Interfaces
Vol. 43, No. 2, March–April 2013, pp. 130–141
ISSN 0092-2102 (print) | ISSN 1526-551X (online)

http://dx.doi.org/10.1287/inte.1120.0642

IBM Blends Heuristics and Optimization to Plan Its Semiconductor Supply Chain

Alfred Degbotse
IBM Corporation, Essex Junction, Vermont 05452, adegbo@us.ibm.com

Brian T. Denton
Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, Michigan 48109, btdenton@umich.edu

Kenneth Fordyce
IBM Corporation, Hurley, New York 12443, fordyce@us.ibm.com

R. John Milne
School of Business, Clarkson University, Potsdam, New York 13699, jmilne@clarkson.edu

Robert Orzell
IBM Corporation, Essex Junction, Vermont 05452, rorzell@us.ibm.com

Chi-Tai Wang
Institute of Industrial Management, National Central University, Jhongli City 32001, Taiwan, Republic of China, ctwang@mgt.ncu.edu.tw

IBM uses operations research techniques to plan its enterprise semiconductor supply chain. The scale and complexity of this planning problem make developing robust supply chain optimization tools a challenge. Pure optimization methods are computationally infeasible, and fast heuristic methods alone generate poor results. Consequently, we developed a method that decomposes the problem by dividing the bills of materials product structure horizontally and vertically into complex and simple portions that are based on the major stages in semiconductor manufacturing and the choices of supply chain paths for building parts. The method then solves the complex portions with a mixed-integer program and the simple portions with fast heuristics that contain small embedded linear programs. A unique pegging algorithm, an explosion heuristic, and an implosion linear program enable coordination among these portions. The result is a unified production, shipping, and distribution plan with no evidence of the original decomposition. This method has helped IBM to improve its asset utilization, customer service, and inventory levels.

Key words: industries: computer, electronic; information systems: management, decision support systems; production, scheduling: planning; programming: linear, integer.
History: This paper was refereed. Published online in *Articles in Advance* October 25, 2012.

IBM has been in the semiconductor business since 1957. It has manufacturing and contract manufacturing facilities in Asia and North America (and until recently in Europe); these facilities make products that range from silicon wafers to complex modules and consist of multiple semiconductor devices (sometimes called dies). Until the 1990s, IBM facilities operated primarily based on their geographic ties. For example, a European facility would supply component parts to local assembly plants in Europe. These regional supply chains were planned independently because a planning system capable of enterprise supply chain optimization did not exist.

To coordinate planning across the extended supply chain, we developed the central planning engine (CPE). Its purpose is to determine a production and shipment plan for the enterprise by using limited material inventories and capacity availability to satisfy a prioritized demand statement. A pure optimization approach would have resulted in an unacceptably long run time—if it would have even been able to solve a problem of this scale; however, a pure heuristic approach—although fast—often makes poor choices when many possible supply chain paths are available to satisfy demand. Consequently, in developing the CPE, we blended the best of two decision

technologies to achieve near-optimal results within a reasonable run time: we use a mixed-integer program (MIP) to determine plans for those portions of the supply chain that involve alternatives (choices) of supply chain paths; we use advanced heuristic methods to determine plans for other portions of the supply chain in which possible paths involve either no choices or straightforward choices, which we determine according to predefined rules (e.g., source 70 percent of this particular part's supply from one particular vendor and 30 percent from another). We refer to these as complex and simple portions, respectively.

Built upon IBM's earlier accomplishments in supply chain optimization (Lyon et al. 2001, Denton et al. 2006), the CPE models enterprise supply chains at a part number (PN), order number, and lot number level of detail. From the time a PN's job is released, the manufacturing line is treated as a black box from which the job is assumed to emerge a lead time later at the completion of that PN. Because the CPE is used for enterprise runs, we do not need to model the operational details of how a PN is built. We express capacity consumption rates as units of capacity of a resource (e.g., total machine hours for a group of similar machines) consumed per piece released or started for a PN. The capacity consumption may be offset by a user-specified time duration subsequent to manufacturing release or start. Jobs are not planned to be released into the line unless the resulting capacity consumed by all jobs does not exceed the capacity available in each period for all resources consumed by the part. When capacity is not available, the job releases are delayed, which results in an implicit extension of the user-specified planned lead time.

In modeling capacity, the CPE's heuristic method uses an unlimited number of one-day periods (buckets). The MIP uses from approximately 40 to 90 periods, depending on the purpose of the run and the available time window to complete the run. For a tactical run, MIP periods might consist of daily buckets for several weeks, followed by two-day, four-day, and weekly buckets totaling a one-year horizon. For a strategic run covering multiple years, MIP periods might begin with weekly buckets, followed by two-week, monthly, and quarterly buckets.

The CPE decomposes the problem by dividing the bills of materials (BOM) product structure in accordance with the major stages in semiconductor manufacturing; it further divides each stage into a complex and a simple portion based on the choices of supply chain paths for building a part. The CPE then uses an MIP to solve the complex portions and fast heuristics to solve the simple portions. Because the decomposition represents a relaxation of the overall problem, subproblems are solved in processing demand from the top of the supply chain to the bottom to create a tentative production and material plan at all levels of the supply chain necessary to satisfy all demand on time, and again in processing this production and material plan from the bottom of the supply chain to the top to create a feasible enterprise supply chain plan.

IBM uses the CPE, which is robust and versatile and meets rigorous business requirements, for its strategic and tactical planning activities. It has become a critical part of daily, monthly, and yearly planning at IBM.

