# IMPReSS: An Automated Production-Planning and Delivery-Quotation System at Harris Corporation—Semiconductor Sector

ROBERT C. LEACHMAN

*Engineering Systems Research Center*
*University of California at Berkeley*
*Berkeley, California 94720*

ROBERT F. BENSON

*Engineering Systems Research Center*
*University of California at Berkeley*

CHIHWEI LIU

*Tyecin Systems, Inc.*
*Four Main Street*
*Los Altos, California 94022*

DALE J. RAAR

*Aeon Decision Engineering, Inc.*
*1290 Oak Hampton Road*
*Holland, Michigan 49425*

IMPReSS, an optimization-based production planning system at Harris Corporation's semiconductor sector, generates capacity-feasible production schedules for a worldwide manufacturing network and quotes product delivery dates in response to customer inquiries. The planning engine of IMPReSS is the Berkeley Planning System (BPS), which models the problem in a form that permits linear programming optimization. BPS embeds formulation techniques for planning the requirements of binning and substitutable products, for representing dynamic capacity consumption by reentrant process flows, and for developing multiple optimization calculations that reflect marketing priorities. It uses a heuristic decomposition strategy to break the overall problem into several manageable calculations. Its implementation raised on-time deliveries of line items from 75 to 95 percent without increasing inventories, enabled the sector to expand its markets and its market share, and helped move the sector from a loss of $75 million to profit of over $40 million annually.

Harris Corporation is an electronics and electronic systems company based in Melbourne, Florida, with annual sales approaching $3.5 billion. The corpo-

0092-2102/96/2601/0006$01.25

INDUSTRIES—ELECTRIC/ELECTRONIC
PRODUCTION/SCHEDULING—APPLICATIONS

ration consists of several sectors in different lines of business. Harris's semiconductor sector has annual sales of $670 million.

Prior to 1989, the semiconductor sector enjoyed a profitable but much smaller business producing sophisticated niche products serving the military and aerospace markets. It was famous for its "rad-hard" (radiation-hardened) process technology, which enables production of devices exceptionally resistant to solar radiation and thus preferred by designers of satellites, spacecraft, missiles, high-flying aircraft, and so forth. A substantial portion of its sales were in components supplied to defense contractors or aerospace companies who were prime government contractors for weapons or space exploration programs. Other products sold to commercial customers also tended to be proprietary,

In late 1988, Harris purchased the GE Solid State (GESS) semiconductor product lines and manufacturing facilities from General Electric; they included both the former RCA Solid State as well as GE-proper semiconductor products and manufacturing facilities. (Subsequent to its 1986 acquisition of RCA, GE had merged the RCA products and facilities with its own GE semiconductor products and facilities, retaining the Solid State name for the merged organization.) This acquisition by Harris roughly tripled the size of the sector in terms of products and manufacturing facilities and substantially increased the proportion of production in competitive commercial product lines, such as automotive and telecommunications products. Harris now had to provide competitive on-time delivery performance over a much greater product mix.

To achieve operational economies, the sector needed to concentrate like process technologies for the Harris, RCA, and GE product lines in common manufacturing facilities and to rationalize the newly combined factory and distribution networks. Given the huge size of the acquisition and the debt load involved, sector managers wanted to make these economies quickly. Unfortunately, the manufacturing databases, control systems, and planning systems in use at GESS and at Harris were different and very difficult to integrate. They could not simply select one set of systems and immediately begin using it to manage all manufacturing facilities and all product lines of the combined company.

After the merger, production planners had to cope with data provided in multiple formats on multiple systems and, in some cases, cope with serious gaps in information. The sector developed a reputation for late delivery. Throughout 1989, a metric measuring the percent of ordered line items delivered within one day of promised delivery date hovered around 75 percent. One survey indicated that 60 percent of the customers wished to replace Harris as a vendor. The following spring, sector sales executives estimated that $100 million in potential sales had been lost in calendar year 1989 because of its noncompetitive delivery performance. Sales continued to decline by about $100 million per year until Harris ultimately turned delivery performance around. In 1989, the sector

In fiscal 1991, the sector reported a loss of $75 million.

started reporting losses which continued to mount in 1990 and 1991; in fiscal 1991, it reported a loss of $75 million.

**A Global Planning Project**

Sector executives took action. In mid-1989 they asked the sector manufacturing systems department to perform a global planning system study. The study was to define the scope and requirements of an efficient, integrated production-planning and delivery-quotation system serving the entire sector, to review available software and planning methodologies, and to make prompt recommendations for an implementation plan.

The study found that the newly expanded sector was selling more than 10,000 finished goods produced in a factory network including more than 30 manufacturing facilities in the United States and Asia. Planning and delivery quotation was decentralized and conducted with a myriad of systems, policies, and personnel. The sector employed two large MRP systems, one for a subset of the former Harris facilities and one for a subset of the former GESS facilities. Many smaller MRP-like spreadsheet analyses were performed by factory planners. Data on demand, work in process, inventory, and capacity were weak in quality and were judged differently by various participants. Working out a plan inevitably involved meetings to negotiate differences, leading to multiple planning iterations. Sector-wide planning cycles were undertaken only once a month and consumed two weeks or more. Quotations and delivery commitments were often little more than judgments made by planners who were forced to work with incomplete information.

Observing practices in other firms in the industry, the study found that most large semiconductor companies had developed their own applications for company-wide production planning, generally following an architecture similar to that of commercially marketed MRP II systems. Developing and implementing these systems had taken years and a large staff. All were operated on main-frame computers, tended to by a large staff who maintained and improved the applications. One company was willing to negotiate selling its planning system to Harris.

MRP logic has serious shortcomings for application to semiconductor manufacturing. Harris's experience with the MRP systems then in use in the sector revealed these weaknesses, even though the systems were well designed (as MRP systems go) and incorporated several desirable features. The sector's research and development department was aware of these weaknesses, and since 1987 it had been funding research at the University of California at Berkeley aimed at developing an automated production-planning system for semiconductor manufacturing. Other semiconductor manufacturers also had been funding this research, begun in 1984.

The research effort at Berkeley had yielded an optimization-based approach to semiconductor production planning. The Berkeley researchers had programmed this approach as a prototype software package that they offered to research sponsors for industrial testing or application. A number of such implementations were carried out, but none on a full companywide scale involving all products and all plants. Based on the results of field experience and direction from industry sponsors, year after year the Berkeley researchers steadily improved the mathematical models and the software embedding these models. This software became known as the Berkeley Planning System (BPS). Harris had implemented BPS in 1987 to handle production planning of three wafer fabrication facilities ("wafer fabs" or just "fabs" for short) located at the Palm Bay, Florida manufacturing site, and the researchers had steadily improved the application based on feedback from the users. It became the regular planning mechanism for all products produced at these facilities beginning in mid-1988.

The sector's experience with the prototype BPS application had been very positive, although it was implemented on a fairly small scale. The factory network in Palm Bay, though small, featured most of the complexities of the companywide planning problem that frustrated the effective application of more conventional planning methodologies: binning and substitution in product structures, hard capacity constraints on reentrant process flows, and market potential far in excess of available capacity, juxtaposed against critical outstanding orders for which on-time delivery must be protected. Factory planners praised BPS as an effective planning tool for this situation.

Given the urgency of the delivery problem, the approach to planning systems taken at other large semiconductor companies—involving a large expenditure, a large staff, and a lot of time—did not appeal to the sector managers. Their feeling was that the sector did not have a lot of time. Karl McCalley, vice-president of Sector Information Systems, argued that the sector would never be able to match the delivery performance of such industry giants as NEC, Intel, Texas Instruments, or Motorola by emulating their approach to planning, simply because it could not hope to match their level of expenditures. If the sector had any hope, it would have to come from a strategy to outsmart them, and so McCalley argued in favor of expanding the BPS application into a companywide planning system. But other sector executive managers hesitated to embrace an operations-research-based technology that they were unfamiliar with, that was not in regular large-scale use at any major semiconductor company, and that was not supported commercially as a planning software product. Fortunately, the sector president at the time, Jon Cornell, did not share these fears. Cornell was familiar with operations research techniques, and in fact, he had experimentally developed a linear programming-based wafer fab scheduling model in his earlier years as a fab manager.

MRP logic has serious shortcomings.

In the spring of 1990, Harris decided to start a full-scale effort to develop and implement an automated sectorwide production-planning and delivery-quotation system, with an enhanced BPS serving as the planning engine of the new system. The manufacturing systems department came up with an acronym for the new system: IMPReSS, standing for integrated manufacturing production requirements scheduling system. IMPReSS was envisioned to provide a worldwide database of factory status and factory capabilities, and of order status and marketing demands. Running off this database would be applications for production planning, delivery quotation, and order entry. The entire system would run on workstation computer hardware. Given the crisis it faced, the sector established an extremely aggressive one-year schedule for project completion. The lead UC Berkeley faculty investigator (Leachman) for BPS took a one-year leave of absence from teaching to work full time directing much of the project. Harris hired a masters graduate (Raar) from Berkeley, who had enhanced the BPS application at Harris as his masters project, as a permanent Harris employee to lead database and system development. Harris recruited many other Berkeley masters students to carry out their masters intern project assignments by helping to develop capacity databases in various factories around the world. At the same time, two doctoral students at Berkeley (Benson and Liu) would develop and program an upgraded version of BPS to handle the companywide problem at Harris. After graduation, Liu would take a permanent position at Harris, serving as the key technical staff member for the planning engine.

Harris undertook an internal publicity campaign to stress the importance of the IMPReSS project to all manufacturing, marketing, sales, and engineering staff. It distributed white polo shirts, buttons, and pens emblazoned with an IMPReSS logo. The sector newsletter highlighted articles describing the project. At the project kick-off, Jon Cornell, wearing his IMPReSS shirt, spoke to the entire sector via closed-circuit broadcast. "IMPReSS is the most important project in the sector, and everyone must do whatever is necessary to expedite the project and to insure its success. The sector will not survive unless we solve our delivery problem. If IMPReSS succeeds, we can succeed. If it fails, we will surely fail."

### Scope of the Planning System

IMPReSS incorporates a number of subsystems and databases that collectively accomplish automated planning (Figure 1). At the center of IMPReSS is the planning engine, a batch application that calculates a companywide production plan when given certain inputs from supporting systems. All the manufacturing areas located around the world can be scheduled in a single planning cycle of the engine.

An on-line system for quotation, order entry, and customer shipment maintains in real time a schedule of product availability, that is, a schedule of the uncommitted portion of the production plan. As inquiries are received from customers, the system calculates the best delivery schedule it can offer and then reserves this supply for the customer, somewhat akin to the way an airline reservation system works. If a customer places an order, it processes the order and includes it on the "order board," a list of current customer commitments; otherwise, the reservation is cancelled and the supply reverts back to the availability. This system also prioritizes shipments to customers from finished goods inventory, furnishing order picking lists to the product distribution centers.

