Planning for the Semiconductor Manufacturer of the Future

Hugh E. Fargher & Richard A. Smith
Semiconductor Process Development Center
Texas Instruments, Inc.
P.O. Box 655012, MS 3635
Dallas, TX 75265

Introduction

Texas Instruments (TI) is currently contracted by the Air Force Wright Laboratory and the Defense Advanced Research Projects Agency (DARPA) to develop the next generation flexible semiconductor wafer fabrication system called Microelectronics Manufacturing Science & Technology (MMST). Several revolutionary concepts are being pioneered on MMST including new single-wafer rapid thermal processes, in-situ sensors, cluster equipment, and advanced Computer Integrated Manufacturing (CIM) software. The objective of the project is to develop a manufacturing system capable of achieving an order of magnitude improvement in almost all aspects of wafer fabrication [1]. TI was awarded the contract in October, 1988, and will complete development with a fabrication facility demonstration in April, 1993.

An important part of MMST is development of the CIM environment responsible for coordinating all parts of the system. The CIM architecture being developed is based on a distributed object oriented framework made of several cooperating subsystems. The software subsystems include: Process Control for dynamic control of factory processes; Modular Processing System for controlling the processing equipment; Generic Equipment Model which provides an interface between processing equipment and the rest of the factory; Specification System which maintains factory documents and product specifications; Simulator for modelling the factory for analysis purposes; Scheduler for scheduling work on the factory floor; and the Planner for planning and monitoring of orders within the factory.

This paper first outlines the division of responsibility between the Planner, Scheduler, and Simulator subsystems. It then describes the approach to incremental planning and the way in which uncertainty is modelled within the plan representation. Finally, current status and initial results are described.

Planner/Scheduler Division of Responsibility

One role of the Planner is to plan and predict work completion dates, given a required confidence level, set of plan goals and the current state of the factory. This requires that the plan representation model factory resource utilization over time, and that the plan be continually updated to reflect unexpected events such as machine failure. This role is not provided by the Scheduler, which performs more locally based decision making.

As part of this role, the Planner is able to warn the user of the impact of unexpected events. For example, the Planner can determine whether work completion dates are slipping, well in advance of their quoted delivery dates. The user can also be warned of any work which has been automatically replanned due to unexpected events, so that they may request changes to the plan if required. Automatic replanning of work will remain an option to be invoked if desired by the user.

The ability to request plan changes is another key Planner role which is not provided by the Scheduler. 'What-if' plan changes refer to requests such as putting a machine on hold or introduction of new work.

Finally the Planner constrains work release into the factory, based on the current plan being executed. This is important since early release of work carries the penalty of increased WIP and early completion of work is undesirable. The high level plan representation does not allow the Planner to determine the precise moment for work release, which may be based on low level factory data such as machine queue sizes. This is an important role for the Scheduler, since work released early will only increase WIP by placing work on a queue. Work release is accomplished by the Scheduler requesting more work from the Planner, with the Planner satisfying the request as best as possible given the work planned for release over the next chosen time interval.

Another role of the Scheduler is to make sequencing decisions for work on the factory floor, based on details such as queue sizes, machine setups, and so forth. Although such decisions may be based on currently planned ship dates, this service cannot be provided by the Planner (which does not distinguish between identical resources in the plan representation). Finally, the Scheduler is responsible for tracking work in process.

- The Planner influences the schedule being executed by constraining work release and predicting work completion dates, which may be used in Scheduler dispatch decisions. However, work released into the factory cannot be directly influenced by the Planner. The Scheduler provides important feedback to the Planner by tracking work in process. This can be used to update cycle time estimates used by the Planner, and to warn of tardy work which may cause replanning.

Planner/Simulator Division of Responsibility

Both the Planner and Simulator systems provide the user with the ability to determine the consequences of 'what-if' requests. However, the allowed requests differ fundamentally between the Planner and Simulator.

Planner 'what-if' requests may be made on a single plan only, and result in incrementally updating the existing plan to satisfy the request. Typically, the existing plan reflects the current state of the factory. Rapid feedback is required, since the requests may refer to the effect of putting a machine down in the near future for maintenance, or the effect of introducing a new hot lot onto the factory floor. These requests must be rapidly evaluated if a manager is to fully benefit, since they may require immediate attention. The ability to have multiple 'what-if' plans open simultaneously will also be important if possible plan options are to be compared.

In contrast to this, Simulator 'what-if' requests are typically performed by running a suite of simulations, using factory conditions possibly selected at random from a set of work release or machine failure distributions. Feedback is not required immediately since simulation results typically refer to changes which are not immediately put into practice. Example requests may include the effect of introducing new machines into the factory, or re-training several of the operators.

