Chapter 19, *Advanced Sensing, Modelling, and Control of Materials Processing*,
Ed. by E. F. Matthys and B. Kushner, TMS 1992.

INTELLIGENT PROCESSING OF MATERIALS: CONTROL MODELS FOR INDUCTION-COUPLED PLASMA DEPOSITION†

James H. Taylor, Paul K. Houpt, L. P. Harris, Hsin-Pang Wang,
GE Corporate R & D, PO Box 8, Schenectady NY 12301;

Edward S. Russell, GE Aircraft Engines,
1000 Western Avenue, Lynn MA 01910

Abstract. The production of advanced engineering materials such as metal-matrix composites via Induction-Coupled Plasma Deposition (ICPD) is still emerging from the laboratory/demonstration phase of development. A key to the commercialization of such materials is the introduction of suitable process control methodologies so that uniformly high-quality materials can be produced with good efficiency. The present plan for accomplishing these goals involves devising an Intelligent Processing of Materials (IPM) approach that replaces the laboratory production mode (which involves custom planning each run and direct control of the process run by human operators) with an automated planning and supervisory control system. The invention of such a system requires the creation, study, and manipulation of process models at several stages in the development and implementation cycle. In particular, this presentation focusses on *modeling for conventional control system design* and on *model-based intelligent control.* Implementation of IPM for this process has just begun - this discussion represents a preliminary plan for system definition and integration.

† This work is supported in part by the Materials Science Division of the Defense Advanced Research Project Agency and the Naval Research Laboratory under contract N00014-90-C-0060.

1. Introduction

Induction-Coupled Plasma Deposition (ICPD) has been investigated recently as a means of producing advanced engineering materials such as metal-matrix composites (MMCs, [1]). This work is motivated by the fact that such materials may be lighter, stiffer and stronger in comparison with more conventional materials, and thus may be key enabling technologies for next-generation aircraft engines, aerospace vehicles, and other high-technology products where weight, size and strength are critically important.

The feasibility of manufacturing MMC monotapes via ICPD has been demonstrated in the laboratory. It has been shown that these can be consolidated via vacuum hot pressing or hot isostatic pressing to produce panels or other components with superior material properties. The production of MMC materials by existing methods, however, is still too costly, slow and labor-intensive to permit their use in commercial products. These factors provide the impetus for the present effort to prepare the process for commercialization.

The present objective of the DARPA/GE ICPD Project is to develop the software and hardware to implement an *intelligent process control system* [2-8] for the synthesis of MMC monotape via induction-coupled plasma deposition. Such a system should improve:

- product *uniformity,*
- product *quality,* and
- process *efficiency.*

MMC monotape quality and uniformity can be ensured by systematizing the generation of each process run definition and guaranteeing that the ICPD process is well regulated during the run. Efficiency, in terms of labor, time and materials, can be improved by devising a simple operator interface that allows the process engineer to specify the desired attributes of the MMC material to be produced and then initiate an automatically-controlled ICPD process run, and by implementing a planning and learning system that substantially reduces the number of trial runs and discarded tapes needed to achieve desired material attributes. The goal of the resulting IPM system is to achieve substantial improvements in throughput of acceptable MMC materials at the lowest possible cost.

A complete ICPD process run under IPM will involve the following steps:

- specifying the desired MMC monotape **material attributes** (e.g., dimensions, porosity, residual stress limit, fiber damage limit),
- running the **ICPD Planner** to generate the required process "recipe" (e.g., desired fiber pre-heat temperature schedule, schedules for droplet and deposit temperature during the spray phase, etc.) and the corresponding control regimen to regulate the process, based on the desired attributes,
- running the **ICPD Supervisor** to execute the process recipe and implement the conventional control strategy required to produce the MMC monotape material,
- **evaluating** the MMC material so produced to see if the specified attributes have been attained, and
- running the **ICPD Learning System** to update the ICPD Planner data base so that the recipes it generates consistently produce the desired attributes.

The ICPD Planner, Supervisor, and Learning System together form an intelligent controller which will enhance MMC process efficiency, quality, and uniformity by automating the recipe-generation process and by ensuring that the ICPD process is controlled to follow the recipe accurately and repeatably. In addition, the Learning System will be capable of expanding the operating boundaries of the process, by refining the models to encompass more of the processing "envelope".

The remainder of this paper is organized as follows: Section 2 overviews the ICPD process for producing MMC monotapes, Section 3 deals with the IPM system functional description and architecture, Section 4 discusses the roles and methods of modeling in IPM, and we conclude in Section 5 with current ICPD Project status, summary and conclusions.

2. ICPD Process Overview

The ICPD process is shown schematically in Fig. 1. The apparatus consists of a reduced-pressure chamber outfitted with a water-cooled quartz tube, an inductively-coupled radio-frequency (RF) plasma spray gun, and a shaft that can rotate and translate the deposit target (mandrel). The plasma gas is fed in upstream of the RF coil and is energized by the induced electromagnetic field. The feed powder is injected axially into the plasma stream using a water-cooled particle injection probe, which is inserted deep into the induction coil to prevent recirculation of the particles. The relatively large diameter of the quartz tube and slow speed of the plasma gas ensures good containment of the particles in the plasma stream and sufficient dwell time to provide excellent particle melting. As the particles are melted by the plasma, they are propelled toward the target, which is covered with an array of reinforcing fibers. Upon reaching the target, the molten droplets infiltrate the fiber bed and rapidly lose heat as they solidify to form the substrate of an MMC monotape. Rotation and translation of the target are controlled to ensure uniform spray coverage.

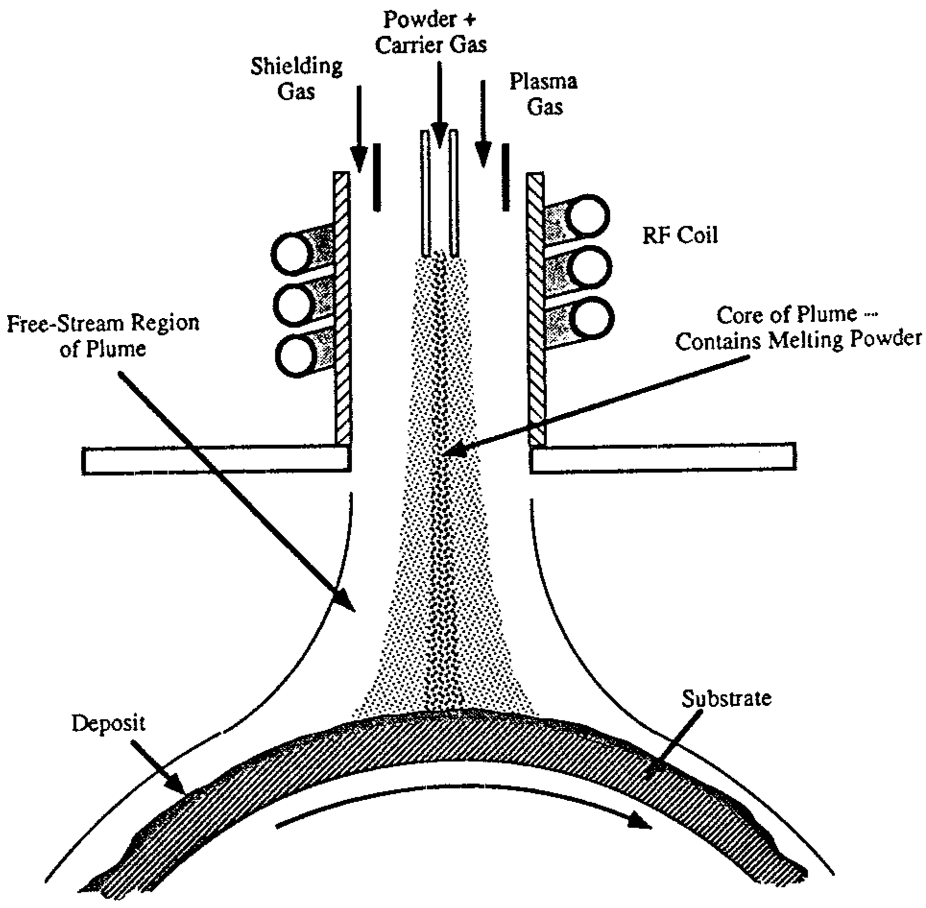

Figure 1. Schematic of Plasma Spray Deposition of MMC Monotapes

The quality of the microstructure produced by the ICPD process is directly related to the processing history. There are several ways by which the properties of the composite can be degraded [3,6]. For example, volatile elements in the molten droplets can evaporate if the temperature environment is too extreme. Temperature differences between the fiber and droplet can produce fiber thermal shock damage and either fiber cracking or spallation of the fiber coatings. Spray deposits often contain porosity in the fiber shadows. The generation of reaction products at the matrix/fiber interface and the production of phase variations in the solidified material can result from thermal exposure during the spray and solidification processes. Residual stresses formed as a result of thermal expansion mismatches among the fiber, matrix and/or mandrel can cause cracking in brittle matrix alloys. Finally, unmelted particles may adhere to the material, causing excessive surface roughness that can lead to difficulties during consolidation. Intelligent control of this complex manufacturing process is needed to reduce these defects and meet stringent quality specifications.

3. IPM System Functional Description and Architecture

For the purposes of intelligent control, ICPD functionality can be partitioned in two dimensions: **time** and **level:**

- in time (sequence) the ICPD process can be broken down into stages or *phases*: initialization, preheat, spray (deposition), post-heat (annealing), and cool-down, and
- in level (hierarchy) the ICPD process is divided into low-level conventional control (for process regulation), higher-level supervisory control (the ICPD Supervisor, to manage the set-point definition and logic involved in conventional control), and intelligent control (the ICPD Planner and Learning System).

Figure 2 shows the breakdown of ICPD into phases and illustrates the definition of a recipe (e.g., substrate temperature varying from T_{si} to T_{sf} according to a specified schedule) and control of substrate temperature by varying the torch plate current. Note that each phase is defined in terms of preconditions, execution, exception-handling, postconditions, and hand-off; the associated control logic is managed by the ICPD Supervisor. Tentative conventional control loops have also been identified for each phase, as shown in this example.

