NISTIR 88-3872

A Functional Approach to Designing Architectures for Computer Integrated Manufacturing

Wayne J. Davis

Professor of General Engineering
University of Illinois
Urbana, Illinois 61801

Albert T. Jones

U.S. DEPARTMENT OF COMMERCE
National Institute of Standards and Technology
(Formerly National Bureau of Standards)
Center for Manufacturing Engineering
Gaithersburg, MD 20899

October 1988

Issued February 1989

NISTIR 88-3872

A Functional Approach to Designing Architectures for Computer Integrated Manufacturing

Wayne J. Davis

Professor of General Engineering
University of Illinois
Urbana, Illinois 61801

Albert T. Jones

U.S. DEPARTMENT OF COMMERCE
National Institute of Standards and Technology
(Formerly National Bureau of Standards)
Center for Manufacturing Engineering
Gaithersburg, MD 20899

October 1988

Issued February 1989

National Bureau of Standards became the National Institute of Standards and Technology on August 23, 1988, when the Omnibus Trade and Competitiveness Act was signed. NIST retains all NBS functions. Its new programs will encourage improved use of technology by U.S. industry.

U.S. DEPARTMENT OF COMMERCE
Robert A. Mosbacher, Secretary

Ernest Ambler, Acting Under Secretary for Technology

NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
Raymond G. Kammer, Acting Director

A FUNCTIONAL APPROACH TO DESIGNING ARCHITECTURES FOR COMPUTER INTEGRATED MANUFACTURING

Wayne J. Davis
Professor of General Engineering
University of Illinois
Urbana, Illinois 61801

Albert T. Jones
Deputy Project Manager, AMRF
National Institute of Standards and Technology
Gaithersburg, Maryland 20899

ABSTRACT

Developing effective CIM architectures is hampered by automation and integration problems. The key to resolving these problems lies in a better understanding of each manufacturing function and how it is related to other manufacturing functions. Our view is that mathematical models can provide this understanding. This paper presents the results of our initial efforts to develop such models. They can be used to guide the development of the technology needed for automation. They also specify the inputs, outputs, and inter-relations needed for integration, regardless of the specific CIM architecture used.

1.0 INTRODUCTION

Throughout history, organization has played a vital role in the success of large manufacturing companies. It provides the means for meeting the strategic goals of the company. Designing an organizational structure to meet those goals involves [18] 1) defining the major functions, 2) decomposing those functions into a set of activities to be performed by individual employees, and 3) developing a managerial structure to coordinate both activities and employees.

Traditionally, hierarchical managerial structures have been used to ensure this coordination. However, the number of levels and the responsibilities of the employees at each hierarchical level vary from one company to another. In some companies, all decisions are made at the top, and lower level employees simply implement them. In others, there may be several levels of decision-making. Employees make decisions based on input from their superiors and exert the control necessary to have subordinates execute those decisions. The former strategy is typical of very small shops, while the latter is imperative in large shops. Each strategy can be successful [18] in meeting the company's needs.

Recent attempts to design and build managerial structures for Computer Integrated Manufacturing (CIM) systems have been based on similar concepts. CIM functions are defined, decomposed into smaller subfunctions, and "logically" grouped to be performed at some level within the managerial structure. Almost all existing CIM structures are hierarchical [26]. A major distinguishing characteristic of these CIM hierarchies is that they do not contain people. They are made up of computers and other automated manufacturing equipment. The goal is to eventually have all decision-making, control, and information processing functions performed by these computers.

Currently, these CIM systems are part of an existing factory. In most instances, all of the decisions related to the jobs they do are still made by factory systems external to the CIM system. This includes which jobs will be done, when they will be done, and the data needed to do them. This implies that CIM hierarchies do very few of the functions they were designed to do. Although it is imperative to expand their capabilities to include more of those functions, it is proving to be far more complex than originally thought. Although there are many reasons for this, two stand out: automation and integration.

Most attempts to automate human-intensive functions such as process control, production scheduling, and process planning have met with little success. Automation has been difficult because we do not understand enough about the functions themselves or the way human beings perform those functions. There is no consensus on the best computer algorithm or procedure to use in place of a human being executing a given function. Furthermore, recent efforts to develop expert systems to address these problems have also stalled. Until such algorithms, procedures, or expert systems are developed, human beings will continue to play an important role in these functions. In fact, Davis [14] has illustrated that the CIM environment may actually enhance the human participation in certain manufacturing functions.

Integrating these automated functions into an "effective" CIM architecture is a multi-faceted problem. First, there is no agreement on 1) the number of levels, 2) the mapping of functions onto levels, and 3) whether functions can be distributed across more than one level [26]. The commonly accepted number of levels ranges from four to six. In some models material handling is supervised at the cell level. In others, it is managed at the shop level. Jackson and Jones [25] have even advocated that planning and scheduling be done at every level.

Second, although hierarchical approaches are commonly used in decentralized control theory [28,34], integrating automated decision-making methodologies into such a structure has been difficult. The best known paradigm, decomposition of mathematical programming [19,27], assumes a static, deterministic, single criterion problem. Manufacturing problems, on the other hand, are dynamic, stochastic, and multi-criteria. In addition, there are several different time scales associated with these functions which impact the way they can be grouped into levels and the way control feedback information can be used to update decisions at each level.

Third, the data needed to carry out these manufacturing functions must be collected, verified, and disseminated using advanced database management systems (DBMS) and sophisticated network communications systems (NCS). There are still many unresolved design and management problems associated with the NCS and DBMS. Finally, since data plays such an integral role in this automated environment, the NCS, and the DBMS must be integrated with whatever decision-making and control structure is used to form an "effective" CIM architecture.

1.1 STATEMENT OF THE PROBLEM

As noted above, automation and integration are intrinsically bound together. Effective methodologies for automation cannot be developed without realizing that the results must be integrated into a working system. Effective integration schemes cannot be developed without realizing what is being integrated together. The key to resolving these issues lies in a better understanding of each manufacturing function and how it is related to other manufacturing functions. Once we have this understanding, we can then address the questions involving the best architecture for a given manufacturing environment.

Our view is that mathematical models are one way to provide this understanding. Developing detailed mathematical models for all manufacturing functions is clearly beyond the scope of this paper. We present the results of our initial efforts to develop such models for a subset of the functions depicted in Figure 1 which in turn is a subset of the functions that would be considered within a CIM implementation. We address what each model does and how it is related to other models, but we impose no other structure. That is, we have purposely done this work in an architecture-free environment. In section 2 we describe models for a single process while section 3 presents models to capture the interactions among individual processes. Section 4 contains models of production planning decisions.

Even though we do not know how to solve all of these models, we feel that their definition is a first step. The models can be used to guide the development of the technology needed for automation. They also specify the inputs, outputs, and interrelations needed for integration, regardless of the specific decision-making and control architecture used. Finally, we have attempted to develop generalized formulations for each function which can be specialized for given applications. Through this approach, the consequences of each assumption pertaining to a given application can be mathematically analyzed through its modification upon the generalized formulation.

2.0 MODELING INDIVIDUAL PROCESSES

We assume that the manufacturing shop floor contains N processes P_n $(n=1,...,N)$. These processes can be machining centers, inspection centers, material handling devices, etc. We also assume that these processes are used in the fabrication of M distinct products ρ_m $(m=1,...,M)$. Modeling these processes is based on the concept of state transition functions.