The Planner system may interact with the Simulator in two distinct modes. First, by providing a static work release plan, generated using some initial factory status, which provides the Simulator with a work release time table. This is particularly important for verifying the plan model and algorithms, since simulated work completion should match plan predictions if the Planner is correctly predicting processing capacity. Second, by providing a dynamic release plan, which is updated in response to simulated events (such as machine failure) during simulation execution. This is important for verifying Planner response times, which must remain small if the Planner is to be truly 'reactive'.

Approach to Incremental Planning

A plan representation has been chosen which models the manufacturing environment in enough detail to achieve the planning functions, while allowing incremental updates due to replanning. The following section outlines the representation, along with the search algorithm used to generate and update plans.

Modelling the Plan

The plan representation is based on the processing capacity of resource groups within the factory, divided into contiguous time intervals. Each resource group has an associated set of processing capabilities which every member of the group is able to perform. Since a single semiconductor manufacturing machine may perform several different processes, a machine may be a member of several different resource groups. Each resource group is represented over contiguous time intervals, where the planned processing commitment and remaining capacity is recorded.

The plan representation does not distinguish which resource, within a resource group, is planned to process a particular piece of work represented within a plan. The representation simply commits processing time for the whole resource group to a particular piece of work. Furthermore, the plan representation does not sequence processing within each time interval, only between time intervals. In this way, the level of detail modelled by the plan is a function of both resource groups and time interval sizes. If resource groups contained only one resource, and all time intervals were shorter than the shortest processing step, the plan representation would reduce to a Gantt chart describing the processing schedule for each resource. If, on the other hand, the entire plan were covered within a single time interval, the representation would reduce to the model frequently used for planning within semiconductor manufacturing [2]. The 'time-phased' representation outlined above lies somewhere between the two extremes.

The plan representation must accurately reflect factory capacity, projected forward from the current clock time. To ensure this, all planned processing for the earliest time interval is removed from the plan representation when the clock time exceeds the time interval upper bound. Planned processing is then compared with the current state of the factory (via the WIP tracking system) and the system user is warned of any work which appears tardy on the factory floor. Finally, the processing capacity of resource groups within the first plan time interval reduce linearly with time, to reflect the constantly increasing clock time.

The Planning Algorithm

The planning algorithm is divided into two parts, that of determining the sequence of work to be planned (given its due-date, customer priority, etc), and incorporating the required processing into the plan representation (given the current resource group commitments, type of planning requested, and constraints imposed on which time intervals processing may be planned for). Planning may use the existing plan representation as a starting point, or some user defined variation if multiple 'what-if' plans are to be explored.

Deciding the sequence of work to be planned ultimately determines the overall product mix, and is determined by an ordered list of goals in which the first unsatisfied plan goal is used to sequence work for planning. The ordered goal list may be thought of as defining the Planner 'strategy'. Each goal sequences work using its associated heuristic, which is designed to guide plan generation in favor of satisfying the goal. All goals have numerical values, which must be met by the plan if the goal is to be satisfied. Once a goal is satisfied, processing moves to the next unsatisfied goal. By 'interleaving' similar goals in the ordered list, the Planner strategy can be used to satisfy several different goals, while ensuring that the plan never deviates much from satisfying any one goal [3].

Once work has been sequenced for planning, it must be incorporated into the time-phased plan representation. The resources required for each processing step must be committed over some time interval so that no resource group is overutilized and all constraints on processing are satisfied. Plan independent constraints, such as processing times and required resource groups, are determined by querying the Specification system. Within these constraints, the planning search algorithm determines precisely in which time interval to commit resource groups for each processing step.

The planning search algorithm uses a work representation in which wafer processing is divided into discrete segments, where each segment represents processing on resources which may be completed within one time interval of the plan representation. Division of wafer processing into segments is performed by calculating which segment each processing step would lie in if processing were distributed evenly over the entire wafer cycle time. Since the wafer cycle time is greater than the minimum theoretical processing time, such a representation accounts for the expected queue time during wafer processing. Each search operation either inserts or removes segments from the plan representation, terminating when all required segments for processing work have been inserted, or when no further processing capacity remains.

The search algorithm uses a modified beam search with chronological back-tracking. Maximum beam width is determined by the ratio of measured wafer cycle time to minimum theoretical cycle time, since the greater the ratio, the greater the choice of time intervals for planning each processing segment. The search space is further reduced by constraining the beam width to increase linearly with search depth. One advantage of this is that solutions which appear unpromising at an early stage in the search are quickly discarded, whereas those which appear more promising are more thoroughly searched. Another advantage is that 'disjoint' plan representations, in which no resources may be available for an extended period of time due to factory shut-down, do not prevent new work from being planned, as long as sufficient processing capacity exists while the factory is operational.