The higher supervisory and intelligent control levels are depicted in greater detail in Fig. 3, including the model-based ICPD Planner that generates the recipe that will achieve the desired MMC attributes, and the ICPD Learning System which either *saves validated plans* for runs where expected and actual results are in close agreement or *"tunes" the Planner models* based on actual results if there are significant discrepancies between predicted and actual behavior. The ICPD Learning System thus works to improve the Planner's ability to predict MMC attributes based on control mode selection and set-point schedule parameters, and broadens the processing envelope that can be accommodated by the IPM system.

4. Roles of Modeling in Intelligent Control

Control in general, and "intelligent control" in particular, must be founded on process knowledge in some sense. There are two distinctly different situations regarding the type of process knowledge that is available and the corresponding strategy for IPM:

- Case 1: process knowledge comprised mainly of experimental runs and operator experience (complete, realistic modeling hard or impossible to achieve) - in this situation, the best course of action may be to emulate a human operator, using an appropriate real-time programming paradigm such as rule-based expert systems or fuzzy logic and an extensive knowledge-capture procedure to codify operator expertise and capabilities.
- Case 2: process knowledge captured primarily in credible models (limited operations experience) - in this case, models can and should play an important role in IPM, as demonstrated below.

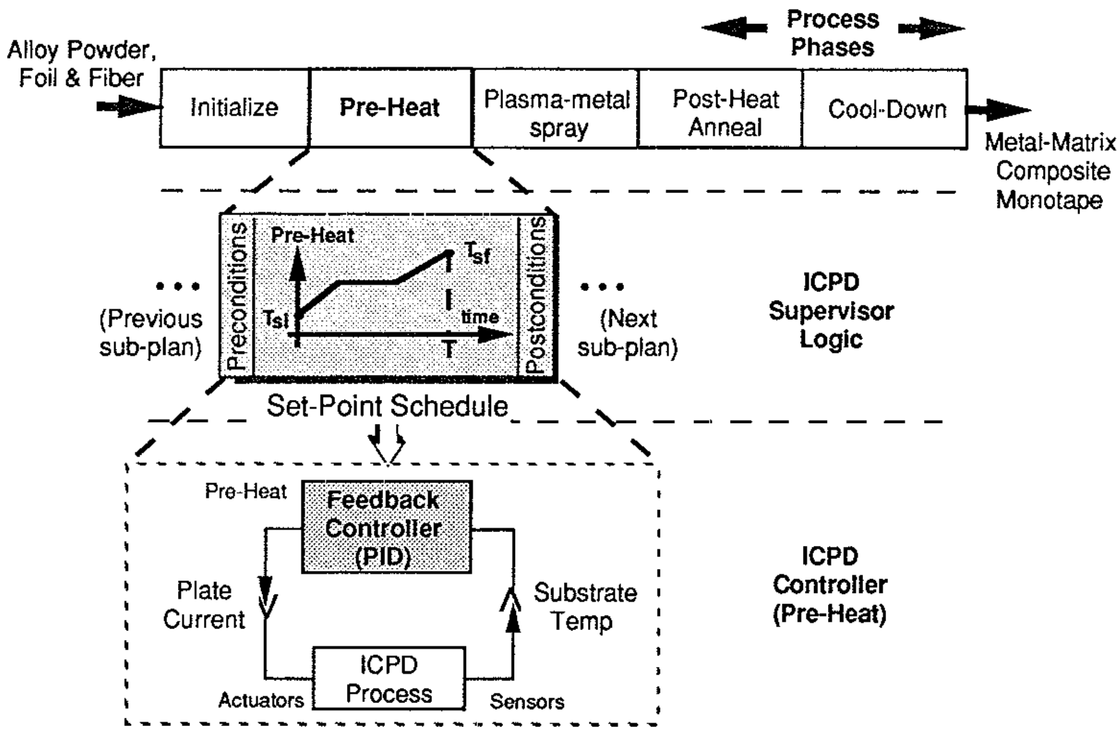

Figure 2. ICPD Phases and Control

In reality, an IPM controls effort usually falls somewhere between these two extremes. In the ICPD Project, extensive physics-based modeling and simulation has been performed in close coordination with experimental studies. Therefore, a great deal of the information needed for IPM has been captured in ICPD models. The decision to invest in modeling was motivated by the need for a system that can be extended beyond present narrow processing experience, and by the belief that physics-based models have the predictive capability required to achieve this goal.

While the emphasis of this presentation is on modeling, it should be mentioned that there are other important issues as well. For example, two critical considerations are sensing and actuation [8]; these are not treated here.

4.1 Overall ICPD Model and Simulation Requirements

ICPD models and simulations are required as the basis for designing and implementing all levels of the ICPD controls hierarchy portrayed in Fig. 3. A high-level breakdown of such a model is portrayed in Fig. 4. The models needed for the design of the conventional control-loop level of the hierarchy and the ICPD Supervisor include **actuator, process,** and **sensor** transient-response models; models for the ICPD Planner include these plus **material attribute models.**

4.1.1 ICPD Actuator, Process and Sensor Models

Actuator, process and sensor models used for controls engineering are generally nonlinear dynamic representations of the behavior of the process, which may be used to generate "time-histories" (records of process variables as they evolve over time in response to varying input signals), by simulation. These are usually "lumped-parameter" models, rather than distributed-parameter, either because the process is inherently lumped or because the process has been modeled using finite-element methods or some other approach that

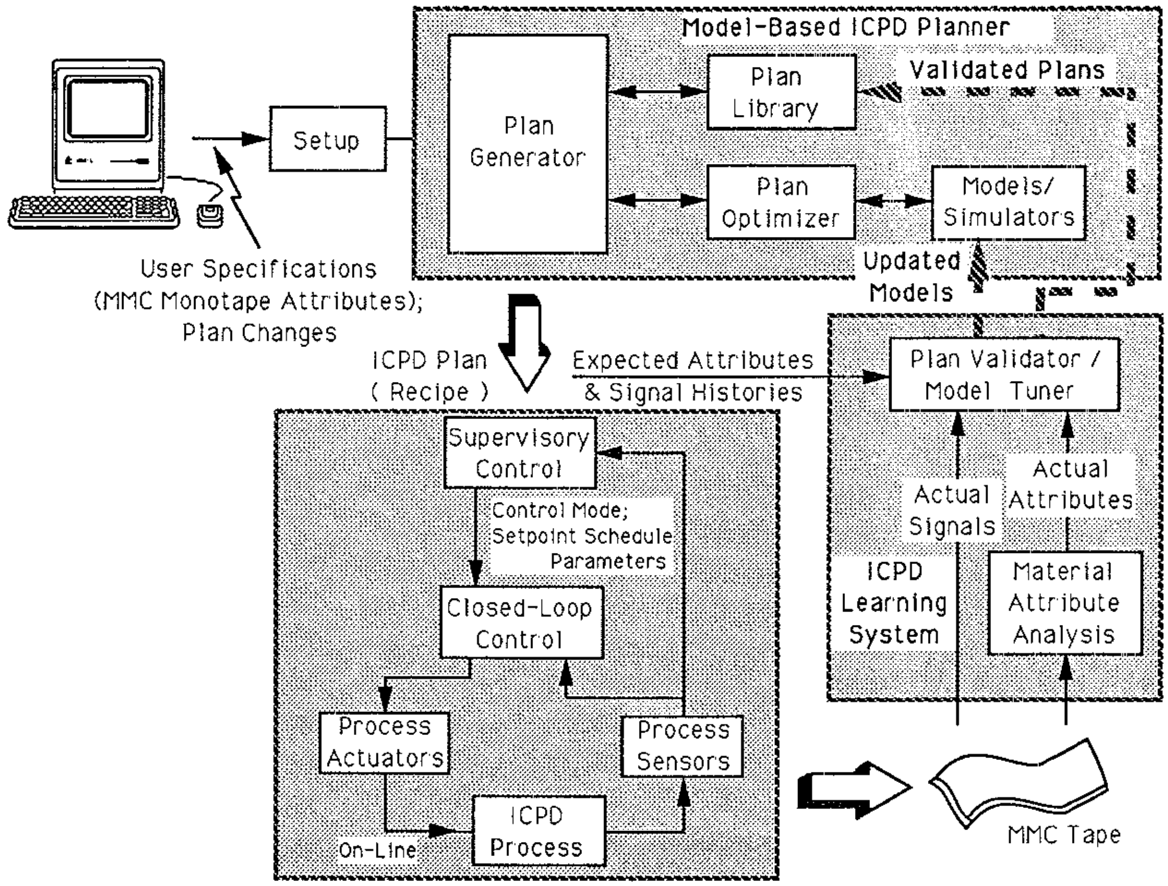

Figure 3. IPM Controls Architecture for ICPD

transforms a distributed-parameter system into lumped-parameter form. A general continuous-time model can then be expressed in the form:

$$\begin{aligned} \dot{x} &= f(x,u)\,, \\ 0 &= g(x,u)\,, \\ y &= h(x,u) \end{aligned} \tag{1}$$

where x represents the "state" of the dynamic portion of the model, u corresponds to process input variables, and y specifies output variables. The state differential equations $\dot{x} = f(x,u)$ govern the evolution of the state over time (e.g., of $T_s(t)$ or substrate temperature as influenced by torch power and other input variables), and the algebraic equations $0 = g(x,u)$ model relations that can be considered to occur instantaneously (e.g., a change in chamber pressure immediately affects the plasma radiation losses). The output equations $y = h(x,u)$ include both measured variables (with sensor scale factors and nonlinearities, if required; for example, a thermocouple reading in volts is a nonlinear function of the temperature being sensed), as well as unmeasurable variables (e.g., particle velocity at impact on the workpiece, which cannot be sensed practically during processing). Note that the model does not have to be structured exactly as in Fig. 4 (with separate blocks for actuators, process, sensors, and attributes) - it is sufficient that the signals labeled $\boxed{B}$ and $\boxed{D}$ be allowed as input variables and that the model outputs include those variables identified as $\boxed{C}$ $\boxed{F}$ $\boxed{E}$ and $\boxed{G}$ in that schematic. A combination of actuators, process and sensors is often called a "plant" model. Highly detailed and realistic models of the ICPD plant have been described elsewhere [3 - 7].

