PROCESS DESIGN AND CONTROL

Dynamic Optimization in a Discontinuous World

Paul I. Barton,* Russell J. Allgor, William F. Feehery, and Santos Galán

Department of Chemical Engineering and Energy Laboratory, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

Many engineering tasks can be formulated as dynamic optimization or open-loop optimal control problems, where we search a priori for the input profiles to a dynamic system that optimize a given performance measure over a certain time period. Further, many systems of interest in the chemical processing industries experience significant discontinuities during transients of interest in process design and operation. This paper discusses three classes of dynamic optimization problems with discontinuities: path-constrained problems, hybrid discrete/continuous problems, and mixed-integer dynamic optimization problems. In particular, progress toward a general numerical technology for the solution of large-scale discontinuous dynamic optimization problems is discussed.

Introduction

The past 5 years have been marked by increasingly widespread application of large-scale dynamic simulation technology by the chemical processing industries. There are now many examples in the literature of major chemical companies constructing and solving plantwide dynamic models of their processes and realizing major benefits (Mani *et al.*, 1990; Evans and Wylie, 1990; Naess *et al.*, 1993; Longwell, 1993; Cole and Yount, 1993; Grassi, 1993; Zitney *et al.*, 1995; Debelak *et al.*, 1995; Papageorgaki *et al.*, 1995). Given that dynamic models of entire processes will become increasingly available in the near future, it is natural to ask what other applications might exploit the resources invested in the development of such models. In particular, can these models also be used to optimize the transients of a dynamic system, and how might such an optimization capability be useful? In fact, many of the oft-quoted applications of dynamic simulation technology are more properly formulated as dynamic optimization problems; i.e., systematic improvement of a dynamic system's performance using optimization techniques as opposed to ad hoc improvement using simulation-based 'what if' studies.

The optimization of a dynamic system according to some performance measure is a classical problem in the calculus of variations, and its extensions by modern optimal control theory (Pontryagin *et al.*, 1962; Kirk, 1970; Bryson and Ho, 1975). Optimal control policies are classified as either *open-loop* or *closed-loop* optimal controls. A closed-loop policy is a stronger form in which the optimal feedback of a dynamic system's states for any initial condition is determined. On the other hand, an open-loop policy determines the optimal time program for the inputs to a dynamic system corresponding to a particular initial condition. This paper will focus on open-loop policies because (a) there are many practical applications of open-loop policies in process design and operation and (b) reliable numerical solution is now possible even for large-scale nonlinear process models. We shall refer to such open-loop problems as dynamic optimization problems hereafter.

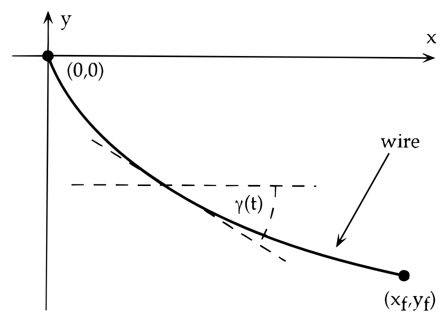

Figure 1. Schematic of the brachistochrone problem.

In recent years, several researchers in the chemical engineering community have taken steps to develop generic algorithmic tools for plant-wide dynamic optimization. The purpose of this paper is to identify the many potential applications of this emerging plantwide dynamic optimization technology, to highlight the discontinuous nature of many problems of engineering importance, and then to review and speculate concerning the development of generic numerical technologies for discontinuous dynamic optimization. The paper begins with an introduction to classical dynamic optimization formulations, moving on to the many potential chemical engineering applications. Next, both some generic properties of discontinuous dynamic systems and strategies for the numerical solution of dynamic optimization problems are reviewed. The remainder of the paper discusses three classes of dynamic optimization problems with discontinuities: path-constrained problems, hybrid discrete/continuous problems, and mixed-integer dynamic optimization problems. In par-

* Corresponding author at Massachusetts Institute of Technology 66-464, 77 Massachusetts Avenue, Cambridge MA 02139. Phone: +1-617-253-6526. Fax: +1-617-258-5042. E-mail: pib@mit.edu.

S0888-5885(97)00738-0 CCC: $15.00
Published on Web 02/12/1998

ticular, the discussion highlights some generic theoretical properties of these problems, progress toward a general numerical technology for the solution of large-scale discontinuous dynamic optimization problems, and directions for future research.

Dynamic Optimization

As a simple illustration of a dynamic optimization, consider the classic brachistochrone problem that was proposed in 1696 by John Bernoulli to challenge the mathematicians of Europe (Figure 1). The object of the brachistochrone problem is to find (in two dimensions) the shape of a frictionless wire that causes a bead, initially at rest, to move under the force of gravity between the origin and a specified final point in minimum time. This can be formulated as a dynamic optimization problem in the following manner (Kraft, 1994):

$$\min_{\gamma(t),\, t_f} t_f \tag{1}$$

Subject to:

$$\dot{x} = w \cos(\gamma(t)) \tag{2}$$

$$\dot{y} = -w \sin(\gamma(t)) \tag{3}$$

$$\dot{w} = g \sin(\gamma(t)) \tag{4}$$

$$x(0) = 0 \qquad y(0) = 0 \qquad w(0) = 0 \tag{5}$$

$$x(t_f) = x_f \qquad y(t_f) = y_f \tag{6}$$

where (1) defines the performance measure for the optimization, which in this case is to minimize the time required to reach the final point. The motion of the bead in a gravity field is described by the differential equations (2−4) (i.e., the dynamic model of the system), where w is the velocity tangential to the wire and g is the gravitational acceleration (a given constant). In this case the control (or forcing function) for the dynamic system is $\gamma(t)$, the angle of the wire to horizontal, which defines the shape of the wire as a function of time. Hence, the decision variables in this problem are the duration of the motion t_f and the control *profile* $\gamma(t)$. The purpose of the optimization is to find, out of all the possible functions $\gamma(t)$/shapes of the wire, that function/shape that minimizes the final time. In fact, one of the key results of optimal control theory is that such optimal functions (and their corresponding state trajectory) exist and are in many cases unique.

To complete the formulation of the brachistochrone problem, it is necessary to include the *point constraints* (5−6). The initial point constraints (5) require the bead to be at rest at the origin at the initial time, and the final point constraints (6) require the bead to move through some other point in two-dimensional space at the final time. Note that these constraints establish relationships between the system's states at a finite number of points in time, hence the term point constraint.

Many systems of interest in the chemical processing industries experience significant discontinuities during transients of interest in process design and operation. As a small example of the optimization of an inherently discontinuous dynamic system, let us study the depth-time profile of a scuba diver. One of the major hazards of diving is decompression sickness, which is caused by bubbles that form in the blood and tissues of the body when the nitrogen dissolved at high pressure supersaturates in a rapid ascent from depth. At the beginning of this century J. S. Haldane was appointed by the British Admiralty to develop safer decompression tables for Navy divers. Haldane experimented with goats and proposed a model describing the uptake and elimination of N_2 as a first-order system where the time constants for different tissues are represented by 'half-times.' In this example only a 5-min half-time tissue will be modeled, yielding (8), where P_5 is the partial pressure of N_2 in the tissue and P is the surrounding pressure.

The control for this problem will be the rate of ascent/descent u, and since pressure P is equivalent to depth, u is equivalent to the rate of change of P (eq 10). Diving tables establish decompression stops at different depths depending on the profile of the dive. These tables follow conceptually the theoretical principles developed by Haldane combined with more recent developments and data. In this simplified example, we will assume that when the partial pressure in the tissue becomes twice the environmental pressure, the diver must make a decompression stop of 4 min, which effectively constrains the admissible control profile (eq 11). Note that these discontinuities in the control profile occur at points in time determined by the state trajectory $P_5(t)$ crossing the state trajectory $2P(t)$, and therefore they are not known a priori but rather are determined implicitly by the solution of the model equations (this type of implicit discontinuity is commonly called a *state event* in the simulation literature (Park and Barton, 1996)). Hence, in general the dynamic optimization has to determine the number and ordering of these implicit discontinuities along the optimal trajectory. In addition, the descent and ascent rates will be constrained (30 and 10 m/min).

Imagine a scenario in which the diver wishes to collect an item from the ocean floor at 50 m with minimum total consumption of air. Clearly, given this objective, we must model the consumption of air. A regular tank volume is $V_t = 15$ L, and it is charged to $P_t(0) = 200$ bar. An average air consumption is $q = 8$ L/m, but the mass consumed varies with the depth (pressure), yielding (9). A parameter t_b is introduced to model the time at which the diver reaches the bottom, and point constraints (14) and (15) force him to travel to the bottom and then surface. Overall, this yields the formulation:

$$\max_{u(t),\, t_f,\, t_b} P_t(t_f) \tag{7}$$

Subject to:

$$\dot{P}_5 = \frac{\ln(2)}{5}(0.79P - P_5) \tag{8}$$

$$\dot{P}_t = -\frac{q}{V_t} P \tag{9}$$

$$\dot{P} = u \tag{10}$$

$$\begin{cases} \text{State A:} & -1 \le u(t) \le 3 \\ & \text{switch to B if } P_5 \ge 1.58P \\ \text{State B:} & u(t) = 0 \\ & \text{wait for 4 min and then switch to A} \end{cases} \tag{11}$$

$$P_t \ge P \qquad \forall t \in [0, t_f] \tag{12}$$

$$P_t(0) = 200 \qquad P(0) = 1 \qquad P_5(0) = 0.79P(0) \tag{13}$$

$$P(t_b) = 6 \tag{14}$$

$$P(t_f) = 1 \tag{15}$$

$$0 \le t_b \le t_f \tag{16}$$

This problem also introduces the notion of *path constraints* that establish relationships between a system's states that must hold throughout the entire time interval of interest and point constraints that only hold at specific points in time. For example, inequality path constraint (12) ensures that the diver can breathe at every point in time throughout the dive.

In a first approach, two admissible control profiles are examined:

1. Dive as quickly as possible to the bottom and then ascend at 5 m/min, making decompression stops as necessary (Figure 2).
2. Dive as quickly as possible to the bottom and then ascend at 10 m/min, making decompression stops as necessary (Figure 3).

Observe that the different admissible controls yield different sequences of decompression stops. The consequences of these control policies are that in the first case the diver is forced to make two decompression stops, augmenting the time and air consumption of the dive, whereas in the second case a more rapid ascent rate only requires one decompression stop. Further, if we only admit constant ascent rates and then if the ascent rate is dropped further, only one stop becomes necessary again. In general, this need to determine the number and ordering of implicit discontinuities along the optimum trajectory appears to be the most difficult issue with the optimization of discontinuous dynamic systems.

Applications of Plantwide Discontinuous Dynamic Optimization

The application of dynamic optimization that probably immediately springs to mind is the determination of optimal operating policies for batch unit operations, for example, a temperature profile that will maximize selectivity to the desired product in a batch reaction involving complex competing kinetics (Denbigh, 1958) or a reflux policy and accumulator dumps for a batch distillation column that maximizes profit subject to purity and recovery constraints (Diwekar *et al.*, 1989). In a real plant, the profiles determined a priori by dynamic optimization can then serve as set-point programs for the temperature control system, reflux control, etc. Although the literature abounds with both analytical and numerical treatments of such problems (see, for example, Rippin (1983) for a review), the natural extension of dynamic optimization formulations to the design of an integrated plantwide operating policy (or *recipe)* for an entire batch process has only been contemplated in recent years. The benefits of plantwide dynamic optimization of batch processes were first demonstrated in Barrera and Evans (1989), and recent work (Charalambides *et al.*, 1995; Charalambides, 1996; Bhatia and Biegler, 1996) has shown that dynamic optimization of relatively sophisticated plant-wide models involving thousands of states offers a promising and tractable approach to these problems. However, to date these formulations have been purely continuous, with the exception of discontinuous control profiles, which are relatively easy to handle with existing technology. It is certain that most batch processes contain implicit discontinuities in their physical behavior (or *physicochemical discontinuities* (Barton and Pantelides, 1994)). For example, many exothermic batch reactions will self-heat to the boiling temperature of the reaction mass unless sufficient temperature control can be applied, or the rate of addition of one of the reactants is regulated (such a situation is not necessarily dangerous since a reflux condenser can then be used to remove the heat of reaction). Similarly, as the pot level drops during the course of batch rectification, heating coils in the pot may become partially exposed, with the consequent discontinuity in the heat transfer law. Further, the design of an integrated recipe will, in general, require sequencing decisions concerning the ordering of candidate control actions in the time domain, and such decisions cannot always be handled by purely continuous formulations. At a level of greater complexity, the development of an optimal integrated batch process design to be implemented in an existing manufacturing facility involves a dynamic optimization with physicochemical discontinuities (i.e., design of the plant-wide recipe) coupled with integer decisions, such as the selection of solvents and reagents from a list of candidates, selection of reaction and separation technology and recycle structure for the process, and selection of equipment items from an inventory available within the plant (Allgor and Barton, 1997; Allgor *et al.*, 1997; Allgor, 1997). A later section of this paper will discuss some preliminary results on these mixed-integer dynamic optimization (MIDO) problems.

