Volume 09 Issue 03 Published, August 3, 2005 ISSN 1535-864X DOI: 10.1535/itj.0903

Intel® Technology Journal

Managing International Supply and Demand at Intel

Intel's Processes for Capacity Planning Optimization

Intel's Processes for Capacity Planning Optimization

Anwar Ali, Technology and Manufacturing Group, Intel Corporation
William J. Campbell, Technology and Manufacturing Group, Intel Corporation
Lance I. Solomon, Technology and Manufacturing Group, Intel Corporation
Megan A. Walsh, Technology and Manufacturing Group, Intel Corporation
James R. Wuerfel, Technology and Manufacturing Group, Intel Corporation

Index words: optimization, linear programming, planning, capacity

ABSTRACT

Customer demands for Intel® products are rarely consistent or predictable but must be fulfilled to the best of Intel's ability. Intel also regularly increases product differentiation and provides additional platform offerings. As a result, product mix, manufacturing equipment (or tool) requirements, and overall business processes at each of Intel's manufacturing plants and across the virtual factory are constantly being updated and adjusted. These practices dramatically impact how demand can be met and how capacity is utilized within both 200 mm and 300 mm Fab/Sort Manufacturing (FSM) and Assembly/Test Manufacturing (ATM).

There are a number of modeling challenges: working with an installed tool base while planning new purchases, the requirement to distribute volume requirements across sites and toolsets, and the ability to re-use tools across sites and between manufacturing processes. These constraints require interaction between multiple groups and separate capacity planning methods, and they have become increasingly difficult to manage. A more systematic and automated approach is called for.

Mathematical optimization models have been groundbreaking in their ability to gather key stakeholders around a repeatable approach. Not only have the optimization models been used to generate solutions to complex tools, they have also been used to foster collaboration between different business organizations at Intel. This has, in turn, greatly increased communication between stakeholder groups and reduced the cycle time required to produce business-ready solutions. The cost savings that resulted from using each of these tools individually as well as cumulatively has been dramatic. The use of these tools has reduced response time remarkably and aided in decisions resulting in over $1.5B of cost avoidance over the last five years.

In this paper, we reveal how an optimization approach provided powerful solutions within the FSM and ATM spaces both strategically and tactically. We also review each of the individual solutions and describe how they work together within Intel's virtual factory network.

INTRODUCTION

Intel's capacity planning process is done at different levels of detail for different time horizons. Decisions for equipment purchase or re-use are made based on target capacity, with protective capacity to support demand variability. Production planning is done for multi-year horizons split across the virtual factory network. Rough-cut capacity planning is done for a multi-month horizon using the split volume for each factory. Here the resource requirement planning is reviewed, and adjustments are made to the production plan, labor, collaterals, and material. Finally, production control is done for multi-week horizons.

The Fab/Sort Manufacturing (FSM) and Assembly/Test Manufacturing (ATM) Industrial Engineers (IEs) and Strategic Capacity Planning (SCP) co-own the multi-year capacity planning, with the IEs owning the capital purchase and re-use process. The ATM IEs also own the multi-month individual factory capacity planning, while manufacturing and planning owns production control.

Since each process has different data needs and business rules, we developed a family of applications for each one, as follows.

(1) Fab Routing Optimizer–performs product splitting and capacity balancing across Fabs.

(2) Sort Volume Optimizer–helps explore wafer sort opportunities to save additional testers.

(3) Capacity Roadmap Optimizer–performs equipped capacity split among multiple ATM factories to minimize tool purchases.

(4) Capacity Model Optimizer–performs detailed individual ATM factory capacity checks and optimizes tool allocation to product.

(5) Re-use Optimization–optimizes inter-factory (Fab, Sort, Assembly, and Test) capital equipment re-use and conversion opportunities.

Partnerships were created between Intel's Operational Decision Support Technology (ODST) group, the Re-use Teams, ATM Industrial Engineering, Strategic Capacity Planning, and TME Strategic Programs and Planning to aggressively address these issues by upgrading existing spreadsheet models to a more automated, faster, and more accurate suite of optimization models. The suite includes the Capital/Capacity Planning System that handles volume splitting, the Capacity Model Optimizer that performs site-based capacity checks, a Re-use Optimizer, an inter-factory capital equipment re-use opportunity optimization tool, the Fab Routings Optimizer that performs product splitting and capacity balancing across Fabs, the Sort Volume Optimizer that helps explore opportunities to save additional testers, and the Sort Allocation Tool that works on a shift level to meet outs and minimize setups. These models span Fab and Sort, Assembly and Test, on through equipment re-use.

FAB ROUTINGS OPTIMIZER

A major component of the periodic planning cycle is the routing of products in each process to individual wafer Fabs. These routings are important, since they impact toolset requirements. Strategic Capacity Planning (SCP) performs these routings with the cooperation of the individual factory capacity planning groups. There are a number of guidelines that need to be followed in performing the routings including under-loading and balancing virtual factory loadings. Prior to the implementation of the routings optimizer, these routings had been performed manually with the aid of ad-hoc spreadsheets without analytically comprehending the impact of product-mix implications on factory lithography capacity. Operational Decision Support Technology (ODST) partnered with SCP to produce an optimization model to semi-automate and aid in routings, comprehend product-mix, and to provide a consistent routings methodology used by all analysts and manufacturing processes. In the following sections, we discuss the routings problem, our approach to it, the distinguishing features of the optimization model, and our next steps.