Semiconductor Manufacturing Background

Semiconductor manufacturing is performed in facilities that require billions of dollars in capital investment. The manufacturing process has four major stages: wafer, device, module, and card. In wafer fabrication, four manufacturing steps—deposition, photolithography, etching, and ion implantation—are repeated as many times as the design requires. Through these steps, a set of three-dimensional layered circuit structures (called integrated circuits or devices) is built on the surface of the wafer (Monch et al. 2011). (Our modeling treats the repetitions of these four manufacturing steps as a single black box because they are all completed within the specified lead time of a single PN.) Next, finished circuits are tested and tagged (to be referenced later in the manufacturing process), wafers are diced, and the outputs are categorized as individual devices. The devices are then bonded to a substrate and packaged to make modules, which are subsequently tested and assembled to make cards. Depending on the customer, semiconductor firms may receive orders for wafers, devices, modules, or cards.

From start to finish, hundreds of operations are necessary to make semiconductor parts. The manufacturing lead times are long; wafer fabrication takes several months to complete. Approximately one-third to one-sixth of the lead time is raw processing time; the remainder is queue time as jobs wait for resources (typically machines) to become available. The long wait times result from the expense of the machines and the variability in arrival and service times. The overall manufacturing process has complicating characteristics such as binning, substitution, and alternate BOM/plant locations. Binning refers to testing and categorizing PNs as fast, medium, or slow.

To illustrate, Figure 1 shows that each untested device UD11 will be characterized as device category D11 (fast) 25 percent of the time, as device category D12 (medium) 55 percent of the time, and as device category D13 (slow) 20 percent of the time. We could also describe this situation as: UD11 bins to D11, D12, and D13 with proportions 25 percent, 55 percent, and 20 percent, respectively. These proportions result from statistical fluctuations in the semiconductor manufacturing process. Substitution, which is often associated with binning, refers to using alternate PNs to finish a manufacturing step. In Figure 1, if a shortage occurs for the medium-speed device D12, the fast-speed device D11 can be used to continue the manufacturing process in place of D12. The term, alternate BOM/plant locations, means that the PN (e.g., module M25 in Figure 1) can be manufactured by multiple processes at multiple sites.

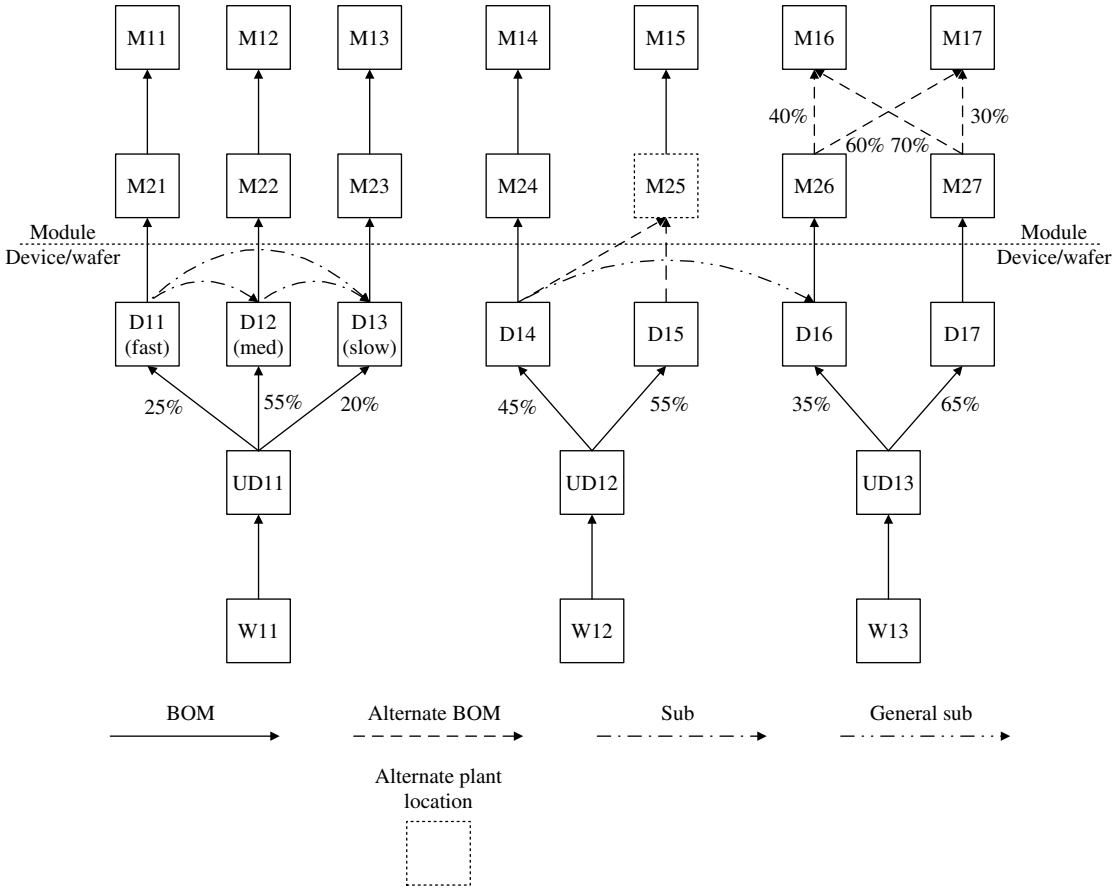

Figure 1: In semiconductor manufacturing, material flows from the bottom of the BOM supply chain (e.g., W11) to the top of the supply chain (e.g., M11). Complexities, such as alternative BOMs, alternative plant locations, binning, and substitutions, result in alternative supply paths for meeting the demand.

The combination of binning, substitution, and alternate BOM/plant locations creates challenging scenarios for semiconductor supply chain optimization. Sequential binning (i.e., one binning activity is followed by another) is one of these challenges. In Figure 1, untested device UD13 bins to create devices D16 and D17, followed by modules M26 and M27 binning to create modules M16 and M17. General substitution refers to substitutions permitted for PNs resulting from different binning activities. In the example in Figure 1, device D14 substitutes for device D16 via a general substitution; in contrast, the activities between D11, D12, and D13 are called simple binning with substitutions.

