On The Optimization Of Differential-Algebraic Process Systems

by

J. E. Cuthrell, L. T. Biegler

EDRC-06-21-86

September 1986

ON THE OPTIMIZATION OF DIFFERENTIAL-ALGEBRAIC PROCESS SYSTEMS

by

J. E. Cuthrell and L. T. Biegler *

CARNEGIE-MELLON UNIVERSITY

To be presented at Annual AIChE Meeting
Miami, FL
November, 1986

Session on Applied Mathematics II

Paper f

* Author to Whom Correspondence should be Addressed

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

The optimization of systems of differential and algebraic equations is required in many important areas of Chemical Engineering. Solution of these optimization problems is usually done either by replacing the differential equation based models with simple algebraic models (as in Flowsheet Optimization) or by using an implicit numerical integration scheme to solve the differential equations (as in process control). Each of these approaches has serious drawbacks when applied to large problems. The use of simple models renders any solution obtained sub-optimal (that is, optimal with respect to the level of approximation the model affords), while optimization involving repeated numerical integration can be prohibitively expensive. Furthermore there is presently no general and accurate method (other'than doing them analytically) to handle optimal control problems which include discontinuous and singular profiles.

This paper describes a method to solve the above type of problems and avoid the stated difficulties. Finite element collocation is used to convert the differential equations to a set of algebraic residual equations with unknown coefficients. Then a nonlinear program is formulated with the residuals incorporated as equality constraints and the coefficients as decision variables. Collocation points are fixed relative to the finite element knots but adaptive knot placement is done to minimize the error due to collocation. The conditions for optimal knot placement are necessary and sufficient and can be incorporated as additional equality constraints in the nonlinear program. All of the equality constraints (i.e. the collocation equations and knot placement constraints) are then solved simultaneously with the optimization problem, thus requiring only a single solution of the approximated model. The effectiveness of this strategy is illustrated by optimizing a reactor with state variable constrain is and steep temperature profiles.

In addition, singular problems and profiles with jump conditions are treated by introducing an extra level of elements (super-elements) to allow for the location of discontinuities. These locations are chosen while solving the overall optimization problem. A simple optimal control problem is also solved to illustrate this strategy.

1. INTRODUCTION

Optimization involving both differential and algebraic equation models represents one of the new frontiers facing optimization techniques. Currently optimization of differential equations and algebraic equations can be done separately by various methods. Successive Quadratic Programming (SQP) (Han (1977), Powell (1977)) and Reduced Gradient techniques (Murtagh & Saunders (1978)) are available for handling nonlinear algebraic problems, while variational calculus (Bryson & Ho (1975)) can be used for differential equation optimizations.

Up to this point, however, no general and accurate method is available for handling differential-algebraic optimization problems. Standard nonlinear programming (NLP) methods cannot be used without resorting to an often expensive numerical integration scheme. Furthermore, few options exist (Sargent & Sullivan (1977)), even in this case, for imposing constraints on continuous profiles and for handling discontinuities and/or singular profiles. Methods based on optimal control theory, conversely, have difficulty handling even simple types of algebraic equations. Moderately sized (say 3 or more state equations) nonlinear two point boundary value problems, for example, composed of nonlinear ODEs and algebraic boundary conditions, can be difficult to solve using numerical integration. Furthermore, repeated numerical integration (often implicit) can make these optimizations very expensive. Additional complexities such as discontinuous/singular profiles, profile constraints, as well as other complex algebraic side conditions (see Bryson & Ho (1975)) can make these problems almost impossible to solve using variational calculus.

In this paper we present a method that solves, under certain conditions, differential-algebraic optimization problems efficiently and accurately. Here the differential equations will be discretized using polynomial approximation and orthogonal collocation. The resulting algebraic equations are then written as part of an NLP which is solved with an SQP optimization technique. Accuracy of the approximation is guaranteed by developing a set of finite element knot placement equations which represent the sufficient conditions for error minimization. These equations also become part of the NLP. In addition to finite elements an extra level of elements, super-elements, is introduced. Inclusion of these extra breakpoints allows for the approximation of discontinuous/singular profiles.

In the remainder of this section a general statement of a differential-algebraic

optimization will be made along with a discussion of some of its features. Section 2 will formulate the problem as an NLP through use of orthogonal collocation on finite elements. A discussion of optimal placement of finite element knots will follow with reference to some standard knot placement methods. Next, in Section 3, the need for super-elements will be outlined and illustrated by example. A final example will be presented in Section 4 to demonstrate the effectiveness of the knot placement equations.

1.1. A NONLINEAR PROGRAM WITH ODE MODELS

A differential-algebraic optimization problem (DAOP) which includes an ODE model (written here for the sake of convenience as an initial value problem) can be posed as:

$$
\begin{aligned}
&\underset{x,Utt),ZCt)}{\text{Min}} && 4>(x,Utt).Ztt)) && \text{(DAOP)}\\
&\text{S.L} && c(x.UCt),Ztt)) = 0\\
& && g(x,Utt).Ztt)) \pounds 0\\
& && \dot{Z}tt) = F(x.Utt).ZttU) && \text{X6CO.1]}\\
& && Z(0) = Z_o\\
& && x^L \pounds x \pounds x^u\\
& && U^L \pounds Utt) \pounds U^u\\
& && z^L \leq> nt) \diamond z^u
\end{aligned}
$$

where

- 4> = an objective function
- g,c = design constraint vectors
- x = decision variable vector
- ZCt) = state profile vector (of dimension M)
- UCt) = control profile
- $x\backslash x^u$ = variable bounds
- $U\backslash U^U$ = control profile bounds
- $Z^L{}_t Z^u$ = state profile bounds

The differential equations, as written, include a possible dependence on some scalar parameters (for example length of a reactor or pressure in a column). These we've chosen to include in the decision variable vector x. The algebraic constraints and objective function in DAOP are written as functional indicating a possible dependence upon the continuous control profile UCt) and state profiles Ztt).