2.1 State Transition Functions

The behavior of process P_n while producing product ρ_m can be modeled by the state transition function $g_{nm}[x_{nm}(t),u_{nm}(t),t]$ where $x_{nm}(t)$ is the state of the process P_n and $u_{nm}(t)$ is the controlling input into the process P_n at time t. Using a sequence of discrete sampling times $\{t_0,t_1,t_2,...\}$ where

$$t_k = t_0 + k\ \Delta t \text{ for } k = 0,1,2,\ ... \qquad (1)$$

we can define the evolution of $x_{nm}(t_k)$ via the recursive relation

$$x_{nm}(t_{k+1}) = g_{nm}[x_{nm}(t_k),u_{nm}(t_k),t_k] \qquad (2)$$

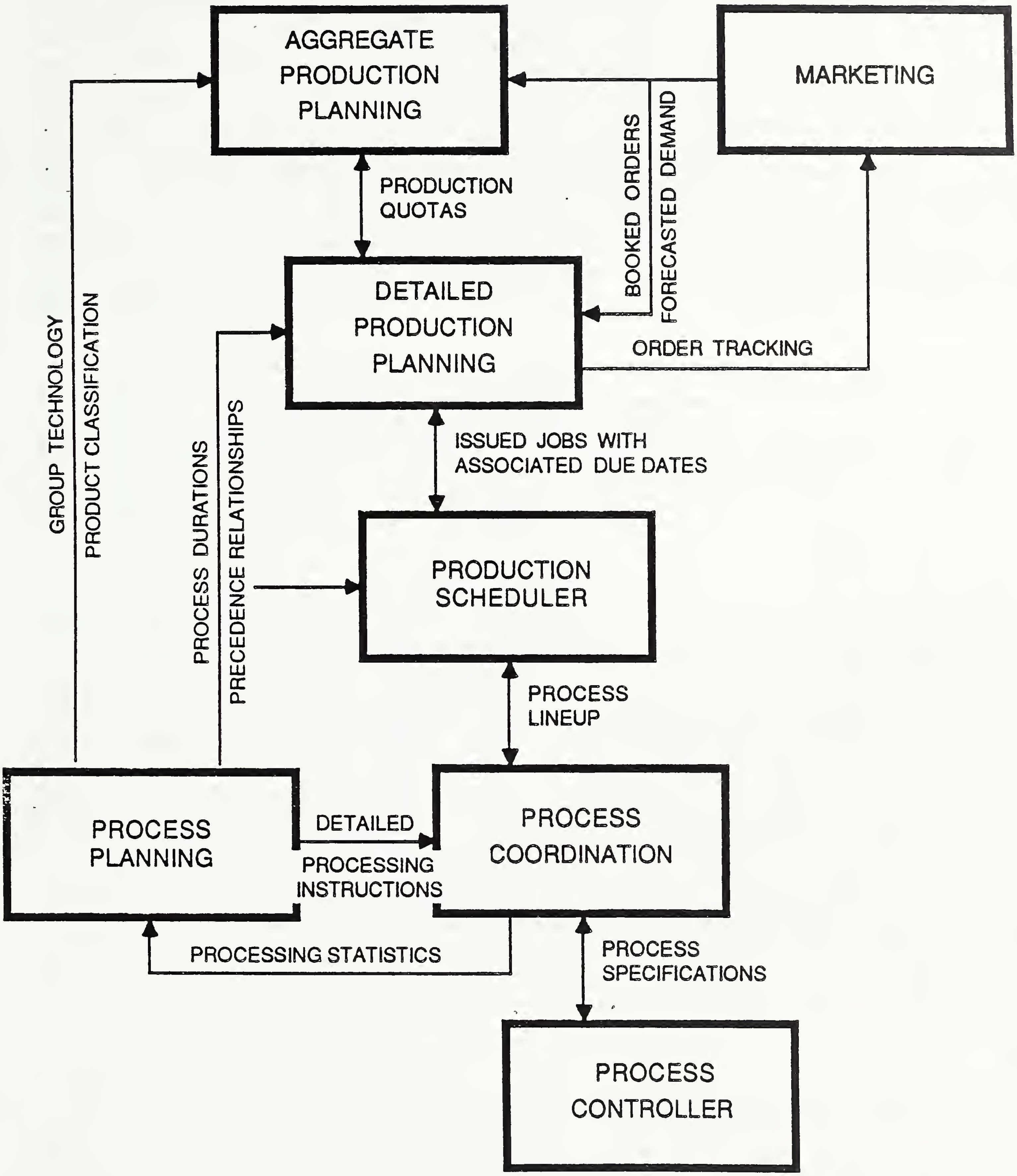

Figure 1--An Abridged Functional Schematic for a Computer-Integrated-Manufacturing Environment

There are several properties of $g_{nm}[\cdot]$ that ultimately govern the complexity associated with determining the controls $u_{nm}(t_k)$ and predicting the states $x_{nm}(t_k)$. The first is the functional dependence on time, which implies that the functional form of $g_{nm}[\cdot]$ can vary over time. The second is the fact that this variability is often stochastic in nature. In metal removal processes, the amount of input energy needed to maintain a constant cutting force can change drastically as the tool becomes worn. Another familiar example is the rolling of hot steel slabs, where the rolling process produces permanent plastic deformation upon the rollers. In both of these examples, it is impossible to predict $g_{nm}[\cdot]$ or the behavior of P_n with certainty. Consequently this means that the best that one can hope to do is to specify the $\text{prob}\{g_{nm}[\cdot] \in G_{nm}\}$ where G_{nm} is a known subset of potential functions for $g_{nm}[\cdot]$.

Another important limitation in modeling manufacturing processes today is the fact that $x_{nm}(t_k)$ is seldom observed directly. Despite the abundant research in this area, on-line process monitoring and control techniques are still not available for many processes. Until this happens, we will continue to measure the output of the process rather than the process itself. This output, $y_{nm}(t_k)$, can be defined by

$$y_{nm}(t_k) = h_{nm}[x_{nm}(t_k), u_{nm}(t_k), t_k] \tag{3}$$

Since the calibration of a measuring device may change randomly over time, the same potential problems arise in estimating $h_{nm}[\cdot]$ as in the estimation of $g_{nm}[\cdot]$ in equation (2), including the determination of the $\text{prob}\{h_{nm}[\cdot] \in H_{nm}\}$. Furthermore, it is rarely possible to measure all the processing outputs in real-time. As an example, consider a turning operation. In this case, measurements such as turning speed and the location of the tool holder can be made in real-time. However, as the tool wears the depth of the cut changes. Therefore, measurements such as the precise depth of the cut or the resulting surface roughness cannot be made in real-time, and must await for the completion or the interruption of the process for a precise measurement. Let T_{nm} and t_0 denote the anticipated duration and initiation time for the processing task; $C_{nm} = t_0 + T_{nm}$ be the anticipated completion time; and $y_{nm}(C_{nm})$ the final measured output. Then the modeler may only have a rough estimate, $y'_{nm}(t_k)$, of the true value for $y_{nm}(t_k)$ or

$$y'_{nm}(t_k) = h^*_{nm}[y_{nm}(t^*), u_{nm}(t^*), t^*, t_k] \tag{4}$$

for $t^* \leq t_k \leq C_{nm}$ where t^* is the last sampling time prior to t_k, $y_{nm}(t^*)$ is the last measured output, and $h^*_{nm}[\cdot]$ is the estimating function. Letting Y_{nm} represent a set of acceptable outputs--usually a set of predefined tolerances--the best information the modeler can hope to supply is the $\text{prob}\{y_{nm} \in Y_{nm} \mid y_{nm}(t^*) \text{ and } u_{nm}(t^*)\}$ where y_{nm} represents the entire output stream arising from the implementation of the processing task.

It seems clear that using these functional relationships to monitor and control manufacturing processes will not be possible in the near future. The task of estimating prob($g_{nm}[\cdot] \in G_{nm}$), prob($h_{nm}[\cdot] \in H_{nm}$), and prob($y_{nm} \in Y_{nm} \mid y_{nm}(t^*)$ and $u_{nm}(t^*)$) will also be difficult. In most cases, these estimates must be based on the experiences gained from past implementations of each particular process upon a given product. This, of course, leads to significant problems in estimating $g_{nm}[\cdot]$ and $h_{nm}[\cdot]$ for a given process upon a new product.