Wafers travel between fabrication steps in discrete containers, each of which can hold as many as 25 wafers. The best production plans result from running the CPE with the lot-sizing option turned on. In this particular case, lot sizing means the wafer starts are in multiples of 25. However, even an efficient algorithm can require a long run time to compute feasible plans with lot-sizing constraints. Consequently, some CPE runs at IBM consider lot sizing; others do not, depending on user preferences for trading-off solution accuracy and run time in a particular situation. Sullivan and Fordyce (1990), Lyon et al. (2001), Denton et al. (2006), and Monch et al. (2011) provide details on semiconductor manufacturing and enterprise supply chain optimization.

Related Literature

Researchers have developed a variety of divide-and-conquer approaches to handle large-scale supply chain optimization problems. Gupta and Maranas (1999) proposed a hierarchical Lagrangean relaxation procedure for an MIP formulated for tactical planning. They used dual information to systematically identify key complicating constraints and divided the problem into smaller and more computationally tractable subproblems.

Frederix (2001) proposed a stepwise optimization methodology to plan the extended semiconductor enterprise: a rough feasible plan is generated by using material requirements planning (MRP) or dynamic scheduling heuristics; he improves this plan by identifying and swapping pairs of critical operations in a specific order, and subsequently runs a branch-and-bound algorithm to find subcontracting options to facilitate make-or-buy decisions.

Lee et al. (2006) suggested an integrated approach by considering semiconductor supply chains as a production chain and a distribution chain. They considered three policies (push, balance, and pull) for the production chain and two policies (push and pull) for the distribution chain; each policy was formulated using mathematical programs. They then evaluated six combined policies at a Korean semiconductor firm.

Leachman et al. (1996) developed a production planning and delivery quotation system, IMPReSS, for Harris Corporation's semiconductor sector. IMPReSS solves linear programs (LPs) and uses a heuristic decomposition scheme to break the problem into tractable pieces. Unlike our decomposition scheme, IMPReSS assigns parts to heuristics and optimization methods based on the parts' categories, irrespective of product structure complexity.

Lin et al. (2000) modeled the extended supply chain of IBM's Personal Systems Group as a network of inventory queues and decomposed the problem by capturing queue interactions based on actual lead times.

Intel conducted capacity planning with reconfigurable kits (e.g., fixtures and jigs) using a two-level methodology developed using mixed-integer programming: midrange (monthly) planning focuses on resource purchasing and demand allocation across the extended enterprise; short-range (weekly) planning optimizes the allocation of machines and kit components at a single factory (Zhang et al. 2006, 2007).

Elements (subroutines) of the CPE have been described in the literature. Most notably, Denton et al. (2006) describe the MIP used within the CPE, and Lyon et al. (2001) provide an overview of the fast heuristic method used within the CPE and the LP embedded within that heuristic. Our paper is the first journal article to describe the CPE's problem decomposition approach, overall algorithm, and process for blending its elements.

The Central Planning Engine

To understand the CPE, an understanding of the basics of explosion and implosion (i.e., techniques

developed to calculate supply chain plans) is helpful. Explosion begins by determining the manufacturing releases (starts) needed to support demand for parts at the top of the BOM supply chain and then exploding these releases to calculate the dependent demand for parts at the next level down in the supply chain. Using Figure 1 as an example, suppose a demand for 30,000 units of module M11 is to be satisfied on day 20 (i.e., 20 days from today). If the lead time of M11 is 4 days and producing each unit of M11 consumes two units of module M21 with no product scrapped, then 30,000 units of M11 must be started on day 16 to meet the demand on time (20 days minus M11's lead time of 4 days equals 16). This is the manufacturing release plan for M11, which is then exploded to create a dependent demand of 60,000 units of M21 to be satisfied by day 16. The same calculation repeats until manufacturing releases have been determined for wafer W11. The result of these explosion calculations is a production and material plan at all levels of the supply chain necessary to satisfy all demand on time.

Because limited capacity or fixed, planned lead times may require new activities to have been performed in the past, the plan that an explosion creates may be temporally infeasible. Consequently, an implosion that performs calculations in the opposite direction of an explosion may also be required. Again referring to Figure 1, an implosion would first determine a feasible plan for wafer W11. It would use this plan to calculate a feasible plan for untested device UD11, and so on, until it determines a feasible plan for all levels of the BOM supply chain. Continuing with this example, suppose 40,000 units of M21 will be available on day 30. If these 40,000 units are used immediately to produce M11, 20,000 units of M11 will be completed by day 34. Because implosion calculations consider capacity and material availability, the resulting plan provides an estimate of future finished goods supply and the dates by which this supply will satisfy customer demands.

Figure 2 illustrates how the CPE generates supply chain plans by performing an explosion followed by an implosion. To perform these two tasks, the CPE first decomposes the problem by dividing the BOM product structure into three stages: card, module, and device-wafer. It further divides each stage

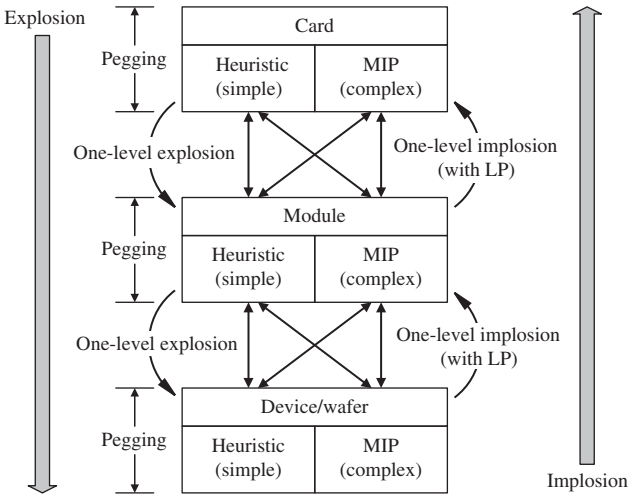

Figure 2: The IBM CPE generates supply chain plans by decomposing the problem into simple material flows (solved with a heuristic algorithm with an embedded LP) and complex material flows (solved with mixed-integer programming). Processing occurs in stages from card to device-wafer via explosion and from device-wafer to card using implosion. Coordination among the stages is achieved through a unique pegging algorithm, a one-level explosion calculation, and an LP-based one-level implosion.