Similarly, many problems in the optimal operation of continuous processes can be formulated as discontinuous dynamic optimization problems. For example, in Debling *et al.* (1994) dynamic simulation is used to predict the effect of altering the changeover policies for product-grade transitions in polyolefin processes. In principle, this problem can be formulated systematically as a dynamic optimization problem in which a sequence of control actions is found to achieve the changeover in, for example, minimum time or with minimum production of off-spec product. In particular, decisions have to be made concerning the selection of a number of control actions from a list of alternatives and their ordering in the time domain. This observation is easily extended to the generic operating procedure synthesis problem, e.g., the design of minimum time or cost start-up and shut-down procedures, and as we move further from the nominal steady state(s) of the continuous process, the discontinuous aspects of the problem become more pronounced and critical. Operating procedure synthesis is the design of a sequence of control actions that will take a plant from some initial state to a goal state subject to a set of physical, safety, and operational constraints on the dynamic path between these states. Although frequently viewed as a feasibility

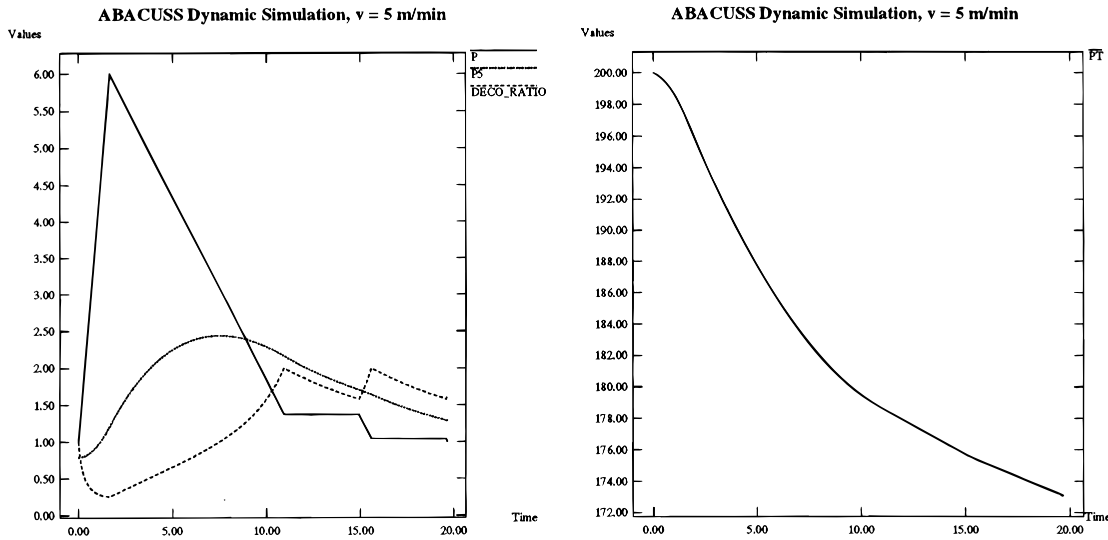

Figure 2. Time profiles for admissible control profile 1.

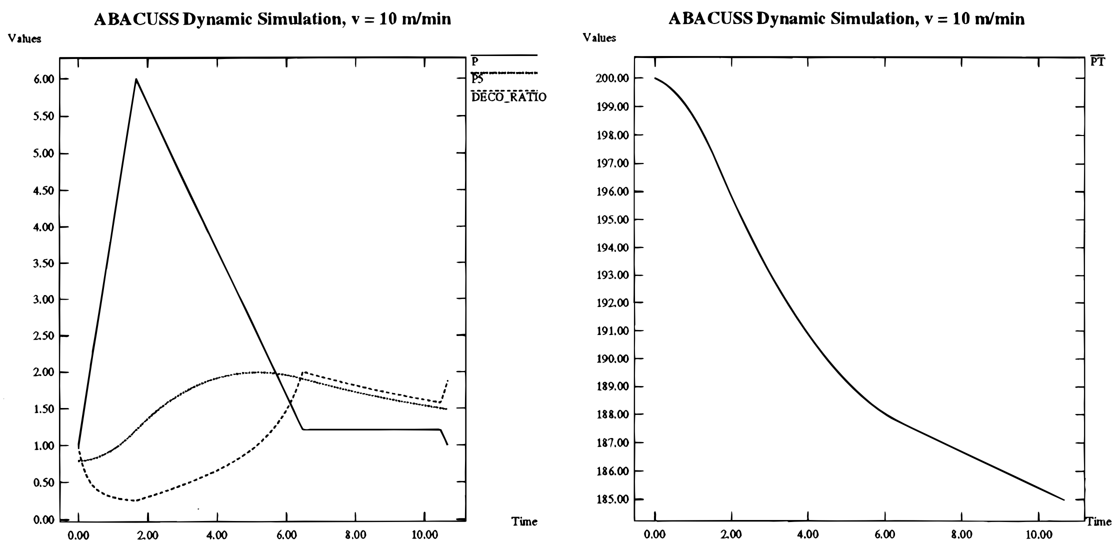

Figure 3. Time profiles for admissible control profile 2.

problem in the literature (Lakshmanan and Stephanopoulos, 1988), clearly this search can be performed in order to minimize or maximize some cost metric, and in a dynamic optimization framework this adds no further complexity (provided one is content with a local extremum) (Galán and Barton, 1997).

As an example of an operating procedure synthesis problem and its formulation as a discontinuous dynamic optimization problem, consider the simple tank changeover operation illustrated in Figure 4a (Rivas and Rudd, 1974; Han *et al.*, 1995). The purpose of this operation is, in minimum time, to take the system between an initial state in which oxygen is flowing through the system to a final state in which methane is flowing through the system. These initial and goal states can be expressed in terms of initial and final point constraints, and a nitrogen stream is available as an inert purge. The differential equations describing the physical system must include a model for the accumulation of mass and energy in the region between the inlet and outlet valves, and flow/pressure/stem position relationships for the open/close valves, including flow transitions between laminar, turbulent, and choked flow regimes (the last of which must be modeled with a discontinuity). In addition, path constraints are necessary to express many important features of the problem. For example, the key constraint is that the path of the species mole fractions within the tank in composition space (a subset of state space) must lie outside some region bounding the explosive limits for oxygen/methane/nitrogen mixtures (see Figure 4b). Furthermore, there will be a minimum temperature constraint (the brittle transition temperature for the vessel) and maximum pressure constraint (the design pressure of the vessel) that must be satisfied throughout the operation. Moreover, the influence of control actions is implicitly constrained by the valve model, particularly in the choked flow regime.

Finally, the control decisions amount to selecting the number and order of valve openings/closings in the time domain. If we assume that the signal to the open/close valves takes only two values (i.e., open or close), then this amounts to searching for the points in time t_i^* at which there is a transition in the signal to each valve (i.e., a set of purely *continuous* decision variables). So, in principle, provided we introduce a sufficient number of these transition parameters as decision variables in the problem formulation (e.g., since we know the oxygen valve is open in the initial state and closed in the goal state, we must allow an odd number of transitions for

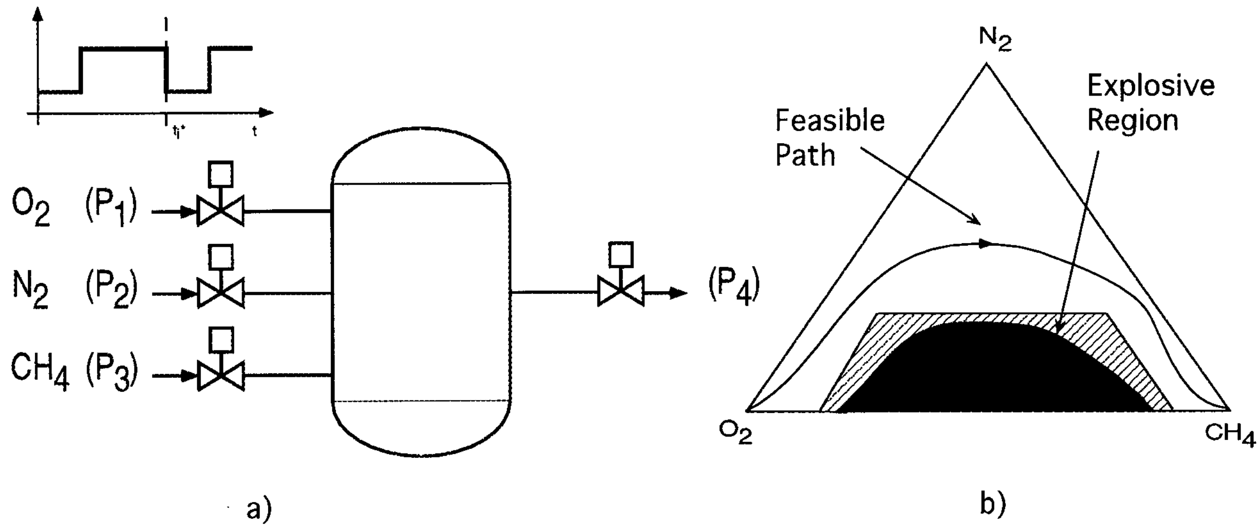

Figure 4. Tank changeover problem: (a) process flowsheet; (b) feasible paths in composition space which must lie outside the region bounding the explosive limits.

the oxygen valve signal, etc.), the dynamic optimization can design a minimum time changeover operation automatically.

There are two intriguing features of this formulation of the tank changeover problem: (a) Given a sufficiently detailed dynamic model of the equipment available, the need to purge the system with nitrogen before introducing methane is effectively inferred from the existence of path constraints in the dynamic optimization formulation. (b) Superficially, it seems as though a purely continuous dynamic optimization formulation can make sequencing decisions and solve operating procedure synthesis problems. In general, we believe that formulation of the relevant path constraints coupled with detailed dynamic models provides an appropriate and unified framework for representing the important features of operating procedure synthesis problems. Ultimately, the need for operations such as a purge are dictated by path constraints and thus can be inferred from path constraints by an optimization formulation. On the other hand, the formulation is not in fact purely continuous: we do not know a priori the number and ordering of path inequality constraint activations/deactivations and physicochemical discontinuities along the optimal trajectory, and as we shall see, this need to search over alternative sequences introduces nonsmoothness into optimization formulations. However, if we indeed do know this sequencing information a priori, we shall demonstrate a smooth optimization formulation arises, even though the dynamics are discontinuous!