2.2 Process Control

To account for uncertainties in the evolution of a process, a self-tuning controller can be used [1,2]. As depicted in Figure 2, a self-tuning controller has two basic elements: a system identifier and a controller. The system identifier uses the recent time history for $u_{nm}[t_k]$ and $y_{nm}[t_k]$, to develop approximations for the $g_{nm}[\cdot]$ and $h_{nm}[\cdot]$, denoted by $g'_{nm}[\cdot]$ and $h'_{nm}[\cdot]$, respectively. These approximations are then used by the controller to generate the next control input. Specifically, the controller computes the difference between the measured (or estimated) output $y_{nm}(t_k)$ and the desired output $y^d_{nm}(t_k)$. This error is used in conjunction with the optimal control law $U^*[y_{nm}(t_k),t_k]$ supplied by the Process Coordinator (PC) (discussed below) to determine the next control input for the process. Thus, the process controller continuously attempts to minimize deviations from the desired output trajectory specified by the PC.

For deterministic systems in which $g_{nm}[\cdot]$ is known with certainty, the system identification element is not needed. For processes where no formulation of the state transition function exists, the controller block would not exist. In this case, the system identification element would attempt to evaluate available system characteristics and a process coordinator would subsequently define the initial process settings or control parameters. After initiation, the process would evolve in open loop fashion. As an example, consider the Basic Oxygen Furnace process in steel-making. Here the charge of molten iron, scrap steel, and alloy additives are introduced into the process. After approximately thirty minutes the finished steel is produced with little or no opportunity to sample the steel's chemistry during the processing duration. As a consequence, there is only a 70 percent probability that the correct steel chemistry will be achieved. Post processing techniques can often generate an acceptable product, but this eventual outcome cannot be guaranteed.

Although limitations on process modeling exist, the self-tuning controller is being adopted in industrial environments. Astrom _et al_. [2] provide an excellent overview of the topic as well as several examples. Watanabe [36] recently reported the development of a similar controller for a milling process.

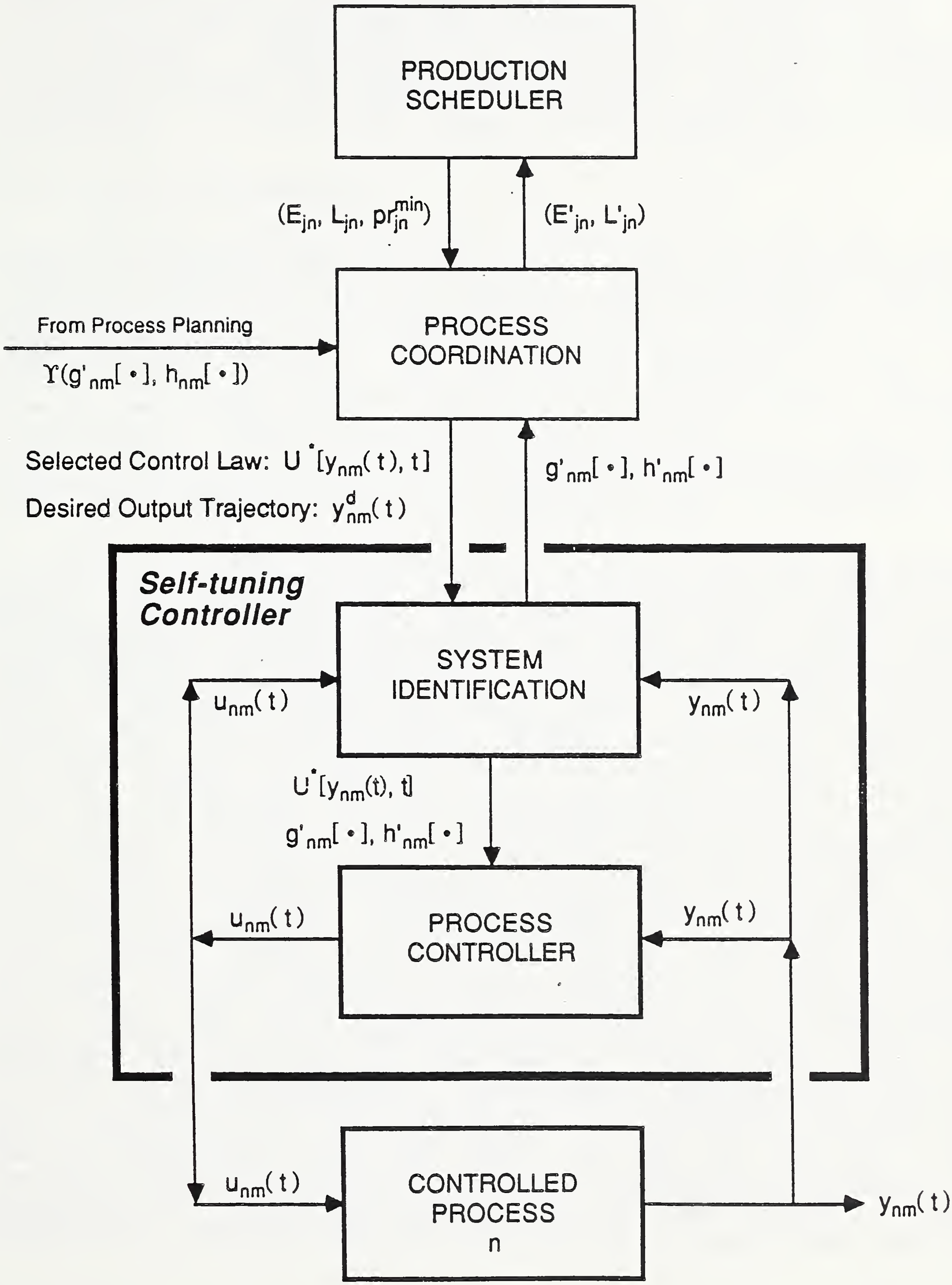

Figure 2--Detailed Schematic for Process Control

Papapanagiotou *et al*. [31] have performed work demonstrating the applicability of self-tuning controllers to controlling the rollers in a hot strip mill for steel-making. Blattner *et al*. [9] have developed a formulation for controlling the blast furnace in steel-making. Their formulation includes not only parameter estimation within the system identification element, but also a simulation to predict the system response under various control strategies.

2.3 Process Coordination

Process Coordination (PC) determines the desired output trajectory $y^d_{nm}(t)$ for a given process and the optimal control law $U^*[y_{nm}(t_k),t_k]$ to be used in achieving that trajectory. Although this law is written as a function of time, it is typically set at the beginning of the processing and remains constant until some major problem arises. The PC detects potential problems by monitoring feedback from the process and its controller. Whenever the PC determines that a new control law and/or desired output trajectory is needed during the implementation of a processing task, ideally it should also estimate the resulting probability that the process will be able to implement that law correctly, and thus, successfully complete the processing task.

To understand the relationship between the PC and the process controller, consider the tool wear example from Section 2.1. First, the PC specifies an initial optimal control law, $U^*[y_{nm}(t_k),t_k]$, which identifies the exact tool, speed, feed, cutter paths, and cutting time to be used in the process. This data is then used in negotiation with the Production Scheduler (PS) to determine start and finish times. But, as the process evolves in time, unanticipated changes in the feed, speed, or cutting time may be required to account for unforeseen tool wear or other changes in the processing environment. These changes, which are a function of the integrated measurements of $y_{nm}(t_k)$, are dictated by the process controller's success in implementing the desired trajectory $y^d_{nm}(t)$. These changes are typically minor with respect to their consequences upon processing durations, but nevertheless, updated completion times are reported to the production scheduler. If the tool breaks, major modifications could follow. First, the PC must determine if the job can be salvaged. If yes, then the PC must specify a new optimal control law with the associated probabilities for successfully implementing the task. These decisions would be based on the last measured value of $y_{nm}(t_k)$, the desired output $y^d_{nm}(t)$, and the current production schedule. Furthermore, operator intervention as well as potential additional material handling may be required. Upon reinitiation, a new completion time must be generated. It can be concluded that the focus of the PC is the total implementation of the processing whereas the process controller is focusing the immediate restoration of the desired output trajectory. We will now give two potential formulations for the

determination of the optimal control law.