into two portions, which are based on the alternative supply chain paths available to build a part: simple and complex. The CPE follows the sequence of card, module, and device-wafer, performing explosions one stage at a time by using an MIP (Denton et al. 2006) for the complex portion and a fast heuristic (Lyon et al. 2001) for the simple portion at each stage. This creates a temporal plan for all levels of the supply chain necessary to satisfy all demand on time. The CPE then follows the sequence of device-wafer, module, and card, performing implosions to generate an enterprise supply chain plan while still using the MIP and the heuristic at each stage. The remainder of this section describes the decomposition of the problem into stages and portions, the MIP and the heuristic that are used at each stage, and the procedures that connect calculations of the supply chain plan from one stage to the next adjacent stage. These connecting procedures are (1) pegging, which passes demand information from the top of a stage to the bottom, (2) one-level explosion, which explodes manufacturing releases from the bottom of one stage to create dependent demand on the adjacent stage downward in the supply chain, and (3) one-level

implosion, which uses the part supply of one stage to constrain the fabrication of parts at the bottom of the adjacent stage upward in the supply chain.

Wafers, devices, modules, and cards represent the main stages (part groupings) in semiconductor manufacturing. Generally, portions of the BOM in one stage are independent of those in other stages in terms of capacity usage and manufacturing processes. Moreover, plant locations for performing these stages might also be different. IBM does its wafer fabrication in Essex Junction, Vermont or Fishkill, New York; it does test and assembly in Bromont, Quebec in Canada, or outsources it to firms in Southeast Asia. These stages thus became targets for problem decomposition; therefore, we adopted card, module, and device-wafer as the stages in the CPE.

Decomposition of Product Structure and Capacity

We leverage the decomposition approach by classifying PNs at each stage as complex or simple to form the stage's complex and simple portions. As we explained in the beginning of this paper, complex portions involve choices of supply chain paths, whereas simple portions involve straightforward choices or no choices. We will now describe how we classify each PN. Complex PNs are those directly involved in sequential binning, general substitution, and alternate BOM/plant locations (see Figure 1), or connected to complex PNs through the BOM supply chain within the same stage. Planners can also manually designate some parts as complex, allowing the MIP to make better decisions about capacity consumption and sourcing alternatives for these parts. Simple PNs are those not classified as complex. This PN classification thus creates a complex and a simple portion within each stage and six portions for the original problem: card-complex, card-simple, module-complex, module-simple, device-wafer-complex, and device-wafer-simple (see Figure 2).

Figure 3 illustrates a PN classification for the module and the device-wafer stages. At the module stage, PNs M15, M34, and M35 are classified as complex because they have alternate BOMs; other PNs in the module stage that have a BOM connection with these PNs are also classified as complex (i.e., all 15 PNs in the upper-right quadrant of Figure 3). Similarly, at the device-wafer stage, devices D11, D32, and D33 are classified as complex (because of their alternate BOM and general substitutions), as are other PNs that have a BOM connection with them (i.e., all 11 PNs in the lower-left quadrant of Figure 3). The remaining PNs (i.e., those in the upper-left and lower-right quadrants of Figure 3) are classified as simple.

Card, module, and device-wafer manufacturing consume different capacity resources. Therefore, allocating capacity between the card, module, and device-wafer stages presents no issue. However, within a given stage, capacity is often shared between the simple and complex portions; thus, it must be divided between them. During the explosion processing of each stage, each portion is treated as if it had the entire capacity available to use (see Figure 2). The capacity requirements load determined through the explosion processing to support the resulting tentative production and material plan in each portion becomes a basis for prorating proportionally the available capacity between the complex and the simple portions at that stage. This capacity allocation is performed prior to the implosion processing; the complex portion will receive, as its share of capacity in each period, the maximum amount from the three quantities listed as follows:

(1) the complex portion's prorated share of the total capacity that is required to support complex and simple PNs for the given period;

(2) the complex portion's prorated share of the total required capacity averaged over the given period, the previous period, and the next period;

(3) the complex portion's prorated share of the total capacity over the entire planning horizon.

This ensures that the capacity that is available to the MIP is at least equal to the prorated share of the complex PNs during the current, intermediate, and long-term time frame. We found that this approach allows the MIP to use capacity that is close to its prorated share because the capacity of a given period is not always consumed. Any capacity not consumed by the MIP is reallocated to the simple portion because the heuristic conducts implosion processing of the simple PNs following the implosion of the complex PNs by the MIP.