Other potential applications of discontinuous dynamic optimization formulations include safety validation of chemical processes and the design of optimal pressure or temperature swing adsorption cycles. For example, some preliminary work on safety validation (Dimitriadis *et al.*, 1994) poses a continuous dynamic optimization problem that, from a space of initial conditions, attempts to find the minimum time in which a system gets into trouble. However, a purely continuous formulation is very limited since most safety critical transients will contain a sequence of many physicochemical discontinuities. Moreover, during such transients it is desirable to study the interaction of the chemical process with its automatic protective system (Park and Barton, 1994), the control actions of which will introduce further implicit discontinuities into the dynamics. Further, it is possible to develop a coupled model of a process and its automatic protective system in terms of a mixed system of differential-algebraic equations and linear inequality constraints in terms of binary variables (Park and Barton, 1994; Park, 1997), which potentially could be used in a MIDO formulation. As regards the optimal design of PSA/TSA cycles, there are at least three problems of increasing complexity: (a) dynamic optimization of a fixed flowsheet in which all sequencing aspects of the problem are fixed a priori, (b) dynamic optimization of a fixed flowsheet in which a search over alternative control sequences is made, and (c) an optimization problem in which a superstructure of alternative bed configurations is established and a search is made for the optimal bed configuration and operating policy simultaneously.

Hybrid Discrete/Continuous Dynamic Systems

In recent years, the study of hybrid (or combined) discrete/continuous dynamic systems has attracted increasing attention in the chemical engineering community and beyond (Grossman *et al.*, 1993; Antsaklis *et al.*, 1995; Alur *et al.*, 1996; Barton and Park, 1997). In particular, industrial application of combined discrete/continuous simulation to many chemical engineering problems, such as the simulation of batch processes, start-up and shut-down procedures and safety critical transients in continuous processes, and periodic processes such as pressure swing adsorption, has been greatly facilitated by advances in simulation technology, particularly that embedded in the gPROMS (Barton, 1992) and ABACUSS [(Advanced Batch And Continuous Unsteady-State Simulator) process modeling software, a derivative work of gPROMS software. Copyright 1992 Imperial College of Science, Technology, and Medicine]. (Allgor *et al.*, 1996) process modeling environments.

The dynamics of many chemical processes can be adequately predicted by the solution of initial value problems (IVPs) in differential-algebraic equations (DAEs) with the general nonlinear form

$$f(\dot{x},x,u,t) = 0 \tag{17}$$

Hence, this is the mathematical formulation supported by most modern dynamic simulators.

Hybrid phenomena can be classified as either *switches* or *jumps* (Branicky *et al.*, 1994). A switch refers to discrete changes in the functional form of (17) as a consequence of *events* that occur instantaneously at a point in time. These different possible functional forms are sometimes known as the *modes* of a hybrid system. The time of occurrence of events may be either defined a priori (*time events*) or defined implicitly by the system state satisfying some condition (*state events*). For example, a class of implicit switches can be expressed with the notation:

$$\begin{cases} f_1(x,\dot{x},u,t) = 0 & \forall t \in [0, t_f]:\ s(x,\dot{x},u,t) > 0 \\ f_2(x,\dot{x},u,t) = 0 & \forall t \in [0, t_f]:\ s(x,\dot{x},u,t) \le 0 \end{cases} \quad (18)$$

where f_1 is a subset of the model equation (17) that is inserted in the overall model when the (scalar) state condition $s(x,\dot{x},u,t) > 0$, and f_2 is inserted otherwise. Note in particular that in this case the state events defining switching times, i.e., $t^* = \{t \in (0, t_f]:\ s(x,\dot{x},u,t) = 0\}$, are not known in advance because they are a function of the system state, and therefore the timing and order of equation switches is also not known in advance. Note that the physicochemical discontinuities discussed above are a class of implicit switches. As a simple chemical engineering illustration, the following equations could be used to model the influence of a weir on the dynamics of a buffer tank:

$$\begin{cases} F_{out} = k_{weir}(h - h_{weir})^{1.5} & \forall t \in [0, t_f]:\ h(t) > h_{weir} \\ F_{out} = 0 & \forall t \in [0, t_f]:\ h(t) \le h_{weir} \end{cases} \quad (19)$$

where h is the level of liquid in the tank, and h_{weir} is the height of the weir. Basically, this formulation models the fact that while the liquid level is above the weir, liquid flow out of the tank will be driven by the head above the weir, whereas while the liquid level is below the weir, there is no liquid flow from the tank. Note that most modern dynamic simulators at least support modeling of switches conforming to formulation (18).

Similarly, jumps refer to discontinuities in the state of a dynamic system as a consequence of events (see Barton and Park (1997) for a precise definition of jumps in DAE systems). For example, a class of implicit jumps can be expressed with the notation:

$$j(x^-,\dot{x}^-,u^-,x^+,\dot{x}^+,u^+,t^*) = 0 \qquad \forall t^* \in [0, t_f]:\ s(x^-,\dot{x}^-,u^-,t^*) = 0 \quad (20)$$

where x^- denotes $\lim_{t\to t^{*-}} x(t)$ and x^+ denotes $\lim_{t\to t^{*+}} x(t)$, etc., and j coupled with (17) defines the jump. For example, the dumping of a quantity of reactant into a batch reactor over a very short time scale could be abstracted as an instantaneous jump in the relevant variable representing the number of moles of the reactant in the reactor. Clearly, in general, jumps and switches can occur simultaneously at an event.

Numerical Solution of Dynamic Optimization Problems

In this section we will briefly review general purpose methods for the numerical solution of continuous dynamic optimization problems, which can be expressed using the relatively general formulation:

$$\min_{u(t),\, t_f} \phi(x(t_f),u(t_f),t_f) + \int_0^{t_f} L(x,u,t)\, dt \quad (21)$$

Subject to:

$$f(x,\dot{x},u,t) = 0 \qquad \forall t \in [0, t_f] \quad (22)$$

$$h(x,\dot{x},u,t) = 0 \qquad \forall t \in [0, t_f] \quad (23)$$

$$g(x,\dot{x},u,t) \le 0 \qquad \forall t \in [0, t_f] \quad (24)$$

$$k_p(x(t_p),\dot{x}(t_p),u(t_p),t_p) \le 0 \qquad \forall p \in \{0, ..., n_p\} \quad (25)$$

where (22) are the DAEs describing the system dynamics, (23) are a set of path equality constraints that must be satisfied over the entire time horizon, (24) are a set of path inequality constraints, and (25) are point constraints at a finite set of times, including initial and end-point constraints.

There is an extensive literature on numerical strategies for the solution of the above dynamic optimization formulation, falling into three classes of methods: *dynamic programming* based approaches, *indirect* methods, and *direct* methods. The dynamic programming approach was first described in Bellman (1957) and was extended to include constraints on the state and control variables in Luus (1990). Indirect methods (Bryson and Ho, 1975) focus on obtaining a numerical solution to the classical necessary conditions for optimality, which take the form of a two-point boundary value problem in differential-algebraic equations. Direct methods transform the infinite dimensional dynamic optimization problem into a finite dimensional nonlinear program (NLP). Within the framework of the direct method, there are two general strategies: the sequential or *control parametrization* method (Sargent and Sullivan, 1977; Kraft, 1985) (see Figure 5) and the simultaneous or *collocation* method (Neuman and Sen, 1973; Tsang *et al.*, 1975), in which both the controls u and x are discretized using polynomials on finite elements, and the coefficients of these polynomials and element sizes become decision variables in a large-scale NLP.

The discussion in the remainder of this paper will focus on the control parametrization approach to dynamic optimization because (a) our efforts to develop a technology for discontinuous dynamic optimization have been directed at extensions of the control parametrization framework and (b) recent advances in numerical sensitivity technology (Maly and Petzold, 1996; Feehery *et al.*, 1997) make control parametrization computationally very efficient in comparison to other approaches. However, it should be noted that our results and their implications will have their analog for other numerical strategies. Control parametrization (Figure 5) reduces the infinite dimensional dynamic optimization (21)–(25) to a finite dimensional problem through approximation of the control profiles $u(t)$ by a finite family of basis functions. For example, in Figure 5 the control is approximated as four piecewise linear functions, and the slopes, intercepts, and lengths of each finite element become the decision variables (or parameters) in the optimization. Similarly, if we wished to admit discontinuous controls, we would merely relax the requirement for continuity of the control from one finite element to the next (i.e., relaxing constraints between coefficients of the polynomials on consecutive finite elements). However, in choosing a number of finite elements and

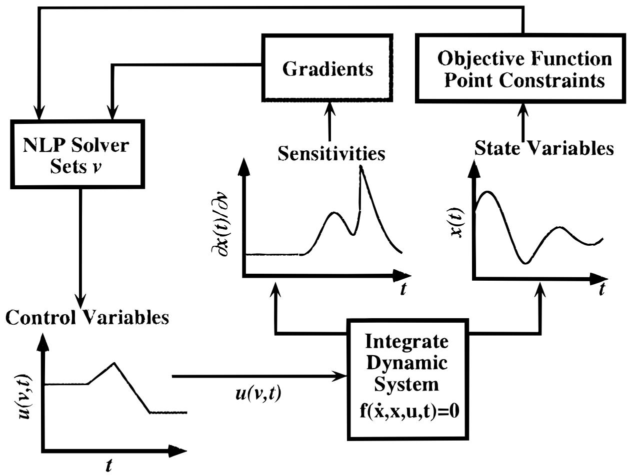

Figure 5. Schematic of the control parametrization approach to dynamic optimization.

allowing discontinuities, we have made a decision concerning the possible sequences of discontinuities over which the optimization will search.

Discretization using control parametrization allows the objective function and point constraints in (21)−(25) to be expressed as composite functions of a finite number of time invariant parameters v (characterizing the basis functions) and leads to a decomposition of the dynamic optimization into two subproblems:

(1) an IVP subproblem, which is the numerical integration of (22) and possibly (23) and (24) for given values of v which yields:

$$\dot{x}(v,t),\ x(v,t) \qquad \forall t \in [0,\ t_f] \tag{26}$$

(2) a Master subproblem, which is a NLP in terms of the parameters v:

$$\min_{v,\ t_f} \phi(x(v,t_f),u(v,t_f),t_f) + \int_0^{t_f} L(x(v,t),u(v,t),t)\ \mathrm{d}t \tag{27}$$

Subject to:

$$k_p(x(v,t_p),\dot{x}(v,t_p),u(v,t_p),t_p) \le 0 \qquad \forall p \in \{0,\ ...,\ n_p\} \tag{28}$$

Note that this NLP will be much smaller than that arising in the collocation approach. Even if there are thousands or tens of thousands of state variables x, the number of control variables u and parameters required to discretize each control will typically be small. Hence, n_v may be on the order of hundreds even in very complex plantwide dynamic optimization problems.

The Master NLP may be solved with either gradient-based approaches (e.g., successive quadratic programming (Vassiliadis, 1993)) or gradient-free approaches such as direct or stochastic search (Carrasco and Banga, 1997). In a gradient-based approach, function and gradient evaluations for the Master NLP can be obtained by simultaneous solution of an IVP subproblem and the related sensitivity systems:

$$\frac{\partial f}{\partial \dot{x}}\frac{\partial \dot{x}}{\partial v_i} + \frac{\partial f}{\partial x}\frac{\partial x}{\partial v_i} = -\frac{\partial f}{\partial u}\frac{\partial u}{\partial v_i} - \frac{\partial f}{\partial v_i} \qquad \forall i = 1,\ ...,\ n_v \tag{29}$$

which yields

$$\frac{\partial \dot{x}}{\partial v_i}(v,t),\ \frac{\partial x}{\partial v_i}(v,t) \qquad \forall i = 1,\ ...,\ n_v,\ t \in [0,\ t_f] \tag{30}$$

followed by application of the chain rule, e.g., for the point constraints:

$$\frac{\partial k_p}{\partial v_i}(v,t_p) = \frac{\partial k_p}{\partial \dot{x}}(v,t_p)\frac{\partial \dot{x}}{\partial v_i}(v,t_p) + \frac{\partial k_p}{\partial x}(v,t_p)\frac{\partial x}{\partial v_i}(v,t_p) + \frac{\partial k_p}{\partial u}(v,t_p) + \frac{\partial u}{\partial v_i}(v,t_p) \qquad \forall p \in \{0,\ ...,\ n_p\} \tag{31}$$

Recent work has demonstrated that local solutions to dynamic optimizations involving thousands of state variables x can be obtained in reasonable computational times on engineering workstations using control parametrization (Charalambides *et al.*, 1995; Charalambides, 1996). For example, our own experience indicates that, given a state-of-the-art sparse large-scale DAE solver (e.g., DSL48S (Barton *et al.*, 1997; Feehery *et al.*, 1997)) coupled with recent algorithmic developments that dramatically increase the efficiency of sensitivity system solution (Feehery *et al.*, 1997) and the use of sparse automatic differentiation techniques to obtain derivatives efficiently (Tolsma and Barton, 1997a,b), dynamic optimization problems with thousands of states can be solved in tens of minutes on desktop workstations. In addition, there is a lot of potential to exploit parallel computing architectures within the control parametrization framework that has hardly been explored yet. Since computer hardware and algorithms are constantly improving, it is probably not long before control parametrization applications with 10,000 and 100,000 state variables will be reported.