2.3.1 <u>On-line Determination of the Control Law</u>. We will begin with the on-line generation of the control law $U^*[y_{nm}(t_k),t_k]$ at the implementation of process P_n upon JOB_j. From the process controller, both $g'_{nm}[\cdot]$ and $h'_{nm}[\cdot]$ have been developed. We next assume that from process planning an admissible set of control laws has been specified which we will denote by $T(g'_{nm}[\cdot],h'_{nm}[\cdot])$. This set is determined off-line and the selected control law $U^*[y_{nm}(t_k),t_k]$ must be contained within this set. For example, $g'_{nm}[\cdot]$ and $h'_{nm}[\cdot]$ could be used to estimate the remaining tool-life. Given this estimate, acceptable ranges for tool speed, cutting depth, and metal removal rates could then be specified to establish $T(g_{nm}[\cdot],h_{nm}[\cdot])$. The methods through which process planning would make this specification will be discussed in the next section. However, in Figure 1, we note that there is feedback to process planning from the process coordinator who monitors each implementation of the given process upon each job.

Another input from process planning is the minimum desirable completion probability pr_{nm}^{min} for the implementation of process P_n in the manufacturing of product ρ_m which places a minimum bound upon the $prob(y_{nm}(t_k) \in Y_{nm})$. Upon the implementation of the processing task, the determination of the acceptability of $y_{nm}(t_k)$ is made by the process coordinator. However, the basis for this determination is specified by the process planner. In most cases, selecting $U^*[y^d(t_k),t_k]$ from $T(g_{nm}[\cdot],h_{nm}[\cdot])$ should necessarily guarantee this probability constraint is met. However, as a coordinative input to the PC, the PS may increase this probability on a given JOB_j which we will denote as pr_{nj}^{min}. Specifically, only the PS can determine the consequences upon the near-term production flow if a given implementation of a process fails. Two additional inputs to the PC are presented by the PS:

E_{jn} - the arrival time for JOB_j at process P_n, and
L_{jn} - the planned pickup time for JOB_j at process Pn.

To assess the consequences upon the planned production flow, it is assumed that the PS has a collection of L objective functions of the form $f^\ell(E_{jn},L_{jn},pr_{nj}^{min})$ for $\ell=1,\ldots,L$ with an overall utility function $W[f^1(E_{jn},L_{jn},pr_{nj}^{min}),\ldots,f^\ell(E_{jn},L_{jn},pr_{nj}^{min})]$. To permit the PC to coordinate its decision with that of the PS, we will assume that there are L' compatible objectives for the PC, $\theta_{jn}^\ell[x_{nm},u_{nm}]$ for $\ell=1,\ldots,L'$ with x_{nm} and u_{nm} denoting state and input stream occuring during the processing $[t_0,t_0+T_{nm}]$ with t_0 being the planned initiation time and T_{nm} being the planned duration. Note that both t_0 and T_{nm} will be variables, not presepecifed constants, within the proposed optimization. An utility function, $\omega_{jn}(\theta_{jn}^1[\cdot],\ldots,\theta_{jn}^{L'}[\cdot])$ will quantify the tradeoffs among those objectives. A formal statement of the real-time optimal control problem can now be given.

$$\text{minimize} \quad \omega_{jn}\{\theta^{1}_{jn}[\cdot],\ldots,\theta^{L'}_{jn}[\cdot]\} \tag{5}$$

subject to the following constraints for $t_k = t_0, \ldots, t_0 + T_{nm}$

$$x_{nm}(t_{k+1}) = g'_{nm}[x_{nm}(t_k), u_{nm}(t_k), t_k] \tag{6}$$

$$y_{nm}(t_k) = h'_{nm}[x_{nm}(t_k), u_{nm}(t_k), t_k] \tag{7}$$

$$u_{nm}(t_k) = U_{nm}[y_{nm}(t_k), t_k] \tag{8}$$

$$U_{nm}[y_{nm}(t_k), t_k] \in T\{g'_{nm}[\cdot], h'_{nm}[\cdot]\} \tag{9}$$

$$\text{prob}\{y_{nm} \in Y_{nm}\} \geq \max\{pr^{min}_{nm}, pr^{min}_{nj}\} \tag{10}$$

$$t_0 + T_{nm} \leq L_{jn} \tag{11}$$

$$t_0 \geq E_{jn} \tag{12}$$

A more detailed discussion of the decomposition procedure is given by Davis and Jones [15]. Nevertheless, some particular points merit discussion here. Note that both $g'_{nm}[\cdot]$ and $h'_{nm}[\cdot]$ are explicit functions of t_k. That is, models for continued process modification (or degradation) can be included within this formulation. These models would be supplied from process planning. However, the stochastic nature of the process evolution makes a deterministic specification of $g'_{nm}[\cdot]$ and $h'_{nm}[\cdot]$ impossible to achieve. To insure the feasibility of the deterministic solution approach, conservative models for process evolution might be employed. Stochastic optimization approaches might also be adopted, but these approaches are complex and computationally intensive, making them difficult to implement in real-time. It can be concluded that the PS's decision-making must always address the uncertainties that remain.

The decision defined in relations (5) through (12) indicate two types of dynamic coordination that are to be employed within the planned interaction of the PC with the PS. The decomposition procedure uses a goal coordinative scheme by having the PS specify E_{jn}, L_{jn}, and pr^{min}_{nj} which serve as the right-hand-side of the process coordinator's constraint sets. Also we note that $\omega_{jn}\{\theta^{1}_{jn}[\cdot],\ldots,\theta^{L}_{jn}[\cdot]\}$ is indexed upon both j and n. This implies that a price directive coordination scheme can also be imposed to permit the PS to modify the PC's utility function on a given JOB_j. To support this type of coordination, additional pricing informations would be required as inputs from the PS to the PC. Using sensitivity analysis approaches from mathematical programming, the PC can study the effects of varying any cost or right-hand-side coefficient. Similarly the PS, who makes the specifications, can determine the consequences of its decision as particular parameters are varied. These variational procedures provide the basis of negotiation which can lead to the real-time modification of E_{jn}, L_{jn}, pr_{nj}^{min} or additional vaues specified within the pricing inforamation.

Finally, the existence of relations (5) through (12) highlight the shortcomings of assuming an invariant control law for every implementation of a given process upon a given product. First, it ignores the true time-varying nature of the process, and second, it eliminates any potential benefits that can be derived from the negotiation between the PS and the PC (see Davis [14]).

2.3.2 Off-line Determination of the Control Law. The off-line determination of the control law represents a subfunction within the larger function of process planning. Ideally, the entire function of process planning should also be addressed. However, space limitations will simply not permit a full discussion. Rather the focus will be directed toward supporting the on-line determination of the control law and serve as a transition to the production flow considerations which will be addressed in the next section.

Process planning provides several important inputs toward the on-line determination of the control law. The first input is the models for the evolution of the state transition and output functions, $g_{nm}[\cdot]$ and $h_{nm}[\cdot]$, respectively. Certainly, analytical modeling of the processes provides a useful input for developing this input. However, the complexity of many manufacturing processes limits this mathematical definition. Perhaps the greatest source of information for generating these models would be gained through prior experience with the processes. As mentioned earlier, there is the potential for feedback from the PC to process planning pertaining to each implementation of the process.