Most other information flows within IMPReSS occur in batch mode. The order

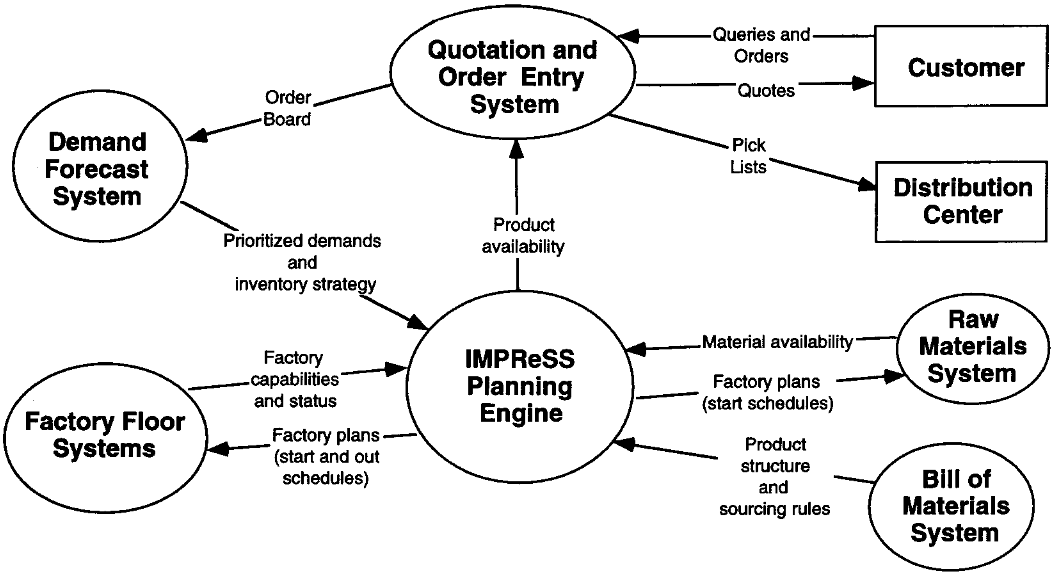


**Figure 1: Information flows between systems that make up IMPReSS are shown as arrows. In a planning cycle, the planning engine receives demand and priority inputs from the demand forecast system. Factory floor systems provide inputs concerning factory capabilities and status (yields, cycle times, capacities, work in process, and static inventory status). The raw materials system provides inputs concerning the availability of materials. The bill of materials system provides inputs concerning the product structure and sourcing rules. The planning engine provides the planned product availability to the quotation and order entry system and factory schedules to the factory floor systems and the raw materials system. An on-line quotation and order entry system provides delivery quotations to customers, accepts customer orders, and sends pick lists to the distribution centers.**

board is passed periodically from the order entry system to a demand forecast system. This system prepares market forecasts for each finished good type, utilizing the current order board as supporting information. These forecasts serve as inputs to production-planning calculations.

Semiconductor manufacturers normally cannot accommodate all market demands promptly within the capacities of their existing manufacturing facilities. To properly guide the loading and product support decisions made in production-planning calculations, the planning engine needs more information than just the total forecast for each product. Forecasts must be partitioned into subsets of demands with different priorities from a sales or marketing viewpoint, and the relative priorities of these demand subsets need to be expressed to the planning engine. As an obvious example, demands representing customer orders should take priority over demands representing the unrealized (and uncertain) portion of sales forecasts. As another example, a forecast for sales of a high-margin custom product should have higher priority than a forecast for sales of a low-margin commodity product, given that the two forecasts are equally reliable. The demands communicated to the planning engine by the forecast system are therefore sorted into priority classes.

A materials system used to track and procure raw materials supplies the planning engine with a schedule of the availabilities inside vendor lead times of scarce raw materials. Such availabilities constrain the production plan. After calculation of the plan, the planning engine supplies the materials system with the planned product starts in each manufacturing area, from which it can calculate material requirements. A basic MRP system is suitable for the materials system application.

For each manufacturing area, factory floor systems maintain the status of work in process (WIP) and static product inventory. These systems include applications to convert WIP into an equivalent projected out schedule (a WIP-out projection) for the manufacturing area. These systems also include databases for maintaining models of factory capability. These models describe the routes followed by products through the factory, including relevant data about the operations on each route, such as manufacturing yields, lead times (hereafter called *cycle times*, the standard terminology in the industry), equipment processing rates, and equipment capacities. They pass these data to the planning engine in a standard format. After calculating a worldwide plan, the planning engine passes back to each factory schedules of *starts* and *outs* (that is, schedules of lot releases and completions) for which it will be held accountable.

> Converting to a standard data structure caused conflicts.

A bill-of-materials (BOM) system supplies the planning engine with the official product structure and sourcing rules. These data specify the factories authorized to produce each final or intermediate product, and each product's alternative source products on the next lower level of the product structure. The BOM data also goes to the order entry system to insure that it does not issue delivery quotations for products currently in engineering "hold" status or not yet passing qualification tests.

Versions of some of the peripheral systems supporting the planning engine existed when the project started, although they required some upgrading. While all the systems are critical to the proper functioning of the overall planning and quotation system, we shall focus in this article on the planning engine, which embodies most of the decision-making logic in the system.

A planning cycle (Figure 2) starts with demand inputs divided into priority classes. Those that represent external or internal delivery commitments are the *order board classes:* confirmed orders, contractually guaranteed supply, scheduled test lots for product development, and so forth. Demand classes that represent replenishment of safety stocks are termed *inventory classes.* Other demand classes are termed *forecast classes.*

The first phase of the overall calculation is requirements planning. The planning engine subtracts worldwide finished goods inventory from the demands to determine net requirements for new finished goods output serving each priority class. Next, the engine performs an MRP-type calculation, working net demands in each class

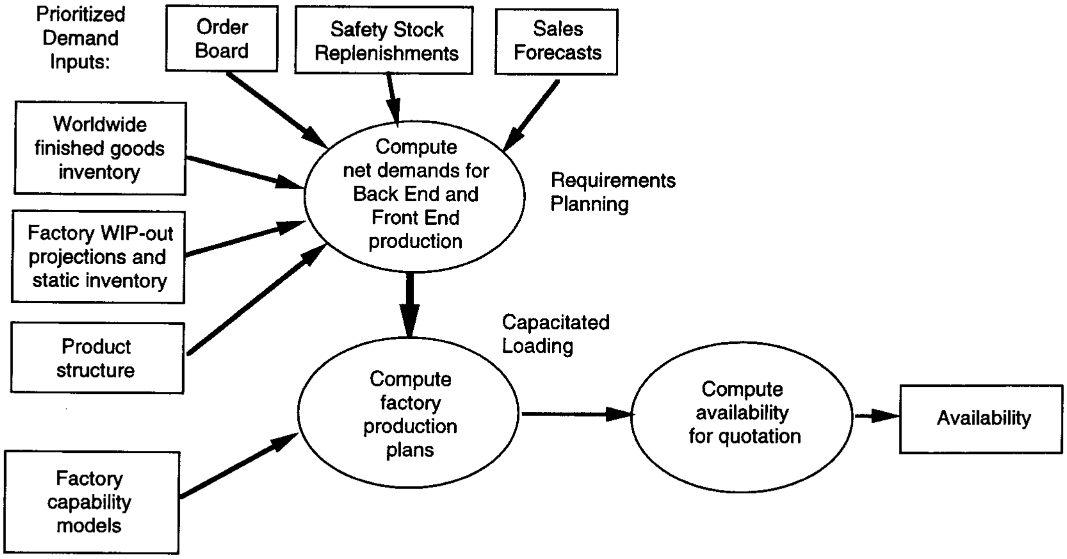

Prioritized Demand Inputs:
Order Board
Safety Stock Replenishments
Sales Forecasts
Worldwide finished goods inventory
Factory WIP-out projections and static inventory
Product structure
Compute net demands for Back End and Front End production
Requirements Planning
Capacitated Loading
Compute factory production plans
Factory capability models
Compute availability for quotation
Availability

**Figure 2: A planning cycle consists of three phases: (1) requirements planning, in which inventory and work in process are subtracted from the prioritized demands to determine net demands for new starts in back-end and front-end production; (2) capacitated loading, in which the net demands are loaded onto the factories according to the priorities subject to resource availabilities; and (3) computation of the availability, in which order board demands are subtracted from the production plan to determine the supply of products available for new delivery quotations.**

backwards through the product structure, with allocations of factory WIP-out projections and static inventory determining prioritized net requirements at each level of the bill of materials.

Semiconductor manufacturing may be thought of as a two-stage process: (1) fabricating integrated circuit structures on silicon wafers and testing the circuits (wafer fab and wafer probe), and (2) slicing wafers into individual devices, packaging the devices and testing them (device assembly and device test). Factories performing these stages are collectively called *front-end plants* and *back-end plants.* The key outputs of requirements planning are prioritized net demands for new production in front-end and back-end plants.

The next phase is capacitated loading, in which the requirements for new factory starts are mitigated by the factory capacity models and materials limitations to determine the best scheduled response to the marketing demands and priorities. After making this calculation for all manufacturing areas, the planning engine combines worldwide planned output of finished goods with the finished goods inventory to form the worldwide supply schedule. Finally, it nets out both the order board and inventory classes of demands from this supply line to obtain an updated availability schedule for use in the quotation and order entry system.

## Challenges of the Semiconductor Planning Problem

Several characteristics of semiconductor manufacturing conspire to make planning production quite challenging. Semiconductor factories are capital-intensive facilities that are operated 24 hours per day, seven days per week. The capabilities and performance of the processing equipment determine capacity. Working overtime or adding staff are not means of resolving infeasibilities in the production plan. The potential market frequently exceeds a factory's capacity (particularly for products that it must produce using the most recent technology). If a firm is to provide on-time delivery, it must limit the market demand it accepts to fit its capacity. Employing planning methodologies (such as MRP) that have difficulty developing efficient, capacity-feasible factory schedules would compromise the firm's planning.

The loading of equipment capacity by semiconductor products is unusually complex. Products are routed through hundreds of steps, requiring weeks to traverse. The routings comprise reentrant process flows in which a product visits a particular equipment type many times for performance of different processing steps, interspersed with steps performed on other types of equipment. For example, wafers following the wafer fab process flow for an 18-layer circuit design must visit the photolithography workstation 18 times. Fab cycle times range up to eight weeks; test cycle times for complex devices can range up to three weeks. This means that newly released production lots compete for capacity with WIP; similarly, product releases made over several weeks compete with each other for scarce capacity.

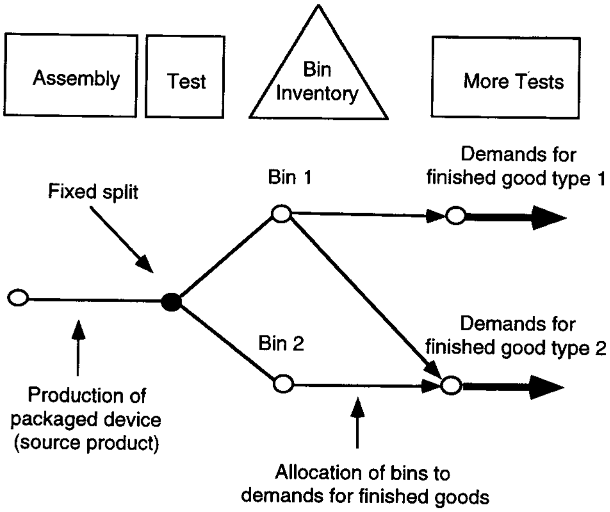
Traditional forms of capacity analysis apply capacity constraints to total factory input rates or output rates in each planning period, without consideration of planned rates in adjacent periods. When product mix is dynamic (as it is at Harris and most semiconductor manufacturers), these forms of analysis are inaccurate.