Path-Constrained Dynamic Optimization

The above discussion has highlighted the fact that path constraints are a common feature of realistic dynamic optimization formulations that address problems of engineering relevance. In particular, we highlighted the critical role they play in defining operating procedure synthesis problems. On the other hand, the reliable and efficient numerical treatment of dynamic optimization formulations containing path constraints remains a persistently difficult problem. In this section we discuss a new approach to path-constrained problems that establishes an equivalent discontinuous dynamic optimization formulation.

Path Equality Constraints. Path equality constraints are equality constraints that must be satisfied over the entire time period of interest. As a simple illustration of a problem with path equality constraints, consider the following reformulation of the equations of motion (2)−(4) for the brachistochrone problem:

$$\dot{x} = u \tag{32}$$

$$\dot{y} = v \tag{33}$$

$$\dot{u} = F\sin(\gamma(t)) \tag{34}$$

$$\dot{v} = -g + F\cos(\gamma(t)) \tag{35}$$

where u and v are the horizontal and vertical velocities, respectively, and F is the normal contact force. Equations 32−35 just describe the forces acting on the bead, so we must add a constraint defining the shape of the wire:

$$-\tan(\gamma(t)) = \frac{v}{u} \quad \left(= \frac{\dot{y}}{\dot{x}} = \frac{dy}{dx}\right) \tag{36}$$

in other words, an equality constraint that forces the bead to follow a certain path.

There are two broad strategies for solving problems containing equality path constraints such as (36). In the first, discretizations of both $\gamma(t)$ and $F(t)$ are treated as degrees of freedom by the Master NLP, and given a particular realization of these profiles, the ODEs (32)−(35) are solved in the IVP subproblem. However, given an arbitrary $\gamma(t)$ and $F(t)$, there is no guarantee that the resulting solution of (32)−(35) will satisfy the path constraint (36). Hence, the Master NLP must take care of adjusting the control profiles so that the path constraint is satisfied at the optimum (e.g., by forcing the integral violation of the constraint to be small). This approach is particularly problematic for control parametrization because, unless a very large number of finite elements and high-order polynomials are used for the controls, it is impossible to find control profiles that track the path constraint to high accuracy. In addition, this approach tends to require many, many NLP iterations (Vassiliadis, 1993), making the computation extraordinarily costly.

On the other hand, another approach is to have the IVP subproblem take the responsibility of satisfying the path constraints (Feehery *et al.*, 1995). In the brachistochrone problem this amounts to augmenting the path constraint (36) to the ODEs (32)−(35) to yield a DAE system (32)−(36). Clearly, by adding an extra equation to the IVP subproblem we lose a degree of freedom, so let us consider the situation in which $\gamma(t)$ is chosen as the decision variable for the dynamic optimization, and F is determined by solution of the IVP subproblem. We now need to study the properties of DAE (32)−(36) with $\gamma(t)$ specified. In particular, DAEs of the form (17) are characterized by their *differential index* (Brenan *et al.*, 1996), a nonnegative integer such that an index = 0 system is by definition an ODE. For example, the brachistochrone DAE (32)−(36) has an index = 2, but most modern dynamic simulators can only solve DAEs correctly with index ≤ 1. In general, the treatment of equality path constraints by the IVP subproblem is likely to lead to *high-index* DAE subproblems (loosely speaking, those with index ≥ 2). But, if reliable methods for solving high-index DAEs were available, this could be an effective approach to solving equality path-constrained problems.

It has been known for some time that high-index DAEs cause pathological difficulties for standard numerical integration algorithms (Petzold, 1982) and require special treatment in the specification of initial conditions (Pantelides, 1988). However, just recently a novel approach known as the method of *dummy derivatives* (Mattsson and Soderlind, 1993) was proposed for the reliable and efficient numerical solution of a broad class of nonlinear DAEs of arbitrary index. In a nutshell, this approach automatically derives and solves a larger index 1 DAE that has an equivalent solution set to the original high-index system, so that the solution of the index 1 system can be substituted for that of the high-index system. On the basis of the original ideas presented by Mattsson and Soderlind (1993), we have made a number of fundamental developments in the dummy derivative method (Feehery and Barton, 1996) that enable it to solve reliably large-scale high-index DAEs with a computational cost close to that required for an index 1 DAE of similar size. Our preliminary numerical results indicate that treatment of path constraints via a high-index IVP subproblem is indeed a very efficient dynamic optimization strategy.

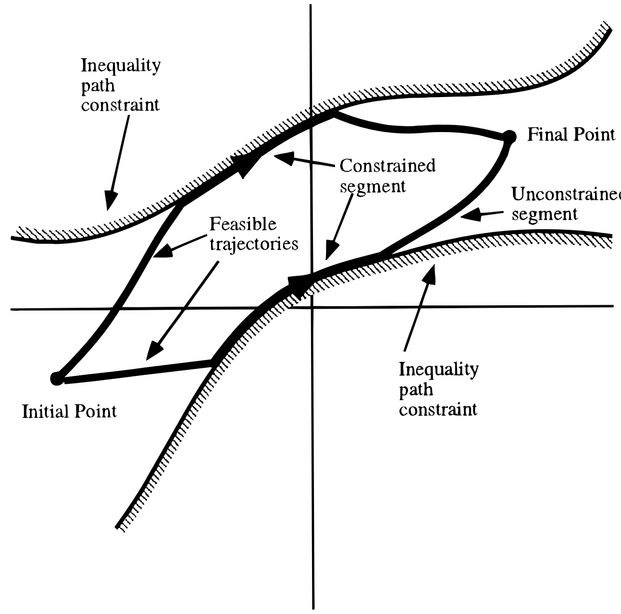

Figure 6. Sample of the feasible trajectories in a constrained state space.

In fact, given a generic dynamic optimization problem (21)−(25), the dynamic model (22) and the equality path constraints (23) can be coupled to yield a DAE with a certain number of degrees of freedom. In any given problem there may be a number of admissible sets of control variables (in the sense that they yield a solvable DAE) to satisfy these degrees of freedom, each yielding an IVP subproblem of different index. For example, in the reformulated brachistochrone problem the alternative choice of $F(t)$ as the decision variable will yield an index 1 IVP subproblem. Unfortunately, this index 1 formulation is not valid for the case in which we are interested in, where $v(0) = 0$ *and* $u(0) = 0$ (see (36)). But, it is interesting that such an unnatural choice of decision variable leads to a formulation that is, in principle, better behaved numerically. In general, given a choice of decision variables, it seems advisable to choose one that yields an index 1 formulation (Vassiliadis *et al.*, 1994), if one exists. Indeed, a modification of the work described in Lefkopoulos and Stadtherr (1993) could be used to provide automated guidance in the selection of the index 1 formulation. However, when no index 1 formulation exists (as is most likely the case with path equality constraints), it is not clear yet whether there are better or worse choices for the decision variables.

Path-Inequality Constraints. As an illustration of the issues associated with the treatment of path-inequality constraints, consider the set of feasible trajectories between two points in a two-dimensional state space, subject to inequality constraints on the path in state space between these points (Figure 6). Each feasible trajectory connecting the two points is composed of a series of constrained and unconstrained segments, where a constrained segment tracks one or more active path constraints. In addition, a trajectory may be discontinuous at the points of constraint activation or

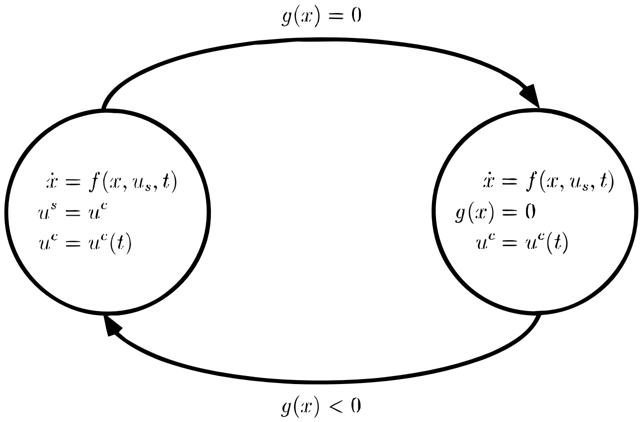

Figure 7. Autonomous switching of hybrid formulation for path-inequality constraints.

deactivation at the boundaries between constrained and unconstrained segments. It is clear that any theoretical or numerical treatment of a path-inequality-constrained problem must determine the sequence of constraint activations and deactivations that characterizes the optimal trajectory.

As in the path-equality constrained case, there are two broad strategies for numerical solution of problems containing path-inequality constraints: either the path-inequality constraints are ignored during solution of the IVP subproblems and the Master NLP attempts to adjust the control so that the solution is feasible at the optimum (Xing and Wang, 1989; Vassiliadis *et al.*, 1994) or attempts are made to remain feasible with respect to path-inequality constraints during solution of the IVP subproblems (Jacobson and Lele, 1969; Feehery and Barton, 1997).

We will consider an approach that attempts to remain feasible with respect to path-inequality constraints during solution of the IVP subproblems that yields an equivalent discontinuous dynamic optimization problem (Feehery and Barton, 1997). During solution of an IVP subproblem, if the state trajectory intersects one of the path-inequality constraints, feasibility may be enforced by augmenting the DAE (22) describing the unconstrained dynamics with the set of active constraints:

$$g_j(x,\dot{x},u,t) = 0 \tag{37}$$

where g_j is the set of all locally active constraints in (24). Each equation in g_j takes up 1 degree of freedom in the augmented DAE, and therefore for each active inequality a control is released and permitted to be determined implicitly by the active inequality. However, an IVP in which the degrees of freedom are fluctuating along the solution trajectory is problematic from the point of view of analysis and solution.

In order to avoid these problems with fluctuating degrees of freedom, we reformulate the path-inequality constrained problem in a higher dimensional space by relabeling the controls u in (21)−(25) as u_c and introducing a new set of variables u_s. For example, consider a problem with a single control and a single path constraint. The following equations will hold whether or not the path constraint is active:

$$f(\dot{x},x,u_s,t) = 0 \tag{38}$$

$$u_c = u_c(t) \tag{39}$$

In unconstrained portions of the trajectory (i.e., when $g(x,\dot{x},u,t) < 0$), the following additional equation holds:

$$u_s = u_c \tag{40}$$

whereas in the constrained portions (i.e., when $g(x,\dot{x},u,t) = 0$), instead the following additional equation holds:

$$g(x,\dot{x},u_s,t) = 0 \tag{41}$$

Note that (40) has been replaced with (41), but the degrees of freedom for the overall dynamic system remain unchanged. Thus, during unconstrained portions of the trajectory the control that actually influences the state (u_s) is equivalent to the forcing function for the system (u_c), whereas in constrained portions of the trajectory u_s is determined implicitly by the active path constraint (41) and is unrelated to u_c. In a control parametrization context, u_c would be prescribed by the control parameters over the entire time horizon, but during constrained portions of the trajectory it would not influence the solution trajectory.