The Routings Problem

Fab product routing is a manually intensive, quarterly activity that requires multiple revisions each cycle. This process requires the SCP Fab process coordinator to juggle yield vehicle direction, New Product Introduction (NPI) and product site alignments, and to finance driven constraints in a unit demand routing system while comparing over/under loads to wafer start capacities. SCP coordinators receive feedback and approval from the Fab planning/sort/yield managers, and then do their best to route products by percentages of the unit demand in response to the stated wafer start capacities.

Routing is a tedious, time-consuming process that requires uninterrupted concentration and constant checks by the analyst.

If photolithography limits are known for certain sites, routers try to accommodate this as well. These limits are commonly expressed as a maximum number of wafer starts, or percentage of product mix, that a factory can produce of a particular product. For example, in many technology nodes, factories have often needed to place limits on products that they could produce. Frequently, the initial routing results did not take into account the impact on lithography of the product mix capacity. Since these routings occur manually, time constraints do not allow this in-depth analysis to occur. The initial routing results are distributed to all involved for feedback within one week. Fab manufacturing engineers evaluated the impact of the routings on their specific sites by reviewing the Rev0 product mix and providing feedback. This feedback was subsequently incorporated into routings that drove further manual manipulation within the routing system to rebalance the Fab loadings while simultaneously maintaining all other constraints. This can be viewed as restarting the routing process. This routing process is completed for the final routings that are then used for the SCP coordinators quarterly publication.

Becoming proficient as an SCP analyst takes a long time, due to the many details and issues involved. With the ODST model the analyst is able to make changes based on feedback and to let the model incorporate them while rebalancing other products and maintaining routing constraints. The model provides a consistent solution strategy and allows analysts to complete their routing tasks more quickly.

Modeling Approach

We implemented a linear programming optimization model in two phases. In the first phase, we utilized the Frontline Systems Premium Solver Platform and Large-Scale LP Solver plug-ins to Excel*. This provided us with maximum flexibility in developing the optimization model. In this environment, we had rapid prototyping capabilities and were able to quickly experiment with different model formulations in order to best meet the SCP requirements. In the second phase, we converted the optimization model From the Premium Large-Scale LP Solver to an ILOG OPL Studio* [2] model and moved the model data from Excel to a SQL2000* database. The OPL/SQL solution was quicker and worked better with larger data sets than the model based on Excel. The model data are housed in a centralized planning database and are connected to other data sources for factory capacity information. Moving the data to SQL2000 provided enhanced data integrity and manipulation capabilities.

Model Features

There are a number of different conflicting objectives that are included in this model. Weights are added to balance the different units of measure and to indicate relative priorities. The model attempts to maximize the minimum loadings of all factories, "smooth" product loadings within a factory over time, and minimize the overloading/underloading of a site, based on input factory goal loadings.

There are a number of fixed constraints that must be met by the model. All product demand must be routed. Additionally, the planner may specify a minimum or maximum number of wafers that need to be routed to individual factories.

This optimization model uses the following input data: individual Fab process capacities, engineering requests, product demand, known factory routings on low-volume products, the previous quarter loadings, Fab overall loadings targets, and product mix information.

The model outputs include individual Fab percentage loadings and the product wafer starts and percentage loadings by Fab in each time period.

In addition to Fab routings, ODST and SCP have been investigating ATM routings. ATM and Fab routings have many differences, and they have become separate projects. One large difference is in how subcontracted routings are handled. One similarity, though, between ATM and Fab routings is in the separation of routings. While Fab routings are process specific, ATM routings are based on platform/package combinations. ATM routings are constructed as separate Assembly and Test models; the output of the Assembly model becomes part of the input to the Test model.

* Other brands and names are the property of their respective owners.

SORT VOLUME OPTIMIZER

As product and process breakthroughs increase the total number of memory bits per wafer on memory processes, the time to test these bits (even at a basic read/write level) grows at nearly the same rate. This requires more testers to support a given wafer start level than previous memory generations have required. The primary way to address this reality is to increase test parallelism (the number of die tested simultaneously). In the first 8" Flash Sort process, multiple die were tested in parallel. Our current testers can test a greater number of die in parallel. Additionally, testers may be able to test even larger numbers of die in parallel in the future. For each of these parallelisms there have been enabling tester/prober (cell) upgrades. In all, there are many unique cell/parallelism (platform) combinations. A single product may be eligible for multiple platforms at each of its three sort test operations (flows). The number of product/flow/platform combinations is practically endless. Given that the cycle time to enable a product on a new platform is fairly long, poor product allocation decisions could waste significant tester resources. It is critical to find product to platform allocations (for each of the multiple test steps), tool purchases (testers and probe cards), and a product conversion plan that ensures maximum sort capability at the minimum cost (capital, expense, and labor). Complicating matters further, these decisions must be made for all factories as well as for each individual site simultaneously.