As stated DAOP cannot be solved directly either by typical nonlinear programming techniques or optimal control methods. With an NLP technique one cannot directly optimize a continuous control profile nor is it possible to impose bounds on continuous profiles. Optimal control methods, on the other hand, generally do not deal with scalar variables nor is it possible to handle efficiently general algebraic constraints (such as c or g).

It is possible, under certain conditions, to handle DAOP by using both an NLP optimization method and a numerical integrator (Cuthrell & Biegler (1985)). This can be done in a straightforward manner only if no control profiles exist and no profile bounds are enforced. Under these conditions DAOP can be solved with a two step procedure where the optimizer chooses x in an outer loop followed by numerical integration of the ODE model in the inner loop. This approach, however, often requires expensive integration of sensitivity equations of the form

$$\frac{\partial \dot{Z}(t)}{\partial x} = \frac{\partial F(x,Z(t),t)}{\partial x} + \frac{\partial Z(t)}{\partial x}\,\frac{\partial F(x,Z(t),t)}{\partial Z(t)}$$

along with the state equations. In general, however, DAOP cannot be solved by any available method. We are aware of only two attempts at solving DAOP. Bapat & Heydweiller (1985) used polynomial approximation to discretize the ODE models in some optimal control problems. For problems containing a discontinuous control profile the time domain was divided into two elements each containing a state and control profile polynomial approximation. Here the location of the discontinuity was included into the optimization as a variable and found by the search procedure. This method, however, met with only limited success since a relatively simple optimization technique was used; moreover the question of accurate profile approximations was not directly addressed. Sargent & Sullivan (1977) considered a form of DAOP by parameterizing the control profile over variable time intervals and transforming state variable constraints into constraints enforced at final time. This method was successful in addressing some of the above problems but still required repeated numerical solution of the ODE model.

To avoid the above problems we present a method in Section 2 which allows simultaneous solution of the optimization problem and ODE model.

2. AN NLP FORMULATION FOR DIFFERENTIAL-ALGEBRAIC OPTIMIZATION PROBLEMS

This section will focus on two key ideas. First, by using orthogonal collocation the differential equations in DAOP can be converted to a set of approximating algebraic equations. And, second, the error of the approximation can be minimized by incorporating into the resulting NLP a finite element method which adaptively chooses the locations of the finite element knots.

2.1- DISCRETIZATION OF THE ODE MODELS

Recall the ODE model presented in DAOP as:

$$ht) = Hxvtt\backslash m)x) \qquad *eco,i] \qquad \text{(i)}$$

$$Z(0) = Z_o$$

Next we make use of the two polynomials written in Lagrange form

$$z_{K+1}(t) = \sum_{i=0}^{K} z_i \, \phi_i(t) \qquad *\&= \prod_{j=0.i}^{K} \frac{(t - t_j)}{(t_i - t_j)} \qquad (2)$$

$$u_K(t) = \sum_{i=1}^{K} u_i \, \psi_i(t) \qquad \psi_i(t) = \prod_{j=1,i}^{K} \frac{1}{(t_i - t_j)} \qquad (3)$$

where z_v CO denotes a (K-^l)th order (degree < K+l) polynomial and u.,Ct) a Kih order (degree < K) polynomial (Note that $<f_t>W$ is a polynomial df degree K and $\backslash f/ U)$ is a polynomial of degree K-1.) The difference in the orders is due to the existence of the initial condition for ZCC). And, the notation j=0,i indicates that j=0....,i-1.i+l*....K. The Lagrange form polynomial has the desirable property that (for z_K+,0t) for example)

$$z_{K+1}(t_i) = z_i$$

which is due to the Lagrange condition

$$\phi_i(t_j) = 6_{it} \qquad (6_{ij} = \text{Kronecker delta}).$$

Since for chemical engineering problems the states and controls represent quantities like temperature or concentration, using Lagrange polynomials produces coefficients z_i and u_i which are physically meaningful quantities. This becomes useful when providing variable bounds, initializing a profile, or interpreting solution profiles. For other types of polynomial forms (e.g. B-splines) the coefficients do not have these features.

Substitution of (2) and (3) into (1) yields the residual equation:

$$R(t) = \sum_{i=0}^{K} z_i \frac{d}{dt} u_{|c}(t)^{\wedge}{}_1(t)^{\wedge}) \quad , \qquad (4)$$

with $z_Q = Z_o$

which still remains a function of time. Discretization of the residual equation is next done through use of one of the Methods of Weighted Residuals (see Villadsen & Michelsen (1978). Finlayson (1972)). We choose the method of collocation which requires

$$\int_0^1 R(t)\, \delta(t - t_i)\, dt = 0 \qquad i=1,\ldots,K \qquad (\delta = \text{Dirac delta}) \qquad (5)$$

since this integral can be written simply as:

$$R(t_i) = \sum_{j=0}^{K} z_j \dot{\phi}_j(t_i) - F(x, u_K(t_i), z_{K+1}(t_i), t_i) = 0 \qquad i=1,\ldots K$$

with $z_0 = Z_o$

With the Lagrange condition, the polynomials evaluated at discrete points reduce to the coefficient at that point, and thus

$$R(t_i) = \sum_{j=0}^{K} z_j \dot{\phi}_j(t_i) - F(x, y, z, t_i) = 0 \qquad i=1,\ldots K \qquad (6)$$

with $z_0 = Z_0$

Similarly the polynomial approximations when substituted into DAOP and evaluated at some t_i become just the corresponding coefficient Using the ODE model (6), DAOP becomes:

$$\begin{aligned} &\underset{x,u_i,z_i}{\text{Min}} && \langle Hx,u_i,z_i) && \text{(NLP1)} \\ &\text{s.t} && g(x,u_i,z_i) \wedge 0 \\ & && c(x,u_i,z_i) = 0 \\ & && RU) = \sum_{j=0}^{K} z_j\, bit)_t - Fixji \wedge X.)_i = 0 \qquad i=k...,K \\ & && z_0 = Z_0 \\ & && X^L \leqslant x \pounds X^u \\ & && U^L :_* u_i \wedge U^u \\ & && Z^L <-_1 z_i S Z^u \end{aligned}$$