The above equations correspond to a hybrid dynamic system that experiences autonomous switching in response to state events. This behavior is illustrated in Figure 7 using the finite automation representation introduced in Barton and Pantelides (1994). Further, in a fashion similar to that of path-equality constrained problems, the differential index of the constrained segments will, in general, differ from that of the unconstrained segments. A proof of this is given in Feehery and Barton (1997); in fact, this result implies that optimal control of ODE systems subject to path constraints requires either implicit or explicit treatment of DAEs. Given this discussion, it is now evident that a path-constrained dynamic optimization with a single control and a single inequality is equivalent to the following hybrid discrete/continuous dynamic optimization problem:

$$\min_{u_c(t),\ t_f} \phi(x(t_f),u_s(t_f),t_f) + \int_{t_0}^{t_f} L(\dot{x},x,u_s,t)\ dt \tag{42}$$

Subject to:

$$f(\dot{x},x,u_s,t) = 0 \tag{43}$$

$$\begin{cases} u_s = u_c & \forall t \in T\text{: } g(x,\dot{x},u_s,t) < 0 \\ g(x,\dot{x},u_s,t) = 0 & \forall t \in T\text{: } g(x,\dot{x},u_s,t) = 0 \end{cases} \tag{44}$$

The extension to multiple controls and inequalities is evident, provided each inequality is matched with a unique control. Note that this hybrid dynamic optimization problem exhibits the following properties: (a) autonomous switching is defined by state events, (b) the number and order of state events at the optimum is unknown a priori, and (c) in general, the differential index of the system fluctuates at the events. The numerical solution of this problem can now be considered as one piece in the puzzle during the development of generic strategies for the solution of hybrid discrete/continuous dynamic optimization problems.

Hybrid Discrete/Continuous Dynamic Optimization

In this section we will discuss some theoretical results that highlight the properties and classes of hybrid discrete/continuous dynamic optimization problems and point the way to a generic numerical capability for the solution of at least some classes of such problems. Given that the sequence of state events is known in advance, Bryson and Ho (1975) [pp 106−108] state the classical necessary conditions for optimality of a class of ODE-embedded hybrid systems experiencing both switches and jumps. One interesting feature of these results is that the influence functions (costates) $\lambda(t)$ will, in general, be discontinuous at state events. Similarly, results for a class of optimizations involving more general ODE-embedded hybrid systems are presented by Branicky *et al.* (1994). The need to know the sequencing information along the optimum trajectory is reinforced by the comment (Bryson and Ho, 1975) [p 109] (referring to the activation of inequality constraints on controls):

> In solving a particular problem, constrained and unconstrained arcs must be pieced together to satisfy all the necessary conditions. At the junction points of constrained and unconstrained arcs, the control may or may not be continuous; if it is discontinuous it is called a corner. ...Corners may occur at any point, but they are more likely to occur at junction points than in the middle of unconstrained arcs. There seem to be no a priori methods for determining the existence of corners.

In the current context, corners can be interpreted to mean any kind of discontinuity in the controls, equations, states, or activation/deactivation of path inequality constraints. From these results we can conclude that there are two features of the decision process: (a) determining the sequence of discontinuities that characterize the optimal trajectory, and (b) given this information, actually finding the optimal trajectory. Any numerical procedure must attempt to automate both features of the problem.

Given these theoretical results that indicate optimal trajectories do indeed exist, we now consider numerical solution using control parametrization. In principle, in Figure 5 we can just substitute a hybrid discrete/continuous simulation for the continuous IVP subproblem. However, the immediate question that springs to mind is, what are the properties of the Master NLP in such hybrid discrete/continuous problems? In particular, smoothness of the Master NLP will be related to the existence and local continuity of sensitivity functions for hybrid discrete/continuous problems via (31). Rozenvasser (1967) derives the sensitivity equations for a limited class of ODE-embedded hybrid systems (similar to those considered by Bryson and Ho (1975)). Ironically, this result has been neglected in the subsequent literature, and we recently derived Rozenvasser's results independently, only subsequently stumbling upon Rozenvasser's work purely by chance! In Galán *et al.* (1997), Rozenvasser's results are extended to quite general classes of ODE- and DAE-embedded hybrid systems and for the first time existence and uniqueness theorems are proven for nonlinear explicit ODE-embedded systems and linear time invariant DAE-embedded systems.

The key consequences of the existence and uniqueness theorems (beyond the fact that sensitivity functions for hybrid systems do indeed exist and are unique) arise from interpretation of the conditions that must be satisfied for the theorems to hold. In particular, one situation in which the conditions break down are so-called 'critical' values for the parameters v at which the state trajectory $x(v,t)$ is tangent to the hypersurface (46) defined by a state condition. Physically, these are the points in parameter space at which state events appear or disappear, i.e., points at which the sequence of discontinuities along a state trajectory changes qualitatively. The fact that the existence and uniqueness theorems do not hold at these critical points in parameter space indicates the strong possibility of nonsmoothness in the Master NLP. In fact, our preliminary numerical investigations, particularly with path-inequality constrained problems in which the sequence of constraint activations and deactivations changes during the search, indicate that these critical points do indeed introduce nonsmoothness, creating severe problems for gradient-based approaches to the solution of the Master NLP.

Another major consequence of these results is that, in general, the sensitivity trajectories (30) jump at the points in time defined by state events (although, as in the case of discontinuous controls, this property in itself does not imply nonsmoothnessof the Master NLP). For ease of presentation, we will present the conditions defining this jump in an ODE-embedded hybrid system, i.e.:

$$\dot{x} = f(x(v,t),u(v,t),t,v) \tag{45}$$

where the vector field f may experience switching as a consequence of state events. In addition, the junction condition (20) will be used to define the (potentially discontinuous) transfer of states at a time t^* defined implicitly by the state condition:

$$s(x^-(v,t^*),\dot{x}^-(v,t^*),u^-(v,t^*),t^*) = 0 \tag{46}$$

Noting that since t^* *is* not independent, but instead determined implicitly by the parameters v and the state condition (46):

$$\mathrm{d}t^* = \frac{\partial t^*}{\partial v}\,\mathrm{d}v \tag{47}$$

so that the partial derivative of (20) with respect to v is given by

$$\begin{aligned}\frac{\partial j}{\partial x^+}\left(\frac{\partial x^+}{\partial v} + \dot{x}^+\frac{\partial t^*}{\partial v}\right) + \frac{\partial j}{\partial \dot{x}^+}\left(\frac{\partial \dot{x}^+}{\partial v} + \ddot{x}^+\frac{\partial t^*}{\partial v}\right) + \\ \frac{\partial j}{\partial u^+}\left(\frac{\partial u^+}{\partial v} + \frac{\partial u^+}{\partial t^*}\frac{\partial t^*}{\partial v}\right) + \frac{\partial j}{\partial x^-}\left(\frac{\partial x^-}{\partial v} + \dot{x}^+\frac{\partial t^*}{\partial v}\right) + \\ \frac{\partial j}{\partial \dot{x}^-}\left(\frac{\partial \dot{x}^-}{\partial v} + \ddot{x}^-\frac{\partial t^*}{\partial v}\right) + \frac{\partial j}{\partial u^-}\left(\frac{\partial u^-}{\partial v} + \frac{\partial u^-}{\partial t^*}\frac{\partial t^*}{\partial v}\right) + \frac{\partial j}{\partial t^*}\frac{\partial t^*}{\partial v} = 0\end{aligned} \tag{48}$$

which defines the jump in the sensitivities. In the absence of an explicitly specified jump (i.e., a switch not accompanied by a jump), it can be shown (Brull and Pallaske, 1992) that, for discontinuities in ODE-embedded systems, the state should be transferred according to

$$x^+ = x^- \tag{49}$$

In this special case, the general condition (48) reduces to

$$\frac{\partial x^+}{\partial v} = \frac{\partial x^-}{\partial v} + (\dot{x}^- - \dot{x}^+)\frac{\partial t^*}{\partial v} \quad (50)$$

which implies that, even if the state variables x are continuous at a state event, in general there will be a jump in the sensitivities.

Both (48) and (50) introduce a new unknown $\partial t^*/\partial v$, the sensitivity of the state event time, which is defined implicitly by the partial derivative of the state condition (46) with respect to v:

$$\frac{\partial s}{\partial x^-}\left(\frac{\partial x^-}{\partial v} + \dot{x}^-\frac{\partial t^*}{\partial v}\right) + \frac{\partial s}{\partial \dot{x}^-}\left(\frac{\partial \dot{x}^-}{\partial v} + \ddot{x}^-\frac{\partial t^*}{\partial v}\right) + \frac{\partial s}{\partial u^-}\left(\frac{\partial u^-}{\partial v} + \dot{u}^-\frac{\partial t^*}{\partial v}\right) + \frac{\partial s}{\partial t^*}\frac{\partial t^*}{\partial v} = 0 \quad (51)$$

Subject to the conditions asserted by the existence and uniqueness theorems, the linear equation (51) can be solved to determine $\partial t^*/\partial v$ and then the linear system (48) is solved to determine $\partial x^+/\partial v$.

Extensions of these results to nonlinear DAEs (which, in general, will require simultaneous consistent initialization with (17) and possibly its first and higher order derivatives (Pantelides, 1988)) are possible (Galán *et al.*, 1997), although the application of these results is limited in the high-index case because the theoretical definition of general junction conditions at discontinuities in high-index DAEs is an unresolved issue (Brull and Pallaske, 1992; Barton and Park, 1997; Gopal and Biegler, 1997). From these results it is evident that sensitivity functions exist for hybrid systems with implicit switching and/or jumps and can be computed efficiently using extensions of existing combined discrete/continuous simulation and sensitivity computation technology (Barton and Pantelides, 1994; Park and Barton, 1996; Feehery *et al.*, 1997; Tolsma and Barton, 1997a).

The consequences of our existence and uniqueness theorem also reveal that there are at least three classes of hybrid discrete/continuous optimization problems in order of increasing difficulty. The simplest class of problems are ones in which the search space is such that the sequence of state events cannot change, and the sensitivity functions do not jump at state events. From a study of (50), we can see that this will occur in ODE-embedded systems whenever there is a switch, but no elements of the vector field f jump and/or when the event time is not a function of the parameter. For example, consider a model for optimal drug scheduling in cancer chemotherapy (Martin, 1992). Here, the dynamic optimization determines the optimal drug delivery profile ($u(t)$) to decrease the size of a malignant tumor as measured at some particular time in the future. The problem can be formulated as (Carrasco and Banga, 1997):

$$\max_{u(t)} x_1(t_f) \quad (52)$$

Subject to:

$$\dot{x}_1 = -k_1x_1 + k_2(x_2 - k_3)H(x_2,k_3) \quad (53)$$

$$\dot{x}_2 = u - k_4x_2 \quad (54)$$

$$\dot{x}_3 = x_2 \quad (55)$$

$$x_2 \le 50 \qquad \forall t \in [0, t_f] \quad (56)$$

$$x_3 \le 2.1 \times 10^3 \qquad \forall t \in [0, t_f] \quad (57)$$

$$x_1(0) = \ln(100) \qquad x_2(0) = 0 \qquad x_3(0) = 0 \quad (58)$$

$$x_1(21) \ge \ln(200) \qquad x_1(42) \ge \ln(400) \qquad x_1(63) \ge \ln(800) \quad (59)$$

where the tumor mass is given by $N = 10^{12}\exp(-x_1)$ cells, x_2 is the concentration of the drug in the body, x_3 is the cumulative effect of the drug, and k_1, k_2, k_3, k_4, and t_f are fixed constants. The most interesting feature of this problem is the discontinuous function $H(x_2,k_3)$, defined as

$$H(x_2,k_3) = \begin{cases} 1 & \text{if } x_2 \ge k_3 \\ 0 & \text{if } x_2 < k_3 \end{cases} \quad (60)$$

In other words, the drug has no effect on the tumor if the concentration is below some threshold. Further, this discontinuity in the model equations occurs at points in time determined by the state trajectory $x_2(t)$ crossing some threshold, where $x_2(t)$ is not known explicitly but rather determined by the solution of the discontinuous model equations. However, the functional form of (53) is such that the vector field does not jump at the discontinuity, so the sensitivities do not jump and the problem can be solved using control parametrization without any particular modification (Martin, 1992).