Many semiconductor product families include various quality grades and design revisions. Thus alternative or substitutable source products are very common in semiconductor product structures. In particular, many intermediate products are the result of *binning,* whereby a distribution of several quality-graded products emerges from testing a single manufacturing lot of source product. Each bin defines a specific range of performance for one or several electrical attributes of performance, for example, a bin definition might be "speed between 30 and 40 megahertz and power consumption less than 100 milliamps." The fractions of the source product falling into each bin of quality are known as the *bin splits.* The bin splits are characteristic of the manufacturing process and must be regarded as prespecified (but probabilistic) for planning purposes. In some cases, alternative testing procedures generate different bin distributions.

Each finished goods type has specific requirements for electrical performance. In general, there are a number of bins whose attributes fulfill these requirements. Conversely, a particular bin is generally suitable for a number of different finished goods types. Suitable source bins for each finished goods type are listed in an accept bin table; such tables form an integral part of the product structure and sourcing rules in the BOM system.

These binning and substitution possibilities in the product structure frustrate the application of MRP logic to perform requirements planning. When product mix is dynamic, it is difficult to establish input parameters that enable MRP logic to generate feasible net requirements for factory releases, let alone efficient requirements (Figure 3).

In studying global planning, the sector manufacturing systems department observed that other semiconductor companies tried to apply MRP logic to binning structures in two ways. One method is to simply ignore the bin splits and equate requirements for the source product to the sum of requirements for the finished goods. For the example in Figure 3, this method would set production of the source product to be 105 in period 1 and 120 in period 2, which would result in a shortage of four units of finished good 1 in the first period. The other method is to select a single bin as the "driver bin" upon which to base the requirements planning calculations. Under this method, the planner chooses one of the types of finished goods and divides its demand by the split for the corresponding bin to determine the quantity of source product needed. For this example, if bin 1 is selected as the driver bin, production of the source product would be computed as 125 in period 1 and 50 in period 2, leading to a shortage of 50 units of finished good 2 in the second period. If bin 2 is the driver bin, production would be computed as 100 in period 1 and 138 in period 2, leading to a shortage of five units


Assembly
Test
Bin Inventory
More Tests
Fixed split
Bin 1
Demands for finished good type 1
Bin 2
Demands for finished good type 2
Production of packaged device (source product)
Allocation of bins to demands for finished goods

Data:

| | Bin Split | Finished Goods Demands t=1 | Finished Goods Demands t=2 |
|---|---|---|---|
| Bin 1/Type 1 | 0.20 | 25 | 10 |
| Bin 2/Type 2 | 0.80 | 80 | 110 |

**Figure 3: In this simplistic example of requirements planning through binning structures, we consider a product structure of one source product, two bins and two finished goods. The demands in time periods 1 and 2 for finished goods types 1 and 2 must be translated into net demands for the source product. The input flow of the source product must satisfy the given, required outflows of finished goods. Bin 1, which corresponds to the electrical requirements of finished good type 1, is suitable for filling demands for either finished good type; bin 2 is suitable only for finished good type 2. The flow of source product is split into bins according to fixed percentages, here, 20 percent to bin 1 and 80 percent to bin 2. For simplicity, there is no initial inventory or work in process, and the manufacturing cycle time is zero.**

of finished good 1 in the first period.

The shortfall of capacity to market demand increases the importance of representing marketing concerns in the planning calculation, since the firm may have to delay or turn away some of the demand. At the same time, it must protect delivery dates for outstanding customer orders and other customer commitments. Thus the planning engine must explicitly consider the relative priorities for accommodating demands from different sources in making its planning decisions.

Forecasting the sales of individual semiconductor products varies in difficulty, depending upon whether the product is custom or commodity-like. Manufacturing cycle times are quite long and they vary depending on the product's complexity. The firm must decide for its various products how far through the manufacturing process it will build product in response to forecasts versus delaying production until it receives actual orders. Given the uncertainty in forecasts, it was clearly desirable for Harris to increase the frequency of replanning from monthly to weekly or even daily.

The sheer scale of the planning problem for a large semiconductor manufacturer is daunting. A full-scale planning calculation at Harris Corporation's semiconductor sector includes on the order of 2,500 wafer types, 6,000 packaged device types, 10,000 finished goods types, 200 types of scarce processing equipment, and 200 types of scarce raw materials. Many products have alternative routings through the manufacturing network. The desired horizon for quotation and planning is 18 months. Consider a planning calculation that is to specify production quantities with weekly resolution for the first two months, monthly resolution out to one year, and in quarters beyond that, for a total of about 20 time periods. With variables representing the quantities of each product released into each factory in each planning period, such a problem would involve about one half million constraints on about the same number of variables if the problem were formulated as a single mathematical optimization model.

### Representing the Product Structure, the Production Processes, and the Manufacturing Network

To develop and implement a practical optimization-based planning system, we needed a standardized data model for the planning problem. Data structures defining the manufacturing network, production processes, and product structures were different among the Harris-proper, ex-RCA and ex-GE portions of the merged company. A simple union of these structures would have led to a large number of levels in the bill of materials, making formulation of practical optimization models very difficult. Moreover, in many cases, the existing structures omitted data that were critical to generating efficient or even feasible production plans.

Our standardized manufacturing process flow for nonbinning semiconductor products (Figure 4) consists of five serial manufacturing processes: base wafer fabrication (in wafer fab), wafer fabrication (wafer fab), wafer electrical probe (probe), device assembly (assembly), and device test (brand, test, and pack). Harris has a number of manufacturing sites around the world. Each site, or even a single manufac-

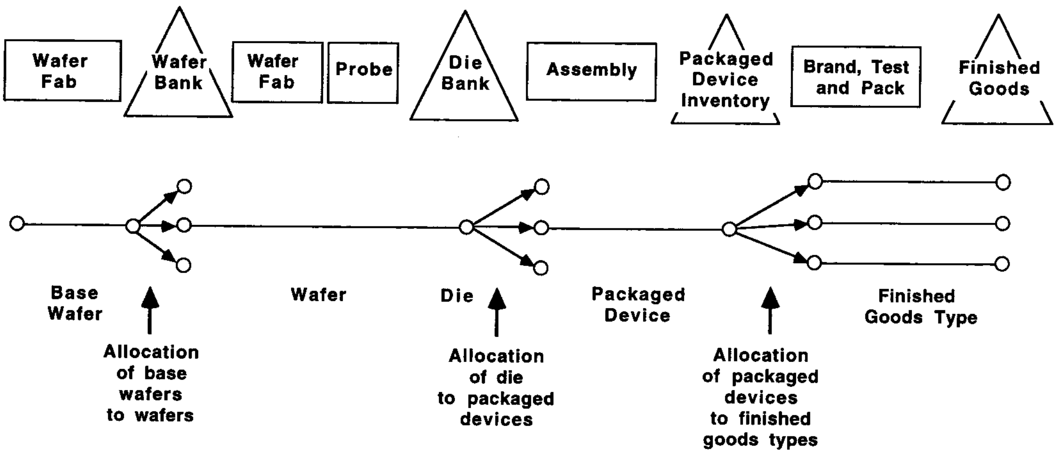

Wafer Fab
Wafer Bank
Wafer Fab
Probe
Die Bank
Assembly
Packaged Device Inventory
Brand, Test and Pack
Finished Goods
Base Wafer
Wafer
Die
Packaged Device
Finished Goods Type
Allocation of base wafers to wafers
Allocation of die to packaged devices
Allocation of packaged devices to finished goods types

**Figure 4: The pattern of arcs and nodes in the figure represents the product structure for non-binning products. Open nodes designate points of change in the product structure. In-line arcs with no arrowheads denote manufacturing process flows in which there are no changes in the product structure. For example, a wafer fab process and a follow-on probe process form a process flow. Arcs with arrowheads on them denote possible allocation flows from completed source product to various follow-on products.**

turing facility at the site, may operate several or all of these processes. Generally, one site integrates the first three processes (the front end), and a different site, the last two processes (the back end).

The first three manufacturing stages process round wafers of silicon, termed *base wafers* in the first stage, and just *wafers* in the second and third stages. The fab processes imprint the wafers with many identical patterns of an integrated circuit; each circuit pattern on the wafer is called a *die.* The probe process electrically tests each die on each completed wafer. Our subdivision of the overall wafer fabrication process is made to allow for the maintenance of an inventory of semi-processed wafers known as *wafer bank.* The inventory of wafers that have completed the third stage is known as *die bank.* Wafer fab processes have hundreds of serial operations; wafer probe processes generally involve only a few steps. Small lots of 10 to 50 wafers flow through the wafer fab and wafer probe processes.

In the assembly stage, the wafers are sliced up into individual chips which are sealed in plastic or ceramic packages to become *packaged devices.* In the next and final stage, the packaged devices are tested, labelled, and packed for customer shipment. The assembly and test manufacturing areas process individual devices in manufacturing lots of 500 to 2,000 devices.

The product structure for semiconductor products is an arborescent (branching) rather than a coalescent type of structure typical of assembled products. In general, a single base wafer is the source product for several types of wafers; a single die type is the source product for several types of packaged devices; and a single packaged device is the source product for several finished goods types. The base wafer stage is vacant for many types of products, but it is useful for planning production of product families involving gate-arrays, read-only memory (ROM) codes, and epitaxial base wafers.

At corporate inventory points, where the product structure changes, the planning engine must allocate completed product to follow-on uses, depending on the demands. The planning engine specifies product starts and outs at these points. Manufacturing activity may generate inventories at other points in the manufacturing network, but they serve no corporate purposes and are simply regarded as queues of work in process (WIP) in planning production. The production planning model assumes that the series of operations between corporate inventory points is to be managed as a continuous flow process, not individually scheduled by global planning. However, we model the string of operations comprising these process flows in enough detail to prevent the planning engine from scheduling starts to a process flow unless it finds that capacity is sufficient to permit their movement as a flow to the next corporate inventory point.

For binned products, the process flows are similar, except that the test process is split into two parts, the *initial test* and the *final test* (Figure 5). The initial test process subjects packaged devices to one or a series of electrical tests to characterize the various electrical attributes of interest. Generally, several combinations of attributes emerge from each manufacturing lot, each termed a *bin.* The inventory of binned packaged devices following completion of the initial test process is known as *class stores.* The planning engine schedules production in the initial test process in terms of packaged devices, with output at class stores expressed in terms of bins.