With NLP1 we can now solve, once the points $\pounds_f$ i=L.$_{Mf}$K have been chosen, very general differential-algebraic optimization problems. An example of a flowsheet optimization problem solved with NLP1 can be found in Cuthrell & Biegler (1985). The location of the points i. i=l,...,K are chosen to correspond to the shifted roots of an orthogonal Legendre polynomial of degree K (hence the term orthogonal collocation (see Villadsen & Stewart (1967)). Orthogonal collocation can be equivalently thought of as a Galerkin procedure with the resulting integrals approximated by an optimal K-point quadrature. It turns out that the quadrature points for this approach correspond to roots of an orthogonal polynomiaL

Essentially the discretization of the ODEs is done over a set of points which can be visualized as follows in Pig. 2-1 for K=3. Here the initial condition specifies the first state coefficient leaving MK residual equations in the K(M+1) unknowns z_i. tt_{it} i=l....K. The control profile coefficients are left as decision variables in the optimization so the algebraic system of approximating equations has dimension MKxMK.

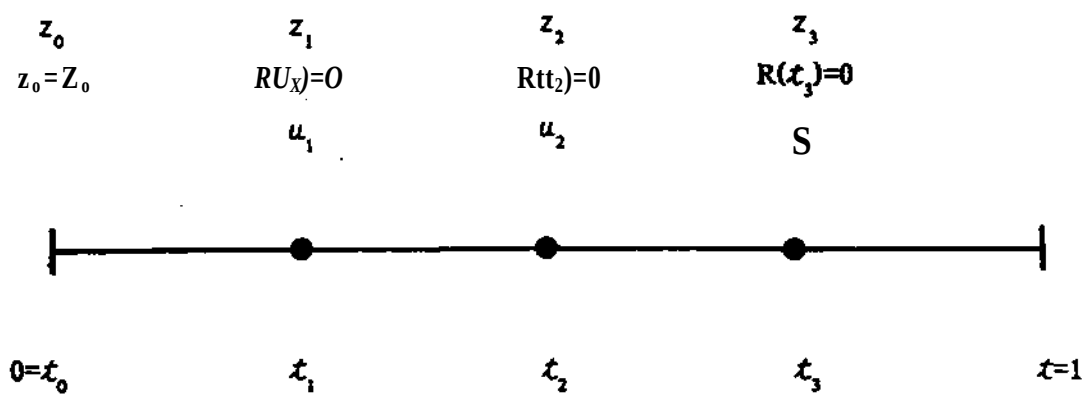

Figure 2-1: Global Collocation c

2.2. EXTENSION OF ORTHOGONAL COLLOCATION TO FINITE ELEMENTS

The global collocation method discussed in the preceding section constructs the state approximation using (K+l)th order polynomials. For functions which are poorly behaved (i.e. rapidly changing in some small region) an accurate approximation using global collocation would require K to be very large. The steep region could then be approximated well at the expense of grossly overapproximating the rest of the function.

An alternative to global collocation uses piecewise polynomial approximations. Here a set of (K+l)th order state polynomials z*.. .CO and Kth order control polynomials u¦Ct) are defined on finite elements. Each finite element Aa.¦ is bounded by two knots &.¦ and a ¦M with Aa.¦≡a.¦M~ a..¦ The distribution of elements now can be chosen so that the approximations are done both efficiently and accurately.

To begin a discussion of orthogonal collocation on finite elements consider Fig. 2-2 (where K=2, and NE=3; NE is the number of finite elements). To preserve the orthogonal properties obtained with global collocation the domain *£[0,l] is mapped into each finite element through the formula (with a =a, <* =b)
I NE+l

$$t = a_i \cdot \# a_M - a_i) \qquad i=i....NE \qquad \text{for } t\in[\alpha_i,\alpha_i\wedge]$$

And the locations of the orthogonal Legendre roots (with $<t_0=0$) are mapped to the points

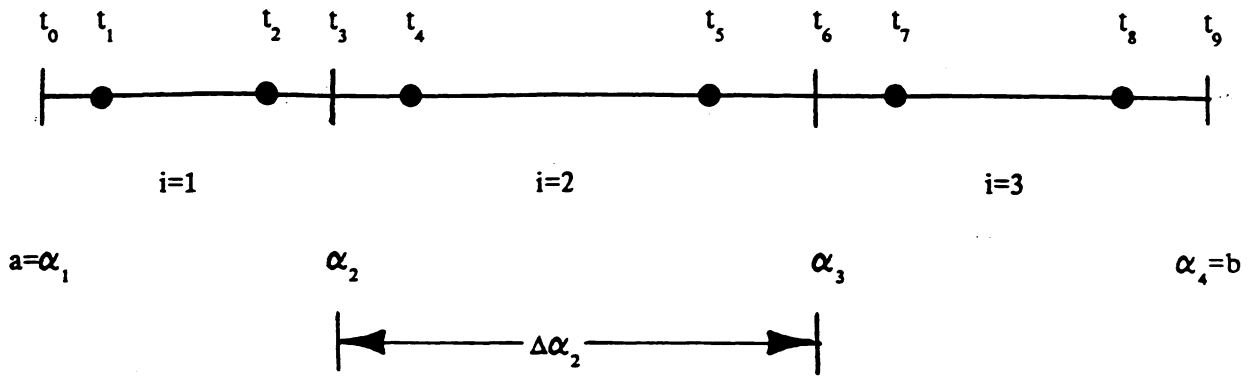

Figure 2-2: Finite Element Collocation

$$t_{(i-1)(K+1)+j} = \alpha_i + \xi_j(\alpha_{i+1} - \alpha_i) \qquad \begin{matrix} i=1,\ldots,NE \\ j=0,\ldots,K \end{matrix} \tag{7}$$