The second class of problems are ones in which the search space is such that the sequence of state events cannot change, but the discontinuities are such that the sensitivities may jump. This class of problems can also be solved using gradient-based optimization techniques provided appropriate modifications of the sensitivity evaluation codes are made.

Finally, the most interesting and complex class of problems are those in which the sequence of state events along the trajectory can change during the search process, which is highly likely to lead to a nonsmooth Master NLP, confounding gradient-based optimization algorithms.

Example. Consider the following hybrid system with two modes and a reversible state condition:

$$S_1: \begin{cases} \mathrm{d}x/\mathrm{d}t = 4 - x \\ s:\ -x^3 + 5x^2 - 7x + p \le 0 \\ j = x^+ - x^- = 0 \end{cases} \quad (61)$$

$$S_2: \begin{cases} \mathrm{d}x/\mathrm{d}t = 10 - 2x \\ s:\ -x^3 + 5x^2 - 7x + p > 0 \\ j = x^+ - x^- = 0 \end{cases} \quad (62)$$

The initial mode is S_1 with initial condition $x(0) = 0$. There is only one parameter p that only appears in the transition condition. In Figure 8 we represent the state condition as a function of the state x for three different values of p. When $p = 3.1$, the state event activates at a value of x close to 3. If $p = 2.9$, then there are (1) a transition to mode 2 around $x = 0.8$, (2) a transition to mode 1 around $x = 1.2$, and (3) a transition to mode 2 around $x = 3$. For $p = 3$ the function is tangent at $x = 1$ and crosses zero at $x = 3$. The first point is singular.

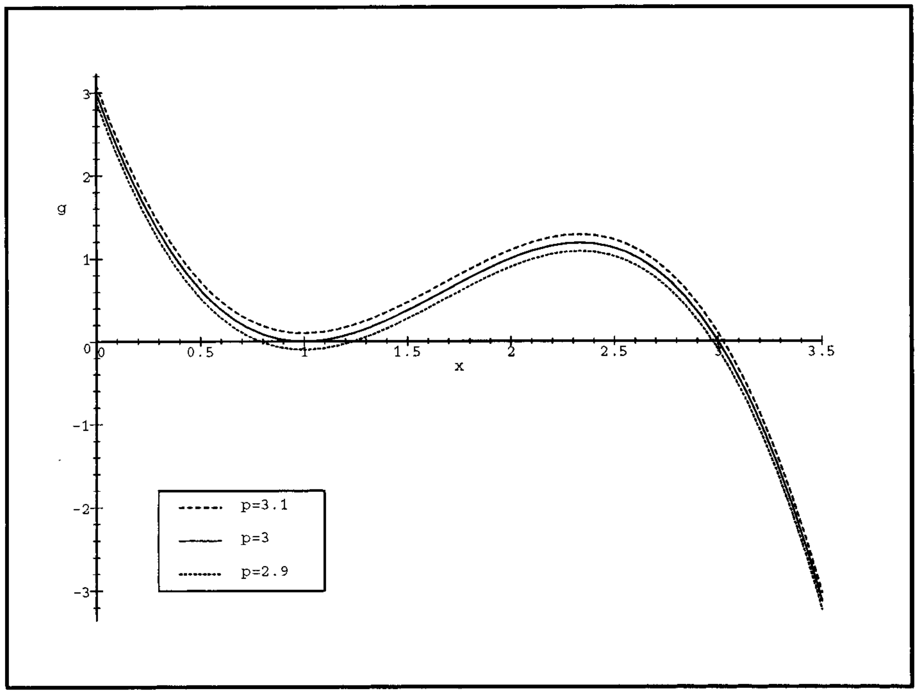

Figure 8. State condition.

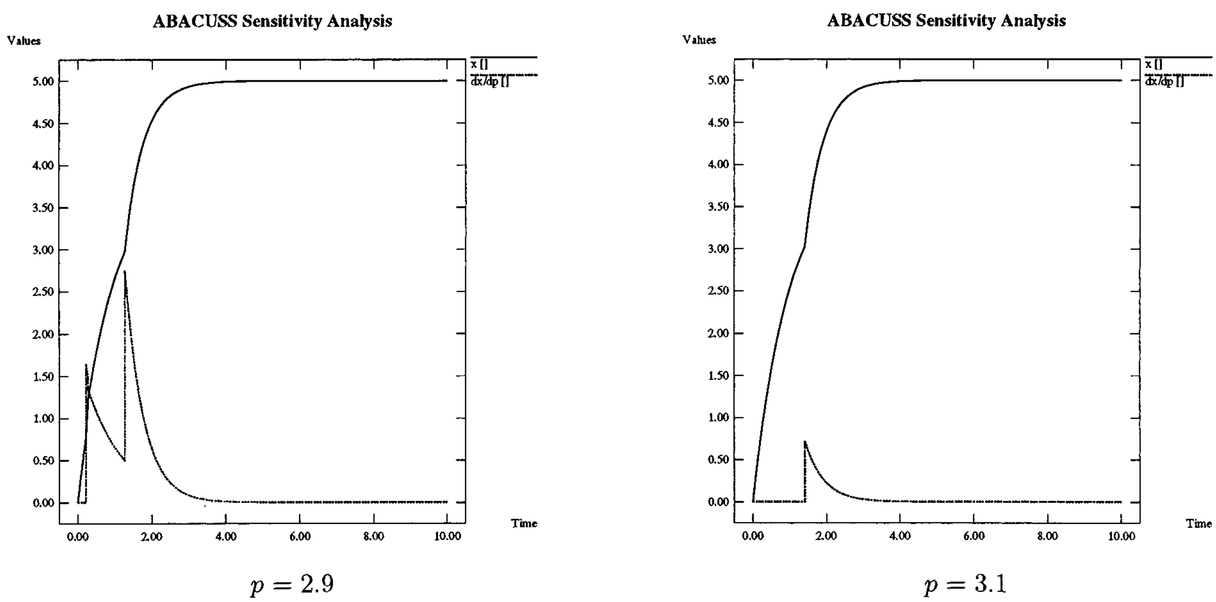

Figure 9. Sensitivities and trajectories for reversible state condition.

It touches 0, switches to the second mode, and immediately changes to mode 1 again. At this point the conditions of the existence and uniqueness theorem are not satisfied.

If we examine the evolution of the state event as a function of the parameter p, we observe that for values less than 3 there are three state event times and they vary continuously. But at $p = 3$ there is a nonsmoothness in the event time: now there is only one event, and the time of the occurrence has jumped from the first event in the previous case. We can see that the changes in the sequence of events are related to these 'critical' points.

The sensitivity functions for $p = 2.9$ and $p = 3.1$ are plotted in Figure 9. In this case, the sensitivity functions are discontinuous at the time of switching. The expression for transfer of the sensitivities is

$$\frac{\partial x^+}{\partial p} = \frac{\partial x^-}{\partial p} - (\dot{x}^+ - \dot{x}^-)\frac{\left[\dfrac{1}{-3x^2 + 10x - 7} - \dfrac{\partial x^-}{\partial p}\right]}{\dot{x}^-} \tag{63}$$

Figure 9 illustrates that the discontinuous effect of the different sequences is perceptible in the trajectory but almost negligible in the long term.

Consider now a system with a nonreversible state condition; i.e., there is a switch from S_1 to S_2, but once there the system remains in that mode:

$$S_1{:} \ = \begin{cases} \mathrm{d}x/\mathrm{d}t = 4 - x \\ s{:}\ -x^3 + 5x^2 - 7x + p \le 0 \\ j = x^+ - x^- = 0 \end{cases} \tag{64}$$

$$S_2{:}\ \{\mathrm{d}x/\mathrm{d}t = 0 \tag{65}$$

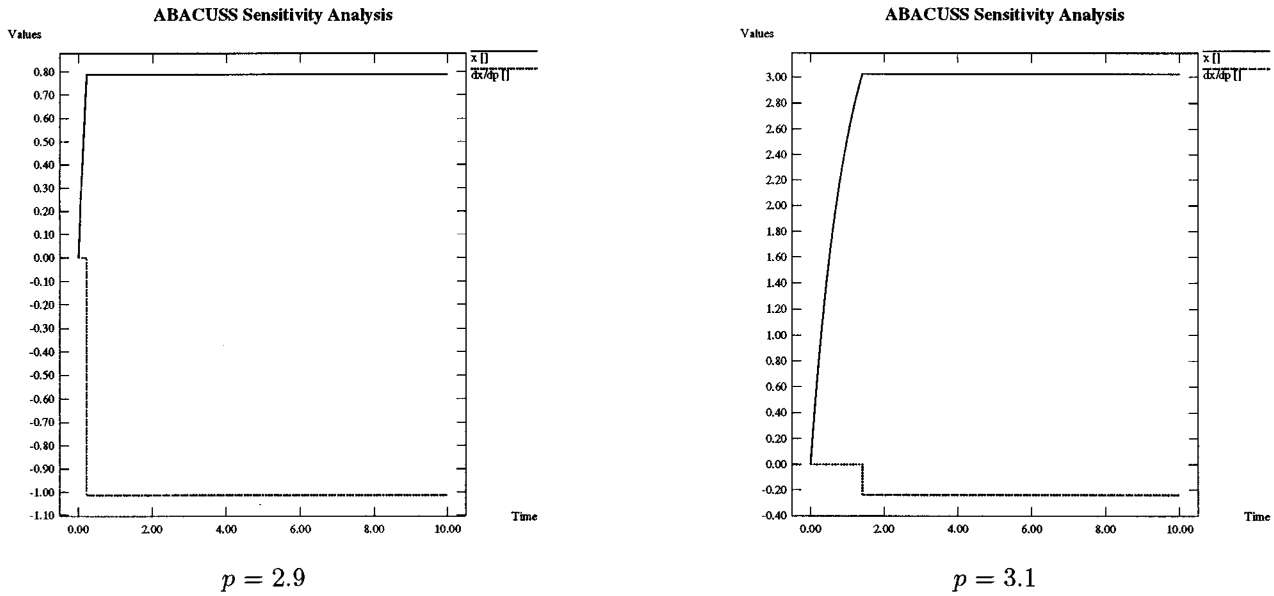

Figure 10. Sensitivities and trajectories for nonreversible state condition.

In Figure 10 the trajectories and sensitivities for this example are plotted. Now the jump in the final values of the state and the sensitivity for the variation of the parameter p is evident.

In essence, there is a combinatorial character to this class of problems. On the other hand, direct or stochastic search algorithms will be less sensitive to this nonsmoothness, so they may offer a practical approach for this class of problems given the current state of technology. Alternatively, since using gradient-based techniques, we can solve efficiently a subproblem in which the sequence does not change; a decomposition in which the sequencing decisions are handled separately may ultimately lead to a more attractive technology.

Mixed-Integer Dynamic Optimization

The existence of design and operational problems that can be formulated as dynamic optimization problems coupled with integer decisions (or mixed-integer dynamic optimization) was discussed above. In the past year or so, some preliminary research on MIDO formulations and solution algorithms has appeared in the literature (Mohideen *et al.*, 1996a,b; Allgor and Barton, 1997; Mohideen *et al.*, 1997).

For example, in (Allgor and Barton, 1997; Allgor, 1997) the class of mixed-integer dynamic optimization problems that conform to the following formulation is considered:

$$\min_{u(t),\, v,\, y,\, t_f} \phi(x(t_f),u(t_f),v,y,t_f) + \int_{t_0}^{t_f} L(x(t),u(t),v,y,t)\,\mathrm{d}t \tag{66}$$

Subject to:

$$f(x(t),\dot{x}(t),u(t),v,y,t) = 0 \qquad \forall t \in [t_0,\, t_f] \tag{67}$$

$$g(x(t),\dot{x}(t),u(t),v,y,t) \le 0 \qquad \forall t \in [t_0,\, t_f] \tag{68}$$

$$k_p(x(t_p),\dot{x}(t_p),u(t_p),v,y,t_p) \le 0 \qquad \forall p \in \{0, ..., n_p\} \tag{69}$$

where $x(t)$ are the continuous variables describing the state of the dynamic system, $u(t)$ are continuous controls whose optimal time profiles on the interval $[t_0, t_f]$ are required, v are continuous time-invariant parameters whose optimal values are also required, and y are a special set of time-invariant parameters that can only take 0−1 values. For example, in a formulation for the optimal development of batch processes y may represent decisions such as the selection of solvents from a list of candidates, the selection of equipment items from an inventory, and the selection of a separation/recycle structure for the process (Allgor, 1997). The point constraints (69) encompass as a special case constraints that only involve the binary variables y, such as logical relationships that must be satisfied (e.g., an item of equipment may only be used for certain processing tasks).