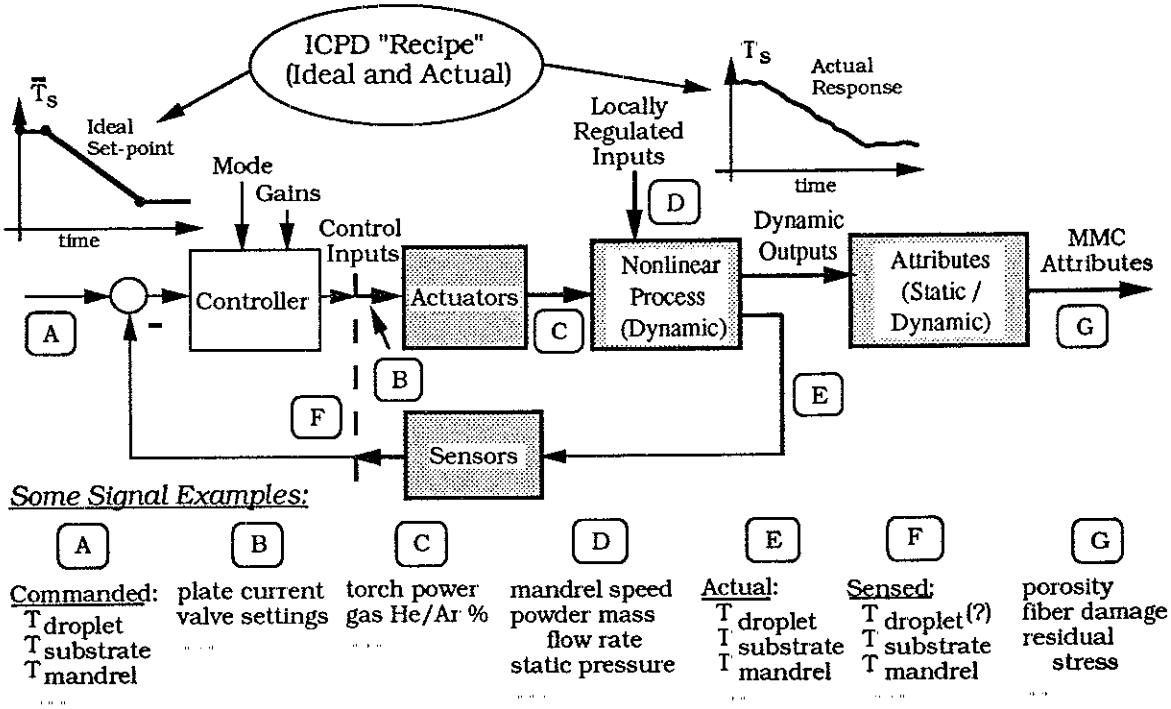

Figure 4. Models Required for IPM of ICPD

4.1.2 ICPD Material Attribute Models

Material attribute models may be dynamic or static characteristics, as appropriate. The requirement is simply that a specified "recipe" can be played through the complete model chain shown in Fig. 4 and the resulting MMC monotape attributes can be predicted with adequate accuracy so the process engineer can study the effects of trying various recipes and, ultimately, the ICPD Planner can iterate to determine the best or most feasible recipe to utilize for the specified run. Attribute models are often nonlinear operators which accept process variable time-histories and detect improper processing conditions. To determine the amount of fiber damage caused by excessive temperature gradients, for example, a material attribute model may monitor the time-history of $\Delta T = T_{droplet} - T_{fiber}$ and produce a measure of excessive ΔT and a corresponding prediction of fiber thermal shock. This modeling effort has also been reported in other publications [6].

4.1.3 Overall Model Integration

The interconnections and interactions of the various parts of the ICPD model are indicated in Fig. 4. A typical list of input and output variables associated with these blocks is provided, to further clarify the nature and scope of these models. In this illustration, a recipe prescribes a desired set-point schedule (command) for substrate temperature ($\bar{T}_s$) in order to achieve specific MMC monotape material attributes. This is converted into a digital signal by specifying $\bar{T}_{s,k} = \bar{T}_s(t_k)$, $k = 1, 2, \ldots$ where t_k denotes the sample time (the command samples in the digital controller might be defined every 0.1 seconds, for example). This signal is input to the control loop where is it compared with sensed substrate temperature $\hat{T}_s(t_k)$, and the controller responds to the corresponding error signal by generating control manipulations of (perhaps) *plate current* and *gas valve settings*. The actuators respond to produce the actual process inputs, i.e., *torch power* and *gas ratio (He/Ar)*, and the nonlinear process variables then respond to these input signals. The output of the process model includes the actual substrate temperature T_s which evolves with time; if good regulation is achieved, it follows the command closely. Finally, the substrate temperature sensor provides a measurement $\hat{T}_s$ which is used to calculate the controller error signal, and the actual signal T_s is fed into the attribute model to be used in conjunction with other process variables for predicting MMC material properties.

4.1.4 Simulation Requirements

From the standpoint of controls engineering, the actuator, process, and sensor models in the control loop in Fig. 4 must:

- be realistic over the entire material processing envelope in which we expect to control the ICPD process;
- model all of the inputs, outputs and process variables that will be involved in its control;
- run interactively in an acceptable time; and
- be linearizable.

Fast, interactive executability is critical, since the design and validation of the control system typically take hundreds of simulation runs. Execution speed is an important factor for the material attribute models, too: simulations that take hours or even many minutes to run will severely hinder exploratory efforts and limit the development of the ICPD Planner (see Section 4.3.1). The ability to obtain linearizations at various operating points is also required, as these will serve as the basis for control architecture and algorithm development (Sections 4.2.6, 4.2.7).

It is usually highly advantageous to have such models interfaced with a general-purpose nonlinear simulator (e.g., SystemBuild [9], SIMULAB [10], SIMNON [11] or ACSL [12]), since setting up and executing simulation runs is then very easy. In addition, such packages can linearize the models, thus providing the needed starting point for control system design. Finally, the controls engineer can model the IPM control system in the same environment, couple it with the actuator, process and sensor models, and validate the IPM controls design by simulation very readily (Section 4.2.8).

The trade-off between realism and speed is a reality of modeling and simulation. This generally leads to the development of models having several levels of detail / realism / speed. These might be typified as:

- ***high-fidelity models,*** based on physics / first principles, carefully validated against experimental runs, and typically very detailed and slow to run;
- ***simplified models,*** derived from or based on the hi-fi model and simplifying assumptions validated by running it, not as accurate but usually much faster;
- ***input / output (I / O) models,*** which may be as simple as first-order "black-box" models obtained by model identification; and
- ***linearized models,*** derived from hi-fi or simplified models by specifying an operating point and taking partial derivatives of model nonlinearities about that point.

All levels of modeling are being used in the ICPD Project. Approaches for generating and validating models of the last three categories are outlined in greater detail in Section 4.2. These secondary models are most frequently used in controls design and implementation; the hi-fi model plays an important role as the "truth model", sparingly used for validating results obtained with other models.

4.2 Modeling and Conventional Controls Design

The following steps characterize a typical approach to the development of a conventional control system given a high-fidelity model of the process to be controlled such as the ICPD Simulator [3 - 7]:

1. exercise the ICPD Simulator to obtain:
   a. a basic understanding of the process dynamics and control requirements,
   b. insight into simplifications that might be made in the model to improve its speed without significantly decreasing its accuracy, and
   c. synthetic input/output data sequences for identifying I/O models if needed;
2. define ICPD process runs needed to test model simplifications and/or to produce real input/output data sequences for generating I/O models; execute these runs;
3. develop simplified physics-based models of the ICPD process, to the extent possible;
4. identify I/O models by processing real and synthetic data logs (obtained as in Step 1 (c) and Step 2), as necessary;
5. compare the real and synthetic data logs and models based on them, to validate the models from Steps 3 and 4 and the hi-fi ICPD Simulator;
6. produce linearized models from the high-fidelity and/or simplified models (I/O models are usually already linear), at a number of operating points; validate their realism for small perturbations about the operating point;
7. perform a preliminary control sensitivity and design exercise based on the validated linearized models to produce low-level control algorithms and supervisory control logic for pre-specified MMC recipes;
8. validate the performance of the preliminary control design by modeling it coupled with the ICPD Simulator and exercising it for a variety of realistic scenarios; iterate if necessary until satisfactory performance is achieved; implement the controller in the actual facility; test and refine as necessary.

These steps are described in greater detail in the sections that follow. Particular emphasis is placed on the creation, validation, and use of nonlinear and linearized models in the process of ICPD control system design. This overall approach is typically very iterative and involves a great deal of "boot-strapping" to achieve the desired results.

4.2.1 Step 1: Understanding the ICPD Simulator

The hi-fi ICPD Simulator is a physics-based model of the ICPD spray process that consists of seven submodels [5,6]: a gas mixture model, a power-supply circuitry model, a detailed free-stream model of the plasma flow in the gun and plume, a central core model for the interaction between the plasma and injected particles, a plume/mandrel heat transfer model, a dimensional and thermal deposit model for the heat flow and mass accumulation on the workpiece, and an attribute model for MMC material quality. Portions of the original detailed hi-fi model were based on finite difference methods and placed a high demand on computer resources. For example, a single static solution of the detailed finite-difference model of the flow and temperature patterns in gun and plume required about an hour of computations on a 10 mips computer. This calculation has to be done repeatedly in simulations of the controlled process as described in Section 4.1.4, making such runs prohibitively time-consuming. This motivated the development of simplified models that are less detailed and accurate but more fast, flexible and understandable. The plan is to use such models in the initial stages of the controls design process and in preliminary predictions of the controller's performance based on adding the controller model to the simplified ICPD Simulator (see Section 4.2.8).

Substantial progress has been made in developing a comprehensive and fast-acting ICPD simplified simulator. Much of this work has been based on the use of physics-based subsystem decoupling and precalculated data bases [7]. Certain additional simplifying assumptions have also been suggested on the basis of detailed ICPD Simulator runs. These approaches and corresponding results are outlined in Section 4.2.3.

4.2.2 Step 2: Defining Experimental Runs

The results from Step 1 have served as the basis for specifying process runs to be executed with the ICPD Facility to produce real input/output data sequences for model validation, simplification, or identification. These data sequences are used to verify that simplifying assumptions are appropriately accurate or for generating I/O models in Step 4, and it is important that they be appropriately defined and logged so that these purposes can be realized. Given the expense of such runs, this model-based approach is highly preferable to cut-and-try experimentation at the facility.