Sort Industrial Engineers (IEs) from all sites would periodically meet and attempt to solve these issues. However, the amount of data and the possible solution sets were too complex to continue to solve manually.

Methodology and System Architecture

The tool itself, the Sort Volume Optimizer (SVO), is comprised of a mixed integer optimization model developed in OPL Studio. The Mixed Integer Program (MIP) is similar to a traditional Linear Program (LP), but contains decisions that must be integer values, adding complexity and, therefore, solve time to the solution space. The SVO was built using Microsoft Access* as the front-end Graphical User Interface (GUI). Visual Basic

* Other brands and names are the property of their respective owners.

for Applications (VBA*) codes in Access and Excel perform data transformation and SQL operations, importing flat files from the capacity model to the SVO database and preparing data for MIP solve. This architecture is shown in Figure 1.

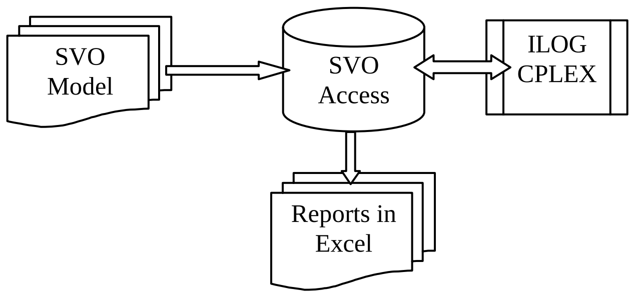

Figure 1: SVO architecture

The Access VBA code also generates data fallout reports to assist end users in debugging data. The VBA code then loads and executes the ILOG OPL Studio compiled model using component object model objects. The optimization results are then written back to the database. Finally, the Product Allocation report, Tool Purchase and Transfer reports, and other detailed reports are created in Excel (see Figure 2).

Model Formulation

The optimization model is comprised of five sets of main decision variables to determine the optimal product platform allocation roadmap, tester and probe card purchase and transfer schedules, product to platform conversion (qualification) schedules, and product cross shipping requirements/schedules. The model can also determine when it is more cost effective to miss volume than it would be to perform product platform conversions and/or to make additional tester and probe card purchases. The decisions are made in order to optimally meet the constraints and rules of the system while minimizing total cost. There are approximately 50 global constraints that must be followed in the decision-making process.

The model requires a great deal of input data in order to be able to optimally perform its decision-making routine. The Sort IEs must provide the cost and penalty information that is required in the objective function, i.e., information regarding the test operations such as run rates, utilization, tool limitations, and site space constraints. In addition, starting inventory and demand information, conversion details, and any resource and site limitations are also required inputs.

* Other brands and names are the property of their respective owners.

Advantages

The new SVO models capacity and capability and solves for the strategic product-to-platform development planning in this complex environment more thoroughly and optimally than previous manual methods allowed. As a result, the SVO has enabled new business processes (such as tool re-use strategies) and has the primary benefit of reducing the decision-making process by six weeks each quarter, thereby enabling a greater than 17% reduction in product-platform development throughput while enabling full utilization of legacy sort test platforms.

CAPACITY ROADMAP OPTIMIZERS

SCP and ATM IEs develop the CPU capacity roadmap over both the short-term and long-term horizons. Previously, allocating product capacity between multiple factories was a very manual and time-intensive process that produced sub-optimal results due to partial information at each stage. SCP generated a roadmap, IEs responded with major tool and space constraints for a revision, and finance checked revenue concentration at the end. The net result was a multi-week turnaround to create a joint SCP/IE roadmap that contributed to capital estimations with relatively inaccurate data. The need for a faster, more agile roadmap and accurate capital estimation process required an overhaul of the complete system. This work is similar in concept to that of Berman and Hood [1].