Replanning due to unexpected resource failure requires reasoning at both the goal list and the search algorithm level. To ensure that resource groups are not overutilized in the plan representation when a resource goes down, currently planned work must be sequenced for replanning. This is performed by removing work until resource utilization levels are not exceeded, and then replanning this work to be released at a later date.

Results

Table 1 illustrates performance when using this algorithm to plan new work into an existing plan. The table shows the fraction of successful search nodes (for which a processing segment was successfully inserted into the plan representation), failed nodes (for which there was not enough processing capacity in the attempted time interval), and backtracked nodes. The results illustrate that even for a highly utilized factory the search required to plan new work, for which there is processing capacity available, is not prohibitive. Furthermore the percentage of backtracked nodes does not continue to increase with committed utilization. In a semiconductor fabrication facility an average of 80% utilization across all machines is considered very high. The results in this case assume that human operators are not a bottleneck resource.

Table1:

Committed Utilization Percent	Successful Node Percent	Failed Node Percent	Backtracked Node Percent
10%	100%	0%	0%
20%	100%	0%	0%
30%	47%	40%	13%
40%	44%	44%	12%
50%	36%	50%	14%
60%	35%	52%	13%
70%	32%	56%	12%
80%	30%	58%	12%

Approach to Modelling Uncertainty

The plan representation must be able to model the uncertainty inherent in work cycle-times, since such cycle-times often form the best available data for planning. The following section outlines the approach taken to representing uncertainty in the planning process.

Domain Uncertainty

Two areas of uncertainty are tackled by the Planner, both corresponding to data which is represented by a probability distribution. The first is wafer yield, which is recorded as the probability of manufacturing n good chips given the starting number. The second is cycle time, which is recorded as the probability of completing all manufacturing steps on a wafer in a given time.

This section outlines how cycle time distributions are used within the Planner.

The objective of the Planner is to predict work completion dates to within some given confidence, which may be used to negotiate with customers. For example, an order may be represented within the plan so that it completes processing on Friday to within a 50% confidence level, but on the following Monday to within an 80% confidence level.

Modelling Uncertainty

Uncertainty is modelled within the Planner by reinterpreting the plan representation in terms of fuzzy sets [4]. Resource group utilization for a given piece of work has a degree of membership within each time interval, which reflects the expected utilization of resources for this work during the time interval. For example, the total cycle time distribution for wafer processing may be interpreted as the probability distribution for completing the final processing step at a given time. This can be modelled within the plan representation by assigning degrees of membership between time intervals to match the given probability distribution for the final processing step. The advantage gained by this interpretation is two-fold. First, computation on fuzzy sets is much less expensive than on probability distributions. Second, cycle time uncertainty within the time-phased representation means that resources committed to processing a given set of wafer steps within one time interval will very likely process some of those steps within other time intervals. This closely matches the concept of membership degree within fuzzy set theory.

To enable the Planner to reason at this level of detail, knowledge of the total processing cycle time distribution is required, as well as some estimate of the distributions required to complete each time interval's worth of processing. Intermediate processing steps for which data is recorded in semiconductor manufacturing are traditionally referred to as 'log-points'. If log-point data were available for processing steps within each Planner time interval, this data could be used to model the distributions for required processing over all time intervals. However, this log-point data may not be available for all processing steps, only the final cycle time. For this reason, the Planner uses an algorithm to estimate log-point cycle times, given the final cycletime which is available as a distribution.

The algorithm attempts to decompose the final cycle time probability distribution into cycle time distributions for each successive time interval throughout a wafer's processing. This is done so that:

- Interval cycle time distribution variance increases with successive intervals, to reflect increasing future uncertainty.
- Interval cycle time variance is bounded by the final cycle time variance.
- The final computed interval cycle time distribution matches the input cycle time distribution.

The algorithm represents distributions using fuzzy numbers and performs all calculations using fuzzy arithmetic. This approach is based on the job shop scheduling system FSS [5] which also uses fuzzy arithmetic to model increasing uncertainty in generating future schedules. A key advantage with this approach is that calculations on distributions can be performed extremely rapidly. The algorithm has been tested against simulated results, as described in the next section.

Once time interval cycle time distributions have been calculated for a given wafer processing route, they are used to 'fuzzify' the resources committed to processing steps during each time interval of the plan representation. This is achieved by using the fuzzification operator (defined for fuzzy set theory) and results in resource utilization being 'smeared out' within the plan representation. This reflects the uncertainty in the time at which planned processing will actually take place in the factory.