Defining such experiments should include recommendations and specifications for:

- the operating regimes to be run to cover anticipated control operating points,
- the signals to be logged,
- the length of the data sequences in each regime,
- the type of input variation, and
- the sampling interval and the accuracy of the data.

The following sketch illustrates a specification used for I/O model identification:

> *Pre-Heat:* initiate this phase with torch plate current I_p = 9 amps. Allow the environment in the chamber to reach steady state. Then apply a square-wave perturbation to I_p with amplitude 1 amp and period 30 sec.; continue the perturbations for 3 cycles. Then increase the perturbation amplitude to 2 amps, double the period (to 1 min.), and continue for 3 more cycles.
>
> Now ramp $I_p(t)$ from 9 to 12 amps in 30 sec. and hold the plate current at that value for 2 minutes. Then start to apply a square-wave perturbation (etc. . . . similar spec as above). During this run log the signals *{ list of variables }* with sample time 0.2 sec.; the errors in the data due to noise and round-off should not exceed *{ list of rms error bounds }*.

This information should be based on iterative experiments using synthetic data logs from the ICPD Simulator (Step 1) and preliminary real data from the MMC Facility, if the latter can be supplied. Further discussion of these issues as specifically related to model identification is included in Step 4 below.

4.2.3 Step 3: Development of Simplified ICPD Process Models

A fast-acting ICPD simulator is under active development. This effort is based on the use of physics-based subsystem decoupling, precalculated data bases, and the application of certain simplifying assumptions suggested by results from the hi-fi ICPD Simulator.

Physics-based Subsystem Decoupling: The fluid dynamic characteristics inside the gun and plume were studied by exercising the ICPD Simulator, and it was determined that the plasma free-stream region and the central core region where the plasma and particles interact can be decoupled through the use of a boundary-layer core model. Free-stream calculations are done whenever the gas mixture changes; all simulations related to determining the influence of gas ratio changes on plasma and particle interactions can thus be conducted using the fast-acting boundary-layer core model. This significantly reduces the computational burden.

Precalculated Data Bases: Observation of phenomena involved in the various submodels shows that the time constants of variables in the power supply, free-stream flow, central core, and plume/mandrel interactions are very fast compared to the relatively long time constants of the heat flow in the workpiece. This permits modeling everything from the power supply to the heat transfer between the plume and workpiece as steady-state or static characteristics governed by the external inputs and the instantaneous temperature distribution and axial location of the workpiece. The simplified ICPD simulator takes full advantage of this, and thus contains only a single dynamic model, for workpiece heat-flow and mass-accumulation phenomena.

The following illustrates the time-savings that have been achieved in performing free-stream calculations within the static part of the simplified model by the use of precalculated data bases: To implement this approach, a number of representative conditions were chosen, based on hi-fi simulator runs, the time-consuming detailed model calculations were performed off-line, and the results stored as a free-stream data base. This data base or look-up table is then used as the basis for interpolation in solving the static conditions as they vary during simulation exercises. In ranges where such a data base is available, the compute time for determining the simulator response to an adjustment of controlled parameters during a spray operation is presently about 30 seconds on the same 10 mips computer, which is substantially closer to that needed in IPM simulation and design activities.

Through the combination of subsystem decoupling and precalculated data bases, the computation time has been successfully reduced by a factor of more than 100 from the time required for the original detailed finite-difference numerical model. In its present form, this fast-acting simulator provides excellent accuracy in representing the ICPD process. However, to gain a detailed understanding of the effects of system parameter and input variations on the dynamic behavior of the system as outlined in Section 4.1, the 30-second response time of the ICPD simulator is still frustratingly slow in terms of fulfilling the needs of the control system design process and of the ICPD Planner. One way to speed up the response is to migrate the simulator to a faster computer; the other approach is to further simplify the ICPD simulator.

Further Simplifying Assumptions: Simple physics-based submodels for the gun, plasma, particle heating, and plasma/workpiece mass and heat transfer are being investigated as follows:

1. A simplified model for ***heat and mass transfer*** to the cylindrical workpiece can be generated by assuming that:
   - the plasma plume properties are uniform over any cross section near the workpiece;
   - the plume cross section defines the heated area of the workpiece;
   - both mass and heat are deposited uniformly over the heated area;
   - the plasma-workpiece heating is determined by a heat transfer coefficient obtained from a theoretical correlation of Nusselt number, Reynolds number, and Prandtl number derived [13] from established boundary-layer theory; and
   - heat loss from the workpiece is by gray body radiation.
2. A model for ***injected particle heating*** can be developed by assuming that:
   - the relative velocity between plasma and particles is small;
   - the particles are approximately spherical;

- the temperature is uniform within any particle; and
- the particle heat balance involves convective heating and radiation cooling.

The convective heat transfer coefficient from plasma to spherical particle at low relative flow speed can be obtained from the literature [14] or from a laminar heat transfer calculation, and the change in emissivity of particles at melting (due to the smoothing of the particle surface by surface tension forces) can easily be included.

3. A very simple model for *plasma plume expansion* can be based on assuming that:
   - momentum and energy fluxes in the plume are constant;
   - the plume has uniform properties over any cross section;
   - the plume expands with constant cone angle to ingest gas from the tank background; and
   - the plasma gas is ideal with constant specific heat.

   In this model, both the plasma velocity and the plasma temperature vary inversely as the plume area. A radial expansion angle of 0.1 radian gives about a factor of 2 reduction between gun and workpiece in plasma speed and temperature due to plume expansion and mixing.

The preceding submodels appear suited for use in IPM controls design. Unfortunately, less success has been attained thus far in modeling the operation of the plasma gun. A preliminary attempt was based on the assumptions that the electrical discharge fills the gun volume; that the plasma properties are uniform within the gun; and that thermal equilibrium prevailed in all dissociation and ionization processes involving the plasma gases. Calculations with this model give reasonable values for the impedance reflected into the power supply oscillator, but unreasonably low values (5000-6000 °K) for the plasma temperature. Examination of temperature and plasma velocity distributions for plasma guns calculated using the ICPD Simulator suggests that the basic error in this model is the assumption of uniformity in plasma properties and flow velocity across the gun diameter. Those solutions show that most of the gas flow occurs in a relatively cold annulus near the outer wall, and that the electrical dissipation is concentrated in an inner annular region of low (perhaps recirculating) flow, containing temperatures up to about 10000 °K. It should be possible to capture these features in a revised model based on the boundary-layer interaction between an inner cylinder of stagnant high temperature gas and an outer annulus of flowing cold gas. All electrical dissipation would occur in the outer edge of the hot gas in a narrow zone whose radial extent is determined by the electrical skin effect and whose axial extent is limited by boundary-layer mixing between hot and cold gases. Further work on this model may be undertaken in the future, if necessary.

4.2.4 Step 4: I/O Model Identification

Models required for control system analysis and design ("control models") can be obtained from several sources. Where possible, one should use the hi-fi nonlinear ICPD Simulator and linear models based directly on it. If the run-time of the ICPD Simulator is excessive, then simplified physics-based models provide the next best alternative. Finally, if there appear to be areas where these models are not available or may be of questionable accuracy then model identification may be used to fill the gap. For example, it is not clear at this time that the ICPD Simulator accurately characterizes substrate temperature variations due to variations in the plate current and gas ratios, due to uncertainty in boundary-layer conditions and heat transfer coefficients; if so, then model identification may be critical for the successful design of ICPD controls. Also, there may be regimes where the ICPD Simulator nonlinear model is not linearizable (e.g., if there are deadzones or other discontinuous nonlinearities) - here, too, model identification may supply useful approximate models.

The basic concepts of model identification are portrayed in Fig. 5. First, it is usually assumed that the transient-response dynamics of the MMC process at a given operating point can be modeled accurately by a simple linear dynamic model. This is generally a realistic assumption, especially for small variations about an operating point, so in most cases this provides a good basis for control system design. Such a model can be represented in transfer function form. As a simple example,

$$G(s) = \frac{K\, exp(-s\, T_1)}{(1 + s\, T_2)} \tag{2}$$

describes the dynamics of a process with low-frequency gain K, a pure delay (delay time = T_1) and a first-order lag (time constant = T_2). Higher-order models might also be required for certain subsystem dynamics. An operating point is defined by an average or "DC" level of the process inputs, denoted u_0 in Fig. 5. The actual input to the process must vary about this point, as shown by the arbitrary step changes in Fig. 5 (a). The output of the process must show a tendency to follow the input, as depicted in Fig. 5 (b). If this is not so, then there is effectively no dynamic interrelation between the input and output signals under consideration ($G(s) = 0$).

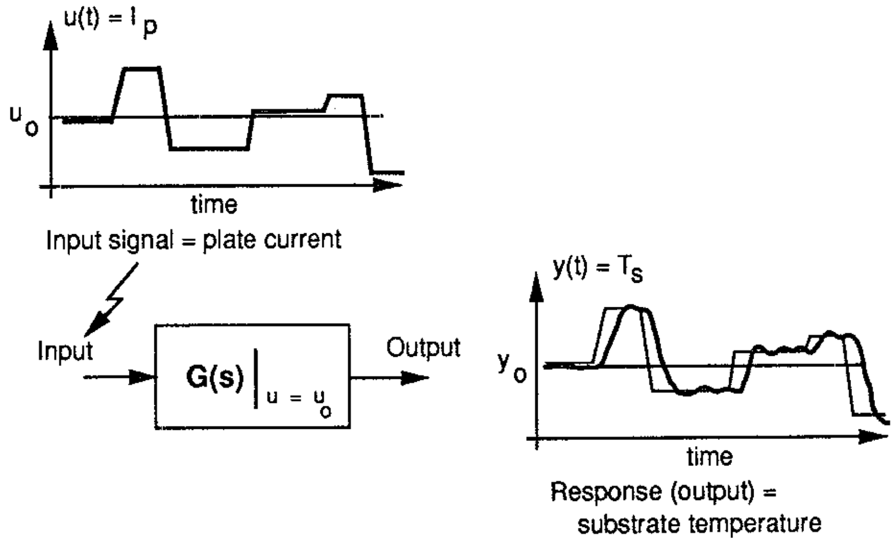

Figure 5. ICPD I/O Model Identification

Real or synthetic data logs as depicted in Fig. 5 (from Steps 1 and 3) are analyzed to identify process I/O models. This procedure involves *parameter estimation,* e.g., determining the values of K, T_1 and T_2 in Eqn. (2) so that the input/output response of the identified model optimally matches that of the real process. Methods such as Least Squares and Maximum Likelihood [15] serve this purpose, and are implemented in algorithmic form in many controls CAD software packages; these can be applied to ICPD data logs to obtain such linear models. The identification of higher-order models may also be pursued, to ensure that accurate models are obtained. Established criteria for determining the optimal model order can be applied (e.g., the Akaiki Information Criterion [16]).