It is convenient at this point, in order to save a considerable amount of rewriting, to define the expression (i-1)(K+1)+j as the new variable [ij]. The expression [ij]≡(i-1)(K+1)+j is not to be confused with the commonly used double subscripts for matrices (e.g. A_{ij} meaning the element in the ith row and jth column). Furthermore, the indices "i" and "j" can themselves take on other characters when the context requires it. For example, [ik] becomes (i-1)(K+1)+k for some i and k, and [i1]=(i-1)(K+1)+1 for some i. For an equivalent derivation of finite element collocation which uses matrix notation see Finlayson (1980). With this convention (7) becomes:

$$t_{[ij]} = \alpha_i + \xi_j(\alpha_{i+1} - \alpha_i) \qquad \begin{matrix} i=1,\ldots,NE \\ j=0,\ldots,K \end{matrix}$$

The Lagrange polynomials can now be expressed as:

$$z^i_{K+1}(t) = \sum_{j=0}^{K} z_{[ij]}\, \phi_{[ij]}(t) \qquad \phi_{[ij]}(t) = \prod_{k=0,i}^{K} \frac{(t - t_{[ik]})}{(t_{[ij]} - t_{[ik]})} \tag{9}$$

$$u^i_K(t) = \sum_{j=1}^{K} u_{[ij]}\, \psi_{[ij]}(t) \qquad \psi_{[ij]}(t) = \prod_{k=1,i}^{K} \frac{(t - t_{[ik]})}{(t_{[ij]} - t_{[ik]})}$$

for i=l,...,NE

For an equivalent representation of eqn (9) without the previous notation change see the Appendix.

The discretized residuals can be immediately written down from eqns (6) as:

$$R(t_{[i\ell]}) = \sum_{j=o}^{K} z_{[ij]}\, \dot{\phi}_{[ij]}(t_{[i\ell]}) - F(x,u_{[i\ell]},z_{[i\ell]},t_{[i\ell]}) = 0 \qquad (10)$$

$$i=1,\ldots,NE \quad \ell=1,\ldots,K$$

with $z_{[I.o]} = z_o$

The calculation of the term $\dot{\phi}_{[ij]}(t_{[i\ell]})$ can be simplified by chain ruling derivatives to obtain:

$$\dot{\phi}_{[ij]}(t_{[i\ell]}) = \dot{\phi}_j(t_\ell) \,/\, \Delta\alpha_i \qquad \begin{matrix} i=1,\ldots,NE \\ j=0,\ldots,K \\ \ell=1,\ldots,K \end{matrix} \qquad (11)$$

And thus eqn (10) is more simply written

$$R(t_{[i\ell]}) = \sum_{j=o}^{K} z_{[ij]}\, \frac{\dot{\phi}_j(t_\ell)}{\Delta\alpha_i} - F(x,u_{[i\ell]},z_{[i\ell]},t_{[i\ell]}) = 0 \qquad i=1,\ldots NE \qquad (12)$$

$$\ell=1,\ldots,K$$

with $z_{[io]} = z_o$

In equation (12) the expression $\dot{\phi}_j(t_\ell)$ can be easily calculated offline (see Villadsen & Michelsen (1978)) since it depends only on the Legendre roots. Now assuming for the moment the variables x and $a_{[i]}$ are fixed. (12) is composed of M(NHKM) equations and M(NE(K+1)) state coefficients. To make the system well posed an additional set of M(NE-l) equations are written to make the polynomials $z_{(i)}$ continuous at the interior knots a_i, i=2,...,NE. This is done by enforcing

$$z^{i}_{K+1}(\alpha_i) = z^{i-1}_{K+1}(\alpha_i) \qquad i=2,\ldots,NE \qquad (13)$$

or equivalently

$$z_{[i0]} = \sum_{j=0}^{K} z_{[i-1j]}\, \phi_j(t=1) \qquad i=2,\ldots,NE \qquad (14)$$

These equations extrapolate the polynomial $z_{K+1}^{i-1}(t)$ to the endpoint of its element and provide an "initial condition" for the next element and polynomial $z_{K+1}^{i}(t)$. Each overall approximation to the state profile is therefore a continuous and piecewise polynomial function of order K+1. Stated mathematically, $z_{K+1}(t) \in P_{K+1} \cap \mathbb{C}[a,b]$ where P_{K+1} denotes the set of all polynomials of order K+1 and $\mathbb{C}[a,b]$ the set of continuous functions. A number of authors construct differentiable and piecewise polynomial approximations, from $z_{K+1}(t) \in P_{K+1} \cap \mathbb{C}^1[a,b]$, to higher order ODE or PDE systems (Finlayson (1980), Gardini (1985)). However, continuous approximations are sufficient for our case particularly since our examples have non-differentiable solution profiles.

Including the ODE model, discretized on finite elements, as well as the continuity at the knot conditions the NLP formulation becomes:

$$\begin{aligned}
&\underset{x,u_{[i\ell]},z_{[i\ell]}}{\text{Min}} && \Phi(x,u_{[i\ell]},z_{[i\ell]}) && && \text{(NLP2)}\\
&\text{s.t.} && g(x,u_{[i\ell]},z_{[i\ell]}) \le 0 && &&\\
& && c(x,u_{[i\ell]},z_{[i\ell]}) = 0 && &&\\
& && R(t_{i\ell}) = 0 && i=1,\ldots,NE && \\
& && && \ell=1,\ldots,K &&\\
& && z_{[10]} = Z_0 && &&\\
& && z_{[i0]} = \sum_{j=0}^{K} z_{[i-1j]}\, \phi_j(t=1) && i=2,\ldots,NE &&\\
& && x^L \le x \le x^U && &&\\
& && U^L \le u_{[i\ell]} \le U^U && &&\\
& && Z^L \le z_{[i\ell]} \le Z^U && &&
\end{aligned}$$

With a_i, i=2....NE fixed, NLP2 can be used to solve most types of differential-algebraic optimization problems. The importance of accuracy of the approximation however has not been discussed. This we take up in the next section where a set of conditions are developed that guarantee minimization of the approximation error at the solution of NLP2.

2.3. DEVELOPMENT OF AN ERROR MINIMIZATION STRATEGY

In this section we present a set of error minimization or knot placement equations. These equations involve the state profile approximations z^{i}.. (t), i=l,...,NE, and the interior knot locations $0C_i$.. i=2,...,NE. To make the presentation of these ideas simpler we shall treat the M-vector Z(t) as having only one element for the next three subsections. In section 2.3.4 this restriction will be relaxed to include multiple states (M>1). At the NLP solution these knot placement equations require the interior knots to be at locations for which the approximation error is minimized. The entire development of these equations is beyond the scope of this paper. However, we begin with a useful theorem from the approximation theory literature and develop from this, with some assumptions, the knot placement equations.