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	
1	Product	Site	Tester	Parallelisr	Flow	1	2	3	4	5	6	7	8	9	10	11
2	1316 Eng_	SF11	1316	x16	Sort 1	733	733	733	508	508	508	508	503	503	503	
3	1316 Eng_	SF11	1316	x16	Sort 2		733	733	733	508	508	508	508	503	503	
4	1316 Eng_	SF11	1316	x16	Sort 2a	733	733	733	508	508	508	508	503	503	503	
5	1316 Eng_	SF14	1316	x16	Sort 1	73	73	192	192	192	192	192	192	192	192	
6	1316 Eng_	SF14	1316	x16	Sort 2		73	73	192	192	192	192	192	192	192	
7	1316 Eng_	SF14	1316	x16	Sort 2a	73	73	192	192	192	192	192	192	192	192	
8	1316 Eng_	SF23	1316	x16	Sort 1	14	14	14	14	14	14	14	14	14	14	
9	1316 Eng_	SF23	1316	x16	Sort 2		14	14	14	14	14	14	14	14	14	
10	1316 Eng_	SF23	1316	x16	Sort 2a	14	14	14	14	14	14	14	14	14	14	
11	4400 Eng_	SF11	4400	x36	Sort 1	501	501	501	485	485	485.51	485	481	481.51	481	
12	4400 Eng_	SF11	4400	x36	Sort 2		501	501	501	485	485	485.51	485	481	481.51	
13	4400 Eng_	SF11	4400	x36	Sort 2a	501	501	501	485	485	485.51	485	481	481.51	481	
14	4400 Eng_	SF14	4400	x36	Sort 1	232.49	232.51	232.51	232.51	232.51	232	232.51	196.51	196	196.51	
15	4400 Eng_	SF14	4400	x36	Sort 2		232.49	232.51	232.51	232.51	232.51	232	232.51	196.51	196	
16	4400 Eng_	SF14	4400	x36	Sort 2a	232.49	232.51	232.51	232.51	232.51	232	232.51	196.51	196	196.51	
17	4400 Eng_	SF18	4400	x36	Sort 1	149.51	149.49	149.49	149.49	149.49	149.49	149.49	149.49	149.49	149.49	
18	4400 Eng_	SF18	4400	x36	Sort 2		149.51	149.49	149.49	149.49	149.49	149.49	149.49	149.49	149.49	
19	4400 Eng_	SF18	4400	x36	Sort 2a	149.51	149.49	149.49	149.49	149.49	149.49	149.49	149.49	149.49	149.49	
20	4400 Eng_	SFD2	4400	x36	Sort 1	1212	1212	1212	1137	1137	1137	1137	1137	1137	1137	
21	4400 Eng_	SFD2	4400	x36	Sort 2		1212	1212	1212	1137	1137	1137	1137	1137	1137	
22	4400 Eng_	SFD2	4400	x36	Sort 2a	1212	1212	1212	1137	1137	1137	1137	1137	1137	1137	
23	ARMAGO:	SF11	1316	x16	Sort 1			36	282.92	266.78	290.97	485.14	485.08	485.08	460.44	
24	ARMAGO:	SF11	1316	x16	Sort 2				36	282.92	266.78	290.97	485.14	485.08	485.08	
25	ARMAGO:	SF11	1316	x16	Sort 2a			36	282.92	266.78	290.97	485.14	485.08	485.08	460.44	
26	ARMAGO:	SF11	1316	x16	Sort 1	40										
27	ARMAGO:	SF11	1316	x16	Sort 2		40									
28	ARMAGO:	SF11	1316	x16	Sort 2a	40										
29	ARMAGO:	SF14	1316	x16	Sort 1		80	80	130	130	130	130	130	130	130	
30	ARMAGO:	SF14	1316	x16	Sort 2			80	80	130	130	130	130	130	130	
31	ARMAGO:	SF14	1316	x16	Sort 2a		80	80	130	130	130	130	130	130	130	
32	CRYSTAL	SF11	4400	x36	Sort 1	62.39	16.21	117.66	160.13	132.1	131.6	139.74	159.63	150.63	141.13	
33	CRYSTAL	SF11	4400	x36	Sort 2		62.39	28.37	180.15	233.42	203.24	203.17	217.55	230.67	223.12	
34	CRYSTAL	SF11	4400	x36	Sort 2a	62.39	24.32	159.32	208.99	179.52	179.31	191.61	206.99	198.96	192.77	
35	CRYSTAL	SFD2	4400	x36	Sort 1	62.39	16.21	83.33	97.73	94.85	95.43	103.74	94.71	96.65	103.27	

ProductAllocation / PlatformReqt / TransTo / TransFrom / TotalTesterReqt / ProdPlatRe

Ready NUM

Figure 2: Product platform allocation report in Excel

Integrated Approach

To achieve this, SCP, ATM IE, and ODST partnered to develop the CPU Capacity Roadmap Optimizer (CRO). This mathematical optimization model integrates the key rules and information to produce a solution that addresses all of the critical requirements during the first pass. With CRO, capacity roadmap allocation is now a joint effort with combined information that generates better roadmaps faster. These roadmaps adhere to key constraints for space, revenue, factory capability, and product and site ramp guidelines with conflicts clearly visible. For direct dollar savings, CRO looks to minimize key tool purchases with re-use up front (versus waiting for IE response) as long as other criteria are satisfied.

The business cycle for updating the roadmap model is a constantly evolving understanding of the inputs and outputs of the process, where each cycle builds on the previous one, and the impact of adjustments is quickly understood. The other key aspect is the feedback loop incorporated into the process. In the initial stages of entering the data, data quality is usually an issue. Whether it be a routine data entry error or significant change in an input that has impacted the solve outcome, the business user needs to be able to identify and resolve data problems quickly. The CRO provides multiple data validation checks to catch routine errors. Also, maximum solve time is an input parameter.

Business Rules (Constraints) and Criteria

The CRO considers many business rules in its optimization: which factories can run each product, factory space utilization, revenue targets, factory capacity goals, worker headcount, and new tool purchases.

The Architecture of the CRO

The CRO captures business rules in an MIP developed in ILOG OPL Studio that runs on the end-users' computers (in three countries) linked to a common SQL database. The CRO architecture enables ease of use and understanding by exporting both input and output data to Excel, displaying objective function components (penalties) to highlight constraints by factory and quarter, running pre-solve data error checks, and managing multiple scenarios.

Figure 3 shows, at a high level, the inputs and outputs of the model and what business group is responsible for maintaining those data.