After binning, the planning engine must decide how to allocate bins among alternative finished goods types for which the bins are acceptable. The planning engine then schedules production in the final test process flow (brand, retest, and pack) in

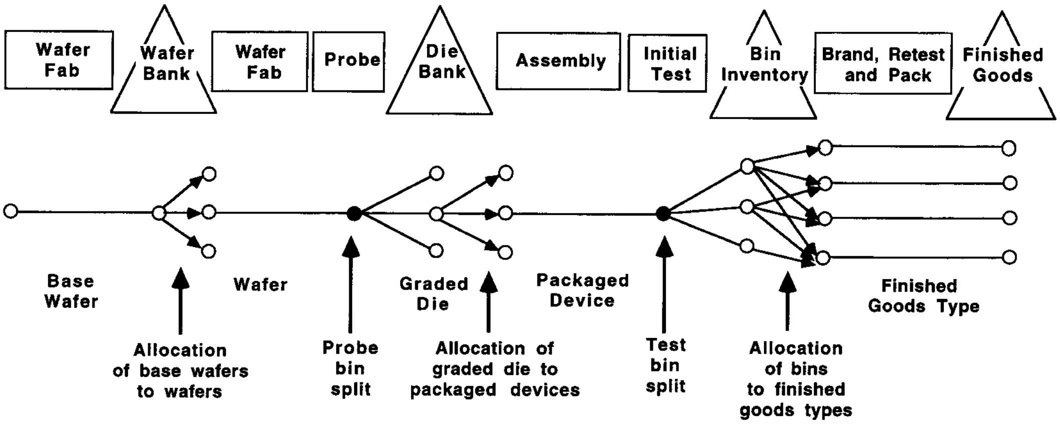


**Figure 5: The pattern of arcs and nodes in this figure represents the product structure for products featuring binning. The solid nodes in the product structure represent bin splits, at which the fractions of source product flowing to output products are prespecified characteristics of the manufacturing process.**

terms of finished goods types. In the case that the final test process includes a *burn-in* operation (prolonged operation of the devices in an oven), binning tests may be repeated in the final test process to find out whether attributes have changed as a result; if they have, devices may be downgraded (assigned to a lower grade bin) and returned to class stores. However, such fallout is typically very small compared to the fallout in the initial test; for planning purposes, it may be regarded simply as yield loss.

Binning also may occur in wafer probe, where a finished wafer may contain two or more usable electrical grades of die. Analogous to the allocation problem after initial test, more than one graded die type may be suitable as source product for a particular packaged device type. Moreover, design revisions may make die from several wafer types suitable for the same packaged device.

The product structure for binned products can be used to model nonbinning products, with appropriate conventions: For nonbinning wafer-probe processes, one graded die type is generated with a 100 percent bin split; for nonbinning packaged devices, one binned packaged device is generated with a 100 percent bin split; and the initial test process flow for such packaged devices includes no operations and has 100 percent yield and 0.0 days manufacturing cycle time. With this convention, there are four standard corporate inventory points (wafer bank, die bank, class stores, and finished goods) and six standard manufacturing process flows (base wafer fab, wafer fab, probe, assembly, initial test, and final test). Two of the process flows (base wafer fab and initial test) can be vacant for some product families.

Harris has a number of front-end and back-end sites. It can produce certain die types at more than one front-end site; much less commonly, it can produce certain finished goods types at more than one back-end site (at most two). Typically it produces low-volume die types in only one front-end fab but often produces high-volume die types in two or more fabs, distributing the manufacturing volume to use capacity efficiently.

The sector produces two broad categories of products, each with their own separate set of factories. Each of the two sets of factories is termed a *manufacturing network,* for which the planning engine can prepare production plans separately.

### Marketing Priorities and Controls

The total demand for any product in a given time period includes different types of demand with different priorities. Using BPS, planners divide the total forecast for each product in each time period into *priority classes* defined by marketing and sales management. Demands for each finished goods type typically appear in most or all of the classes. BPS first schedules production to meet the demands in the top priority class as on-time as is feasible, then schedules additional production to meet second priority demands as on-time as is feasible, and so forth.

Each class belongs to one of three types: order-board classes (including customer commitments and firm demands), inventory replenishment classes (replenishment of safety stocks), and forecast classes (projections of future customer demands). In strict priority order, order-board classes precede inventory classes, which precede forecast classes. In this way, BPS provides maximum service to previous customer commitments and replenishes safety stocks before making product available for future customer commitments. In principle, any number of classes may be tendered to BPS; in practice, about four or five classes is enough to generate desirable plans, and further partitioning tends to have no effect on generated plans. For example, planners might define separate classes for order board, undated order requests, safety-stock replenishments, the reliable portion of sales forecasts as yet unconsumed by orders, and the remainder of sales forecasts. Dividing forecasts into two classes insures that availability is generated for more reliable forecasts first, with additional availability generated for more speculative forecasts only as capacity permits.

To the IMPReSS team, the poor data quality came as a shock.

To resolve resource competition within each class, BPS refers to prices defined for each finished goods type, indicating the marketing value of the different product demands in the same class. For order board or inventory classes, these prices are used in capacitated loading models (discussed below) to define lateness costs per unit time associated with delaying support for the products from the committed time periods. For forecast classes, the price specified for a product is used in the loading models to represent the unit revenue available from sales of output supplied in or after the time period in which a demand is forecasted. In practice, the price for each finished goods type is equated to its average selling price (ASP).

Another parameter that expresses marketing policies is the *build-to-level code* defined for each finished good. It indicates the corporate inventory point to which production can be carried out without corresponding customer orders on hand. Using this parameter, marketing management can control inventory risk. In operation, BPS uses this parameter to limit production starts in the first time period of the production plan and to erase unrealized forecasts.

For example, a build-to-level code equal to die bank means actual back-end production is build-to-order only. Forecast demands for the finished good during the cycle time for the back end plus the length of the first planning period are not realizable (unless already matched by on-hand orders), and so BPS resets them to zero. BPS also limits back-end production starts in the first period of the plan to only those necessary to fill orders. In all following periods of the plan for back-end production, BPS plans production in response to both orders and forecasts. In this way, BPS plans future availability of the finished good, enabling automated quotation of delivery dates for future orders. Before the beginning of the second period, the production plan will be regenerated, and again BPS will schedule the back-end production in the first period of the new plan to be solely in response to orders received.

In summary, to reflect marketing priorities and concerns, the planning engine's capacitated loading decisions are guided by the build-to-level codes, the partitioning of demands into priority classes, and the defined product prices. However, the planning engine responds to these priorities only to the extent that the plans are WIP feasible and capacity feasible.

**Requirements Planning for Binning and Substitution Product Structures**

We cast the requirements-planning problem (Figure 3) as a simple linear programming problem. We introduce variables for the production of source product and the allocation of bins to finished goods in each time period. We define constraints to insure inventory balance of each finished goods type and of each bin in each time period. (Slack variables of these constraints represent back orders of finished goods types and inventories of bins.) Cash flows in the objective function include the sales revenue of each finished goods type (forecast demands case), backorder costs for supply that is delivered late (order-board demands case), inventory holding costs for excess bin output, and the incremental cost of producing additional source product.

The details of the formulation depend on the nature of demands. If they represent customer commitments, then we formulate the problem as the minimization of the costs incurred to meet demands as ontime as possible. Orders for delivery inside the manufacturing cycle time may inevitably be late if there are insufficient initial inventories and WIP. Therefore, balance constraints for periods inside the cycle time must allow back orders. But beyond the manufacturing cycle time, it is always feasible to schedule requirements meeting demands, and so the balance constraints for such periods do not include backorder variables.

On the other hand, if the demands consist solely of forecasts not yet consumed by orders, we need not respond to all demands. Indeed, it may be unprofitable to schedule total requirements for certain finished goods types, depending on the mix of demands relative to the bin splits. As an extreme example, suppose the split to bin 1 is 0.0001 and to bin 2, 0.9999, and the market for finished goods type 1 is large but that for type 2 is small. Fully responding to the market for finished goods type 1 may cost much more than the revenue it brings in. Thus to deal with forecast demands, we formulate the demand satisfaction constraints to allow back orders both inside and outside the cycle time. In this way, we plan availability for only the portion of forecast demands that is profitable.

For multiple demand classes, we should not solve the requirements planning problem independently for each class: the optimal binning strategy for the demand mix in one class may be a very inefficient starting point for handling the demands in the next lower priority class. To cope with this situation, we formulate the requirements planning problem for all demand classes in BPS as a single linear program, including separate variables and constraints indexed by demand class and combining the objective functions for each class into a single objective. In such a consolidated formulation, we define the demand quantities in the constraints for each class to be cumulative over all higher priority classes, and the variables for each class in each time period to represent the total production, allocation, inventory, and backorder levels serving the indexed class as well as all higher priority classes. We also add constraints expressing consistency between cumulative requirements calculated for consecutive classes to the formulation. Solving such a formulation defines the production requirements for servicing the demands in all cumulative classes. We give the details of a basic version of the linear programming formulation prepared by BPS for requirements planning in the appendix.

BPS formulates a separate linear programming model for each family of finished goods types (those made from the same source packaged devices). Generally, these families consist of tens or in extreme cases hundreds of finished goods types; thus, BPS solves many relatively small linear programs to perform the overall requirements planning calculation. Moreover, BPS plans families with no binning using conventional MRP techniques. In operation, the BPS software examines the structure of each product family and determines whether to apply optimization or MRP techniques. BPS then combines the results of linear programming and MRP calculations to state the net production requirements for packaged device output servicing all finished goods demands in each class.

In the most general case, a finished goods family may have multiple source packaged devices, each with alternative process flows with varying manufacturing costs, cycle times, and bin splits. The accept bins for a finished goods type may include bins obtainable from more than one source packaged device or process flow. For example, it is common for the same basic packaged device to be put through an elaborate testing procedure to generate a large set of bins, or through a simpler testing procedure that generates only lower grade bins. The simpler testing procedure affords a savings in test capacity requirements that could be attractive, depending on the mix of demands. The BPS software incorporates an extension of the formulation in the appendix to this case, using differences in production cost as a proxy for differences in capacity consumption.

### Capacitated Loading of Reentrant Process Flows

To generate feasible factory schedules, the planning system must analyze the workloads on such factory resources as processing equipment. Because of the reentrant process flows, the analysis must represent the distribution of resource loads through time associated with scheduled product starts. The BPS software formulates constraints using a rate-based, continuous-time model of production that approximates the discrete, lot-based production activity actually taking place in the factories.

In the rate-based model, we assume the scheduled starts in each time period are uniformly distributed over the entire period, moving through the factory as a continuous flow. We thus view the starts schedule for a process flow as a step function defined continuously over the entire planning horizon. The corresponding cumulative starts plan is a piecewise linear curve (Figure 6). Actual starts of the process flow are a series of events corresponding to lot releases.