Since IBM implemented the CPE, the proportion of PNs designated as complex has usually varied from about a quarter to two-thirds. When this proportion

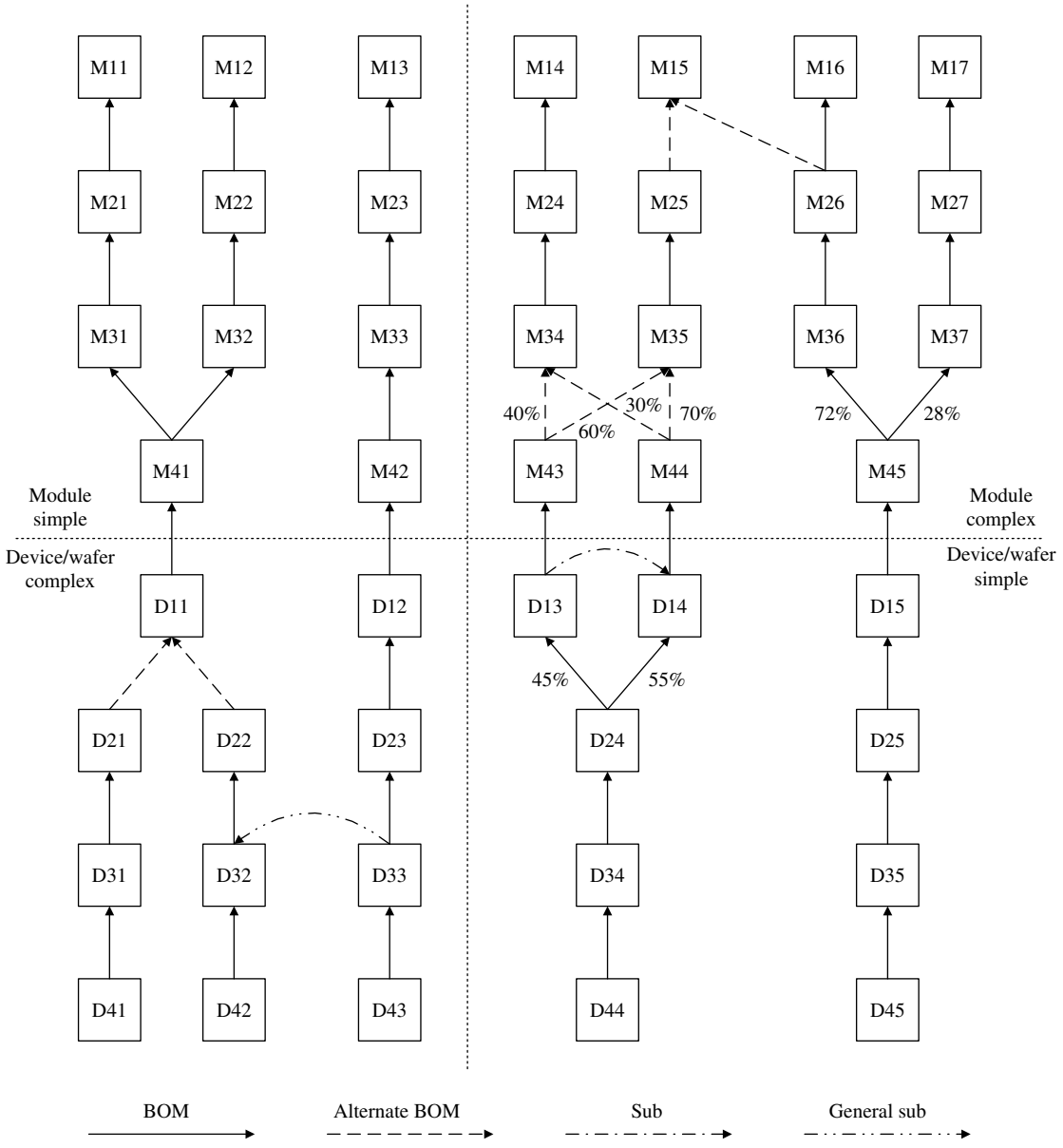

Figure 3: Parts at the module and device-wafer stages are classified as simple (i.e., straightforward supply chain path decisions within the stage) or complex (i.e., alternative supply chain paths caused by alternative BOMs or general substitutions).

rises above 70 percent, the business has tended to predesignate all parts in a stage as complex. This permits a better usage of capacity because it is not divided between two portions of a stage. Over the last decade, the substantial improvement in run-time performance (from hardware and algorithms) has enabled the planning community to gain experience using the MIP to plan all parts of a stage. Generally, large MIP models are vulnerable to run times that spike because of what initially appears to be small adjustments in the input data. The heuristic remains dramatically faster than the MIP, which is particularly helpful with what-if runs. Although we expect that the MIP will have a larger role in supply chain

optimization over time, the decomposition approach remains critical at present.

Factory planning teams use a wide variety of methods to estimate capacity (i.e., availability and consumption rates) and manufacturing lead time. Some methods are as simple as reviewing the actual lead time on completed lots over the previous 30 days, basic spreadsheet analysis of engineering estimates of the consumption rate per part per machine, or manufacturing estimates of machine availability. Many, but far from all, factories incorporate aspects of the lead time and capacity trade-off curve. Some use stochastic simulation models of their manufacturing line; others, such as the IBM 300 mm wafer fabrication facility in East Fishkill, New York (Brown et al. 2010), are slowly adopting advanced queuing network methods.

CPE Core Engines—Advanced Heuristic and the MIP

The CPE uses a fast and novel heuristic algorithm to determine supply chain plans for parts classified into the simple portions. This algorithm models simple binning with substitutions, as illustrated between device PNs D13 and D14 in Figure 3. The heuristic runs an explosion followed by an implosion. The explosion is based upon traditional MRP logic supplemented with the solution of small LPs at simple binning points to obtain optimal required starts; Lyon et al. (2001) discuss formulation details about this LP. The explosion calculates the capacity required to support the starts; if necessary, it moves them earlier in time to become capacity feasible.

During the heuristic's implosion, plant locations are processed in a smart sequence, such that manufacturing locations are processed first, then distribution locations, and then receiving locations. Furthermore, the heuristic processes PNs according to a sequence that considers both BOM relationships and parts that share capacity resources, thus ensuring effective capacity allocation. In determining the parts sequence, the first priority is to satisfy the BOM relationships so that components are processed before their assemblies; the second priority is to satisfy the substitution relationships so that parts that substitute for other parts are processed before parts that are being substituted. Only after those priorities have been satisfied are the capacity relationships considered. To consider together all PNs sharing capacity may not be possible during the implosion. Denton et al. (2003) describe how this sequencing is accomplished.