Key Features

The CRO is adaptable to a changing business environment because of its user-defined penalties, ability to enable smoothing and rounding constraints, and its ability to make tradeoffs between runtime and business rules. The architecture has proven to be easy to maintain and is extendable to advancing business requirements. The user can smooth to a previous roadmap to reduce churn between cycles or to calculate the penalties on a manual roadmap. The ability to "what-if" provides a richer understanding of the relative impact of such things as space, tool costs, and revenue. The CRO provides a graph of high-level roadmap results to speed up understanding for both users and management.

With these features in place, a joint business user and technical team analysis was conducted to determine the correct penalty levels. This was done with a simple run of the model looking at the percentage each penalty consumed. By having the percentages, the joint team was able to dial the individual penalty values up and down to determine the correct level, based on business drivers and technical accuracy. This process is repeated periodically to ensure that the levels still accurately reflect current business priorities.

Since the detailed information about tool level consumption per product is available, the IEs are able to verify and validate the numbers with manual calculations. In cases where the model and the manual calculations do not match up, continuous improvement is possible.

Benefits

The CRO provides "better" roadmaps by consistent enforcement of constraints, and by eliminating sub-optimization with partial information. It also provides a build plan to plan continuity for major product transitions and factory ramps. Roadmaps can now be created for SCP/IE in 40% less time, supporting capital procurement requirements quickly. With this new tool, a richer set of "what-ifs" can be considered in the same time that one analysis was done previously. The quality of information has greatly improved through integrated information and models: the data on tool productivity metrics and space utilization are checked for quality earlier in the process. Since CRO supplies constraints and management summary data, users can visualize solve quality, and management can get more complete data upfront to facilitate explicit management discussion of constraints and tradeoff options. The net dollar impact of better data, integrated processes, and additional "what-ifs" is estimated at ~$13M, and these also identify opportunities to save multiple testers in the mid-range time horizon.

As a result the business units have a more effective business process by employing a joint SCP/IE roadmap early in the process, allowing the IE's analysis to focus on clean roadmaps with no glaring space/capital gaps and ensuring consistent rules for each version; in other words, each cycle builds on previous rules. This translates into increased productivity. SCP and IE work-hours per quarter were reduced by 11%. Moreover, the workload has shifted from overloaded site resources to ATM IE/SCP for data-population and solves.

RE-USE OPTIMIZER: OPTIMIZING SUPPLIER DOCK DATES AND TOOL RE-USE

Optimizing the delivery of Fab capital to meet process priorities is a critical aspect of meeting the production ramp. A key component in keeping wafer costs down is the re-use and conversion of existing capital equipment to meet the needs of new generations of processes across a worldwide network of manufacturing facilities. There are a large number of variables when making allocation decisions: release and required dock dates, supplier dock dates for new tool purchases, conversion costs, conversion time, new purchase prices, tool handedness and draft requirements, grant and lease considerations, transportation costs, etc. Allocation is not only complex, it is large (involving thousands of tools worth several billion dollars).

Originally, there was a manual process that attempted to facilitate equipment re-use periodically across the different factories at Intel. This manual process has now been replaced with the Re-use Optimizer.

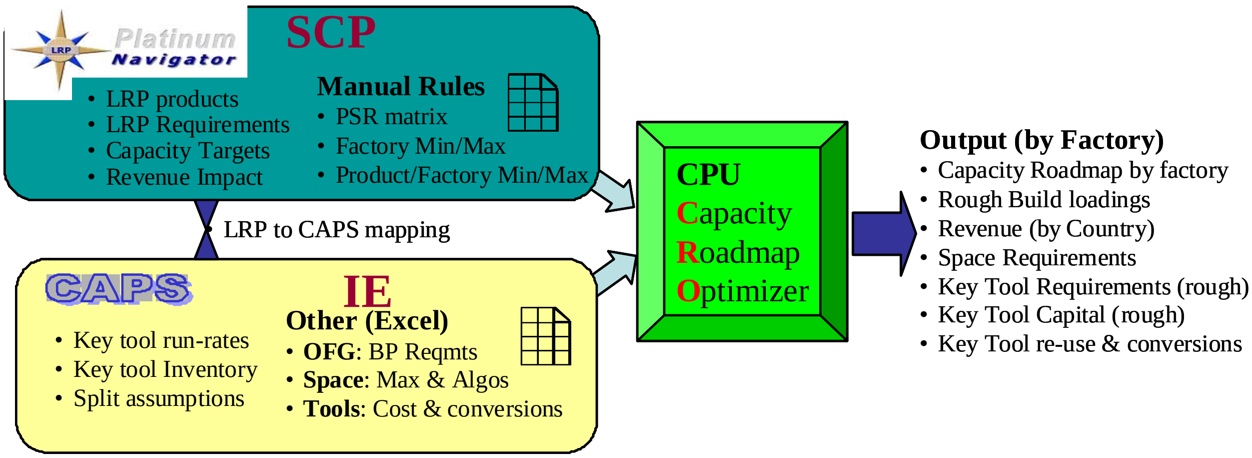

Figure 3: Capacity roadmap inputs/outputs

Re-use Optimization Goals

The Re-use and Allocation team partnered with ODST to change from the manual methods to an automated system using optimization to simultaneously minimize the impacts of supplier tool lateness and to maximize re-use tool opportunities. This optimization model automates the current manual process of allocating supplier dock dates and re-use allocations in a single step.