This procedure is illustrated in Fig. 6, where the two time-history records correspond to scaled torch plate current (trace 1) and mandrel thermocouple reading (trace 2). The output data records were detrended (to remove a slow quadratic tendency $a_0 + a_1 t + a_2 t^2$ that was unrelated to the perturbation). Note that these records are far from ideal - the thermocouple data is very low-gain and noisy; yet a good first-order I/O model was obtained. Higher-order model identification was also carried out; the Akaiki Information Criterion indicated that the resulting model was less desirable than the first-order case.

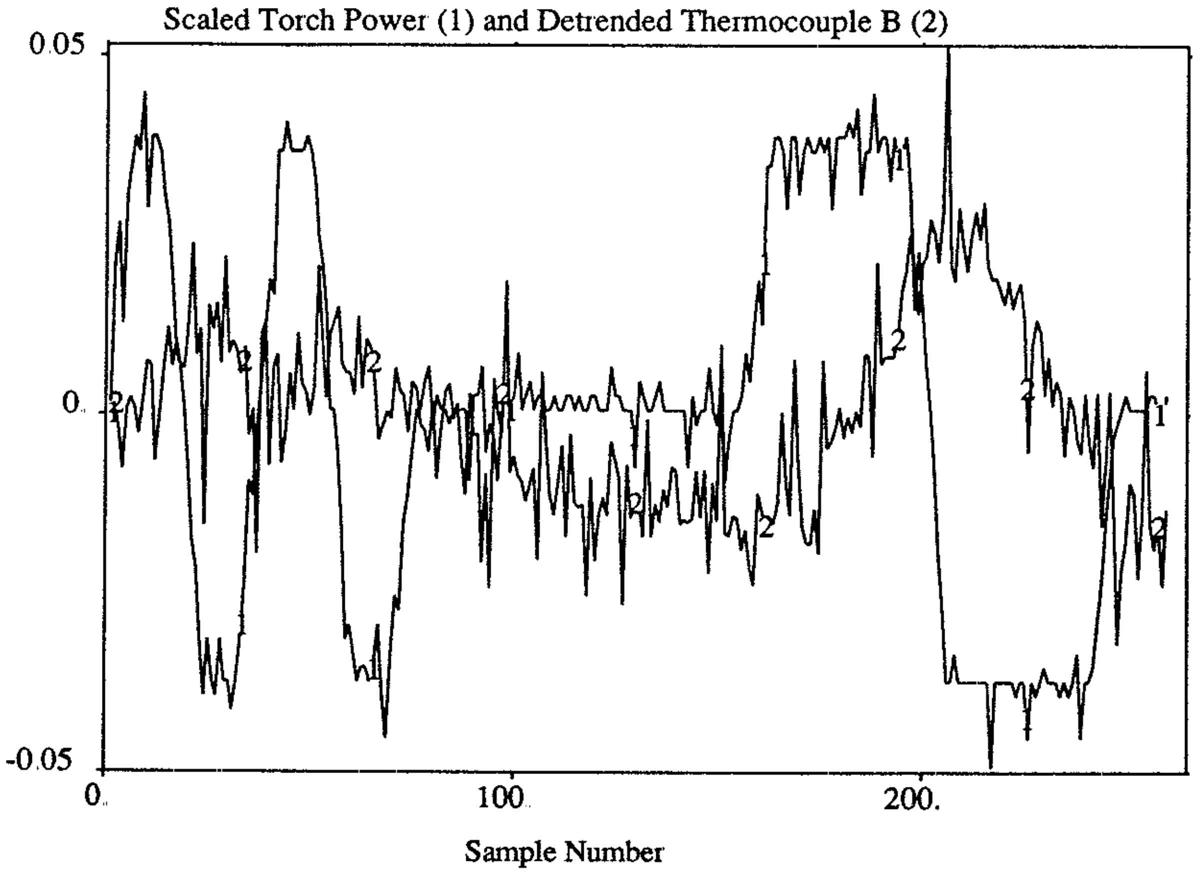

Figure 6. Signals for I/O Model Identification

The input and output signals in Fig. 5 also serve to illustrate some of the issues mentioned in Step 3, namely the definition of suitable experiments for control model identification. The signal $u(t)$ must exhibit transients of the right frequency content, i.e., if the input varies too quickly then the system will not be able to respond, and if the input varies too slowly then the faster transients of the system will not be excited. In either case, the corresponding low- or high-frequency dynamics cannot be identified successfully. The sampling rate must also be matched to the system dynamics - for example, if the data is taken too infrequently, then high-frequency content of the transient response will be obscured.

4.2.5 Step 5: Model Validation

One approach for validating simplified or identified models from Steps 3 and 4 is simulation and cross-verification. In other words, the same input signals that were applied in the process runs (e.g. $u(t)$ = *plate current* as in Fig. 5 (a)) are applied to the secondary model and to the ICPD Simulator. The simulation responses (e.g., $y(t)$ = *substrate temperature* in Fig. 5 (b)) are obtained and inspected to see if the secondary model response and the output of the ICPD Simulator faithfully replicate the behavior of the real process. These comparisons of real and synthetic data logs and models based on them thus serve to validate both the models from Steps 3 and 4 and the ICPD Simulator. If there are any subsystems or signals for which either real process or ICPD Simulator data cannot be obtained, then validation must be based on whatever data is available.

Another technique for validating models is to compare them in terms of frequency response, based on real process runs and on synthetic data from the ICPD Simulator. Such frequency-domain approaches are particularly important for controls design, since "robust designs" (designs that will still work effectively if the models and actual process are not identical [17]) are based on frequency-domain criteria. For the real process and for nonlinear ICPD models, the most reliable way to make this comparison is to obtain the process response to sinusoidal inputs and use Fourier analysis of the output to produce Bode plots of the models' frequency response (gain and phase characteristics); convenient software has been developed to accomplish this for nonlinear simulation models [18]. In the case of linearized models (Eqn. 4, see Step 6) the frequency response is obtained by direct application of transformation methods:

$$G(j\omega) = C_0(j\omega I - A_0)^{-1} B_0 + D_0 \tag{3}$$

and for I/O models as in Eqn. (2) one obtains $G(j\omega)$ simply by substituting $s = j\omega$. However these frequency-domain models are obtained, they can be compared and acceptance criteria might involve bounds on the differences between the magnitude and phase characteristics over the band of frequencies important to the design process.

The two approaches outlined above should be applied judiciously to the various models obtained in Steps 3 and 4. Models considered will only be those required for control system analysis and design (see Fig. 4). The bottom line should be to obtain ***good enough*** models with the minimum expenditure of effort.

4.2.6 Step 6: Model Linearization

A critical step in the control system design process is conventional linearization, also called Taylor series linearization, which can be expressed in terms of Eqn. (1) and an operating point (x_0, u_0) as:

$$\begin{aligned}
\dot{\delta x} &= \frac{\partial f}{\partial x}\,\delta x + \frac{\partial f}{\partial u}\,\delta u \\
&\triangleq A_0\,\delta x + B_0\,\delta u \\
\delta y &= \frac{\partial h}{\partial x}\,\delta x + \frac{\partial h}{\partial u}\,\delta u \\
&\triangleq C_0\,\delta x + D_0\,\delta u
\end{aligned} \tag{4}$$

where δx, δu, δy represent perturbations or "small signal" variations around x_0, u_0, and $y_0 = h(x_0, u_0)$, respectively. (Note that the algebraic relations in Eqn. 1 have been omitted, for notational simplicity.) The arrays $[\partial f / \partial x]$ etc., are evaluated as the partial derivatives at x_0, u_0, and are matrices; the subscript '0' stresses the dependence of the arrays upon the operating point. Such models are extremely important to the control system design task, as the primary task of the lowest-level control algorithms is regulating about a specified operating point where a linearized model is usually realistic. (Linearized models are valid wherever the partial derivatives exist, i.e., wherever the model nonlinearities are smoothly differentiable, and if the excursions from the operating point are sufficiently small. If non-differentiable elements such as relays are present, then one must account for their effect in some other way.)

An advantage of interfacing the ICPD Simulator with SystemBuild, SIMULAB, SIMNON or ACSL is that the matrices A_0, B_0, C_0, D_0 can be evaluated directly by taking the partials *numerically*, e.g., in the scalar case the partial of f with respect to x is obtained by evaluating f for $x \pm \delta$ and taking a central difference:

$$\frac{\partial f}{\partial x} \cong \frac{f(x+\delta) - f(x-\delta)}{2\delta} \tag{5}$$

Most nonlinear simulation packages [9-12] have more or less sophisticated algorithms for linearization, including higher-order approximations and automatic selection of δ to achieve accurate results (see also [19]).

4.2.7 Step 7: Control System Design

The ICPD Simulator and the validated control models from Step 5 will be used for control sensitivity analysis and design in a number of ways. The nonlinear models will be exercised to determine the sensitivity of each important process output variable (each output that must be regulated to achieve acceptable MMC monotapes) to variations in each input that can practically be actuated. In addition to these simulation experiments, linearized models will be extracted and control-theoretic approaches such as relative-gain-array (RGA) analysis [20] will be applied to determine the so-called control architecture, i.e., which inputs are important for MMC attribute control and which loops should be closed to achieve good regulation. Tentative control architectures for the various ICPD process phases have been proposed based on experimental observations of the process; for example, Fig. 7 shows the preliminary control loop for the spray phase. These results need to be validated by linearization and further analysis.