For each product level in the sequence (beginning at the bottom of the BOM), using the material plan established during the explosion as a starting point to be modified, the heuristic's implosion matches demand with supply chain assets, including work in process (WIP) projected to stock, purchase orders, and planned receipts from new starts. The implosion adjusts the plan to account for limited material and capacity availability. When choices must be made, the heuristic allocates material and capacity based upon the demand class priorities associated with the planned manufacturing starts. The concept of demand class recognizes that some demands are more important to satisfy than others (e.g., firm customer orders with high profit margins are more important than forecasted demand of low profit margin products).

The following four steps describe the heuristic's matching process at a particular level in the sequence.

(1) Sort the explosion-created starts at this level by start date and PN.

(2) Push the planned start date of any back-ordered starts forward in time to the plan effective date (typically today).

(3) In sorted sequence moving forward across time, assign each start to the earliest date equal to or later than its (Step 2-adjusted) planned start date that has available component supply and capacity to support that start date. If no remaining component supply or capacity is available on any of these dates, check if supply and capacity can be redeployed from a start that is being used to satisfy lower-priority demands. If so, push that lower-priority start later in time using the same logic and redeploy its component supply and capacity so that the higher-priority start can be released closer to the on-time date.

(4) Once the start has been assigned to a date, reserve the component supply and capacity consumed by that start. These starts may be delayed further if they are preempted by starts with more important demand class priorities.

The CPE's heuristic algorithm has advantages over simple greedy algorithms, which take demands in priority sequence (most important demands first) and assign each in turn to activities required to satisfy

them, booking inventory and capacity. Although such greedy approaches can work well in satisfying high-priority demands, the resulting allocation of inventories and capacities can create unnecessary short falls for the lower-priority demands. In contrast, our heuristic first conducts an explosion to consider demands of all priorities, followed by an implosion to satisfy the lower-priority demands as best possible; if necessary, it preempts their supplying activities to free capacity or inventory to meet the higher-priority demands. Consequently, our heuristic will do as well as the greedy approach in satisfying high-priority demands, but will do better in satisfying less important demands.

For more complex product structures, including sequential binning, general substitutions, and alternate BOM/plant locations, the CPE solves an MIP to minimize the overall supply chain costs. The MIP leverages customized optimization code to provide superior performance. Denton et al. (2006) describe the formulation of this MIP and the methods for solving it. The MIP contains constraints such as back-order balancing (ensuring that demand unsatisfied in one period is back-ordered to the next), inventory balance of product flows (ensuring that quantities arriving in inventory stay there until they leave), capacity limitations for starts, and sourcing balance (between multiple sources and locations). The run time required by this MIP is much longer than that required by the heuristic: in one experiment, we tried solving the entire supply chain problem with a single MIP; however, because the resulting model had over 10 million decision variables, it could not complete within 24 hours. This motivated our decomposition approach to allow a more effective use of the MIP and the heuristic.

During the explosion processing of the complex parts at one stage, the MIP must include some parts from the adjacent stage(s) downward in the supply chain. For example, consider module PN M15 in Figure 3. Because M15 could be built using either component M25 or M26, it must be planned using an MIP. When that MIP executes, its optimal decision of how many units of M15 to produce using M25 and how many to produce using M26 depends upon the in-stock and in-process inventories of M25 and M26 and of all their components and subcomponents extending through the supply chain down to device PNs D44 and D45. Consequently, all these PNs in the quadrants on the right side of Figure 3 will be processed using a single MIP during the explosion of the module-complex portion.

Exploding Down the Supply Chain

After the explosion of the card or the module stage, two supporting algorithms are invoked so that the explosion processing can continue to the module or the device-wafer stage. First, a pegging algorithm is run to associate end-item demand information with bottom level of stage manufacturing starts. In the example in Figure 3, after the module MIP runs during the explosion, the resulting planned customer shipments of module PNs M14, M15, M16, and M17 are associated with the end-item demand they are supporting. The pegging algorithm associates these demands on M14, M15, M16, and M17 with the manufacturing releases at the bottom of the module-complex portion that support them (i.e., M43, M44, and M45). This enables the priorities of these demands to be passed down with the dependent demands on D13, D14, and D15, which support the releases of M43, M44, and M45.

In explaining the pegging algorithm, we first define terms we call assets and needs. Assets are on-hand inventory and any items that add to inventory (e.g., WIP projected to stock, purchase orders, planned receipts from new starts, receipts resulting from part substitutions, and planned shipments from other plants). Needs are any items that subtract from inventory (e.g., disbursements of components for parent-assembly planned starts, planned shipments to customers, shipments to other plants, and withdrawals resulting from part substitutions). Both assets and needs exist by PN and plant location and were created by processing the solutions from the MIP and the heuristic explosions. The pegging algorithm executes the following steps at each BOM level from the top down.

(1) Match each asset with the needs it supports (i.e., by PN and plant location), using the assumption that inventory is consumed in a first-in-first-out sequence. This creates matching asset-need records. When multiple needs are being satisfied by a single asset, create a matching record for each need; likewise, when

multiple assets are being consumed by a single need, create a matching record for each asset. In all cases, the matching asset-need record contains the quantity of the asset that fulfills the same quantity of need for that record.

(2) Match the asset-need records with the final end-item demands and dependent demands that are supported by the needs in those asset-need records. This creates matching asset-need-demand records. When matching a single asset-need record with multiple demands, create a matching record for each demand; likewise, when matching a single demand with multiple asset-need records, create a matching record for each asset-need record. When a single asset concurrently fulfills multiple demands (because of binning), apply the priority of the most important demand to the resulting asset-need-demand records.

(3) If the current BOM level is not the bottom level of the stage, then for each asset-need-demand record where the assets are planned manufacturing releases (starts), explode these starts to create dependent demands that are identified (pegged) with the final end-item demand they are supporting using the BOM and the demand section of that asset-need-demand record.