We can map the starts curve for a process flow into an output curve for the process flow, given assumed values for yields and cycle times for the starts. Such an output curve defines the target output curve

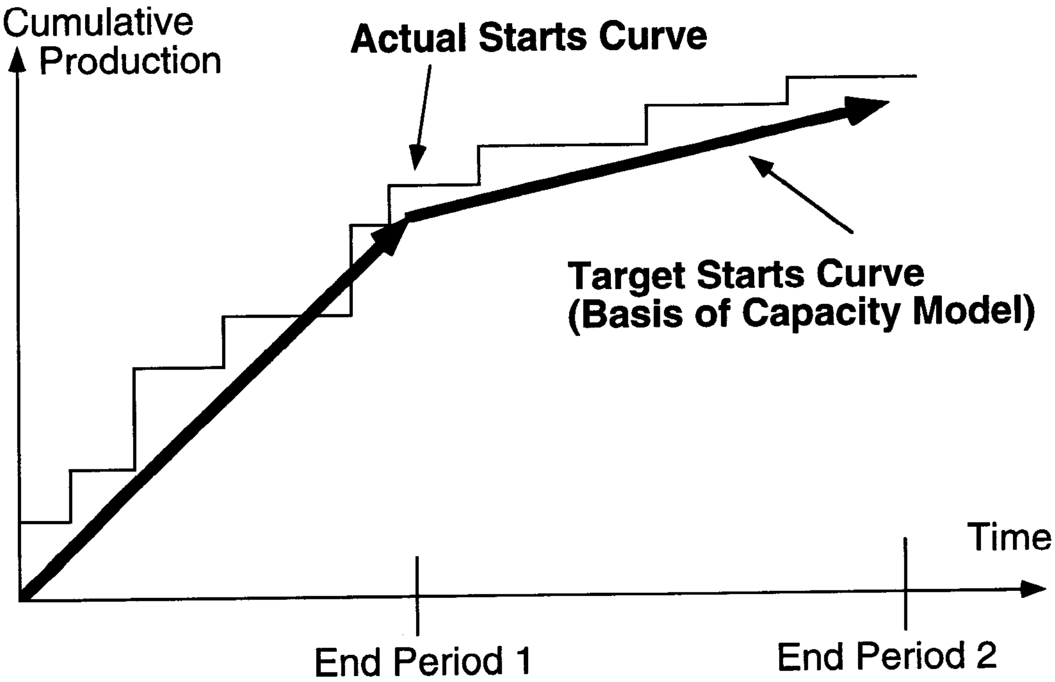


**Figure 6: The scheduled production starts in each planning period are modeled as a constant rate. The target starts curve generated by BPS is thus piecewise linear. The actual starts curve is a stair function with steps corresponding to releases of production lots. BPS assumes the total quantity of wafer starts in each period is distributed uniformly over the period.**

against which we can judge manufacturing's on-time delivery performance. Analogously, we can map the starts curve into a workload curve at each operation in the process flow.

We will use a simple example to explain the BPS model for capacity analysis. A particular machine type performs four different operations in a particular wafer fab process flow. Table 1 shows the quantity, average equipment efficiency, and hours worked per week for the P&E 240 photolithography machines in factory F4. The average equipment efficiency, statistic indicates the percentage of time these machines can be expected to process workloads. The total hours per week the F4-P&E 240 resource can run wafers is

$(7)(120)(0.66) = 554.4.$

Table 2 shows the machine rates for each operation in process flow F4-P411, the average survival yield and the average cycle time from the start of the process flow to the start of the operation. Suppose we schedule wafer starts in continuous time. Let $x(t)$ denote the average rate of wafer starts of process flow F4-P411 at time $t$. If all starts proceed through the process exactly according to the average statistics, then the rate of loading of the machine at time $t$ from starts of process flow F4-P411 is

$$0.0175x(t - 0.368) + 0.0211x(t - 1.330) + 0.0258x(t - 1.744) + 0.0228x(t - 2.290).$$

| Machine Type | Quantity in Service | Hours Worked per Week | Average Equipment Efficiency |
|---|---|---|---|
| P&E 240 | 7 | 120 | 0.66 |

**Table 1: Data is presented for a simple example of modeling capacity consumption by a process flow. The P&E 240 lithography machine is used to perform four steps in process flow P411 in wafer fab F4. The factory floor system at F4 reports statistics concerning the P&E 240 machines.**

Now suppose the planning time periods are each exactly one week in length. Consider the load in week 3 on the F4-P&E 240 machines from performance of operation 9 on process flow F4-P411. The cycle time up to operation 9 is 1.744 weeks; hence starts made between time 0.256 and time 1.256 undergo operation 9 during week 3. By the uniformity assumption, 74.4 percent of the starts in week 1 and 25.6 percent of the starts in week 2 that survive to reach operation 9 will do so during week 3. Now let $x_t$ denote the starts of process flow F4-P411 in week $t$, $t = 1, 2, \ldots$ . Then the load from performance of operation 9 of F4-P411 in week 3 is

$$0.0258[0.744x_1 + 0.256x_2].$$

If we derive similar expressions for the other steps of process flow F4-P411, the total load in week $t$ on the F4-P&E 240 machines from process flow F4-P411 is

$$\begin{aligned}&0.0175[0.632x_t + 0.368x_{t-1}]\\&\quad + 0.0211[0.670x_{t-1} + 0.330x_{t-2}]\\&\quad + 0.0258[0.256x_{t-1} + 0.744x_{t-2}]\\&\quad + 0.0228[0.710x_{t-2} + 0.290x_{t-3}].\end{aligned}$$

Collecting terms, the capacity constraint for machine type F4-P&E 240 takes the following form:

$$\begin{aligned}&0.01106x_t + 0.2718x_{t-1} + 0.04235x_{t-2}\\&\quad + 0.0066x_{t-3} + \text{(similar expressions for}\\&\quad \text{the other F4 process flows using the P\&E}\\&\quad \text{240 machines)} \leq 554.4.\end{aligned}$$

Here, terms with subscripts smaller than one refer to starts made in time periods already past, that is, to WIP. Thus the capacity model can comprehend the competition for capacity between WIP and new starts. The coefficients on starts variables in the

| Process Step ID | Cycle Time up to Step (weeks) | Yield up to Step (percent) | Processing Rate (units per hour) | Resource Load per Wafer Start (machine hours) |
|---|---|---|---|---|
| 4 | 0.368 | 97.98 | 56 | 0.0175 |
| 7 | 1.330 | 95.10 | 45 | 0.0211 |
| 9 | 1.744 | 92.76 | 36 | 0.0258 |
| 12 | 2.290 | 88.95 | 39 | 0.0228 |

**Table 2: The F4 factory floor system also reports statistics concerning the steps of P411. For each step, the resource load per wafer start is calculated as the yield up to the step divided by the processing rate.**

capacity constraints cannot be expressed in closed form but depend on the actual numerical values of cycle times and planning period lengths.

We have illustrated the capacity model for the simplest case in which all time periods have equal length and in which cycle times and yields are constant. The BPS model actually admits time periods of varying length (expressed in terms of working days) and cycle times and yields that vary over time. It uses the cycle times to map the end points of the time periods for planned output and planned operation loads onto the continuous time line for planned starts, and then calculates coefficients on starts variables by observing the fractions of periods that are intercepted, in a way similar to that discussed above. Leachman [1993] provides mathematical formulas for these coefficients.

Staff members in each manufacturing facility at Harris maintain the data required for capacity analysis. These data concern both resource availabilities and operation processing times. For each scarce resource in each planning time period, they identify the quantity in service, average efficiency, and hours worked per working day. The data also includes the factory working calendar. For all steps in all process flows that utilize scarce resources, they identify the resources loaded, the processing rate (units per hour), the cycle time from process start (expressed in fractions of working days), and the yield from process start. They also identify the overall yield and cycle time for each process flow. All parameters are allowed to be time varying. From these data, BPS automatically generates the capacity constraints of the capacitated loading models.

These capacity constraints are but one element of the model. One must append constraints for scarce raw materials, demand constraints, and an objective function. Moreover, the model must reflect the marketing priorities and controls.

In BPS we model the consumption of raw materials to occur at the start of process flows and express such constraints in terms of starts variables. As for demand constraints, we use the same kind of continuous time analysis described for establishing machine capacity constraints to express the linear coefficients on start variables that define the output of process flows in the given planning periods. Given these coefficients, we construct the demand constraints in a straightforward way.

To incorporate the marketing controls, BPS solves a series of linear programming problems, one for each demand class, starting with the highest priority class. The objective function used in problems for each order board and inventory class minimizes total lateness (back-order) costs plus inventory costs. The objective function used in the problems formulated for forecast classes maximizes discounted cash flow, considering discounted revenues from expected product sales less discounted costs for production and inventory.

Each new management team reaffirmed IMPReSS as a key strategic endeavor.

The problem formulated for each class must not undo or trade off the customer service planned for demands in higher priority classes with the customer service to be provided to the class under consideration. To observe this marketing requirement and also to economize on computational effort, BPS formulates each capacitated loading problem using cumulative demands, cumulative over both demand and over higher priority demand classes. BPS places bounds on back-order variables to insure that customer service to higher-priority demands is not disrupted.

The other marketing control on the planning engine is the build-to-level code, which controls the response of the manufacturing lines to sales forecasts unsupported by orders. BPS implements this control simply by enforcing upper bounds on the production starts variables in the first time period for manufacturing stages located beyond the build-to-level point. BPS sets the values of the upper bounds equal to the optimal values of the starts variables in the solution of the formulation for the last order board class. The bounds are enforced in the formulations for all following demand classes.

The only differences in BPS formulations for consecutive classes are (1) right-hand side demands are increased for some products, (2) upper bounds on some variables are changed, and (3) objective function coefficients may be changed. If the values of back-order variables are increased to match the increments in demand, then the optimal solution for the previous class is a feasible starting solution for the current class. Thus solving the series of linear programs for the several classes is actually like solving only one linear programming problem, with pauses after optimizing each demand class to adjust the right-hand side and variable bounds, the values of the back-order variables, and perhaps the objective function coefficients, before reoptimizing the formulation. In practice, the time required to plan four or five demand classes is only about twice the time required if all demands were placed in a single class.

We provide the details of a basic version of the linear programming formulation prepared by BPS for capacitated loading in the appendix.

### Heuristic Decomposition Scheme for the Overall Planning Calculation

Because the planning calculation is so large, we needed some sort of decomposition scheme. Indeed, all large semiconductor companies decompose the overall planning calculation into more tractable pieces. We use various means within BPS:

—We perform separate requirements planning and capacitated loading calculations.

—We perform separate capacitated loading calculations for the front-end and back-end portions of the manufacturing network.

—We perform parallel capacitated loading calculations for different back-end facilities.

All such approaches to decomposition can erode optimality, and so we exercised care in devising a decomposition strategy. We used the following rationale for the heuristic decomposition scheme within BPS:

(1) Only a small number of finished goods types can be produced in more than one back-end site. Without too much loss of optimality, we can preallocate demands for these finished goods among the alternative back-end sites. (Strict optimality would re-

quire simultaneous capacitated loading and requirements planning.)

(2) We can define costs as a reasonably good proxy for differences in the consumption of manufacturing capacity by alternative source product flows feeding class stores. (Since we use discounted costs per start, differences in yields and cycle times are reflected even when costs per start are the same.) Thus we can divorce planning requirements back to die bank from capacitated loading without too much loss of optimality.