Although a manufacturing system may be faced with a variety of products, the physical implementations of a given process are typically composed of machine-level instructions which are universal to all products. Hence, the process planning function is to translate the engineering design for the considered product into these machine-level instructions. The first stage of the planning effort is to determine ordered sequence(s) of the processes which are to be employed. From this step, the input and output configuration for the given product at each processing stage emerges.

Several issues must be considered in generating the manufacturing sequence. The first consideration is the selected process's capability in making the desired transformation. To this end, not only must the physical processing limitations be considered, but also the process's projected availability. Often a given process's capability is not unique, and choices must be made. In this case, the process's reliability in performing the desired task and the associated cost are considerations. The implications for material handling are also major concerns. This fact leads to production flow consideration.

Given the input and output configuration for a given product at each processing stage, the next step is to develop the instruction set that will permit the process to complete the transformation. Typically certain processing parameters will be determined during this phase while others can remain undefined. For example, in milling a flat surface, a choice of the cutting tool and the number of machining passes could be determined. This specification would generate the desired output trajectory that the tool would attempt to follow. One could, however, reserve the specification of the speed that the tool traverses this trajectory as a variable. Using historical processing information, process planning could develop acceptable values to be employed given the current measurements of $g'_{nm}[\cdot]$ and $h'_{nm}[\cdot]$, thereby defining $T(g'_{nm}[\cdot],h'_{nm}[\cdot])$.

Process planning should also specify the means by which the acceptability of a given process's implementation will be determined. For example, bounds might be defined for the maximum deviation between the measured and desired output trajectory. In short, quality assurance is a critical component of process planning. Ideally, it is desired that the determination of a successful implementation of a process will insure that the output of that process will be an acceptable input to the subsequent processing steps and that ultimately the product will perform as designed. Two major obstacles remain to this realization: The inability to accurately model the processes and the inability to make accurate real-time measurements for the output of the process. To date, off-line inspection is still a primary mechanism for determining the acceptability of a process's output.

Using the set $T(g'_{nm}[\cdot],h'_{nm}[\cdot])$ and the desired output trajectory $y^d_{nm}(t_k)$ estimates of the processing duration and the consumption of processing resources can also be made. These data are essential to the production planning functions above the PC in Figure 1. In addition, the precedence relations determining the routing that a product will traverse the processes are also essential to the production planning functions.

3.0 MODELING INTER-PROCESS INTERACTIONS

3.1 Combining Process Models

Thus far, the discussion has focused upon a single process. The state of the entire shop floor system at time t_k is represented by the composite state vector

$$(x_{1,m(1)}(t_k), \ . \ . \ . \ . \ , x_{N,m(N)}(t_k))$$

where $m(n)$ for $n=1,\ldots,N$ is the number of the product being fabricated at process n and the state transition function for the process P_n is given by $g_{n,m(n)}[\cdot]$. Similarly, the output from this collection of processes is given by the composite output vector

$\{y_{1,m(1)}(t_k), \ldots, y_{N,m(N)}(t_k)\}$.

We note that as soon as process P_n completes one product and begins another, these representations are no longer valid. For each process we define a universal set of transition functions

$$\underline{S}_n = \{G_{n1}[\cdot],\ldots,G_{nM}[\cdot],\phi\} \qquad (13)$$

where ϕ is the idle state which could result from planned or unplanned events (i.e. the process breaks down). Assuming the next JOB_j arriving at process P_n will request product ρ_m, then $G_{nm}[\cdot]$ is the collection of potential process transition functions from which the implemented $g_{nm}[\cdot]$ will be chosen by the PC. If E_{jn} and L_{jn} represent the anticipated start and finish times for process P_n to work on JOB_j, then they also indicate the times where the transition function for the manufacturing system may change. These times can be derived from the current production schedule (see section 3.3).

3.2 Production Flow

For each product ρ_m, process planners must determine potential routing sequences. A given sequence yields a set of precedence relationships among processes and operations stating the order in which processes will be applied in the fabrication of the product. We can model this production flow using a process transfer function, $T_m(n)$, which determines the successor process to process P_n for product ρ_m. We define $n' = T_m(n)$ provided process n' immediately succeeds process n in the routing for product ρ_m. The exact form of the transfer function and the complexity of these process planning activities depend on the type of manufacturing system.

In a pure flow shop, all jobs proceed through the processes in exactly the same order. We can number the processes in a manner such that $T_m(n) = n+1$ and $T_m(N) = \phi$ for any $m = 1,\ldots,M$. Here ϕ indicates that there are no remaining processes in the sequence. In this case, process selection and operation assignment is trivial.

In the more general flow line, or multiple path flow shop, we again number the processes such that $T_m(n) = \{n' \cup \phi\}$ for any $m = 1,\ldots,M$ where $n' > n$. Whenever a process is required, it must always be used after a fixed subset of the preceding (lower numbered) processes. But, not all processes are employed in the manufacture of a given product ρ_m. Once the product is identified, process selection and operation assignment for JOB_j is straightforward.

In a job shop, there is usually no preferred path among the processes. This implies that the range of successors for process P_n is contained within the complimentary set of processes $\underline{N}^c(n) = \{\phi,1,...,N\} - \{n\}$. Further specification of the successor set to process P_n is often impossible. For the job shop, process selection and operation assignment can be quite complex. For a given product ρ_m, there will be a feasible set of processes which can be employed, each having a capacity constraint, which can be used in performing a given operation. The collection of feasible processes could lead to multiple routings which differ both in the processes that they employ and the order in which the processes are visited. The lack of ordering among the processes with the potential for multiple routings for a given product ρ_m considerably complicates the production scheduling function. The selection of the routing to be employed in the manufacturing of JOB_j often depends upon the current state of the manufacturing system and the queued jobs awaiting production. The lack of an ordering of processes also complicates the material handling constraints which again are a major component of the production scheduling function.

In the "preferred path job shop" or flexible manufacturing cell the goal is to partition the entire set of processes into disjoint subsets with little or no interaction. This is done using group technology [23] and significantly reduces the set of potential successors to a given process P_n. The adoption of this approach simplifies both the process selection subfunction and the material handling concerns. Therefore, the production scheduling problem is also simplified. However, the complexity of scheduling function for the flexible manufacturing system typically remains greater than that for the flow line configuration.

Routing information with estimated processing durations is used by the production scheduling function to determine the start and finish times for the operations at various processes.

3.3 Production Scheduling

Another important part of the inter-process coordination is production scheduling (PS). A formal statement of the PS problem is as follows: Assume that a set of jobs JOB_j $(j=1,...,J)$ has been issued to the PS with associated due dates D_j $(j=1,...,J)$ and that JOB_j requires the production of a specific product ρ_m. Also, assume that the processes P_n $(n=1,...,N)$ are on-line. Then, if we define

E_{jn} $(j=1,...,J;\ n=1,...,N)$ = earliest start (arrival) time for P_n upon JOB_j,

L_{jn} $(j=1,...,J;\ n=1,...,N)$ = latest finish (pick-up) time for P_n upon JOB_j, and

pr_{jn}^{min} - the minimum acceptable probability for completing the processing task upon JOB_j by process P_n by L_{jn},

the production scheduling problem is to maximize the utility function

$$\text{Max} \quad W[f^1(E_{11},\ldots,E_{JN};L_{11},\ldots,L_{JN};pr_{11}^{min},pr_{JN}^{min}),\ldots,f^L(\cdot)]$$

where $f^{\ell}(\cdot)$ for $\ell=1,\ldots,L$ are the criteria to be considered in the optimization. These criteria may include minimizing tardiness, maximizing production throughput and maximizing process utilization among others. Several constraints can be considered including due date, material handling, resource availability, precedence relations, and alternate routings. The job list and due dates are supplied by production planning. The remaining data are derived from off-line production planning and from negotiation with the PC.