All currently reported techniques for the numerical solution of MIDO problems rely on extensions of decomposition approaches for solving mixed-integer nonlinear optimization problems (MINLP) (Floudas, 1995). In such algorithms (see, for example, Figure 11), the solution process is composed of an iteration between two subproblems: a Master problem which yields a lower bound on the solution and an update for the integer decision variables, and a Primal problem which solves the optimization problem for fixed values of the integer decision variables and yields a upper bound on the solution. Given this decomposition, a correct algorithm should generate a nondecreasing sequence of lower bounds and a nonincreasing sequence of upper bounds that will converge to the optimum in a finite number of iterations. In the context of the MIDO formulation (66)−(69), the Primal problem (i.e., solve (66)−(69) with y fixed) is clearly a conventional dynamic optimization problem that can, in principle, be solved with any of the numerical procedures discussed above. On the other hand, the derivation of a generic Master problem for MIDO that possesses the strict properties required for convergence is probably impossible except in the case of the most trivial of dynamic models.

In the context of the integration of design and control, Mohideen *et al.* (1996a,b; 1997) formulate an ad hoc Master problem that unfortunately has the property of convergence to an arbitrary point which is not even guaranteed to be a local minimum. Alternatively, in

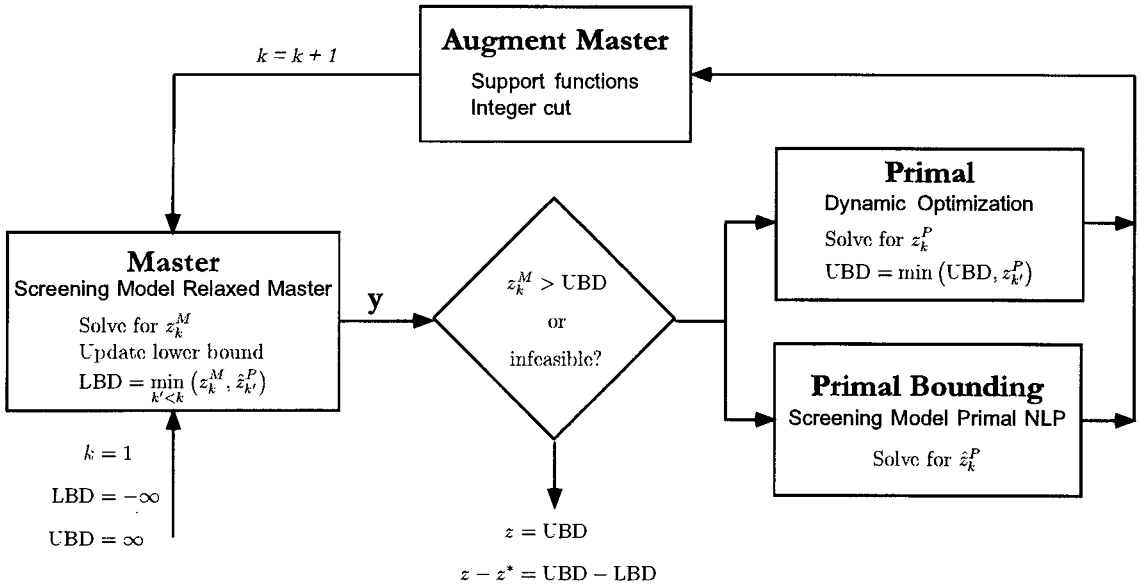

Figure 11. Decomposition approach for mixed-integer dynamic optimization.

the context of optimal batch process development (Allgor and Barton, 1997; Allgor, 1997), the use of a MILP or convex MINLP *screening* model as the Master problem in MIDO is proposed. The screening model is a simplified model derived by a separate modeling effort that is guaranteed to yield a rigorous lower bound on the solution to formulation (66)−(69). This latter approach has more desirable properties in that it (a) provides a rigorous pruning of the discrete decision space, (b) terminates in a finite number of iterations, (c) is guaranteed to systematically improve the solution, and (d) on termination yields rigorous information on the distance from the best solution found to the global solution. On the other hand, the screening approach requires large quantities of domain-specific information. The further development of MIDO algorithms and formulations is likely to receive much research attention in the coming years.

Conclusions and Future Directions

We have considered early steps in the development of generic numerical technologies for the solution of discontinuous plantwide dynamic optimization problems. Such a technology will have many applications to problems encountered in process design and operations. Examples include the selection of optimal integrated plant-wide operating policies and equipment allocation in batch processes, the automated design of operating procedures and even design for operability, the integration of process and controller design, and the optimal design and operation of periodic processes such as pressure swing adsorption or reverse-flow reactors.

Even in this past year, the computational efficiency of technologies for obtaining local numerical solutions to large-scale continuous dynamic optimizations has increased dramatically, facilitating the solution of problems with thousands of state variables (Charalambides *et al.*, 1995; Charalambides, 1996) within a matter of minutes on engineering workstations (Feehery *et al.*, 1997). Even so, a number of exciting research issues remain. In terms of the size of problems amenable to analysis, the size of the DAE (22) is unlikely to be limiting (at least in the case of control parametrization where problem sizes with 10 000−100 000 states are accessible even with existing technology). On the other hand, a large number of parameters in the Master NLP, which may arise from many controls in a plantwide model, higher order basis functions, many finite elements, and/or different finite element grids for different controls, is potentially a major bottleneck. In particular, the Master NLP becomes a relatively large (maybe 1 000−10 000 variables) dense NLP, and given the state of the art in NLP technology, this is likely to limit plantwide dynamic optimization efforts. Further, if there truly are 1000−10 000 parameters in the optimization, the sensitivities indicate that partial derivatives of the point constraints with respect to *all* the parameters exist (eq 31), thwarting approaches such as range and null space decomposition SQP (Vasantharajan and Biegler, 1988; Vasantharajan *et al.*, 1990). Hence, what will be needed to push this envelope is a technology that can solve dense NLPs with several thousands of parameters and for which we do not have the explicit functional form of the constraints and objective. The multimodality of the NLPs that arise from direct strategies for dynamic optimization problems also poses a major challenge (Luus *et al.*, 1992; Banga and Seider, 1996), particularly if we attempt to mitigate this multimodality in large-scale plantwide formulations. While we can obtain local solutions to large-scale problems relatively efficiently, methods that provide some hope of always finding good local optima are very costly and do not scale well to large-scale problems.

On the other hand, dynamic optimization problems appear particularly well suited to exploiting parallel computing architectures. For example, we are developing a framework in which advanced computing architectures are exploited at multiple levels of the control parametrization problem decomposition. At the top level, careful problem formulation (e.g., decoupling of batch units through intermediate storage) can lead to a number of independent IVP subproblems that can be solved in parallel. Then within each IVP subproblem the sensitivity equations can be solved in parallel (Maly and Petzold, 1996). Finally, parallelization at the lowest level can include the linear algebra (Zitney and Stadtherr, 1993) and function and derivative evaluation using parallel automatic differentiation techniques (Bischof, 1991). Another interesting development is the optimization of (parabolic) partial differential equation embedded systems (Petzold *et al.*, 1996). This area has barely been explored and will pose many new challenges; for example, what consequences do the properties of hyperbolic systems have?

The numerical optimization of discontinuous dynamic systems is a technology very much in its infancy. We have identified two broad classes of problems: hybrid discrete/continuous dynamic optimization problems in which the dynamics are discontinuous and mixed-integer dynamic optimization problems that are in addition coupled with integer decision variables. Further, classical path-inequality constrained problems can be reinterpreted in a hybrid discrete/continuous dynamic optimization framework. New theoretical results (Galán *et al.*, 1997) indicate that gradient-based optimization of certain classes of hybrid discrete/continuous dynamic optimization problems is indeed possible but also highlight the difficulties in solving probably the most important class of problems, those in which the optimization effectively makes sequencing decisions. We believe that the study and development of technologies to make sequencing decisions during a dynamic optimization is the key issue in the practical realization of many of the applications discussed above. Two avenues for research that immediately spring to mind are bifurcation style studies of the critical points in parameter space at which the sequence of events changes qualitatively and decomposition approaches in which the sequencing decisions are made in a coupled subproblem. Finally, some preliminary work on mixed-integer dynamic optimization problems is appearing in the literature, and this will also be an area of exciting developments in the next few years. Ultimately, in the case of both hybrid discrete/continuous dynamic optimization and mixed-integer dynamic optimization, it is the combinatorial nature of these problems that challenges us for the future.

Acknowledgment

This work was supported by the U.S. Department of Energy under Grant DE-FG02-94ER14447, the National Science Foundation under Grant CTS-9703623, and Cátedra Repsol of the Universidad Politécnica de Madrid.

Literature Cited

Allgor, R. J. Modeling and computational issues in the development of batch processes. Ph.D. Dissertation, Massachusetts Institute of Technology, Cambridge, MA, 1997.

Allgor, R. J.; Barton, P. I. Mixed integer dynamic optimization. *Comput. Chem. Eng.* **1997**, *21,* S451−S456.

Allgor, R. J.; Barrera, M. D.; Barton, P. I.; Evans, L. B. Optimal batch process development. *Comput. Chem. Eng.* **1996**, *20,* 885−896.

Allgor, R. J.; Evans, L. B.; Barton, P. I. Screening models for batch process development: I. Design targets for reaction/distillation networks. *Chem. Eng. Sci.* **1997**, submitted.

Alur, R., Henzinger, T. A., Sontag, E. D., Eds. *Hybrid Systems III;* Lecture Notes in Computer Science 1066; Springer-Verlag: Berlin, 1996.

Antsaklis, P., Nerode, W., Kohn, A., Sastry, S., Eds. Hybrid Systems II; Lecture Notes in Computer Science 999; Springer-Verlag: Berlin, 1995.

Banga, J. R.; Seider, W. D. Global optimization of chemical processes using stochastic algorithms. In *State of the Art in Global Optimization: Computational Methods and Applications*; Floudas, C. A., Pardalos, P. M., Eds.; Kluwer Academic Pub.: Dordrecht, The Netherlands, 1996.

Barrera, M. D.; Evans, L. B. Optimal design and operation of batch processes. *Chem. Eng. Commun.* **1989**, *82*, 45−66.

Barton, P. I. The modelling and simulation of combined discrete/continuous processes. Ph.D. Dissertation, University of London, London, 1992.

Barton, P. I.; Pantelides, C. C. Modelling of combined discrete/continuous processes. *AIChE J.* **1994**, *40*, 966−979.

Barton, P. I.; Park, T. Analysis and control of combined discrete/continuous systems: Progress and challenges in the chemical process industries. *AIChE Symp. Ser.* **1997**, *93*, 102−114.

Barton, P. I.; Allgor, R. J.; Feehery W. F. DSL48S a large-scale differential-algebraic and parameteric sensitivity solver. Technical report, Massachusetts Institute of Technology, 1997; ABACUSS Project Report No. 97/1.

Bellman, R. *Dynamic Programming;* Princeton University Press: Princeton, NJ, 1957.

Bhatia, T.; Biegler, L. T. Dynamic optimization in the planning and scheduling of multi-product batch plants. *Ind. Eng. Chem. Res.* **1996,** *35*, 2234−2246.