(3) The most important bottlenecks in the manufacturing network are in the front end. Capital expense is highest here, and because of the arborescent product structure, a front-end bottleneck generally constrains availability of many more types of finished goods than does a back-end bottleneck. Thus it makes more sense to do front-end capacitated loading before performing back-end capacitated loading. (A back-end capacitated loading calculation performed beforehand would need to be redone to reflect the limited die supply generated by front-end capacitated loading.)

(4) Because the number of wafer types is small compared to the number of back-end products, a worldwide capacitated loading calculation including all front-end sites in a manufacturing network is practical.

(5) If we preallocate demands producible in multiple back-end sites as in (1), then we can carry out separate capacitated loading calculations for each back-end site in parallel, provided we allocate die supply calculated by the front-end capacitated loading calculation among the back-end sites. Capacitated loading calculations for large back-end sites would require very large linear programs, but much smaller than an LP for calculating the entire planning problem at once.

The heuristic decomposition strategy we used relies on a series of five planning modules to perform the overall planning calculation (Figure 7). Module 1 performs MRP calculations to determine worldwide net requirements for final test starts for each demand class, netting out finished goods inventory and final test WIP. Next, Module 2 performs a mixture of linear programming and MRP calculations to determine net requirements for new shipments to assembly areas from die bank. Then, Module 3 performs capacitated loading of front-end sites, including the allocation of planned die output to the various back-end sites. Subsequently, Module 4 performs capacitated loading of back-end sites, constrained by resource capacities, by raw materials availability, and by planned die availability calculated by Module 3. IMPReSS performs a separate Module 4 calculation for each back-end site. Finally, Module 5 collects the outputs of the various Module 4 calculations and nets out all demand classes that are not forecasts to calculate the availability schedules.

We omit details here, but the decomposition scheme requires Modules 1 and 2 to compute the average revenue per unit start of each source product planned to service the cumulative demands for each class. In this way, it translates average selling prices for finished goods into prices for final test starts and ultimately into average prices for die types. Such revenues for source products serve as the price parameters used in Modules 2 and 3.

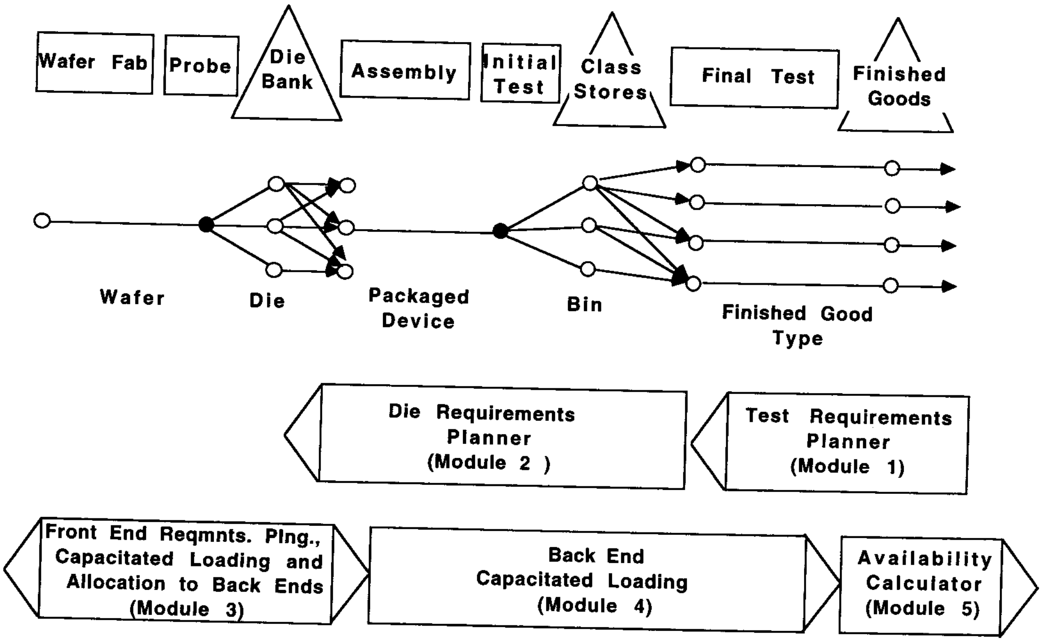

Wafer Fab
Probe
Die Bank
Assembly
Initial Test
Class Stores
Final Test
Finished Goods
Wafer
Die
Packaged Device
Bin
Finished Good Type
Die Requirements Planner (Module 2 )
Test Requirements Planner (Module 1)
Front End Reqmnts. Plng., Capacitated Loading and Allocation to Back Ends (Module 3)
Back End Capacitated Loading (Module 4)
Availability Calculator (Module 5)

Figure 7: The Berkeley Planning System consists of five planning modules that decompose the overall planning problem into tractable pieces. Modules pointing to the left in the figure perform requirements-planning calculations, while modules pointing to the right perform capacitated-loading and availability calculations.

The decomposition scheme requires the derivation for each demand class of net requirements for die shipments to each back-end site, serving as the demand inputs to Module 3. BPS uses a continuous-time analysis to convert net requirements for packaged device output into net requirements for device starts, from which it determines net die requirements using the requirements planning techniques described earlier. To translate required output into starts, BPS departs from conventional MRP methodology to perform a continuous-time analysis using noninteger manufacturing lead times. It establishes requirements schedules that insure each factory can meet demands on time if its production quantity in each of the given planning periods is distributed as a constant rate over the period (Figure 8). Moreover, it avoids the excess inventory that would result from a conventional MRP planning calculation incorporating rounded-up integer lead times. Leachman and Goncalves [1989] describes the algorithm that derives the starts curve from the output curve.

We originally formulated the allocation of die to back-end sites that occurs within Module 3 as a separate optimization model, but subsequently we incorporated it into the capacitated loading formulation. We indexed die demands tendered to the formulation by back-end site, and incorporated the binning and substitution product structures at die bank into the formulation.

Our decomposition scheme effectively breaks the planning problem into tractable pieces. The dimensions of the various cal-

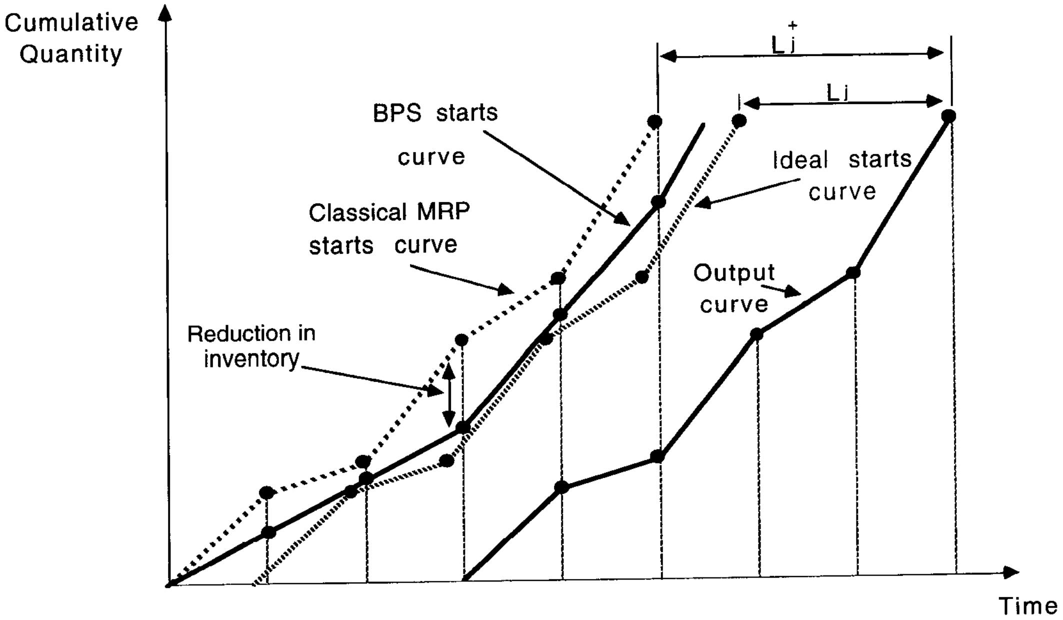


**Figure 8: BPS performs rate-based requirements planning. It uses a noninteger lead time corresponding to the process flow cycle time to translate the output requirements curve into a starts requirements curve. The ideal starts curve is the direct translation of the output curve by the noninteger lead time, but it is not rate based in the given planning periods. The BPS starts curve is a rate-based schedule that lies below the usual MRP curve computed using rounded-up integer lead times.**

culations in the modules work out as follows:

—Module 2 contains many relatively small linear programming (LP) calculations; there are no finished goods families with LP's larger than 2,000 rows. Harris used a single workstation computer to perform a companywide calculation (later changed to two computers in parallel processing different product families).

—Module 3 solves the front-end capacitated loading problems; two manufacturing networks are optimized in parallel on two computers. The largest LP has about 160,000 rows.

—Module 4 solves capacitated loading formulations for five back-end sites on separate computers in parallel; the LP for the largest back-end site includes about 160,000 rows.

—Modules 1 and 5 do not include LP calculations.

We programmed the various modules at UC Berkeley in FORTRAN and C Shell scripts to run in a UNIX environment. In all, BPS includes more than 50,000 lines of code.

### IMPReSS Implementation

The most difficult aspect of implementation was converting data used in planning to conform with a standard data model. Reflecting the decentralized planning and control in the three predecessor companies, various planners and managers defined the boundaries of process flows and the inventory points of the product structure in different ways. Many systems and planning staff members understood semiconductor manufacturing in terms of those data structures, however approximate or inefficient. BPS requires a standardized, specific data structure. Converting to this data structure caused conflicts with long-held intuitions, conventions, and data structures extant in factory floor systems. The change required seemed unreasonable to many. If sector executives had not expressed urgency to move the project forward, this issue might have stopped the project entirely.

As part of the IMPReSS project, Harris installed a commercial demand forecasting software, defined and populated a BOM database, and upgraded the existing order entry system to provide on-line delivery quotation and reservation capability. The IMPReSS project team developed relational data bases at all manufacturing sites as well as at sector headquarters to manage the massive amount of data needed for company-wide planning, including interfaces to all relevant factory floor and marketing systems. These interfaces mapped factory data from the inherited structures into the standardized data structures BPS required. The team developed software to transfer data between factory databases and the central database, to download data to BPS from the central database and to upload results, and to transfer output data from one module run to serve as input data for the next. The team also developed user interfaces for the factory and central databases.

We installed the completed planning engine software at Harris in October 1991. The team completed the IMPReSS central database and the factory databases at about the same time. As we integrated data and fed it to the planning engine for trial runs, we found many gaps and inconsistencies in factory, marketing, and BOM data, making it difficult or impossible to carry out a planning cycle. We had to devise and program many database checks to reveal data errors and to reduce the overall data set down to a consistent and complete subset that could be fed as input to the planning engine. In the fall of 1991, typically less than half of the Harris data set passed such checks. To the IMPReSS team, this poor data quality came as a shock. Clearly, Harris could not switch over to the IMPReSS automated planning within the one-year schedule established at the start of the project. Morale sunk and management frustration increased as sector losses continued to mount despite a tremendous year-long effort.