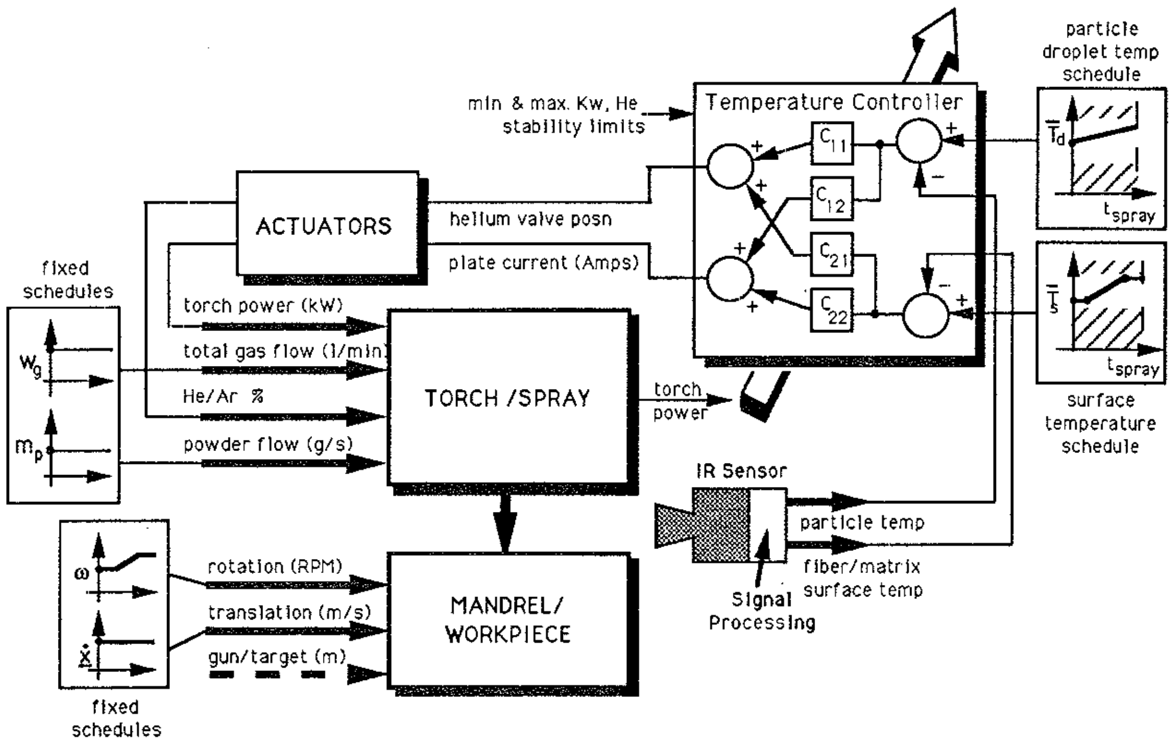

Figure 7. ICPD Control Architecture (Spray Phase)

Several concepts are important to understand in the context of controls development. First, there is the issue of *processibility* or *processing envelope*, which defines the limits that must be imposed on the ICPD process during regulation. For example, there is a rather small "window" in the relation of fiber temperature to droplet temperature where acceptable material attributes can be obtained, as shown in Fig. 8. The IPM control system must ensure that the process recipes and regulation keep the process within this window. Generally, this is done by incorporating *limit protection logic* in the controller.

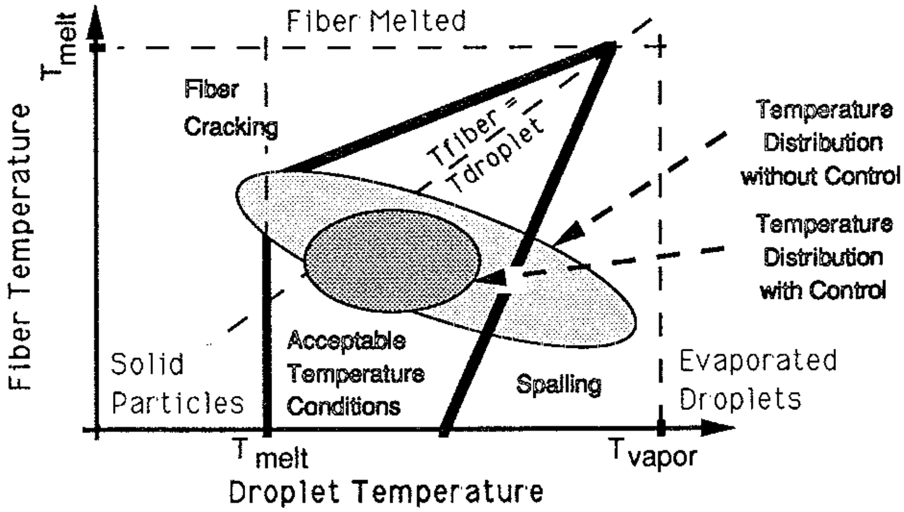

Figure 8. ICPD Processing Envelope

Then, there is the issue of *operating point*. Various "operating points" must be defined in the space of process variables, in recognition of the fact that a nonlinear process has different dynamic behavior depending on the state of the process. For example, it is likely that the effect on substrate temperature due to a small variation in torch plate current will depend on the current itself (e.g., a small variation around 9 amps may produce a different response than the same variation about 12 amps), as well as on the present gas mixture ratios ($H_2/He/Ar$). The degree of nonlinearity, the dimensionality of the process variable space (e.g., I_p vs H_2/Ar vs He/Ar), and the granularity (number of points in the space), of course, has to be determined by detailed study using the ICPD Simulator and/or experimental process runs. The controls schematic in Fig. 7 shows (hypothetically) that the control algorithm for temperature control may have to be parameterized according to the present value of torch power.

Finally, the relative interaction time-scales will be important for specifying the control architecture and algorithms. For example, it is known that the torch dynamics are very fast (so a step-change in plate current results in a rapid change in plasma and droplet temperature), the fiber temperature reacts a little more slowly, and the substrate temperature is much slower. This has important implications on the controller requirements for regulating the process so as to remain in the processing window shown in Fig. 8; these factors can only be quantified by linearization and linear analysis as outlined above.

These factors greatly influence process modeling and simulation for IPM. It is clear that controls engineers require ICPD models that are fast-running, so that exploring the dynamic behavior of the ICPD process over the processing envelope, determining process sensitivities, linearizing or generating synthetic data sequences for I/O model identification, and validating candidate control algorithms can be done in a reasonable time.

Conventional control algorithms are obtained for each loop closure at various operating points, governed by the factors outlined above. There are several possible outcomes in this stage of control system design: It may be that linear control can be used with limit protection logic, it might be possible to synthesize a simple nonlinear controller based on insight into the properties on the process, or it might be necessary to utilize a "gain-scheduled" controller in which supervisory control logic is developed for switching among various control algorithms as the operating conditions change over the duration of a process run. In any case, such a design is done for pre-specified MMC recipes, and the form of the control algorithm and supervisory control logic specifications will be difference equations suitable for direct implementation in the ICPD digital control system using the ISI platform AC-100 [21]; the use of SystemBuild for Step 8 (below) will ensure that implementation will be an automatic translation and download to the AC-100 hardware.

4.2.8 Step 8: Control System Simulation Validation

The performance of the control system design from Step 7 will be validated by modeling the control algorithms and ICPD Supervisor in SystemBuild in conjunction with the ICPD Simulator and exercising the resulting ICPD control system simulation for a variety of realistic scenarios (a variety of pre-specified MMC process recipes). The Step 7/Step 8 process is iterated if necessary until satisfactory performance is achieved. Once a validated design for the conventional controller is obtained, it can be transitioned for implementation on the actual process.

4.3 Model-Based Intelligent Control

The "brain" of the IPM environment depicted in Fig. 3 is comprised of the ICPD Planner and Learning System modules. The Planner will operate to generate a process "recipe" via logic and model-based optimization, and the Learning System maintains and refines the data base (models and validated plans) used by the Planner. More specifically, the IPM framework portrayed in Fig. 3 includes the following submodules:

- Setup - translates the "easy" MMC monotape attributes into run parameters, e.g., monotape dimensions into mandrel dimensions;
- Plan Generator - controls the iteration of the MMC recipe generation system; uses the Plan Library to obtain a preliminary plan, and the Plan Optimizer to converge on the final plan;
- Plan Library - a growing repository of real and synthetic ICPD process runs that serves as the basis for the initial guess and subsequent iterations to determine the process recipe for the present run;
- Models/Simulators module - contains the actuator, process, sensor and attribute models which are run for the current candidate recipe to generate the expected signals (time-histories of the process variables) and MMC material attributes;
- Plan Optimizer - uses the gradient of the material attributes with respect to plan parameters to generate a new parameter set; and
- Plan Validator and Model "Tuner" - compares the expected and actual MMC material attributes and process signals (time-histories) to either validate the plan that was just executed and store it in the Plan Library for future use, or update the models/simulators.

The objectives of these modules are to systematize the operation of the ICPD process, to improve the consistency and quality of MMC so produced, and to continually expand the IPM system's processing "envelope" by building a growing data base of successful MMC processing plans and improving the validity and predictive capability of the models. The details of the design and implementation of these modules are very preliminary at the present time; the results of the conventional controls design activity will be essential in fleshing out the ideas presented below.

4.3.1 ICPD Planner Definition

The ICPD Planner overviewed in Fig. 3 involves several types of knowledge and logic:

1. heuristic / experience-based translation of material attributes to process variables (e.g., MMC monotape thickness may govern some or all of the following: the amount and rate of metal powder feed, mandrel speeds (stroke and rotation), and the duration of the spray phase);
2. locating the plan from the data base that most nearly matches the present material attribute specification;
3. model-based plan optimization (if necessary and possible), to determine an optimal or feasible MMC recipe; and
4. heuristic / experience-based scrutiny of the final proposed plan to screen out recipes that are unlikely to yield good results.