The presented production scheduling statement has included the events which will insure the satisfaction of the above cited constraints. The reader may note a particular focus toward the events associated with material handling which is a major concern of the PS. Indeed, it is these constraints that often lead to a distributed scheduling function throughout the manufacturing organization. For example, consider a manufacturing shop consisting of several manufacturing cells. At the shop level, there could be an automated guided vehicle system to move the parts from one cell to another. Within a given cell, there could be a dedicated robot for material handling to move the part among various processes within the cell. Both material handling systems would typically possess their own dedicated controllers and coordinators. In this sense, they are processes. A distributed scheduling approach would assume that there would be a scheduling function at both the shop and cell levels. At the shop level, the cell would be regarded as a process from a material handling point of view. Through negotiation with the cell level scheduler, the residence time for the job at the cell would be determined. To implement this task, the cell level scheduler would consider the material handling constraints within the cell and the durations associated with the subprocesses that would occur within the cell. On the other hand, a centralized scheduling approach would schedule all processing tasks considering the constraints imposed by all material handling systems. The adopted approach will likely depend upon the size of the manufacturing system and the resulting number of material handling constraints that must be considered.

An exact mathematical representation of the objectives and constraints for a given production scheduling problem is quite complex. For a generic representation of the PS problem, and a detailed formulation, the reader should consult [29,33]. For a survey of mathematical programming approaches to solving the production scheduling problem, the reader should consult [21,32]. For a summary of some of the recent work in this area the reader is referred to [24].

Although it is usually assumed in the literature, the production scheduler rarely knows the durations that will be used in the various operations. Hence, the stated start and finish times are only estimates of the true start and finish times. As discussed earlier, the PC can often change the durations either to speed up or slow down the process. This implies the potential for negotiation between the production scheduler and the PC. Davis and Jones [14,15] have recently developed a scheduling framework which allows this type of negotiation.

4.0 MODELS FOR PRODUCTION PLANNING

A manufacturing system is driven by two stochastic inputs: materials from vendors and demands from customers. These inputs generate a stochastic output, namely finished goods. The overall objective is to optimally track the input demand with the output of finished products. In the preceding sections, we have addressed models of the processing aspects of this problem. We now discuss issues related to planning. We include planning strategies, aggregate planning, and detailed planning.

4.1 Production Planning Strategies

Today, two complementary production planning strategies are often discussed: push and pull.

4.1.1 Push. The push strategy attempts to push each order through the system just before its due date occurs. The planning begins with this due date and works backward to schedule the times at which each required process must begin and end. The planning also considers the capacity and availability constraints of each process in the manufacturing system as well as the material handling requirements. Constraints pertaining to the availability of input materials must also be considered.

The push strategy is often adopted within the MRP-II (Manufacturing Resource Planning) strategy of production planning and scheduling. MRP-II, however, is itself a methodology for CIM. Typically, an MRP-II implementation would consider business planning, sales planning, production planning, inventory planning, resource allocation, master production scheduling, material requirements planning (MRP), and capacity planning as an ensemble of planning functions. A detailed discussion of each of these functions and their associated mathematical relations could easily constitute another paper. The interested reader is referred to Orlicky [30] and Fox [17] for an overview of these functions.

4.1.2 Pull. An alternative approach, which has found more application in the flow shop environment, is the pull strategy for which the Kanban approach is one potential implementation [25]. This approach begins by defining an ideal state for the manufacturing processes and then releases orders to them so that the ideal state is maintained. For example, one Kanban approach would be to limit the number of orders that have been processed by a given process, but have not yet completed processing at the successor process. In the flow shop environment, this Kanban implementation is particularly useful since it allows a given process to function at maximum capacity whenever bottlenecks do not exist at successor processes, but inhibits processing when excessive processing output is building up in the system. Consequently, an inherent benefit of the Kanban approach is to stabilize the response characteristics of the overall system.

4.1.3 Remarks. The designation of push versus pull strategy is unfortunate as it appears to imply that one or the other can be applied. The fact is that both strategies can be implemented simultaneously. Recently, Just-in-Time (JIT) has become nearly synonymous with the Kanban approach as an attempt to minimize Work-in-Progress (WIP) inventory [10,16,22]. Certainly, push strategies can also address this performance criteria. From a system's point of view, the Kanban approach is most likely to improve overall system performance when the probability that a given process will be critical is nearly constant. In the job shop environment, this may not be the case. For example, it may be desirable to increase the WIP for a given process when the process is likely to become critical for an extended period in the near future. The Kanban approach can constrain this buildup and perhaps eventually limit overall throughput. As demonstrated, minimization of WIP may not be the only performance criteria that must be considered. Ironically, an often cited feature or the Kanban approach is its minimization of scrapped production. This results from the fact that the successor process often serves as a quality check for its predecessors. When processing faults arise in the Kanban environment, they can often be recognized and corrected before excessive WIP is generated. This observation again confirms the reality that in manufacturing systems, there are alternative performance criteria which must be considered.

4.2 Aggregate Production Planner

Aggregate Production Planning (APP) attempts to generate production quotas for individual products or groups of products to be manufactured over an extended planning horizon. In addition, the APP must specify target inventory levels for considered buffers. These decisions are made subject to capacity constraints and both the real and forecasted demand for finished goods.

4.2.1 Problem Formulation. As discussed above, the APP's decision can be based upon individual products or an aggregate product grouping. To minimize the introduction of additional notation, the formulation for the APP's decision will be given in terms of the basic products ρ_m ($m=1,...,M$). Assuming that T planning periods will be considered, the primary decision inputs into the APP are the actual and forecasted demands for each basic product type in each planning period τ, denoted by $d_m(\tau)$ for $m=1,...,M$ and $\tau=t+1,...,t+T$. Let $a_{nm}(\tau)$ represent the anticiapated production capacity for process P_n consumed in the production of one unit of product ρ_m in period τ while $c_n(\tau)$ will represent the anticipated availability of process P_n in period τ ($\tau=t+1,...,t+T$). Finally, letting $\rho_m(\tau)$ represent the planned production of product ρ_m in period τ, the basic production capacity constraint is given as

$$\sum_{m=1}^{M} a_{nm}(\tau)\, \rho_m(\tau) \leq c_n(\tau) \text{ for } n=1,...,N \text{ and } \tau=t+1,...,t+T \qquad (14)$$

Note this formulation assumes that the production quota $\rho_m(t)$ for the current period t has already been submitted to the Detailed Production Planner (see below) for implementation. Also in the more general case, we might desire to differentiate between regular and overtime production capacity.

We next consider a set of balance constraints for material inventory. Let $I_m(\tau)$ be the level of planned inventory for product ρ_m in planning period τ while $B_m(\tau)$ represents any backorder for product ρ_m that results from production planning for period τ. The resulting inventory constraints have the form

$$\rho_m(\tau) + I_m(\tau-1) - B_m(\tau-1) - I_m(\tau) + B_m(\tau) = d_m(\tau) \qquad (15)$$

for $m=1,...,M$ and $\tau=t+1,...,t+T$. In addition, the production planner can impose production smoothing limiting the fluctuations in $\rho_m(\tau)$ from period to period as well as inventory smoothing constraints which limit the fluctuations in inventory. An additional set of constraints, which must be considered but are difficult to include due to their product specificity, are the constraints dealing with the availability of input materials for each product type. Finally, there may be constraints in addition (15) relating the production in a given period $\rho_m(\tau)$ to the demand in that period $d_m(\tau)$ and the inventory and backorders from the previous period(s).