Methodology

The optimization methodology can be represented by an LP. The model can be described using an assignment problem, a special class of LP. Many other types of model formulations can be found in Winston [3]. Assignment problems can be characterized by a cost matrix composed of the cost of assigning each supply point to each demand point. Equation 1 displays the general form of the balanced assignment linear programming problem. The first line of the equation indicates that the objective function is to minimize the total cost of the assignments. The cost of assigning the i^{th} supply point to the j^{th} demand is denoted by c_{ij}. The decision variables (assignments) are denoted as x_{ij}. The second and third lines are the constraints on the decision variables, requiring that the total supply and demand requirements be met. Lastly, the fourth line indicates that no fractional allocations are allowed. That is, each supply must fill exactly one demand, and that each demand is filled by exactly one supply.

$$\min \sum_{i=1}^{i=m} \sum_{j=1}^{i=j=n} c_{ij} x_{ij}$$

$$\text{s.t.} \sum_{j=1}^{j=n} x_{ij} = 1 \; (i = 1,2,\ldots,m) \quad (\text{Supply Constraints})$$

$$\sum_{i=1}^{i=m} x_{ij} = 1 \; (j = 1,2,\ldots,n) \quad (\text{Demand Constraints})$$

$$x_{ij} = 0 \text{ or } x_{ij} = 1 \; (i = 1,2,\ldots m;\ j = 1,2,\ldots n)$$

Equation 1: LP representation of assignment problem

In re-use, supply points are synonymous with existing supplier dock dates or excess tools; demand points are forecast tools, and the costs are a function of the lateness of a supplier dock date allocation and the cost of re-using and converting a tool. The constraints in this system are that each excess tool must be assigned to no more than one required tool, and that each required tool must be allocated exactly one excess tool or supplier delivery slot for a new tool purchase. It is possible, of course, to place more than one tool on hold and to also purchase more than one tool.

Model Specifications

In designing a solution to the complex task of performing the allocation of all Fab capital equipment including re-use opportunities, we use the following inputs to the model:

- Capital equipment forecast quantities, excess tool lists, and equipment costs.
- Tool conversion costs and re-use kit lead-times.
- Factory customer-determined process priorities.
- Special tool circumstances and requirements.

We have made the following assumptions in the determination of the Re-use Optimization cost matrix:

- Non-lithography forecast tools must be outside of new purchase lead-time plus a fixed duration to accept re-use tools.
- Demo time on factory installed re-use tools.
- Re-use tool shipping duration.
- Existing forecast tool allocations within a specified allocation start-date do not change.
- Current allocation lateness less than or equal to a user-specified parameter is ignored.
- Current allocations are given a small weighting factor to reduce "churn"–large changes in factory allocations for small gains.
- Excess purchase orders are given priority allocation in order to minimize cancellation costs (even if cancellation cost is zero).
- Additional rules depending upon geographical locations where the Fabs operate.

SuperSTARS* has been built using the MS SQL 2000 database and Microsoft Visual Basic 6* to generate cost matrix, the ILOG OPL Studio algebraic modeling language, CPLEX* for the optimization engine, and Excel for formatting and providing a user interface to the automated allocation.

Benefits

To date, over $1.5B worth of re-use has been managed through the system. The Re-use Optimizer has been able to effectively manage more detail than human planners could: the people using the system bring their expert knowledge about information that cannot be stored in the current system. The Optimizer presents a first-cut at the allocations, and the users are allowed to override and change allocations as needed. This process has allowed the re-use planning cycle to occur in a few days by using computer networking tools. In the past, the re-use process took multiple weeks of face-to-face meetings, with individuals traveling from around the world to manage this process.

CAPACITY MODEL OPTIMIZER

Periodically a new Build Plan (BP) request is received. Since ATM consists of multiple factories worldwide, the demand is split between various factories by using high-level capacity analysis. To ensure the BP is supportable, a more detailed analysis is performed for each factory (Figure 4) before adjustments are made for an unsupportable BP.

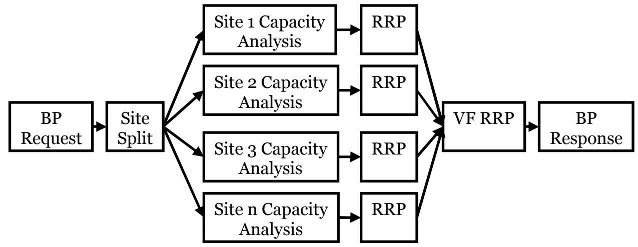

Figure 4: Weekly Build Plan process

The detailed capacity checks are performed for all steps in each factory, including steps with alternate tools. The route and required tool at each step is product specific; therefore, optimizing tool allocation to the right product is important to maximizing capacity. We applied a LP to replace the iterative manual tool allocation process done previously.

Methodology

ATM IEs use a metric model, based on Microsoft Excel, to calculate aggregated capacity for each tool when running a specific product. The capacity per tool is expressed as a runrate in thousands of units per week. The BP is also expressed in the same units, thus all capacity analysis is performed in weekly buckets for a multi-week time horizon.