These steps are shown schematically in Fig. 9. Step 1 is executed in the 'Setup' module; this provides the translation of user specifications or 'Desired MMC Attributes' (DMA) into the information needed to invoke the 'Plan Matcher'. The Matcher searches the plans in the data base to find the closest fit to the input specification, i.e., to find a plan ***Pk*** so that the 'Actual MMC Attributes' (AMA) are nearly equal to the current DMA. If a plan is found that produced material that is sufficiently close to the specifications (DMA = AMA within specified tolerances), then that plan is executed; otherwise the closest plan is extracted for further consideration:

- if the closest plan is an OPTIMIZABLE_PLAN, then model-based optimization is used to refine it; the result is displayed to the operator for approval or modification;
- otherwise the closest plan is displayed for possible use or modification by the operator.

In either case, the operator is given the opportunity to preview and accept / modify / reject the plan. Plans that have been adjusted, whether by optimization or operator intervention, are given a final check to see if there are any obvious flaws, e.g., the plan may nearly match a recipe that failed in the past or violate some other known limit or condition.

Note that every plan in the data base will be categorized in such a way that the above judgments are simply made (see Fig. 10 and Section 4.3.2). For example, we have an OPTIMIZABLE_PLAN if the material attribute models were accurate enough so that the predicted and actual material attributes (PMA and AMA) are in good agreement (attribute 'z' = 1) and if the process model was accurate enough so that the predicted and actual process variable time-histories (PTH and ATH) are in good agreement ('y' = 1). If so, Step 3 of the above plan-generation process is executed. This involves the use of models, and is thus the most germane to this presentation.

The present scheme for model-based plan optimization is to generate a preliminary plan by Steps 1 and 2 above, and then exercise the models in the Planner data base to determine adjustments required to make the plan feasible or optimal in some sense. Feasibility involves finding a recipe that is predicted to achieve the present objective (PMA = DMA), while an "optimal" plan might additionally accomplish this in minimum time, for example. This will be done with a standard optimization approach, i.e., perturb the recipe, run the models to obtain material attribute sensitivities, and adjust the plan accordingly.

A number of factors must be addressed in order for this approach to be practical:

- the "plan space" must be made as small as possible,
- the material attributes must vary smoothly over the plan space, and
- the Planner models must execute very quickly.

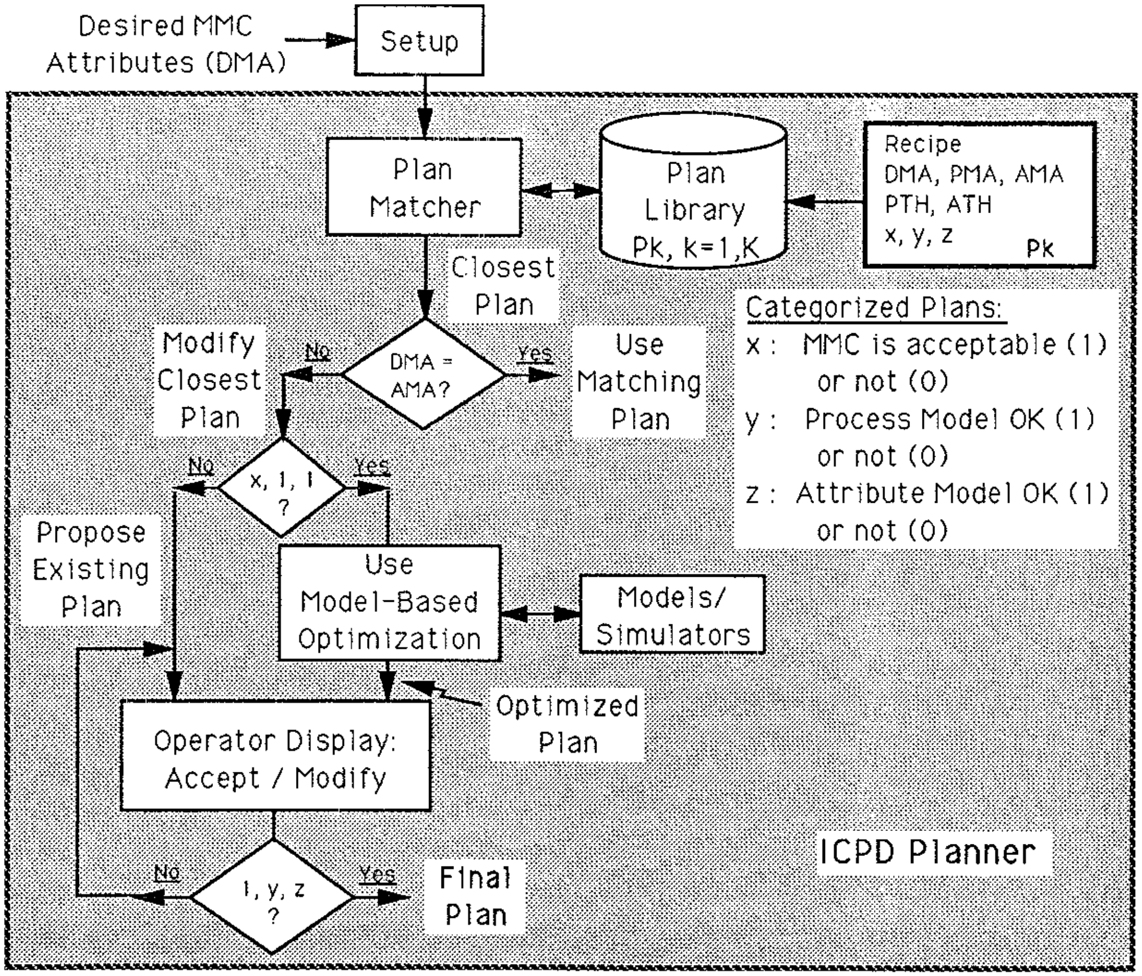

Figure 9. Model-Based ICPD Planner

The plan space dimensionality will critically determine the magnitude of the Planner's task. Optimization over a very large number of parameters will be prohibitive even if the simulators require little computational effort. For this reason, the number of material attributes being optimized must be modest, and an ICPD process recipe must be defined by as few parameters as possible. The most efficient representation of a recipe is to specify each control set-point variable by a table of time-points and values. This is illustrated in the recipe "icons" in Fig. 7; see, for example, the "surface temperature schedule" block. The number of time-points will have to be kept to a minimum; at controller sample times between these points the digital controller (ICPD Supervisor) will interpolate to generate the actual required current set-point. Based on current process understanding, it is expected that a typical recipe will be parameterized by fifty to 100 tabular values.

The characteristics of the material-attribute surfaces in the recipe parameter-space will be important to the speed and convergence of the Plan Optimizer. If the conventional control algorithms are properly designed, this should not be a problem. The sensitivity and control architecture studies mentioned in Section 4.2.7 are conducted precisely to determine the best process inputs to vary in order to achieve the control objectives, namely produce an MMC monotape with the desired material attributes. With proper scaling and cross-coupling in the control loops, the optimization problem should be easy to solve numerically.

Finally, the execution time of Planner simulators must clearly be very short. This will require the use of ICPD process models that are as simple as possible. This is also a pressing requirement for the success of the Learning System, so we will discuss this issue in more detail below.

4.3.2 *ICPD Learning System Definition*

The basic functions of the Learning System are to ***maintain the plans*** and ***improve the models*** in the data base used by the ICPD Planner. The first task involves categorizing plans and adding them to the data base once it has been determined whether or not the results of a run were satisfactory and whether or not there were significant disparities between the Planner's predictions and the actual results. If there are discrepancies, then the second task will entail using automatic and/or manual methods to refine the simulator models. In this way, it should be possible to expand the working envelope of the overall IPM system as different material attribute specifications are proposed and trial recipes executed. This is not to say that one should expect miracles, e.g., that new powders or fibers could be accommodated without trial and error manual learning to incorporate new process knowledge in the data base. It should be possible, however, to produce monotapes with incrementally better material attributes, to produce a monotape that is a little thicker than any previously obtained, or to reduce processing time or raw material consumption, for example.

Classification of plans and refinement of models in the ICPD data base should be a straightforward task. The logic for this is depicted in Fig. 10. There are three characteristics to consider, as mentioned in the Planner discussion above:

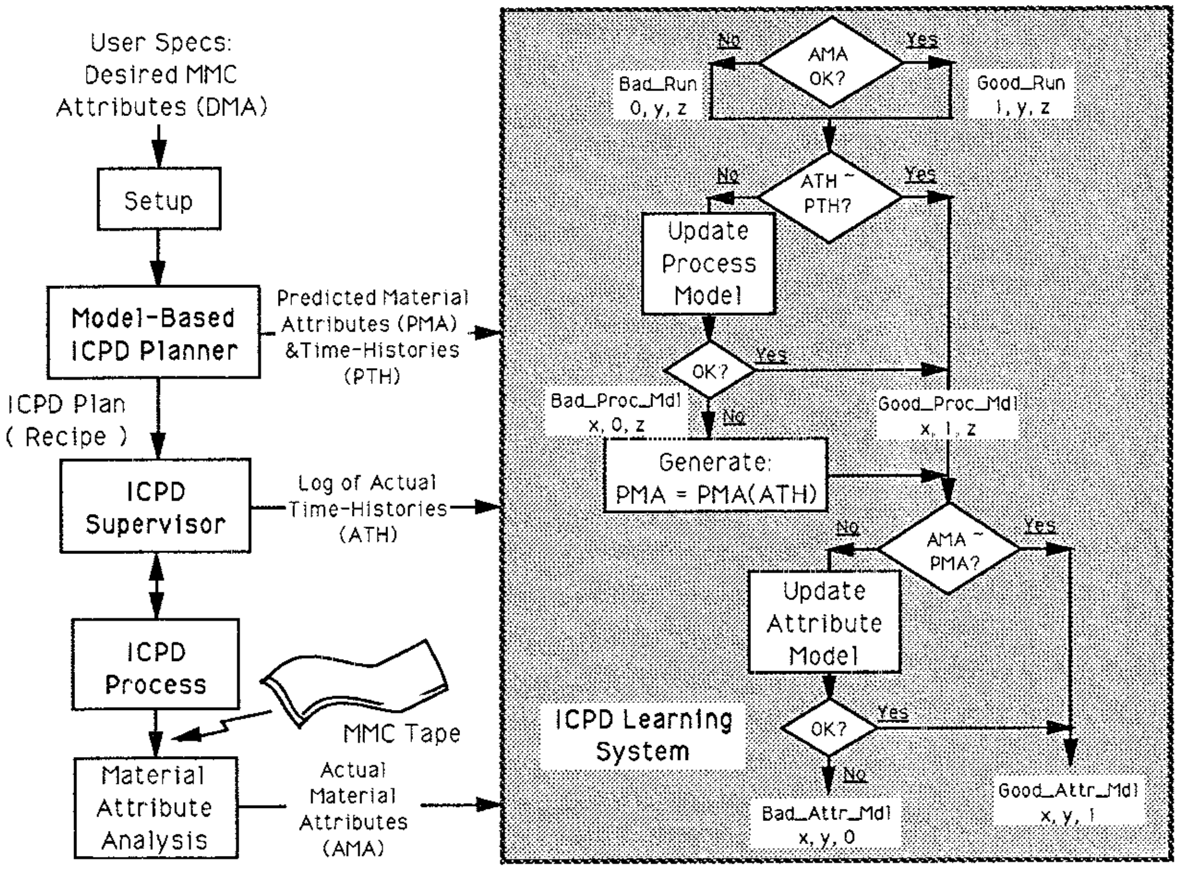

Figure 10. Learning System for ICPD

1. *Run Results*: if the MMC monotape is defective, set attribute 'x' to 0; otherwise 'x' = 1. A sample may be judged to be acceptable (1) even if the DMA were not met, as long as it might be accepted now or in the future; clearly defective tapes (e.g., with broken fiber) are rejected (0).