The APP will typically have a collection of objective functions to be optimized. Let

$$\Phi^{\ell}[\{\rho_m(\tau), I_m(\tau), B_m(\tau), d_m(\tau) \mid \tau=t+1,...,t+T; m=1,...,M\}] \quad \text{for } \ell=1,...,L''$$

represent the collection of objectives to be optimized subject to the above constraints. We also assume that the utility function $\Omega[\Phi^1(\cdot),...,\Phi^{L''}(\cdot)]$ has been defined to represent the comprise among those objectives. This is a stochastic optimization problem since the demands $d_m(\tau)$, the process consumptions $a_{nm}(\tau)$, and the process availabilities $c_n(\tau)$ are never known with certainty.

4.2.2 Potential Problems. The inclusion of these nondeterministic (or stochastic) elements of this strategic problem is a major concern. The resulting decision must be robust in nature, considering all possible contingencies. As one approach to develop deterministic approximations to stochastic optimizaions, Charnes and Cooper [11,12] introduced chance-constrained optimization. Several shortcomings exist in the adoption of this approach. First, the inclusion of the chance-constrained elements significantly complicates the structure of the problem. For example, linear constraints can become nonlinear when the probabilistic definition of the constraint is made. To develop an appreciation for the complexity that results when deterministic approximations are developed for stochastic formulations of the APP problem, the reader is referred to Bitran and Yanasse [8]. Second, the decision optimizes only the expected values, foregoing an extensive contingency analysis. This fact is particularly troubling since the APP's decision is often strategic in nature. Finally, the approach currently considers a single objective only.

Davis and West [13] recently merged the approaches of Monte Carlo simulation and mathematical programming to develop a method for strategic project scheduling. Using Monte Carlo simulation, one thousand potential linear programs were generated and solved. Subsequently, an empirical probability density function for the optimal solution was developed which could be employed in contingency analysis. Although their approach was computationally expensive, the comparison of the method to other stochastic decision-making approaches provided potential avenues for simplifying the approach. The need for a contingency analysis was clearly defined. To date, the authors are unaware of any reported investigations of chance-constrained, multi-criteria optimization approaches.

4.2.3 Feedback and Updates. Although the APP plans for the production periods t+1 through t+T only the product quotas $\rho_m(t+1)$ for $m=1,\ldots,M$ are implemented. During the next planning period, feedback indicating the actual production from the previous period is considered to update resulting inventory and backorder information, and a new $\rho_m(t+1)$ will be generated. Thus, the APP must respond to the deviations between actual and planned production quotas. Furthermore, the APP must continually update its planning as forecasted demand becomes realized with actual booked orders. However, the time scale upon which this updating must proceed is on the order of a week or more. Finally, the APP must continually learn processing parameters through the acquisition of real production data that arises from its previous planning.

4.3 Detailed Production Planner

The Detailed Production Planner (DPP) considers the specified production quotas over a shorter horizon, and issues a request that specific products be produced along with their associated due dates for finished production. Before the request for a given product is issued, the DPP ensures with some probability that all essential inputs for the product will be available when required during production and that the essential processes to manufacture the product will be available. The DPP also attempts to execute all the inventory policies established by achieving the preplanned target inventory levels. A variety of models exist to implement the detailed planning function, and our discussion will summarize three which illustrate a dichotomy of approaches.

Bitran et al. [6,7] have developed a three-level model to implement the aggregate/detailed production planning functions. The basis of their approach is to develop an aggregation of individual product items into production families and then into production types. Their aggregate production planning problem considers the aggregate production cost subjected to a similar set of constraints as discussed above except the decision is defined with production types as the decision variables. The second level of their hierarchy represents a disaggregation of the top level's decision over a shorter horizon with production families serving as decision variables. At this level, setups are considered. The third level disaggregates production families into individual production items. In their approach, each of the two lower levels' problems is posed as a knapsack problem within the constrained specifications arising from the solution of the problem above it. In this manner, the disaggregation approach becomes a single pass algorithm with no formal mechanism for negotiation and feedback among the hierarchical levels.

Axsatter and Jonsson [5] have developed an alternative disaggregation approach. Axsatter [3,4] first developed a methology for aggregating both processes and products. In his aggregation scheme, the errors arising from the aggregation are quantified and procedures which minimize this error are defined to generate an optimal approximate aggregation. Axsatter and Jonnson then developed a detailed recursive relationship relating production of products and parts in a given period to inventory and production in subsequent periods. With this recursive relationship and the approximate aggregation scheme, aggregate recursive relationships are then derived. A solution to the aggregate relationships is then defined to provide an aggregate control strategy. Using MRP principles, this aggregate production control strategy is disaggregated into a master production schedule for the production of individual parts and products.

Gershwin [20] developed his aggregation scheme based upon the dynamics of the manufacturing system. Specifically, he argued that dynamic responses can be decomposed into slowly varying and rapid components. He then states, as has been demonstrated here, that the dynamic response arises from activities which are characterized by their initial and terminal events. By looking at the frequency at which the initial events for a given type activity occur, he argues that these frequencies for different event types will naturally cluster into groups. That is, certain events will occur at a frequency which is significantly greater than the characteristic frequency of the next group, which in turn is significantly greater than that of the next group. The structure of the disaggregation is to schedule the least frequent event group first at the highest hierarchical level. The decisions at this level then specify a constant system response for the next level which in turn schedules the next most frequent events. Finally, the most frequent events are scheduled at the lowest level.

As demonstrated, there are a variety of disaggregation schemes for transferring an aggregate production schedule into a detailed job list for production scheduling. In all cases, capacity planning was considered. All approaches also assume that the lower level will consider a time scale that is shorter than the considered time scale of the decision above it, and therefore, considered events will occur more frequently. Both Bitran et al. [7] and Axsater and Jonnson [5] have discussed the relationship of their approach to the MRP functions cited earlier. Still in viewing this literature, a universal modeling approach and a functional definition for detailed production planning has not yet emerged.

5. SUMMARY

This paper discusses a collection of mathematical models of various decisions which impact the design and implementation a CIM system. We have included objectives, constraints, decisions, and control strategies inherent in every CIM system. Although this list is incomplete, hopefully it will provide a contribution to the automation and integration problems being addressed in CIM. The models can be used to guide the development of the technology needed for automation. They also provide the inputs, outputs, and interrelations needed for integration, regardless of the specific CIM architecture used.

We plan to continue this modeling work. We are particularly interested in data management and communications and the impact they have on the models described above. For example, timing requirements on data may introduce constraints into the decision-making similar to the material handling constraints considered in scheduling. In addition, we plan to investigate CIM architecture questions in more detail.

REFERENCES

[1] Astrom, K.J., "Theory and Applications of Self-Tuning Regulators," **Automatica**, Vol. 13, 1977, pp.457-476.

[2] Astrom, K.J. and Wittenmark, B., **Computer Controlled Systems: Theory and Design**, Prentice-Hall, Englewood Cliffs, New Jersey, 1984.

[3] Axsater, S. "On the Design of the Aggregate Model in a Hierarchical Production Planning System," **Engineering and Process Economics**, Vol. 4, 1979, pp. 89-97.

[4] Axsater, S. "Aggregation of Product Data for Hierarchical Production Planning, **Operations Research**, Vol. 29, No. 4, 1981, pp. 744-756.

[5] Axsater, S. and Jonsson, H., "Aggregation and Disaggregation in Hierarchical Production Planning," **European Journal of Operations Research**, Vol. 17, No. 3, pp. 338-350.

[6] Bitran, G. R., Haas, E. A. and Hax, A. C., "Hierarchical Production Planning: A Single Stage System," **Operations Research**, Vol. 29, No.4, 1981, pp. 717-743.

[7] Bitran, G. R., Haas, E. A., and Hax, A. C. "Hierarchical Production Planning: A Two Stage System," **Operations Research**, Vol. 30, No. 2, 1982, pp. 232-251.

[8] Bitran, G. R. and Yanasse, H.H.,"Deterministic Approximations to Stochastic Production Problems," **Operations Research**, Vol. 32, No. 5, pp. 699-1018.

