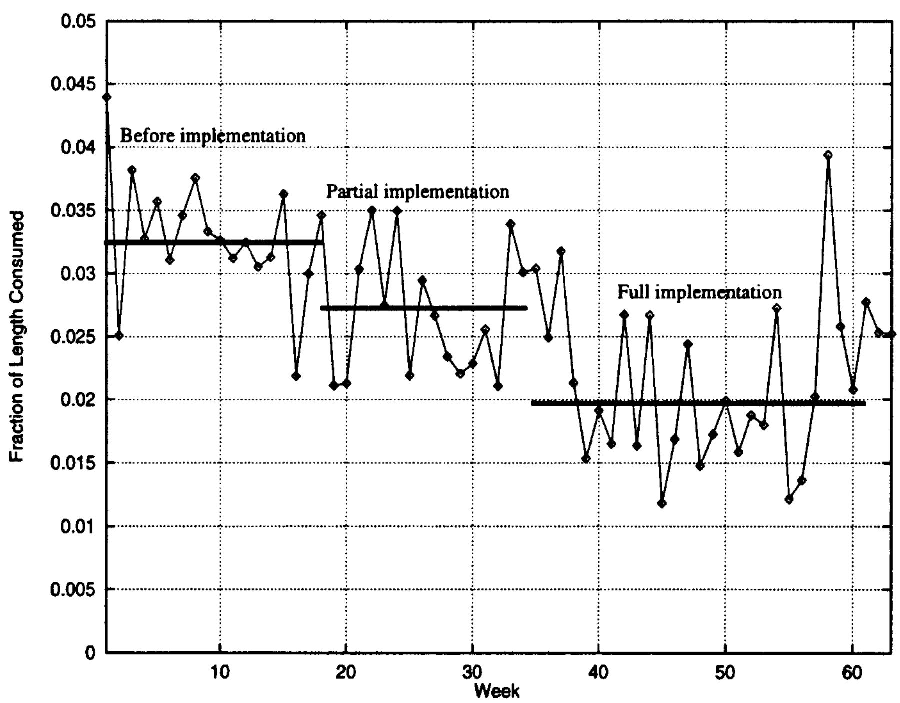
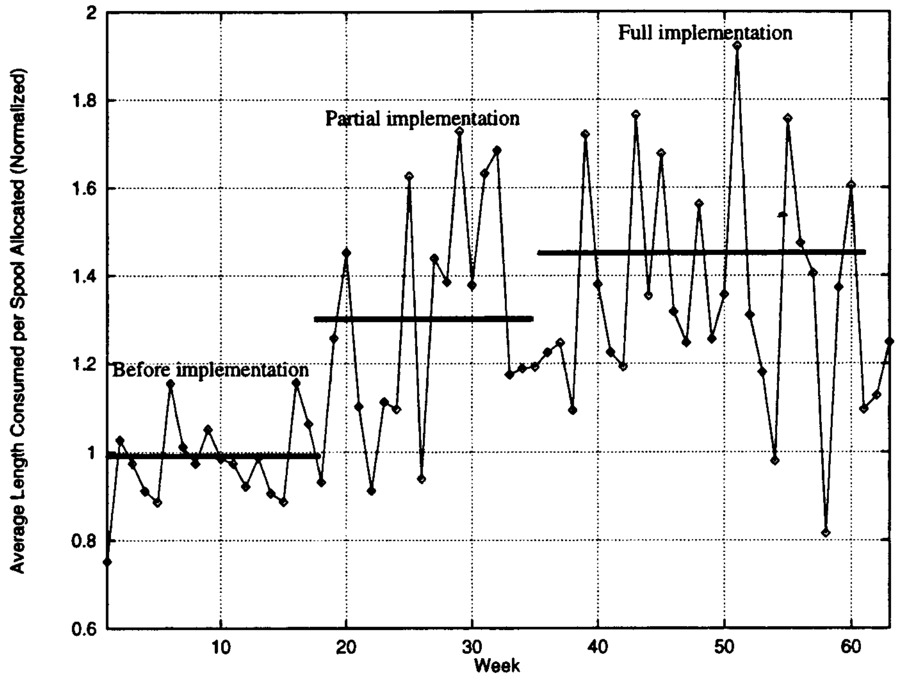


Figure 6 **Average Length of Fiber Consumed Per Spool Allocated**



together, rather than bundled loose as we have described.