2. *Process Model Validity*: First, the *actual* controller outputs (commands) from the run are compared with the Planner's recipe. If there are significant disparities, the operator is notified that the recipe was not properly executed. Then the actual controller outputs are supplied to the actuator/process/sensor model to generate a predicted response. If these results agree within specified tolerances, the process model is validated for the run (attribute 'y' set to 1); otherwise an attempt is made to update the process model to bring the predictions into accord with the run data (below). If the model cannot be refined to yield sufficiently accurate predictions, then the operator is notified and 'y' = 0.

3. *Attribute Model Validity*: If the process model is not validated, then "synthetic" predicted material attributes (PMA) are generated by running the actual time-histories (ATH) through the material attribute models; otherwise, the original PMA are used for comparison with the actual material attributes (AMA). If the PMA agree with the AMA within specified tolerances, the material attribute models are validated for the run (attribute 'z' set to 1); otherwise an attempt is made to update the material attribute models to bring the predictions into accord with the run data (also, see below). If the models cannot be refined to yield sufficiently accurate predictions, then the operator is notified and 'z' = 0.

At the end of this evaluation and updating process, the plan and results are stored in the data base for future use by the Planner.

The actuator/process/sensor or "plant" models (Fig. 4) used by the Planner must be fast in execution time and formulated in such a way that automatic methods for refinement are feasible. Based on preliminary experience with the ICPD Simulator and real process data and I/O model identification, we expect that the Planner's plant dynamic models can satisfy both requirements. We believe that most of the process dynamics are so fast relative to the time scales involved in determining material attributes that they may be neglected, i.e., replaced by a static nonlinear table-look-up functions (Section 4.2.2). During each phase (preheat, spray etc. - see Fig. 2) the number of important control loops is small - we may need to consider plant models with no more than two to four inputs and outputs at any time. (There are secondary control loops, for example governing the static pressure in the chamber, that can be neglected in this context; these are shown at point $\boxed{\text{D}}$ in Fig. 4.) The end result of these considerations is that the Planner's plant models can be expressed either as low-order physics-based nonlinear models or as low-order multi-variable I/O models that are a simple extension of Eqn. (2), e.g., the relation between input u_j and output y_i may be of the form

$$G_{i,j}(s) = \frac{K_{i,j}\, exp(-s\, T_{1,i,j})}{(1 + s\, T_{2,i,j})} \tag{6}$$

where the parameters $K_{i,j}$, $T_{1,i,j}$ and $T_{2,i,j}$ are based on the phase and operating point and obtained by table-look-up functions.

Evaluating and refining such models based on new time-history data is a straightforward task. The predicted and actual time-histories (PTH, ATH, Fig. 10) are segmented into data sequences that correspond to specific phases, and only those phases where there are significant discrepancies between PTH and ATH are considered. Within each phase, only those outputs y_i that differ meaningfully are treated; denote one such output as y_{id}. The input/output data for that variable is then processed by a *recursive* parameter identification algorithm to refine the model (e.g., the parameters $K_{id,j}$, $T_{1,id,j}$ and $T_{2,id,j}$ in Eqn. (6) are updated via recursive least squares or maximum likelihood algorithms). The reason for using a recursive algorithm is that the resulting updated model is the best that can be obtained *for all input/output data up to the present time*, i.e., it optimally increments the parameter estimates based on the past history of the model's usage as well as the most recent run.

Material attribute models will be maintained and refined in the same manner as outlined above: predicted and actual material attributes will be compared; disparities determined, if any; and recursive parameter identification used to update model parameters as needed. These models are also quite simple in form, and will be parameterized as efficiently as possible (e.g., by elementary nonlinear functions with a few parameters or piece-wise-linear table look-up functions) to facilitate this process.

5. Summary and Conclusion

The implementation of an IPM system can be based primarily on empirical process knowledge (from operators and process engineers), or on process models. Several factors, especially limited operational experience and the need to expand beyond the known operating envelope, have motivated us to base the ICPD IPM system on process models.

The reliance upon models as the main foundation for intelligent controls design and implementation has a profound influence on the entire controls engineering effort. The approach described above is based on having a high-fidelity model of the process, and involves generating and validating several types of secondary models (simplified models, input/output or "blackbox" models, and linearized models). Modeling requirements were also discussed briefly. Following that, we outlined the use of models in four distinct realms: process simulation; analysis and design; IPM design validation; and in the IPM system itself (the ICPD model-based planner).

Sections 3 and 4 comprise the present "road-map" for designing and implementing a state-of-the-art IPM system for ICPD. From inception through implementation, process models play a central role in this system. We believe this system shows great potential for realizing the project goals, namely, improving product uniformity and quality, increasing process efficiency, and expanding the limits of the present operational envelope (e.g., producing MMC materials with dimensions and material attributes that have not been achieved so far).

REFERENCES

[1] Siemers, PA, and Jackson, JJ, "Ti3Al/SCS-6 MMC Fabrication by Induction Plasma Deposition", Titanium Aluminide Composite Workshop, Orlando FL, May 1990.

[2] Parrish, PA and Barker, WG, "The Basics of the Intelligent Processing of Materials", *JOM, 42,* July 1990, p.14-16.

[3] Backman, DG, "Metal-Matrix Composites and IPM: A Modeling Perspective", *JOM, 42,* July 1990, p. 17-20.

[4] Backman, DG, Russell, ES, Wei, D and Pang, Y, "Intelligent Processing for Metal Matrix Composites", in *Intelligent Processing Materials,* ed. by Wadley, HNG and Eckhart, WE, published by the Minerals, Metals and Materials Society, Warrendale PA, pp. 17-39, 1990.

[5] Wang, HP, Perry, EM, and Lillquist, RD, "Intelligent Processing of Materials for Plasma Deposition", *Proc. of 1990 Spring Meeting of the Materials Research Society,* San Francisco, CA, April 1990.

[6] Russell, ES, Wei, DY, Pang, Y, and Backman, DG, "Modeling of MMC Plasma Deposition Processing", TMS Fall Meeting, Detroit MI, October 1990.

[7] Wang, HP, and Perry, EM, "A Fast-Acting Process Simulator for Intelligent Plasma Deposition", in *Transport Phenomena in Material Processing, HTD-Vol. 132,* ed. by Charmichi, MK et al, ASME, pp. 99-108, 1990.

[8] Wang, HP, Perry, EM, Lillquist, RD, and Taylor, JH, "Elements of Intelligent Process Control for Plasma Deposition", *Journal of Materials, Vol. 43,* No. 1, pp. 22-25, January 1991.

[9] Integrated Systems, Inc., *$MATRIX_X$/SystemBuild Tutorial,* 2500 Mission College Blvd., Santa Clara, CA 95054-1215, 1990.

[10] The MathWorks, Inc., *SIMULAB User's Guide,* Cochituate Place, 24 Prime Park Way, Natick, MA 01760, 1991.

[11] SSPA, Inc., *SIMNON Reference Manual,* Gothenburg, Sweden / Engineering Software Concepts, Inc., Palo Alto, CA 94301.

[12] Mitchell and Gauthier Assoc., *ACSL Reference Manual,* Concord MA 01742, 1987.

[13] Kays, WM and Crawford, ME, *Convective Heat and Mass Transfer* (Second Edition), p. 141, McGraw Hill, New York, NY, 1980.

[14] Rohsenow, WM and Choi, HY, *Heat, Mass and Momentum Transfer,* p. 292, Prentice-Hall, Englewood Cliffs, NJ, 1961.

[15] Ljung, L, *System Identification, Theory for the User,* Prentice-Hall, Englewood Cliffs, NJ, 1987.

[16] Akaiki, H, "Canonical Correlation Analysis of Time Series and the Use of an Information Criterion", in *System Identification Advances and Case Studies,* ed. by Mehra, RK and Lainiotis, DG, Academic Press, New York, NY, pp. 27-96, 1976.

[17] Maciejowski, JM, *Multivariable Feedback Design,* Addison-Wesley, Reading MA, 1989.

[18] Taylor, JH, "Computer-Aided Control Engineering Environment for Nonlinear Systems", *Proc. 3rd IFAC Symposium on CAD in Control and Engineering Systems,* Lyngby, Denmark, August 1985.

[19] Taylor, JH, "Environment and Methods for Robust Computer-Aided Control Systems Design for Nonlinear Plants", *Proc. Second IFAC Symp. CAD of Multivariable Technological Systems,* West Lafayette, Indiana, September 1982; Internal GE CRD Report (more detailed), available upon request, 1988.

[20] Nett, CN and Manousiouthakis, V, "Euclidean Condition and Block Relative Gain", IEEE *Trans. on Automatic Control, AC-32,* May 1987, pp. 405-407.

[21] Integrated Systems, Inc., *Using the AC-100,* AC-100 User's Guide V2.4.02, 2500 Mission College Blvd., Santa Clara, CA 95054-1215, 1990.