[9] Blattner, J.L., Mika, L.M., Tanski, J.A. and Schiller, T.L., **A Blast Furnace Dynamic Model--Reconstruction of the PLAIC Blast Furnace Model (Volumes I,II and III)**, PLAIC Report Number 149, Purdue Laboratory for Applied Industrial Control, Purdue University, W. Lafayette, Indiana, 1986.

[10] Buzacott, J.A., 1968, "Prediction of the Efficiency of Production Systems without Internal Storage," **International Journal of Production Research**, Vol. 9, 1968, pp. 173-188.

[11] Charnes, A. and Cooper, W., "Chance-Constrained Programming," **Management Science**, Vol. 16, No. 1, 1969, pp. 73-79.

[12] Charnes, A. and Cooper, W. W., "Response to 'Decision Problems Under Risk and Chance Constrained Programming: Dilemmas in the Transition'," **Management Science**, Vol. 29, No, 6, 1983, pp. 750-753.

[13] Davis, W.J. and West, S.M., "An Integrated Approach to Stochastic Decision-Making: A Project Scheduling Example," **IEEE Trans. on Systems, Man, and Cybernetics,"** Vol. SMC-17, No. 2, 1987, pp. 199-209.

[14] Davis, W.J., "Evolving Apllications for Artificial Intelligence in Real-Time Production Scheduling," **Proc. of the ASME Computers in Engineering Conf.**, San Frnacisco, California, Aug. 1-4, 1988, pp. 215-224.

[15] Davis, W. J. and Jones, A. T., "A Real-time Production Scheduler for a Stochastic Manufacturing Environment", **International Journal of Computer Integrated Manufacturing**, Vol. 1, No. 2, 101-112, 1988.

[16] Finch, B.J. and Cox, J.F., "An Examination of Just-In-Time Management for the Small Manufacturer: with an Illustration," **International Journal of Production Research**, Vol. 24, 1986, pp. 329-342.

[17] Fox, K.A., "MRP-II Providing a Natural 'Hub' For Computer-Integrated Manufacturing,' **Industrial Engineering**, Vol. 16, No. 10, 1984, pp. 44-50.

[18] Galbraith, J.K., **Organizational Design**, Addison-Wesley, Menlo Park, California, 1977.

[19] Geoffrion, A.M., "Elements of Large-Scale Mathematical Programming, Parts I and II," **Management Science**, Vol. 16, No. 11, 1970, pp. 652-691.

[20] Gerswhin, S.B., **A Hierarchical Framework for Discrete Event Scheduling in Manufacturing Systems**, Technical Paper LIDS-P-1682, Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, Massachusetts, 1987.

[21] Graves, S, C., "A Review of Production Scheduling," **Operations Research**, Vol. 29, 1982, pp. 646-675.

[22] Hall, R. W., **Zero Inventories**, Dow Jones-Irwin Co., Homewood, Illinois, 1983.

[23] Ham, I., Hitami, K. and Yoshida, T., **Group Technology Applications in Production Management**, Kluwen Publishing Company, Boston, Massachuttes, 1985.

[24] Jackson, R. H. and Jones, A. T., (eds.), **Proceedings of the Symposium on Real-time Optimization in Automated Manufacturing Facilities**, National Bureau of Standards Special Publication 724, Gaithersburg, Maryland, 1986.

[25] Jackson, R. H. and Jones, A. T., "An Architecture for Decision Making in the Factory of the Future", **Interfaces**, Vol. 17, No. 6, 1987, pp. 15-28.

[26] Jones, A.T. and Whitt, N., ed., **Proceedings on Factory Standards Model Conference**, National Bureau of Standards, Gaithersburg, Maryland, November 20-21, 1985.

[27] Lasdon, L.S., **Optimization Theory for Large Systems**, Macmillan, New York, 1970.

[28] Libosvar, C. M., "Hierarchies in Production Management and Control", MIT Technical Report, LIDS-P-1734, January, 1988.

[29] McPherson, R. F. and White, K. P., "A Management Control Approach to the Manufacturing Planning and Scheduling Problem", **Proceedings of the Symposium on Real-time Optimization in Automated Manufacturing Facilities**, National Bureau of Standards Special Publication 724, Gaithersburg, Maryland, 1986, pp. 145-166.

[30] Orlicky, J., **Materials Requirement Planning**, Mc-Graw Hill, New York, 1975.

[31] Papapanagiotou, I., Maliotis, G. and Koivo, A.J., **A Self-Tuning Controller for Rolling Mill Processes**, Report No. 141, Purdue Laboratory for Applied Industrial Control, Purdue University, W. Lafayette, Indiana, 1984.

[32] Raman N., "A Survey of the Literature on Production Scheduling as it Pertains to Flexible Manufacturing Systems", National Bureau of Standards Report NBS-GCR-85-499, U.S. Department of Commerce, National Bureau of Standards, Gaithersburg, Maryland, 1985.

[33] Rodammer, F. A. and White, K. P., "Design Requirements for a Real-time Production Scheduling Decision Aid", **Proceedings of the Symposium on Real-time Optimization in Automated Manufacturing Facilities**, National Bureau of Standards Special Publication 724, Gaithersburg, Maryland, 1986, pp. 401-422.

[34] Saleh, A., "Real-time Control of a Flexible Manufacturing Cell", unpublished Ph.D dissertation, Lehigh University, 1988.

[35] Sugimori, Y., Kusunoki, K., Cho, F. and Uchikawa, S., "Toyota Production System and Kanban System: Materialization of Just-in-Time and Respect-for-Human System, **International Journal of Production Research**, Vol. 15, pp. 553-564.

[36] Watanabe, T., "A Model-Based Approach to Adaptive Control Optimization in Milling," Trans. of ASME: Journal of Dynamic Systems, Measurement and Control," Vol. 108, No. 1, 1986, pp. 56-64.

NBS-114A (REV. 2-8C)

U.S. DEPT. OF COMM. BIBLIOGRAPHIC DATA SHEET (See instructions)	1. PUBLICATION OR REPORT NO. NISTIR 88-3872	2. Performing Organ. Report No.	3. Publication Date FEBRUARY 1989

4. TITLE AND SUBTITLE

A Functional Approach to Designing Architectures for CIM

5. AUTHOR(S)

Wayne Davis, Albert Jones

6. PERFORMING ORGANIZATION (If joint or other than NBS, see instructions)

NATIONAL BUREAU OF STANDARDS
U.S. DEPARTMENT OF COMMERCE
GAITHERSBURG, MD 20899

7. Contract/Grant No.

8. Type of Report & Period Covered

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS (Street, City, State, ZIP)

National Institute of Standards & Technology
Bldg. 220 Room A-127
Gaithersburg, MD 20899

10. SUPPLEMENTARY NOTES

☐ Document describes a computer program; SF-185, FIPS Software Summary, is attached.

11. ABSTRACT (A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here)

Developing effective CIM architecture is hampered by automation and integration problems. The key to resolving these problems lies in a better understanding of each manufacturing function and how it is related to other manufacturing functions. Our view is that mathematical models can provide this understanding. This paper presents the results of our initial efforts to develop such models. They can be used to guide the development of the technology needed for automation. They also specify the inputs, outputs, and interrelations needed for integration, regardless of the specific CIM architecture used.

12. KEY WORDS (Six to twelve entries; alphabetical order; capitalize only proper names; and separate key words by semicolons)

automation, computer integrated manufacturing, functional decomposition, hierarchy, mathematical models, process control, production planning, prod. scheduling

13. AVAILABILITY

[X] Unlimited
☐ For Official Distribution. Do Not Release to NTIS
☐ Order From Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402.
[X] Order From National Technical Information Service (NTIS), Springfield, VA. 22161

14. NO. OF PRINTED PAGES

29

15. Price

$12.95