After the start of the IMPReSS project, the sector experienced two rounds of turnovers in key executive positions, including the sector president. Despite the lateness of the project, each new management team reaffirmed IMPReSS as a key strategic endeavor of the sector, and the project continued to move forward.

Over the last quarter of 1991 and the first two quarters of 1992, Harris greatly improved data quality and completeness. The team assigned every piece of data required for system operation an owner who was responsible for correcting the inconsistencies and gaps found by the automated database checks. With larger data sets now passing the checks, we could test the application software more fully. We found and eliminated bugs in the planning engine and database software. We transferred the final version of BPS, incorporating the final corrections, from the university to Harris in May 1992. In late 1992, Harris had improved planning data to the point that about 75 percent of the sector's products could be planned in automated cycles; this figure exceeded 90 percent a year later and 97 percent a year after that. By the end of 1992, the sector's manufacturing facilities generally accepted IMPReSS production plans, and Harris made delivery quotations according to the system's calculations in lieu of manual means.

IMPReSS generates an official production plan each weekend that revises the availability used in the quotation system and provides official production schedules for all manufacturing facilities. A cutoff for data maintenance Saturday morning at midnight Eastern time triggers the start of a planning cycle; all manufacturing sites in the US and the Far East transfer factory status and capacity data over the Internet to sector headquarters. The planning cycle is targeted to be completed by early Sunday afternoon, in time to provide production schedules to plants located in the Far East Monday morning.

Over two years, Harris missed the Sunday completion target only once, because network communications failed, blocking the transfer of data into headquarters from factories in the Far East. In addition to the official weekend planning run, Harris typically carries out one or two more planning cycles during the week for off-line analysis or perhaps in response to an urgent need to replan.

Using IBM's Optimization Subroutine Library (OSL) software for linear programming to solve the BPS formulations on a battery of five Model 560 RS-6000 IBM workstations, Harris completed the typical full-scale weekly planning calculation for a one year planning horizon during 1994 in under 17 hours, including about five hours for Module 2, six hours for Module 3, five hours for Module 4, and 0.25 hours each for Modules 1 and 5. For the very large-scale capacitated loading calculations, we used interior point algorithms to solve the LPs. A reprogrammed version of the planning engine released in March 1995 that exploits more parallel processing has reduced the planning cycle time to about nine hours.

**Costs and Results**

Tallying the costs of the IMPReSS project, one-time costs totaled about $3.8 million, *including $0.7 million for software,* $1.5 million for computer hardware, $1.4 million for consulting, and $0.2 million for project travel by Harris staff. Annual recurring costs amount to about $0.6 million, including software maintenance and the addition of five staff members to a central planning organization.

From the beginning of 1993 through early 1995, Harris semiconductor sector maintained a 94 to 95 percent on-time delivery performance, one of the best scores in the industry for a high-volume, broad-mix manufacturer. This represents a dramatic turnaround from pre-IMPReSS days. The number of delinquent order line items fell from 5,000 in 1990 to less than 100 by late 1993, a level continuing to the present. Harris achieved these improvements while holding sector inventories constant as a percentage of sales.

Over time, customers became convinced of the reality of this remarkable improvement; basically, Harris has transformed its image from one of the worst in the industry for on-time delivery to perhaps the best. Customer surveys now indicate that 85 percent of the sector's customers would recommend Harris as a supplier to other semiconductor customers, and 100 percent intend to continue buying from the sector. The sector has made major sales to new customers, such as Fujitsu in Japan, who before IMPReSS implementation would not consider Harris as a vendor because of unreliable delivery. Over the past couple of years, sector sales executives estimate that Harris has essentially lost no sales because of noncompetitive delivery performance, in marked contrast to the $100 million it lost annually during 1989–91. Sector sales have risen from a low point of $530 million in fiscal year 1992 to almost $700 million for fiscal year 1995, despite a major decline in defense-business sales. And most important, the sector arrested its financial losses and replaced them with growing profits: $20 million in fiscal 1993, $30 million in fiscal 1994, and $42 million in fiscal 1995. IMPReSS has clearly played a key role in turning around the fortunes of the sector.

Harris also uses IMPReSS for capital equipment planning. The ability to integrate capacity requirements across all factories in IMPReSS-generated plans has improved the effectiveness of capital investments. Sector executives estimate that savings in capital expenses during 1993 and 1994 afforded by the use of IMPReSS more than recovered the IMPReSS project costs.

## Acknowledgments

Countless current and former Harris employees contributed to the success of the IMPReSS project. While it is impossible to acknowledge everyone, it seems appropriate to acknowledge the contributions of key staff members of the sector manufacturing systems department: Randy Burdick (formerly director of sector manufacturing systems), Ted Dimbero (current director), Steve Traines (formerly manager of planning systems), Jim Witte (current manager), Jennifer Bienvenue, Tom Gillespie, Sean McNunn, Sam Murty, Chris Paluszek, and Clay Siefert.

We carried out the BPS research effort from 1984 to the present supported by gifts and grants to the University of California at Berkeley from Harris Corporation Semiconductor Sector; Samsung Electronics Co., Ltd.; Advanced Micro Devices, Inc.; Intel Corporation; Digital Equipment Corporation; SGS Thomson Microelectronics; Cypress Semiconductor, Inc.; Semiconductor Research Corporation; and the State of California MICRO program. We have made the BPS software available free of charge for unlimited internal use by all major sponsors of the research.

Leachman [1993] describes the BPS modeling techniques in detail, and Leachman and Raar [1994] describe its implementation as IMPReSS. Hackman and Leachman [1989] describe rate-based modeling used in BPS, and Leachman and Carmon [1992] describe additional characteristics of capacity modeling used in BPS.

Over the years, many graduates of UC Berkeley have made significant contributions to this effort; portions of the model-

ing logic were developed in the doctoral dissertations of Goncalves [1987], Hung [1991], and Liu [1992]. Other students programming BPS since 1984 include Robert Benson, Michael Mizrach, Renato Monteiro, and Willie Weng.

Many UC Berkeley graduates also contributed to developing the IMPReSS capacity databases at Harris. Student interns at various Harris manufacturing sites during the IMPReSS project included Sean Cunningham, Yi-Feng Hung, Kirsten Kinne, Dennis Kung, Paul Nilsson, Phyllis Shabe, Chia-Li Wang, Dawn Wang, and Willie Weng. UC Berkeley students Dale Raar and Charles Swenberg programmed the database interfaces for the first BPS installation at Harris.

All of us from Berkeley owe a debt of thanks to Harris for an unforgettable industrial adventure.

## APPENDIX: Linear Programming Formulations in BPS

### Requirements Planning of Binning Products

We formulate the case of a single source product generated by a single process flow. Associated with the source product, we allow an arbitrary number of bins serving an arbitrary number of finished goods according to a given accept bin table. For simplicity of exposition, we first formulate the case where all demands for which net requirements are to be computed belong to only one demand class, labelled class $r$. We define the parameters of the formulation as follows:

$w$ = the last time period in which projected WIP-outs of bins of the source product will enter class stores, and the first possible period of bin supply at class stores from planned starts of the source product. (We allow this overlap.)

$a_i$ = the bin split to bin type $i$, that is, the expected fraction of completed source product ending up in bin type $i$, defined for all bin types $i$ of the source product.

$d^r_{jt}$ = the demand in class $r$ for finished goods type $j$ in period $t$, $t = 1, 2, \dots, T$.

$x_t$ = the projected WIP-out of source product in period $t$, $t = 1, 2, \dots, w$.

$p^r_{jt}$ = the discounted unit revenue for demands in class $r$ of finished goods type $j$ in period.

$t, t = 1, 2, \dots, T$.

$c_t$ = the discounted unit production cost of the source product.

$h_{it}$ = the holding cost per unit of bin type $i$ in class stores inventory at the end of period $t$,

$t = 1, 2, \dots, T$.

We define the variables of the requirements planning formulation as follows. (The superscript $r$ on each variable designates that the variable belongs to the formulation to plan requirements for demands in class $r$.)

$X^r_t$ = the quantity of source product to be completed in planning period $t$,

$t = w, w + 1, \dots, T$.

$Y^r_{ijt}$ = the allocation of bin type $i$ to finished goods type $j$ in time period $t$, defined for each bin type-finished goods type combination appearing in the accept bin table,

$t = 1, 2, \dots, T$.

$I^r_{it}$ = the inventory of leftover bins of type $i$ at the end of period $t$, $t = 1, 2, \dots, T$.

$BO^r_{jt}$ = the amount of back orders of finished goods type $j$ at the end of period $t$,

$t = 1, 2, \dots, T$.

We use the notation $i \in j$ to denote that bin type $i$ is an accept bin for finished goods type $j$. We also use the notation $j \in i$ to denote that finished goods type $j$ is a possible use of bin type $i$. We then define the objective function and constraints of the basic requirements planning formulation as follows:

Maximize

$$\sum_{j} \sum_{i\in j} \sum_{t} p^r_{jt} Y^r_{ijt} - \sum_{t} c_t X^r_t - \sum_{i} \sum_{t} h_{it} I^r_{it}.$$

Subject to

(1) $I^r_{i,t-1} + a_i(x_t + X^r_t) - \sum_{j\in i} Y^r_{ijt}$

$= I^r_{i,t}$, all $i$, all $t$.

(2) If demands are in order board or inventory classes:

$\sum_{i\in j} Y^r_{ijt} - BO^r_{j,t-1} + BO^r_{j,t}$

$= d^r_{jt}$, all $j$, $t = 1, 2, \ldots, w - 1$.

$\sum_{i\in j} Y^r_{ijt} - BO^r_{j,t-1} = d^r_{jt}$, all $j$, $t = w$.

$\sum_{i\in j} Y^r_{ijt} = d^r_{jt}$, all $j$, $t = w + 1, \ldots, T$.

(2) If demands are in forecast classes:

$\sum_{i\in j} Y^r_{ijt} - BO^r_{j,t-1} + BO^r_{j,t} = d^r_{jt}$, all $j$, all $t$.

(3) $X^r_t \geq 0$, $Y^r_{ijt} \geq 0$, $BO^r_{j,t} \geq 0$, $I^r_{i,t} \geq 0$,

all $i$, $j$, $t$.

The objective of the formulation measures discounted cash flows for sales revenues, production costs, and inventory holding costs. The coefficient on an allocation variable in the objective function is the discounted unit revenue associated with the finished goods type receiving the allocation. There are no costs on back-order variables in the formulation. The objective is thus one of maximizing discounted cash flow, including discounted revenue from allocation to demands, less the discounted costs of production of source product and the inventory holding costs for bin inventories. Constraints (1) enforce inventory balance of each bin in each time period. Constraints (2) measure the demand satisfaction of each finished goods type. The particular form of the these constraints also depends on the type of demand class involved, as discussed in the text of the article. Constraints (3) are the usual nonnegativity constraints.