After the pegging algorithm, the planned starts (at the bottom level of the stage) in the matching asset-need-demand records (i.e., pegged starts) are exploded—the quantity of the pegged start is multiplied by the quantity-of-component-per-assembly value to create dependent demand on the next stage. This process is called one-level explosion because it is done only for the one BOM level between two adjacent stages. It is similar to a one-level MRP explosion; however, it will also pass end-item demand information contained in the pegged starts at one stage to the dependent demand it creates on parts in the next stage. These dependent demands are viewed as independent demands from the perspective of the lower-stage processing. Continuing the example in Figure 3, with explosions all done for the module stage, the pegging algorithm creates pegged starts for modules M41 through M45, which the one-level explosion process then explodes to create demand on devices D11 through D15, respectively (see Figure 4).

Therefore, as the CPE's explosions progress, end-item demand information gets passed down from the top to the bottom of the current stage's BOM supply chain through the pegging algorithm; thus, it can be used by the one-level explosion process between stages. This passing of demand information is important so that lower levels of the supply chain can be planned with considerations of the end-item demand class priorities. Following one-level explosion, the explosion of the next stage begins until the process reaches the device-wafer stage.

Imploding Up the Supply Chain

The end of the CPE explosions triggers the start of the CPE implosions, one stage at a time. Starting from the device-wafer stage and terminating at the card stage, the implosion processing mirrors the explosion processing. In Figure 2, explosion passes demands downward from the top of the BOM supply chain, and implosion limits production to that permitted by

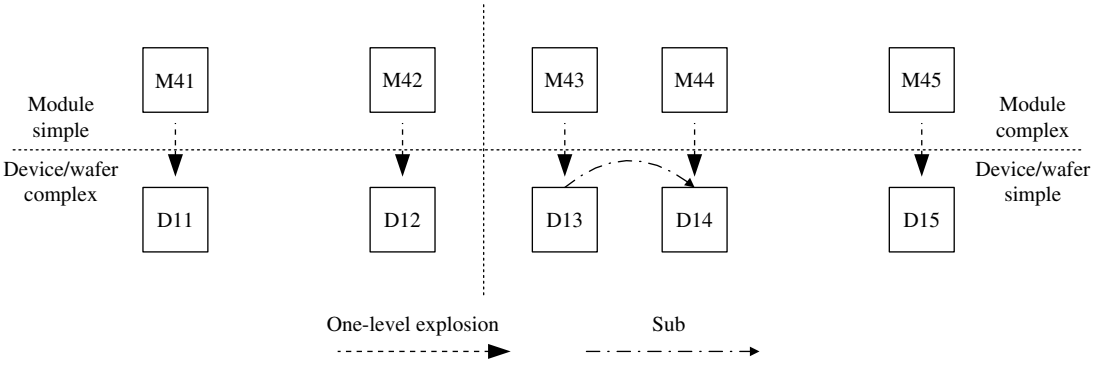

Figure 4: One-level explosion of the planned starts of parts at the bottom of the module stage (e.g., M41) creates dependent demand for device parts (e.g., D11).

available component supply and capacity, proceeding upward from the bottom of the BOM supply chain. At each stage, the implosion invokes the MIP and the heuristic to execute a capacitated run, matching assets with the explosion-determined requirements for the complex and the simple portions, respectively. Unlike the explosion of the complex portions, the MIP does not implode PNs beyond the current stage.

Between the implosions of two adjacent stages, a one-level implosion algorithm executes an LP to allocate supply from PNs at the top of the BOM of one stage to the bottom level starts of its adjacent stage upward in the supply chain. In Figure 3, a one-level implosion takes the customer shipments (i.e., supply) resulting from the MIP and heuristic implosions of devices D11 through D15 as input and allocates them to modules M41 through M45 based on using as demand the desired module releases calculated by the module MIP, heuristic explosions, and any independent demand that customers place directly for the devices. Denton et al. (2006) describe the formulation of this LP model; however, when used for one-level implosion, the model does not use lot sizing, substitutions, or capacity. Furthermore, the customer shipments resulting from the MIP and heuristic implosions at the device-wafer stage are used as the only receipts in the LP model. No new devices may be manufactured, and the BOM and other input data used by the one-level implosion process contain only the PNs at the top of the device-wafer stage and the bottom of the module stage.

As a final CPE step, postprocessing consolidates outputs from all six portions to create a single, coherent supply chain plan, which is further processed to create reports for the supply chain planners. Hegde et al. (2004) describe additional details on the CPE algorithm and some of its underlying elements.

Impact

The CPE has played a major role in planning IBM's semiconductor supply chain. It balances between runtime constraints and solution quality, and handles data that range from long-term strategic (yearly) planning to short-term operational (daily) scheduling. For daily usage, the CPE can solve enterprise data in a few hours. Multiple data sets can be solved in a single day. IBM planners regularly use the CPE to conduct tasks such as what-if analysis, material and capacity planning, asset allocation, order commit, and scenario comparisons.

The CPE and its associated changes in business processes have improved the efficiency of IBM's semiconductor supply chain, providing the following benefits.

- On-time deliveries to commit date increased by 15 percent.
- Asset utilization increased by 2–4 percent of costs.
- Inventory decreased by 25–30 percent.

The CPE has linked and synchronized supply chain activities, successfully transforming IBM's semiconductor extended enterprise to become a tightly coordinated group of manufacturing and support facilities. A major US semiconductor firm has also used the CPE to optimize its supply chain. Importantly, the CPE set a new standard and serves as a base for future innovations including the mitigation of the "illusion of capacity."