2,3.1. JACKSON'S THEOREM

$$E_K\wedge(Z(t)\ ;\ [\alpha_i.\alpha_i,,]> * \text{const}\ At_i^{***1}\ |z^{(K*l)}(t)|_i \qquad (15)$$

(for the complete derivation of this equation see Rivlin (1969))

where

- $E_{\_\_}$.(Z(t) ; [a.,a. ,]) represents the local error between a polynomial of (K+lHh order and the function Z(t) over some region $[a_i, a_{i+1}]$.
- const is a calculable (but physically of little use) constant dependent only upon K.
- Att_i^{**1} is the (K+l)th power of the ith finite element length.
- $|Z^{CK}\wedge\circ(t)|_i$ is the max-norm of the (K+l)th derivative of the function Z(t) in the interval $[a_i, a_{|M}]$.

2.3.2. AN ERROR MINIMIZATION PROBLEM

To minimize the approximation error we formulate the following NLP.

$$\underset{a_i}{\text{Min}} \quad \underset{j}{\text{Max}} \quad Aa_j \, |zf^{K*I)}(t)|^{aw>} \qquad i=2....NE \qquad j=l,....NE \qquad \text{(NLP3)}$$

$$\text{SL} \qquad Aa_j \; £ \; €$$

$$\text{with} \qquad a_1 - a_1 \; a_{NE+1} = b$$

$$€ = \text{a small positive constant}$$

NLP3 is now posed conceptually as a constraint in the optimization of NLP2. That is, we minimize $\cdot fott[_i£]^{**}[_i/])^{m}$ *HLP2* over x, $ci \wedge j. \; z_{f_i \wedge y}$ subject to the indicated constraints and subject to NLP3 being minimized. This "inner-outer" optimization problem can be greatly simplified, provided $Aft_j > €$ holds, by writing the necessary conditions for NLP3 as:

$$Att_i \quad \backslash t^{TM} \backslash l \wedge^{x)} \quad = \quad \text{constant} \qquad i=l....,NE \qquad \text{(16)}$$

It is easy to show (see Fiacco (1976)) that (16) also represents the sufficient opiimality conditions for NLP3. These equations can now be used as basis for a set of knot placement equations which can be included in NLP2 as equality constraints.

2.3.3. DEVELOPMENT OF KNOT PLACEMENT EQUATIONS

Numerous strategies exist in the approximation theory literature for choosing a "good" distribution of knots when using finite elements. An excellent review, with accompanying numerical comparison, is given in Russell & Christiansen (1978). The method presented here is a slightly modified version of one developed by deBoor (1978) and based on (16). It LS not a fully rigorous method, involves convergence of highly nonlinear equations, and can fail under circumstances which vail be outlined as we proceed. However, both references report the method to be quite effective on a number of examples and our results support this conclusion.

Recall that equation (15) represents a local approximation error. A much more detailed analysis of polynomial approximation using collocation at Legendre roots (deBoor (1974) and deBoor & Swartz (1973)) results in the addition of the global term $o(Aa_i^{**1})$. This term. however, is difficult to quantify and vanishes for small Att_r We therefore must assume it is negligible, and focus instead our efforts on the local term. The notion of equidistributing the local term, that is, distributing the local term equally over the entire range [a,b] forms the

basis for most of the knot placement methods we are aware of, even many not based on (16). Our task therefore is to develop a simple strategy which equidistributes the error given by (16).

Two difficulties immediately arise from equation (16). First, the infinity norm must be calculated, and we also need derivatives of the function Z(t), which itself is unavailable since it is the true solution of the differential equations.

Notice, firstly, that the solution to (16), in terms of a_i, is asymptotically equivalent (i.e. as $Aa_i \to 0$) to the solution of:

$$\int_{\alpha_i}^{\alpha_{i+1}} |Z^{(K+1)}(t)|^{1/(K+1)}\, dt = \int_{\alpha_{i+1}}^{\alpha_{i+2}} |Z^{(K+1)}(t)|^{1/(K+1)}\, dt \qquad i=1,\ldots,NE-1 \qquad (17)$$

For a proof of this equivalence see Russell & Christiansen (1978) and Pereyra & Sewell (1975). As a result, an integral approximation to the infinity norm is used. Calculation of (17) is made very simple by replacing the $|2?^{K<l>1)}(t)|$ term by the following piecewise constant approximation:

$$\cdot\cdot - \begin{cases} \dfrac{2|A9\ll X_{J/2})|}{\ll r\ \alpha_1} & \text{on } [\alpha_1,\alpha_2] \\ i=1 \\ \dfrac{[[***_{\bullet}\text{-},*>|}{\alpha_{i+1}-\alpha_{i-1}} + \dfrac{|^{Ae(a_{j+,/2})}|}{\alpha_{i+z}-a_1} & \text{on } [a_i.a_{i+1}] \\ i=2,\ldots,NE-1 \\ \dfrac{2|\Delta\Theta(\alpha_{i-1/2})|}{\alpha_{i+1}-\alpha_{i-1}} & \text{on } [\hat{i}\alpha_{NE},\alpha_{NE+1}] \\ i=NE \end{cases} \qquad (18)$$

where $6(a_{i+I/2}) = (\ z^{(K)}_{K+1}\ V \qquad i=1,\ldots,NE$

= the highest non-zero derivative of $z^i_{K+1}(t)$ in $\Delta\alpha_i$

The function $s_(\ )$ is obtained by deBoor with the important assumption that $z'_{K+l}(t) \cong Hi)$ over $[a_i \wedge_{|+I}]$. This aUows $Z^{<K*l)}(t)$ to be replaced in (17) by $z^{**\circ}_{K+1}(t)$. However, zj^/i) is only (K+l)th order (degree K) and thus possesses only K non-zero derivatives; its Kth denvauve is in fact a constant The function $zj\wedge^{*\circ}_i(t)$ is next constructed using divided differences resulting

in the function s_t. Substituting s_i into (17) gives a final set of working knot placement equations:

$$h_i = s_i^{K+1} Aa_i - s_{i+1}^{K+1} A_{j+1} = 0 \qquad i=1,\ldots NE-1 \qquad (19)$$

which can easily be included into NLP2 as equality constraints, replacing problem NLP3. Equation (19) represents necessary and sufficient optimality conditions.