To extend the formulation to the case of multiple demand classes, we formulate the requirements planning problem for all demand classes as a single linear program, including separate variables and constraints indexed by demand class, and including the objective functions for each class combined into a single objective. In such a consolidated formulation, we define the demand quantities in the constraints for each class to be cumulative over all higher-priority classes and the variables for each class in each time period to represent the total production, allocation, inventory, and back-order levels serving the indexed class as well as all higher-priority classes. In addition to the constraints discussed above, we add constraints expressing consistency between cumulative requirements calculated for consecutive classes to the formulation, as follows:

$$\sum_{\tau=1}^{t} X^r_\tau \geq \sum_{\tau=1}^{t} X^{r-1}_\tau, \quad r = 2, 3, \ldots, R, \text{ all } t.$$

These constraints state that at the end of each time period, the cumulative production of each source product to meet the demands in classes $1, 2, \ldots, r$ must be greater than or equal to the cumulative production of each source product to meet the demands in classes $1, 2, \ldots, r - 1$. The result of solving such a formulation defines the production requirements servicing demands in classes $1, 2, \ldots, r$, where $r$ ranges up to the total number of demand classes $R$.

**Capacitated Loading of Prioritized Demands**

We illustrate the basics of the proposed capacitated loading procedure for a single-stage system, that is, assuming there is only one process flow between raw materials start and finished goods. Suppose we are given demand classes 1, 2, . . . , $R$. We define $R$ linear programming models to load demands in classes 1, 2, . . . , $r$, $r = 1, 2, \ldots, R$, respectively. The basic strategy is to solve the loading models in numerical order. Model 1 loads demands in class 1 as on-time as possible. Model 2 then loads demands in both classes 1 and 2 as on-time as possible. We constrain the solution of loading model 2 to support demands in class 1 at least as much as does the solution of model 1. This process of incremental loading is continued until an overall production plan is specified by the optimal solution to model $R$.

For simplicity of exposition, we assume products and process flows are one-to-one. We introduce the following variables for the $r$th formulation:

$x^r_{it}$ = starts of product $i$ (process flow $i$) planned from loading demands in classes 1, 2, . . . , $r$ in the period ending at time $t$, defined for all $i$ and for $t = 1, 2, \ldots, T$.

$I^r_{it}$ = inventory of completed product $i$ at time $t$, relative to demands in classes 1, 2, . . . , $r$, defined for all $i$ and for $t = 1, 2, \ldots, T - 1$.

$BO^r_{it}$ = back orders of product $i$ at time $t$, relative to demands in classes 1, 2, . . . , $r$, defined for all $i$ and for $t = 1, 2, \ldots, T$.

We also introduce the following shorthand notation for various linear combinations of the process starts variables that arise when we map process outs and operation loads to process starts:

$\widehat{x^r_{it}}$ = output of product $i$ (process flow $i$) planned from loading demands in classes 1, 2, . . . , $r$ in period ending at time $t$.

$\widetilde{x^r_{ijt}}$ = amount processed through operation $j$ of process flow $i$ planned from loading demands in classes 1, 2, . . . , $r$ in period ending at time $t$.

Leachman [1993] provides formulas for these linear combinations as a function of given cycle time, yield, and factory calendar data. These formulas constitute the *dynamic production functions* modeling semiconductor processing flows.

We introduce the following parameters for the $r$th formulation:

$D^r_{it}$ = cumulative demand for product $i$ (process flow $i$) at time $t$, cumulative both over classes 1, 2, . . . , $r$ and over time, defined for all $i$ and for $t = 1, 2, \ldots, T$. $d^r_{it}$ denotes the demand in classes 1, 2, . . . , $r$ for product $i$ in period $t$.

$BO^{r-1}_{it}$ = optimal value of the back-order variable for product $i$ at time $t$ in the linear program for demands in classes 1, 2, . . . , $r - 1$.

$\overline{BO^r_{it}}$ = upper bound on the back orders of product $i$ at time $t$, relative to demands in classes 1, 2, . . . , $r$. We formulate the requirement that service to demands in classes 1, 2, . . . , $r - 1$ must not be diminished when solving loading model $r$ as the condition that

$$BO^r_{i,t} \leq \overline{BO^r_{it}} = BO^{r-1}_{i,t} + D^r_{it} - D^{r-1}_{it}.$$

This constraint expresses that fact that back orders can rise by no more than the increment in demand in the current class; otherwise, higher-priority demands are being back ordered more than necessary. Note that this constraint does not require a row in the formulation matrix for loading model $r$; instead, it is invoked by placing the simple upper bound $\overline{BO^r_{it}}$ on the back-order variable $BO^r_{i,t}$.

$C_{kt}$ = capacity of resource $k$ in period $t$, expressed in units of resource hours.

$a_{ijkt}$ = hours of resource type $k$ required to perform operation $j$ of process flow $i$ in time period $t$, per unit of process flow $i$.

$p^r_{it}$ = discounted price for sales in period $t$ of product $i$ in demand class $r$.

$c_{it}$ = discounted unit cost for starts of product $i$ in period $t$.

$h_{it}$ = holding cost for inventory of completed product $i$ at the end of period $t$, including the difference between discounted revenues in periods $t$ and $t + 1$.

$b^r_{it}$ = cost per unit back ordered of product $i$ in period $t$, based on $p^r_{it}$.

Each of the $R$ loading models seeks to maximize discounted cash flows subject to demand and capacity constraints. Capacitated loading model $r$ is formulated as follows:

Maximize

$$\sum_i \sum_t p^r_{it}\widehat{x^r_{it}} - c_{it}x^r_{it} - h_{it}I^r_{it} - b^r_{it}BO^r_{it}.$$

Subject to

$$(1)\ \sum_i \sum_j a_{ijkt}\widetilde{x^r_{ijt}} \leq C_{kt}, \quad \text{all } k, \text{ all } t,$$

$$(2)\ \widehat{x^r_{it}} + I^r_{i,t-1} - BO^r_{i,t-1} - I^r_{i,t} + BO^r_{i,t} = d^r_{it},$$

all $i$, all $t$,

$$(3)\ BO^r_{i,t} \leq \overline{BO^r_{i,t}}$$

$$= BO^{r-1}_{i,t} + D^r_{it} - D^{r-1}_{it}, \quad \text{all } i, \text{ all } t,$$

$$(4)\ x^r_{it} \geq 0, \quad BO^r_{i,t} \geq 0, \quad I^r_{i,t} \geq 0, \quad \text{all } i, \text{ all } t.$$

The objective function of the formulation maximizes cash flows from sales, manufacturing and inventory. Back-order costs are nonzero only in formulations for order board and inventory classes of demands, while sales prices and unit costs for production starts are nonzero only for formulations of forecast classes. In the objective of formulations for forecast classes, discounted sales revenue is computed as discounted price times production output; however, output left in inventory at the end of the period is assessed an inventory holding cost that includes the loss in discounted revenue from the current period to the next. In all formulations, inventory variables are not defined in period $T$; that is, all production output is sold eventually. In this way, all sales are assigned the correct discounted revenue.

Constraints (1) express the resource capacity limits. Constraints (2) measure the inventory and back-order positions of each product relative to its demands. Constraints (3) express upper bounds on the back-order variables, insuring that service provided to higher priority demand classes, as determined in the solution to the previous formulation, is maintained in the solution to the current formulation. Constraints (4) express the usual nonnegativity conditions on variables.

An additional constraint must be added to the formulation requiring the production system to enter a steady state at the horizon. The usual implicit ending condition in planning models is that no more production starts will be made after time $T$. For the multi-period process flows considered here, variables representing process starts near the horizon are relatively less constrained under such an ending condition. (If indeed no production starts will be made after time $T$, a large batch may be started in the last time period, since such starts do not have to compete with any future starts for capacity. Thus, the usual ending condition leads to unreasonable results in an optimal solution.)

This steady-state horizon condition is enforced as follows: first, process starts in all time periods falling within a cycle time of the horizon are constrained to be at the same rate; this is accomplished simply by using the same variable to represent them, appropriately scaled for differences in the lengths of time periods. To insure that these starts are fully constrained, an extra time period $T + 1$, at least as long as the longest process cycle time, is added to the formulation, and constraint types (2) and (3) are enforced for this extra period. Demands in the extra period are set to be at the same rate as in period $T$ by prorating the period $T$ demands according to the ratio of period lengths. Inventory variables are excluded for both period $T$ and period $T + 1$, insuring there will be no overproduction.

It is straightforward to extend the formulation to include material capacity constraints and to accommodate alternative process flows for the same product. The formulation can also be readily extended to the multistage case, but with the following concern about constraints expressing inventory balance between consecutive stages modeled using process starts variables: In the case that process cycle times are fractional relative to the planning time grid (which is usually the case), inventory balance equations need to be enforced at fractional points of time in addition to the usual grid points [Hackman and Leachman 1989]. The formulation can also be modified to handle the case that interstage inventory balance involves binning and product substitution. This is accomplished by integrating the formulation structure discussed in the first section of the appendix with the structure presented in this section.

**References**

Goncalves, J. F. 1987, "Continuous time analysis of production planning and scheduling models," PhD diss., College of Engineering, University of California at Berkeley.

Hackman, S. T. and Leachman, R. C. 1989, "A general framework for modeling production," *Management Science*, Vol. 35, No. 4 (April), pp. 478–495.

Hung, Y. F. 1991, "Corporate-level production planning with simulation feedback of parameters," PhD diss., College of Engineering, University of California at Berkeley.

Leachman, R. C. 1993, "Modeling techniques for automated production planning in the semiconductor industry," in *Optimization in Industry*, eds. T. A. Ciriani and R. C. Leachman, John Wiley and Sons, Chichester, UK, pp. 1–30.

Leachman, R. C. and Carmon, T. F. 1992, "On capacity modeling for production planning with alternative machine types," *IIE Transactions*, Vol. 24, No. 4 (September), pp. 62–72.

Leachman, R. C. and Goncalves, J. F. 1989, "Rate-based materials requirements planning with noninteger lead times," Report ESRC 89-4, Engineering Systems Research Center, University of California at Berkeley (February).

Leachman, R. C. and Raar, D. J. 1994, "Optimized production planning and delivery quotation for the semiconductor industry," in *Optimization in Industry 2*, eds. T. A. Ciriani and R. C. Leachman, John Wiley and Sons, Chichester, UK, pp. 63–72.

Liu, C. 1992, "A modular production planning system for semiconductor manufacturing," D. Eng. diss., College of Engineering, University of California at Berkeley.

In Harris's presentation at the Edelman Award competition, Phil Farmer, President and Chief Operating Officer of Harris Corporation, stated "The biggest problem we had in the semiconductor sector right after the GE merger was on-time delivery. Our on-time delivery was running about 75 percent, which was not acceptable. Implementation of IMPReSS raised our on-time delivery to 95 percent, so from that point of view the investment was very worthwhile. IMPReSS also allows us to plan our capital investments more wisely, and the savings in capital investment alone has exceeded the cost of implementing IMPReSS."