The total requirement for tool *t* is defined as

$$T_t = USD_t + NSM_t + TR_t$$

Equation 2: Tool requirements

where USD_t and NSM_t are the Unique Scheduled Downtime and Non-Standard Material tool requirement for tool *t*, respectively, and

$$TR_t = \sum_{p,s} \frac{BP_p * Alloc_{pts} / 100}{RR_{pts} * Yield_p / 100}$$

Equation 3: Revenue requirements

is the revenue requirement for tool *t*. BP_p=Build Plan volume for product *p*, $Alloc_{pts}$=Allocation for product *p*, segment *s* and tool *t*, RR_{pts}=Runrate for product *p*, segment *s* and tool *t*, and $Yield_p$=yield for product *p*.

Total tool requirement, T_t, must not exceed the available tool inventory, I_t.

$$T_t < I_t$$

Equation 4: Tool inventory constraints

When the BP has been met and excess tool capacity is available, the unused tools are expressed as available

---

* Other brands and names are the property of their respective owners.

upside capacity for each step. Excess capacity allocation is based on product priority. Low-priority products will be allocated capacity to meet BP only, medium-priority products to installed capacity, and high-priority products to meet or exceed installed capacity. When multiple products have similar priorities, the excess capacity ratio is made equal among those products.

The supportable capacity for each product is determined by selecting the lowest step capacity. When IEs present the capacity statement to the resource requirement planning forum, the most impacting limiting steps are shown. From this, the forum members have a good idea of what they need to focus on, if more upsides are required.

Model Formulation

Since the problem has multiple conflicting objectives, goal programming is used. Slack variables are defined for each goal and penalized according to the importance of the goal. Initially a prototype LP model was developed and demonstrated to the customers using a solver for Microsoft Excel. The model solves one time period for a limited number of products and steps. The penalty values are tuned for various scenarios until the model meets customers' requirements. The model is then migrated to ILOG OPL Studio and expanded to comprehend all products and steps in a factory.

Optimization Model Architecture

CMO was built using Microsoft Access as the front-end user interface. Visual Basic for Application codes in Access and Excel perform data transformation and SQL operations, importing flat files from the capacity model to the CMO database and preparing data for LP solve (Figure 5).

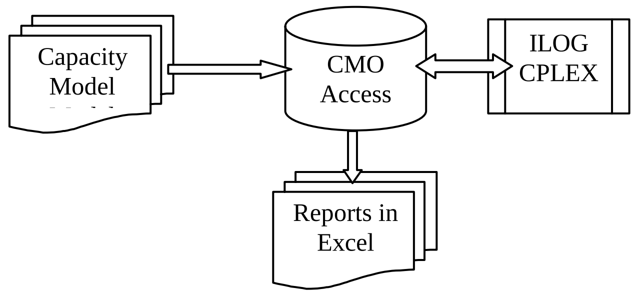

Figure 5: CMO architecture

The Access VBA code also generates data fallout reports to assist the end users in debugging their data. The VBA code then loads and executes the ILOG CPLEX compiled model using COM object. The CPLEX LP results are then written back to the database. Finally, the Limiter Chart and other detailed reports are created in Excel (Figure 6).

CM1 Optimizer Version 1.35 (July 30th, 2003)

FC, FC-BGA2, TUALATIN, MBL, 512, KB, C (FE)

Workweek	WW2103	May'03(Par	WW2203	WW2303	WW2403	WW2503	WW2603	Jun'03	Q2'03(Part
Nominal WW	168.00	168.00	168.00	168.00	144.00	156.00	168.00	804.00	972.00
Target	19.84	19.84	5.62	5.62	4.82	5.22	5.62	26.90	46.74
BP	15.45	15.45	1.74	1.74	1.50	1.62	1.74	8.35	23.80
Limiter 1	EPOXY		TEST	TEST	EPOXY	TEST	DIE PLATE		
Limiter 1 Capacity	20.60	20.60	6.98	7.00	5.96	6.52	7.09	33.56	54.15
Delta	5.15		5.23	5.26	4.47	4.90	5.35		
% Burst	33.30%		299.79%	301.23%	298.73%	302.37%	306.56%		
Limiter 2	CTL		EPOXY	EPOXY	DIE PLATE	EPOXY	CTL		
Limiter 2 Capacity	22.08	22.08	6.98	7.00	6.08	6.52	7.46	34.03	56.11
Delta	6.63		5.23	5.26	4.58	4.90	5.71		
% Burst	42.94%		299.79%	301.23%	306.37%	302.37%	327.24%		
Limiter 3	APL		DIE PLATE	DIE PLATE	CTL	DIE PLATE	EPOXY		
Limiter 3 Capacity	22.62	22.62	7.06	7.11	6.39	6.62	7.55	34.72	57.34
Delta	7.17		5.31	5.36	4.89	5.00	5.80		
% Burst	46.38%		304.55%	307.37%	326.96%	308.52%	332.52%		
Limiter 4	DIE PLATE		CTL	CTL	TEST	CTL	APL		
Limiter 4 Capacity	22.90	22.90	7.45	7.48	6.42	6.96	7.63	35.95	58.85
Delta	7.45		5.71	5.73	4.93	5.34	5.89		
% Burst	48.21%		327.02%	328.42%	329.30%	329.81%	337.45%		
Limiter 5	TEST		APL	APL	APL	APL	CURE		
Limiter 5 Capacity	24.00	24.00	7.63	7.65	6.54	7.14	9.13	38.09	62.09
Delta	8.55		5.89	5.91	5.05	5.52	7.38		
% Burst	55.35%		337.30%	338.41%	337.54%	340.66%	423.01%		