2.3.4. KNOT PLACEMENT EQUATIONS WITH MULTIPLE STATE PROFILES

A modified form of equation (19) can be used if it is desirable to include influence of the errors from multiple profiles. This results in the knot locations being at positions for which some overall error is equidistributed.

Recall that the state profile vector $Z(t)$ has dimension M which shall now be treated explicitly. Approximations of these states on finite elements then yields the polynomials $(z_j, {}_{K+1}(t))$ for i=1....NE, m=1....M. Equation (18) can then be constructed for each state profile with the equations (19) becoming:

$$\sum_{m=1}^{M} \qquad \sum_{m=1}^{M}$$

for i=1....NE-1

Equation (20) incorporates the errors, in each element i, associated with all states m=1....M. into the knot placement equations through a 2-norm. (The 2-nonn of $sJ^{(K*\circ}$ is required over the theoretically more desirable max-norm because of differentiability considerations.)

2.4. AN NLP METHOD FOR OPTIMIZING DIFFERENTIAL-ALGEBRAIC SYSTEMS

This section simply states the NLP formulation which can accurately optimize differential-algebraic systems of equations. Included are the residual approximations, the continuity at the knots equations, as well as the knot placement equations.

$$\underset{x,u_{[i\ell]},z_{[i\ell]},\alpha_i}{\text{Min}} \quad \Phi(x,u_{[i\ell]},z_{[i\ell]}) \qquad \text{(NLP4)}$$

$$\text{s.t.} \quad g(x,u_{[i\ell]},z_{[i\ell]}) \le 0$$

$$c(x,u_{[i\ell]},z_{[i\ell]}) = 0$$

$$R(t_{i\ell}) = 0 \qquad i=1,\ldots,NE \quad \ell=1,\ldots,K$$

$$z_{[10]} = Z_0$$

$$z_{[i0]} = \sum_{j=0}^{K} z_{[i-1j]}\, \phi_j(t=1) \qquad i=2,\ldots,NE$$

$$h_i = 0 \qquad i=1,\ldots,NE-1$$

$$\alpha_{i+1} \ge \alpha_i + \epsilon \qquad i=1,\ldots,NE$$

$$x^L \le x \le x^U$$

$$U^L \le u_{[i\ell]} \le U^U$$

$$Z^L \le z_{[i\ell]} \le Z^U$$

The procedure used in NLP4 to position the knots differs from previous ODE solving methods that use orthogonal collocation on finite elements. deBoor (1978) selects the knots in an outer loop and then solves the collocation equations. Repositioning of the knots is then done to obtain a better approximation, with the procedure terminating when knot movement is very small. Ascher et. al. (1979), in the FORTRAN package COLSYS, use a similar procedure. Here an extrapolation technique is used to bound the approximation error which requires repeated solutions of the approximated problem. Based on this error the knots are either repositioned or new knots are introduced (which increases the size of the problem) by bisecting all intervals.

Our method differs from deBoor (1978) and Ascher et. al. (1979), in that we solve the knot placement and collocation equations simultaneously. However, we use the more desirable Lagrange basis polynomials as opposed to the B-splines used in each above reference. The simultaneous procedure we use has distinct advantages that will be pointed out in Section 4. The ability to change the size of the problem by halving intervals, however, is not a feature of NLP4.

At this point we first discuss a type of problem NLP4 cannot solve and a number of changes necessary for the method to handle more general types of problems.

3. EXTENSION OF THE METHOD TO PROBLEMS WITH DISCONTINUITIES

3.1. AN EXAMPLE - A FAILURE OF NLP4

A number of problems that occur frequently in the area of optimal control are those which have discontinuous and/or singular arc control profiles. Consider the following example problem:

$$\begin{array}{lll} \underset{U(t),t_f}{\text{Min}} & t_f & \text{(PI)} \\ \text{s.L} & \dot{Z},(t) = Z_2(t) & Z\text{^}O)\text{^}) \quad \text{^}(\text{^}=300 \\ & \dot{Z}_2(t) = U(t) & Z_2(0)=0 \quad Z_2(t_f)=0 \\ & -2 * U(t) \leq 1 & \end{array}$$

which represents the problem of finding the minimum time a car requires to cover 300 distance units, starting and stopping at rest, subject to acceleration limits of -2 and 1. This problem is linear in U(t) and has an analytic solution with a bang-bang control profile. The optimal value of t_f is 30 and the point of discontinuity (or switching time t_s) occurs at 20. The optimal state profiles are drawn as solid lines in Figs. 3-2 and 3-3 with the complete analytical solutions given below

$$\begin{array}{lll} \text{For } t\text{€}[0.20): & U(t) = 1 & \\ & Z,(t) = 1/2\ t^2 & \\ & Z_2(t) = t & \\ \text{For } t\text{€ }(26,30]: & U(t) = -2 & \\ & z\text{^}t) = -e + 60t - 600 & \\ & Z_2(t) = 60 - 2t & \end{array}$$

Note that the optimal control profile is a piecewise constant function and the state profiles are simply piecewise polynomials of degree 2 or less. Problem PI was solved using NUM. with both 2 and 3 finite elements, and with K=l Since Z^t) is a quadratic at the optimal solution. both approximations were constructed using quadratics. This results, given that the optimal $u_K(t)$ is found accurately, in the approximations for both Z^t) and $Z_2(t)$ being exact The solutions

obtained using NLP4 are presented in Table 3-1 and Figures 3-1 to 3-6. For the state profiles the analytical solutions are indicated with solid lines and the approximations with broken lines. For the control profile only the approximate solutions are presented as solid lines.