However, despite all of these changes, the basic value function approach we have developed in this paper can still be used, and studied, as part of the core decision logic embedded within the next generation of allocation algorithms.[1]

[1]This work was supported, in part, by a grant from Lucent Technologies to Georgia Tech. In addition, Adelman was supported by a Department of Energy Pre-Doctoral Fellowship for Integrated Manufacturing and by the University of Chicago Graduate School of Business. Nemhauser was supported by NSF grant DDM-9115768. The authors would like to thank Larry Northcraft for many helpful suggestions in the early stages of this work. We would also like to thank the three anonymous referees whose suggestions substantially improved this paper, and David Gay, co-author of AMPL (Fourer et al. 1993), for his valuable assistance throughout the implementation.

References

Adelman, D. 1997. Remnant inventory systems. Ph.D. Dissertation, Georgia Institute of Technology, School of Industrial and Systems Engineering, Atlanta, Georgia.

——, G. L. Nemhauser. 1996. Price-directed control of remnant inventory systems. *Oper. Res.*, to appear.

Bellman, R. E. 1957. *Dynamic Programming.* Princeton University Press, New Jersey.

Bellman, R. E., S. E. Dreyfus. 1959. Functional approximations and dynamic programming. *Mathematical Tables and Other Aids to Computation* **13** 247–251.

Bertsekas, D. P., J. N. Tsitsiklis. 1996. *Neuro-Dynamic Programming.* Athena Scientific, Belmont, Massachusetts.

Clements, D. P., J. M. Crawford, D. E. Joslin, G. L. Nemhauser, M. E. Puttlitz, M. W. P. Savelsbergh. 1997. Heuristic optimization: A hybrid AI/OR approach. In *Proceedings of CP97: Constraint-directed scheduling.* Schloss Hagenberg, Austria.

CPLEX. 1995. *Using the CPLEX Callable Library.* CPLEX Optimization, Inc.

Dobson, G., U. S. Karmarkar, J. L. Rummel. 1987. Batching to minimize flow times on one machine. *Management Sci.* **33**(6) 784–799.

Fourer, R., D. M. Gay, B. W. Kernighan. 1993. *AMPL: A Modeling Language For Mathematical Programming.* Boyd & Fraser Publishing Company, Danvers, Massachusetts.

Godfrey, G., W. B. Powell. 1997. An adaptive approximation method for stochastic, dynamic programs, with applications to inventory and distribution problems. Technical Report SOR-97-10, Dept. of Civil Engineering and Operations Research, Princeton University, Princeton, New Jersey.

Gue, K. R., G. L. Nemhauser, M. Padron. 1997. Production scheduling in almost continuous time. *IIE Transactions* **29**(5) 391–398.

Johnston, R. E. 1992. Dimensional efficiency in cable manufacturing:

Problems and solutions. *Mathematical and Computer Modelling* **16**(1) 19–35.

Lee, H. L., C. S. Tang. 1997. Modelling the costs and benefits of delayed product differentiation. *Management Sci.* **43**(1) 40–53.

Murr, M. R. 1992. Some stochastic problems in fiber production. Ph.D. Dissertation, Rutgers University, New Brunswick.

Nandakumar, P., J. L. Rummel. 1998. A subassembly manufacturing yield problem with multiple production runs. *Oper. Res.* **46** S50–S64.

Nemhauser, G. L., M. W. P. Savelsbergh, G. C. Sigismondi. 1994. MINTO, a mixed INTeger Optimizer. *Oper. Res. Lett.* **15**(1) 47–58.

——, L. A. Wolsey. 1988. *Integer and Combinatorial Optimization*. John Wiley, New York.

Northcraft, L. P. 1974. Computerized cable inventory. *Industrial Engineering* **6**(2) 45–49.

Schiesel, S. Material shortage affects prices of optical cables. *New York Times*, November 4, 1996.

Accepted by Michael Magazine; received January 7, 1997. This paper has been with the authors for 11 months for 3 revisions.