Figure 6: Capacity statement showing the top five limiters

Results

CMO reduced the time needed to prepare capacity statements by 25% while greatly improving the quality and consistency of the process. This comes from reducing the number of iterations required to determine the allocation of tools to products. With CMO, the optimal values are determined automatically. CMO also converts complex business rules into mathematical models and ensures all factories are using consistent methods to declare their capacity statement. The data fallout reports enforce data quality checks in the IE capacity model, improving data quality for use by other capacity solvers.

SUMMARY OF RESULTS

We have shown how this small subset of optimization techniques has enabled cross-site and cross-organizational teams to produce better solutions in a smaller amount of time. The models not only help to automate these different decision-making processes, but provide people common methodologies to collaborate and discuss different solutions and to produce the best results for Intel. The optimization team has also produced many other optimization tools to help in other areas such as wafer purchasing and individual tool-level improvement models. These models have saved Intel a great deal in capital cost avoidance over the past five years, and they have also reduced the time it takes to produce solutions to these difficult problems.

ACKNOWLEDGMENTS

We thank Devadas Pillai and Edward J. Yellig for their vision and belief in optimization modeling. They have allowed and encouraged us to take the risks necessary to deploy new technologies in new areas. We especially thank our peer reviewers. We also thank the many people who worked in making these different projects successful:

Fab Routings Optimizer: Kim Johnson, Jennifer McNeill, Jeff Zino, and Gina Gerave.

Sort Volume Optimizer: Jake Aranda, Roger Pharr, Kosta Rozhenko, and Janna Smith.

CRO: Marcelo Castro, Roberta Roberts, Arlene Say, and Ed Zawacki.

Reuse Optimizer: Iris Rivera, Siroos Sokhan-Sanj, Beth Adkison, Barbara Haney, Leo Monford, Matt Walters, Bill Baldinger, Kalpesh Shah, Viswas Nataraj, and Neeta Pathrabe.

Capacity Model Optimizer: Gizelle Constantino, Chris Birely, Yeoh Hooi Yeam, Tan Teong Wah, and Contessa Preclaro.

REFERENCES

[1] Bermon, S., and Hood, S. J., "Capacity Optimization Planning System (CAPS)," *Interfaces*, vol. 29, pp. 31-50, September-October 1999.

[2] *ILOG OPL Studio User's Manual*, ILOG, France.

[3] Winston, W., *Operations Research: Applications and Algorithms, Second Edition*, PWS-KENT, Boston, (1991).

AUTHORS' BIOGRAPHIES

Anwar Ali is a staff engineer in ODST Malaysia. He leads a team of O.R. practitioners in applying discrete event simulation and mathematical modeling techniques to improve manufacturing productivity. Anwar also collaborates with Malaysia universities in initiatives to apply science to manufacturing. He holds a Masters degree in Decision Science. His e-mail is anwar.ali at intel.com.

William Campbell is a senior manufacturing operations modeling engineer. Bill is currently involved in optimization and simulation primarily in the capacity planning and scheduling space. Bill holds a Masters degree in Industrial Engineering from Purdue University and an MBA from Arizona State University. His e-mail is william.j.campbell at intel.com.

Lance Solomon is the Assembly/Test program coordinator for ODST. Lance's current focus areas include full factory simulation, data integration, and mathematical optimization supporting capacity planning within Assembly/Test factories. Lance holds a B.S. degree in Mathematics from the Pennsylvania State University and an M.S. degree in Industrial Engineering/Operations Research from the University of Texas at Austin. His e-mail is lance.i.solomon at intel.com.

Megan Walsh is a senior manufacturing operations modeling engineer and manager in ODST. She received her Master's degree from Arizona State University in the area of Operations Research in 2000 with a focus on Stochastic Linear Programming. She has been working at Intel in the areas of optimization and simulation since then. Areas of interest include product allocation assignment, quality and reliability modeling, and mathematical optimization to support high-volume manufacturing. Her e-mail is megan.a.walsh at intel.com.

James Wuerfel is an Intel technologist and ODST's optimization program coordinator. Jim's current modeling activities include mathematical optimization and scheduling. Areas of interest include equipment re-use, tool-level optimization modeling, and lot-to-tool scheduling. Jim holds a Masters degree in Industrial Engineering from Purdue University and is currently

pursuing a Ph.D. degree at Arizona State University. His e-mail is james.r.wuerfel at intel.com.

THIS PAGE INTENTIONALLY LEFT BLANK

For further information visit:

developer.intel.com/technology/itj/index.htm