Bischof, C. H. Issues in parallel automatic differentiation. In *Automatic Differentiation of Algorithms: Theory, Implementation and Application;* Griewank, A., Corliss, G. F., Eds.; SIAM: Philadelphia, PA, 1991.

Branicky, M.; Borkar, V. S.; Mitter, S. K. A unified framework for hybrid control: Background, model, and theory. Proceedings of the 33rd IEEE Conference on Decision and Control, Lake Buena Vista, FL, 1994.

Brenan, K. E.; Campbell, S. L.; Petzold, L. R. *Numerical Solution of Initial Value Problems in Differential-Algebraic Equations;* SIAM: Philadelphia, 1996.

Brull, L.; Pallaske, U. On differential algebraic equations with discontinuities. *Z. Angew. Math. Phys.* **1992**, *43,* 319−327.

Bryson, A. E.; Ho, Y. *Applied Optimal Control*; Hemisphere: New York, 1975.

Carrasco, E. F.; Banga, J. R. Dynamic optimization of batch reactors using adaptive stochastic algorithms. *Ind. Eng. Chem. Res.* **1997**, *36,* 2252−2261.

Charalambides, M. S. Optimal design of integrated batch processes. Ph.D. Dissertation, University of London, London, 1996.

Charalambides, M. S.; Shah, N.; Pantelides, C. C. Optimal design of integrated batch processes. I. preliminary process design. AIChE Annual Meeting, Miami, FL, 1995.

Cole, J. B.; Yount, K. B. Applications of dynamic simulation to industrial control problems. *Adv. Instrum. Control* **1993**, *48* (2), 1337−1344.

Debelak, K. A.; Karsai, G.; Chen, G.; Zitney, S.; DeCaria, F. O. Application of dynamic simulation to an industrial process control system. AIChE Annual Meeting, Miami, FL, 1995.

Debling, J. A.; Han, G. C.; Kuijpers, F.; VerBurg, J.; Ray, W. H. Dynamic modeling of product grade transitions for olefin polymerization processes. *AIChE J.* **1994**, *40*, 506−520.

Denbigh, K. G. Optimum temperature sequences in reactors. *Chem. Eng. Sci.* **1958**, *8*, 125−132.

Dimitriadis, V. D.; Shah, N.; Pantelides, C. C. Model-based safety verification of discrete/continuous chemical processes; In *AIChE Annual Meeting,* San Francisco, 1994.

Diwekar, U. M.; Madhavan, K. P.; Swaney, R. E. Optimization of multicomponent batch distillation columns. *Ind. Eng. Chem. Res.* **1989**, *28*, 1011−1017.

Evans, S. F.; Wylie, P. A plant simulator for the THORP nuclear fuel reprocessing plant at Sellafield. *Dynamic Simulation in the Process Industries;* UMIST: Manchester, 1990.

Feehery, W. F.; Barton, P. I. A differentiation-based approach to simulation and dynamic optimization with high-index differential-algebraic equations. In *Computational Differentiation Techniques, Applications, and Tools;* Berz, M., Bischof, C., Corliss, G., Griewank, A., Eds.; SIAM: Philadelphia, 1996.

Feehery, W. F.; Barton, P. I. Dynamic optimization with state variable path constraints. *Comp. Chem. Eng.* **1997**, in press.

Feehery, W. F.; Banga, J. R.; Barton, P. I. A novel approach to dynamic optimization of ODE and DAE systems as high index problems. AIChE Annual Meeting, Miami Beach, FL, 1995.

Feehery, W. F.; Tolsma, J. E.; Barton, P. I. Efficient sensitivity analysis of large-scale differential-algebraic systems. *Appl. Num. Math.* **1997**, *25*, 41−54.

Floudas, C. A. *Nonlinear and Mixed-Integer Optimization Fundamentals and Applications*; Oxford University Press: Oxford, U.K., 1995.

Galán, S.; Barton, P. I. Dynamic optimization formulations for operating procedure synthesis. AIChemE Annual Meeting, Los Angeles, CA, 1997.

Galán, S.; Feehery, W. F.; Barton, P. I. Parametric sensitivity functions for hybrid discrete/continuous systems. *Appl. Num. Math.* **1997**, submitted.

Gopal, V.; Biegler L. T. A successive linear programming approach to consistent initialization and reinitialization after discontinuities of differential-algebraic equations. *SIAM J. Sci. Comput.* **1997**, in press.

Grassi, V. G. Dynamic simulation as a tool to integrate process design and process control *Adv. Instrum. Control* **1993**, *48* (2), 1346−1365.

Grossman, R. L., Nerode, A., Rawn, A. P., Rischel, H., Eds. *Hybrid Systems*; Lecture Notes in Computer Science 736; Springer-Verlag: Berlin, 1993.

Han, C.; Lakshmanan, R.; Bakshi, B.; Stephanopoulos, G. Nonmonotonic reasoning: the synthesis of operating procedures in chemical plants. *Adv. Chem. Eng.* **1995**, *22*, 313−376.

Jacobson, D. H.; Lele, M. M. A transformation technique for optimal control problems with a state variable inequality constraint. *IEEE Trans. Autom. Control* **1969**, *5*, 457−464.

Kirk, D. E. *Optimal Control Theory: An Introduction;* Prentice-Hall: Englewood Cliffs, NJ, 1970.

Kraft, D. On converting optimal control problems into nonlinear programming problems. *Comput. Math. Prog.* **1985**, *15*, 261−280.

Kraft, D. Algorithm 733: TOMP—fortran modules for optimal control calculations. *ACM Trans. Math. Software* **1994**, *20*, 262−281.

Lakshmanan, R.; Stephanopoulos, G. Synthesis of operating procedures for complete chemical plants—I. Hierarchical, structured modelling for nonlinear planning. *Comput. Chem. Eng.* **1988**, *12*, 985−1002.

Lefkopoulos, A.; Stadtherr, M. A. Index analysis of unsteady-state chemical process systems—I: An algorithm for problem formulation. *Comput. Chem. Eng.* **1993**, *17*, 399−413.

Longwell, E. J. Dynamic modeling for process control and operability. *Adv. Instrum. Control* **1993**, *48* (2), 1323−1336.

Luus, R. Application of dynamic programming to high-dimensional nonlinear optimal control problems. *Int. J. Control* **1990**, *52*, 239−250.

Luus, R.; Dittrich, J.; Keil, F. J. Multiplicity of solutions in the optimization of a bifunctional catalyst blend in a tubular reactor. *Can. J. Chem. Eng.* **1992**, *70*, 780−785.

Maly, T.; Petzold, L. R. Numerical methods and software for sensitivity analysis of differential-algebraic systems. *Appl. Num. Math.* **1996**, *20*, 57−79.

Mani, S.; Shoor, S. K.; Pedersen, H. S. Experience with a simulator for training ammonia plant operators. *Plant/Oper. Prog.* **1990**, *10*, 6−10.

Martin, R. B. Optimal control drug scheduling of cancer chemotherapy. *Automatica* **1992**, *28*, 1113−1123.

Mattsson, S. E.; Soderlind, G. Index reduction in differential-algebraic equations using dummy derivatives. *SIAM J. Sci. Stat. Comput.* **1993**, *14*, 677−692.

Mohideen, M. J.; Perkins, J. D.; Pistokopoulos, E. N. Optimal design of dynamic systems under uncertainty. *AIChE J.* **1996a**, *42*, 2251−2272.

Mohideen, M. J.; Perkins, J. D.; Pistokopoulos, E. N. Optimal synthesis and design of dynamic systems under uncertainty. *Comput. Chem. Eng.* **1996b**, *20*, S895−S900.

Mohideen, M. J.; Perkins, J. D.; Pistokopoulos, E. N. Towards an efficient numerical procedure for mixed integer optimal control. *Comput. Chem. Eng.* **1997**, *21*, S457−S462.

Naess, L.; Mjaavatten, A.; Li, J. O. Using dynamic process simulation from conception to normal operation of process plants. *Comput. Chem. Eng.* **1993**, *17*, 585−600.

Neuman, C. P.; Sen, A. A suboptimal control algorithm for constrained problems using cubic splines. *Automatica* **1973**, *9*, 601−603.

Pantelides, C. C. The consistent initialization of differential/algebraic systems. *SIAM J. Sci. Stat. Comput.* **1988**, *9*, 213−231.

Papageorgaki, V.; Garber, W.; Lui, S. An integrated tool for steady state and dynamic simulation of cryogenic processes. AIChE Annual Meeting, Miami, FL, 1995.

Park, T. Formal verification and dynamic validation of logic-based control systems. Ph.D. Dissertation, Massachusetts Institute of Technology, Cambridge, MA, 1997.

Park, T.; Barton, P. I. Towards dynamic simulation of a process and its automatic protective system. *International Symposium and Workshop on Safe Chemical Process Automation;* AIChE/CCPS: New York, 1994.

Park, T.; Barton, P. I. State event location in differential-algebraic models. *ACM Trans. Mod. Comput. Simul.* **1996**, *6*, 137−165.

Petzold, L. Differential/algebraic equations are not ODE's. *SIAM J. Sci. Stat. Comput.* **1982**, *3*, 367−384.

Petzold, L.; Rosen J. B.; Gill, P. E.; Jay, L. O.; Park, K. Numerical optimal control of parabolic PDEs using DASOPT. Proceedings of the IMA Workshop on Large-Scale Optimization, 1996.

Pontryagin, L. S.; Boltyanskii, V. G.; Gamkrelidze, R. V.; Mishchenko, E. F. *The Mathematical Theory of Optimal Processes;* Interscience Publishers: New York, 1962.

Rippin, D. W. T. Simulation of single and multiproduct batch chemical plants for optimal design and operation. *Comput. Chem. Eng.* **1983**, *7*, 137−156.

Rivas, J. R.; Rudd, D. F. Synthesis of failure-safe operations. *AIChE J.* **1974**, *20*, 320−325.

Rozenvasser, E. N. General sensitivity equations of discontinuous systems. *Autom. Remote Control* **1967**, 400−404.

Sargent, R. W. H.; Sullivan, G. R. The development of an efficient optimal control package. *Proc. 8th IFIP Conf. Optim. Tech., Pt.* 2 1977.

Tolsma, J. E.; Barton, P. I. Efficient computation of sparse Jacobians. *SIAM J. Sci. Comput.* **1997a**, accepted for publication.

Tolsma, J. E.; Barton, P. I. On computational derivatives. *Comput. Chem. Eng.* 1**997b**, in press.

Tsang, T. H.; Himmelblau, D. M.; Edgar, T. F. Optimal control via collocation and non-linear programming. *Int. J. Control* **1975**, *21*, 763−768.

Vasantharajan, S.; Biegler, L. T. Large-scale decomposition strategies for successive quadratic programming. *Comput. Chem. Eng.* **1988**, *12*, 1087−1101.

Vasantharajan, S.; Viswanathan, J.; Biegler L. T. Reduced successive quadratic programming implementation for large-scale optimization problems with smaller degrees of freedom. *Comput. Chem. Eng.* **1990**, *14*, 907−915.

Vassiliadis, V. S. Computational solution of dynamic optimization problems with general differential-algebraic constraints. Ph.D. Dissertation, University of London, London, 1993.

Vassiliadis, V. S.; Sargent, R. W. H.; Pantelides, C. C. Solution of a class of multistage dynamic optimization problems. 2. Problems with path constraints. *Ind. Eng. Chem. Res.* **1994**, *33*, 2123−2133.

Xing, A.; Wang, C. Applications of the exterior penalty method in constrained optimal control problems. *Opt. Control Appl. Methods* **1989**, *10*, 333−345.

Zitney, S. E.; Stadtherr, M. A. Frontal algorithms for equation-based chemical process flowsheeting on vector and parallel computers. *Comput. Chem. Eng.* **1993**, *17*, 319.

Zitney, S. E.; Brull, L.; Lang, L.; Zeller, R. Plantwide dynamic simulation on supercomputers. *AIChE Symp. Ser.* **1995**, *91*, 313−316.

Received for review October 22, 1997
Revised manuscript received January 7, 1998
Accepted January 8, 1998

IE970738Y