Once work has been planned for a wafer with a given processing route, the final cycle time distribution is used to quote the completion date to within a given confidence level. For example, if 50% of the final time interval processing has been planned to complete by Friday, the wafer may be quoted to complete on Friday with a 50% confidence level. In fact, the confidence level associated with any delivery date may be quoted.

Finally, measured cycle time distributions provide one important method for feedback to the Planner from the outside world. Cycle time distributions may be updated incrementally as wafers complete processing for each type of manufactured technology. Furthermore, since cycle times are closely related to WIP and product mix, distributions used for planning should be chosen to reflect current conditions. However, planning work in semiconductor manufacturing has shown the difficulty in predicting cycle times up-front, which are highly sensitive to conditions such as resource status and WIP levels.

Results

Table 2 illustrates the cycle time mean and variance, for part of a processing sequence completing during a given time interval, calculated using simulation and the proposed fuzzy arithmetic algorithm. The simulated CT mean and variance were calculated by performing a series of simulations, forward in time, based on known time interval cycle time distributions. The resulting final cycle time distribution (at time interval number 5) was then plugged into the algorithm to generate the set of estimated intermediate time interval cycle time distributions. The algorithm estimated time interval distributions were then compared with the simulated distributions by measuring their mean and variance. Time units are measured in numbers of time intervals. Agreement between simulated and fuzzy means remains close, while agreement between simulated and fuzzy variance improves over several time intervals. Agreement improves as CT variance increases due to the greater number of members in the fuzzy number used to represent the distribution. We intend to explore several possible variations on the algorithm in an attempt to improve agreement.

Table2:

Time Interval	Simulated Mean	Fuzzy Mean	Simulated Variance	Fuzzy Variance
1	1.11	1.00	0.10	0.00
2	2.21	2.04	0.20	0.04
3	3.30	3.10	0.28	0.16
4	4.40	4.07	0.37	0.37
5	5.48	5.48	0.45	0.45

Current Status

A prototype CIM system was built as one of the first tasks of the CIM program. This helped with the overall system design, as well as provide a platform in which to plug prototype subsystems and get feedback from potential users. However, only small parts of each subsystem had been designed at this stage.

All CIM subsystems have now been designed and documented, and are currently being implemented in Smalltalk. The MMST Planner is currently about 25% of the way through the development phase. Interfaces between subsystems have not yet been completed, so many of the results shown above have relied on 'stubbing' subsystem functionality external to the Planner. Functionality has been stubbed to match the expected external system performance as closely as possible, and is based on a detailed scenario analysis for MMST [6]. In particular, wafer processing requirements and resources have been chosen to reflect those described in the analysis.

The Planner mechanism that requires the most development is the 'what-if' capability. Several design approaches have been documented, although determining the best approach (for example, in terms of speed of response) will require experimental measurements which may only be obtained by implementation.

Finally, full CIM installation and integration within a TI fabrication facility remains as the final stage in the MMST program.

Conclusion

A reactive planning system for semiconductor wafer fabrication has been designed and partially implemented, as part of the MMST program, jointly funded by TI, Air Force Wright Laboratory and DARPA. The planning system has been designed to maintain a plan which is constantly up to date with the factory environment, and which can reason with uncertain data such as processing cycle time distributions. The planning algorithm generates plans using a variation on the traditional beam search, and models uncertainty using a fuzzy set approach. Initial results indicate that the system is able to incorporate new work into an existing plan without incurring a large amount of computationally expensive backtracking. However, further work will be required to verify plan results in an existing wafer fabrication environment, and to integrate the Planner with the rest of MMST.

Acknowledgements

This work was sponsored in part by the Air Force Wright Laboratory and DARPA Defense Science Office under contract F33615-88-C-5448.

References

[1] J.McGehee, D.Johnson & J.Mahaffey: 'Semiconductor manufacturing: a Vision of the Future', Texas Instruments Technical Journal, vol.8, no.4, pp.14-26, 1991

[2] PMDS Technical Report CSC-TR89-004, Texas Instruments internal report, 1989

[3] PMDS Memo 91-DR-01, Texas Instruments internal report, 1991

[4] A.Kaufmann & M.Gupta: 'Introduction to Fuzzy Arithmetic', Van Nostrand Reinhold Company, New York, 1985

[5] R.Kerr & R.Walker: 'A Job Shop Scheduling System based on Fuzzy Arithmetic', Proc. of 3rd Int. Con. on Expert Systems & Leading Edge in Prod. & Operations Man. pp.433-450, 1989

[6] J.McGehee: 'Scenario Analysis', Texas Instruments internal report, 1991