	NE=2	NE=3
t_f $(=\alpha_{NE+1})$	30.5154	30.2933
α_i i=2,...,NE	15.2577	12.9002,22.2971

Table 3-1: Results for P1 using NLP4

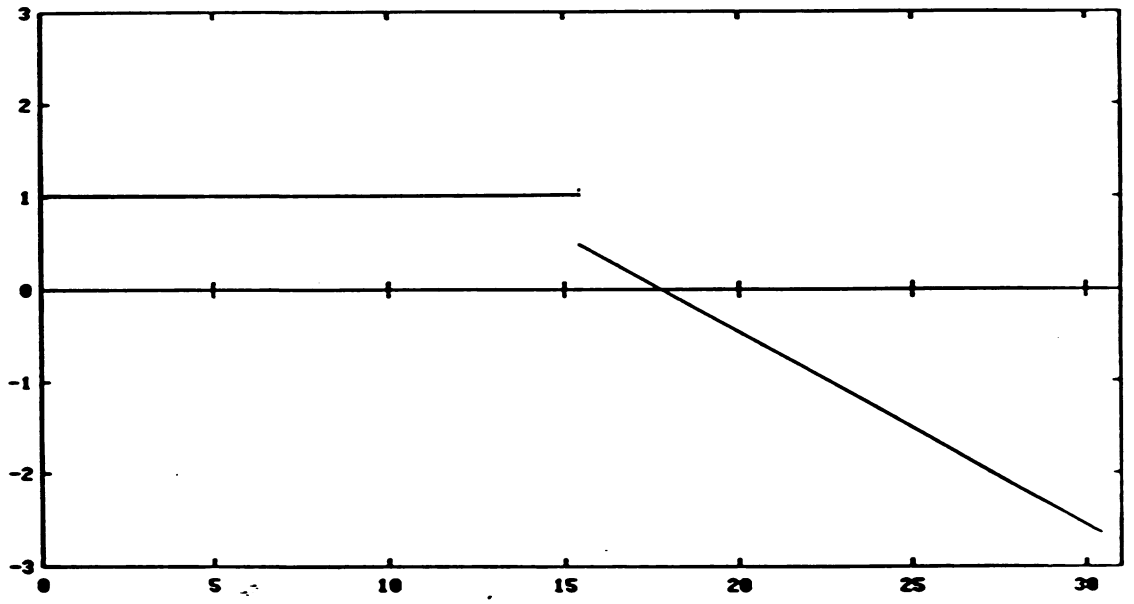

Figure 3-1: Control Profile $u_K(t)$ with NE=2

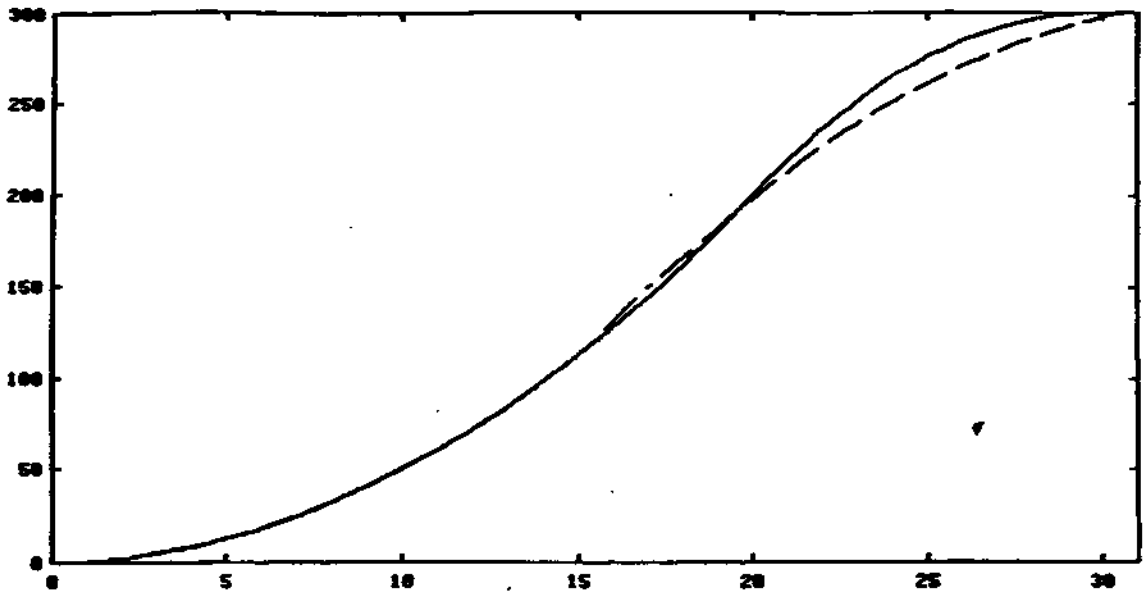

Figure 3-2: State Profile $Z_1(t)$ & $(z_{v}$,(t» , with NE=2