Acknowledgments

Authors are listed alphabetically on the opening page of this article. We thank the anonymous associate editor and referees for their helpful comments. Any large scale successful application such as the CPE requires the dedicated involvement of many people—too many to list. Most critical were the direct managers who championed the CPE development (Peter Lyon and Barbara Wesolowski) and deployment (Stu Reed).

References

Brown SM, Hanschke T, Meents I, Wheeler BR, Zisgen H (2010) Queueing model improves IBM's semiconductor capacity and lead-time management. *Interfaces* 40(5):397–407.

Denton B, Forrest J, Milne RJ (2006) IBM solves a mixed-integer program to optimize its semiconductor supply chain. *Interfaces* 36(5):386–399.

Denton BT, Hegde SR, Orzell RA (2003) Method of calculating low level codes for considering capacities. US Patent 6,584,370, filed May 14, 1990, issued March 23, 1993.

Frederix F (2001) An extended enterprise planning methodology for the discrete manufacturing industry. *Eur. J. Oper. Res.* 129(2):317–325.

Gupta A, Maranas CD (1999) A hierarchical Lagrangean relaxation procedure for solving midterm planning problems. *Indust. Engrg. Chemistry Res.* 38(6):1937–1947.

Hegde SR, Milne RJ, Orzell RA, Pati MC, Patil SP (2004) Decomposition system and method for solving a large-scale semiconductor production planning problem. US Patent 6,701,201, filed August 22, 2001, issued March 2, 2004.

Leachman RC, Benson RF, Liu C, Raar DJ (1996) IMPReSS: An automated production-planning and delivery-quotation system at Harris Corporation—Semiconductor Sector. *Interfaces* 26(1):6–37.

Lee YH, Chung S, Lee B, Kang KH (2006) Supply chain model for the semiconductor industry in consideration of manufacturing characteristics. *Production Planning Control* 17(5):518–533.

Lin G, Ettl M, Buckley S, Bagchi S, Yao DD, Naccarato BL, Allan R, Kim K, Koenig L (2000) Extended-enterprise supply-chain management at IBM Personal Systems Group and other divisions. *Interfaces* 30(1):7–25.

Lyon P, Milne RJ, Orzell R, Rice R (2001) Matching assets with demand in supply-chain management at IBM Microelectronics. *Interfaces* 31(1):108–124.

Monch L, Fowler JW, Dauzère-Pérès S, Mason SJ, Rose O (2011) A survey of problems, solution techniques, and future challenges in scheduling semiconductor manufacturing operations. *J. Scheduling* 14(6):583–589.

Sullivan G, Fordyce K (1990) IBM Burlington's logistics management system. *Interfaces* 20(1):43–64.

Zhang Z, Zhang MT, Niu S, Zheng L (2006) Capacity planning with reconfigurable kits in semiconductor test manufacturing. *Internat. J. Production Res.* 44(13):2625–2644.

Zhang MT, Niu S, Deng S, Zhang Z, Li Q, Zheng L (2007) Hierarchical capacity planning with reconfigurable kits in global semiconductor assembly and test manufacturing. *IEEE Trans. Automation Sci. Engrg.* 4(4):543–552.

Verification Letter

Janice Ebel, Manager, Supply Chain Operations, IBM Systems and Technology Group, 100 River Road, Essex Junction, Vermont 05452, writes:

"This letter supports the manuscript submitted by Degbotse et al. entitled, 'IBM Blends Heuristics and Optimization to Plan Its Semiconductor Supply Chain.'

"The manuscript describes our Central Planning Engine which is in productive use today and has been used successfully for several years to:

- calculate a detailed world wide supply multiple times per week, available for customer commitments
- provide production and shipping direction to all manufacturing lines
- optimize material and capacity allocation
- execute what-ifs
- used for strategic as well as operational planning."

Alfred Degbotse is a senior engineer at IBM where he has developed OR solutions to large-scale advance planning and scheduling problems for semiconductor manufacturing. More recently, he has been consulting on business analytics and optimization to promote their use within IBM. He received his BS in math from the University of Science and Technology (Ghana), his MS in statistics from West Virginia University, and his PhD in operations research from Virginia Tech.

Brian T. Denton is an associate professor in the Department of Industrial and Operations Engineering at University of Michigan, in Ann Arbor, MI. Previously, he has been an associate professor in the Department of Industrial and Systems Engineering at NC State University, a senior associate consultant at Mayo Clinic in the College of Medicine, and a senior engineer at IBM. His primary research interests are in optimization under uncertainty and applications to health care delivery and medical decision making. He completed his PhD in management science at McMaster University, his MSc in physics at York University, and his BSc in chemistry and physics at McMaster University in Hamilton, Ontario, Canada.

Kenneth Fordyce joined IBM in 1977 and serves as a senior computational decision scientist with a focus on planning, scheduling, and dispatch for semiconductor wafer fabrication and packaging manufacturing operations from assigning a lot to a tool to "end to end" supply chain planning. He has a PhD in administrative and engineering system from Union University.

R. John Milne joined Clarkson University in 2010 after a 26-year career at IBM where he was a master inventor and senior technical staff member. His research focuses on the application of operations research to decision problems in supply chain management. He is the Neil '64 and Karen Bonke Assistant Professor in Engineering Management at Clarkson.

Robert Orzell is a senior engineer at IBM Microelectronics. He earned BS and MS degrees in math at the University of Vermont. He has been applying LP technology and heuristics in planning semiconductor manufacturing since the mid-1980s.

Chi-Tai Wang is an assistant professor in the Institute of Industrial Management at National Central University (Taiwan). Before that, he had a nine-year career at IBM (Essex Junction, Vermont) where he was an advisory engineer/scientist manufacturing developing decision making computer applications for the management of IBM's semiconductor manufacturing supply chains using OR technology and heuristics. His current research focuses on OR applications in areas such as supply chain management and facility layout planning. He also studies industrial dynamics and corporate strategy of Asia's high tech manufacturing industries.