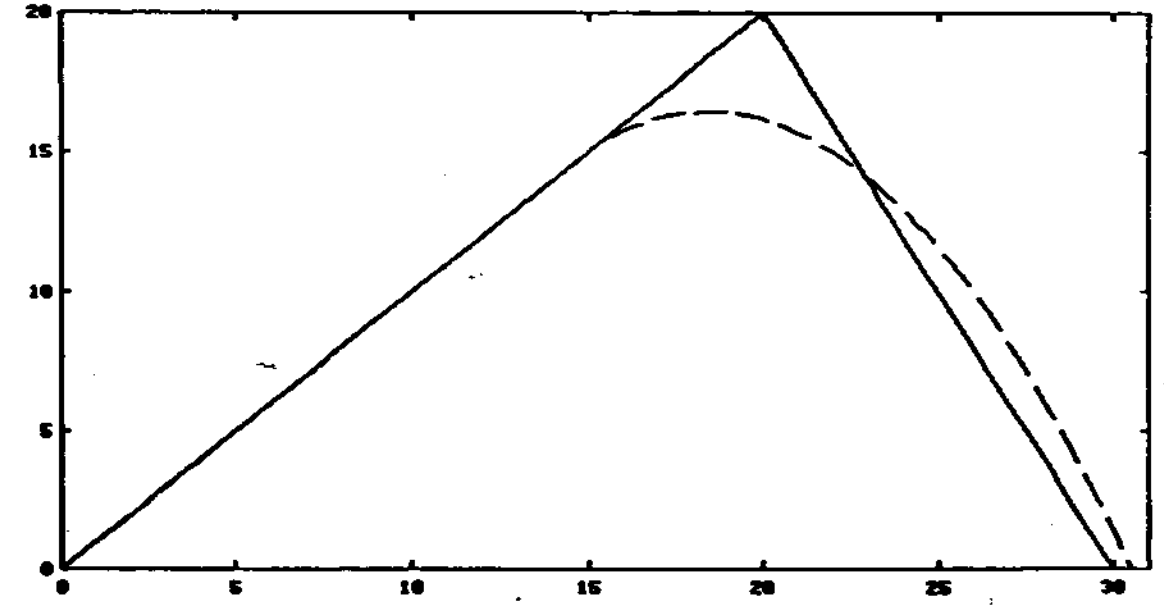

Figure 3-3: State Profile $Z_2(t)$ & $(z_{K+1}(t))_{m\cdot 2}$ with NE=3

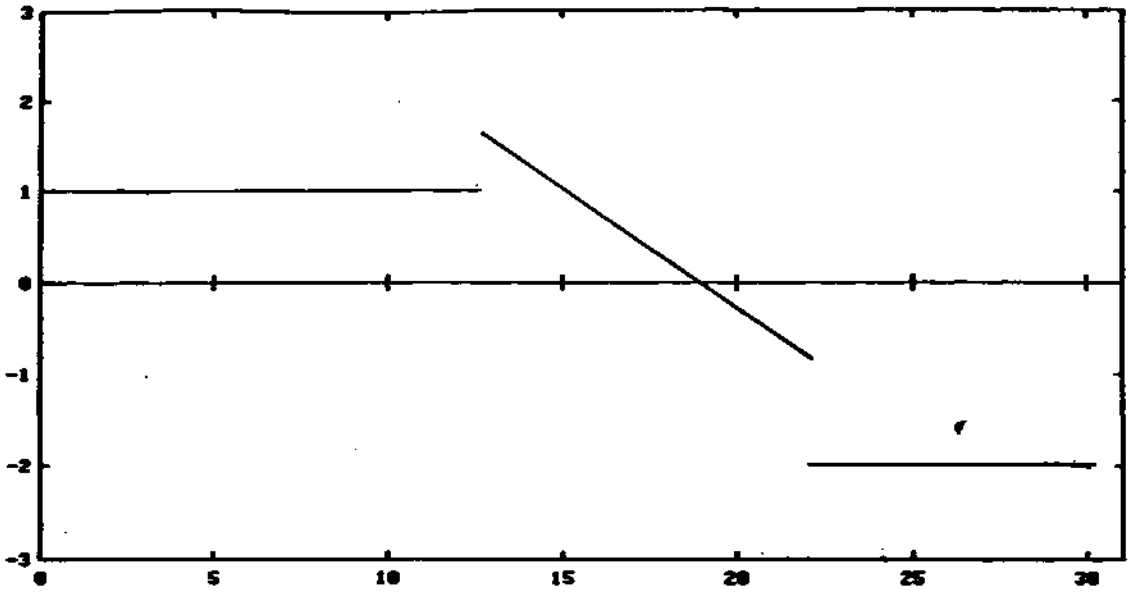

Figure 3-4: Control Profile $u_i(t)$ **with NE=3**

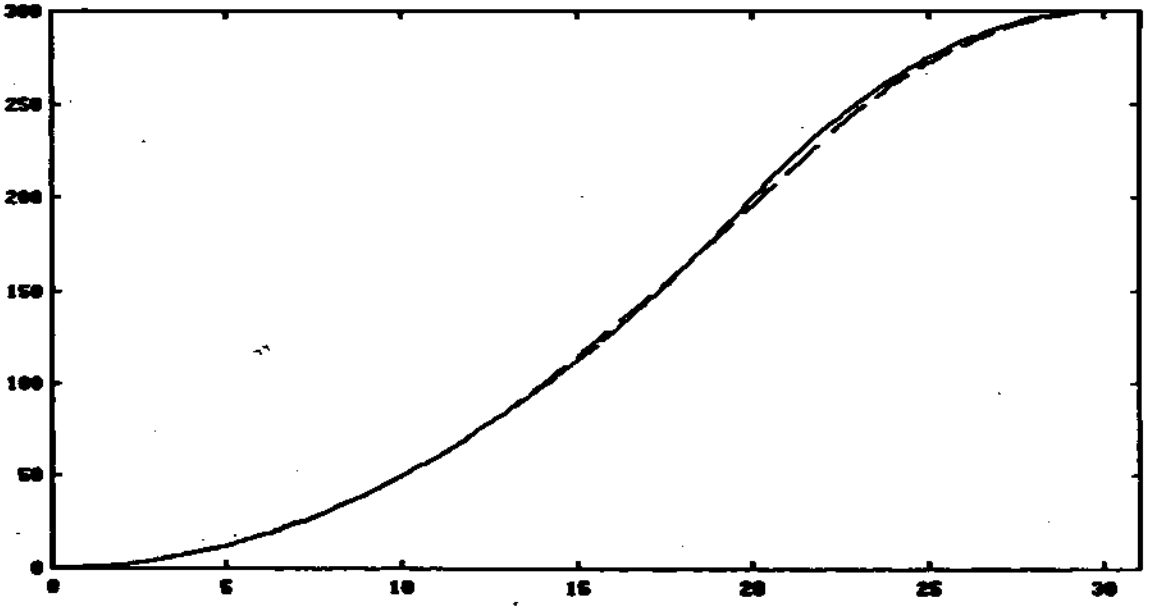

Figure 3-5: State Profile $Z_i(t)$ & $(z_i(t))$, with NE=3

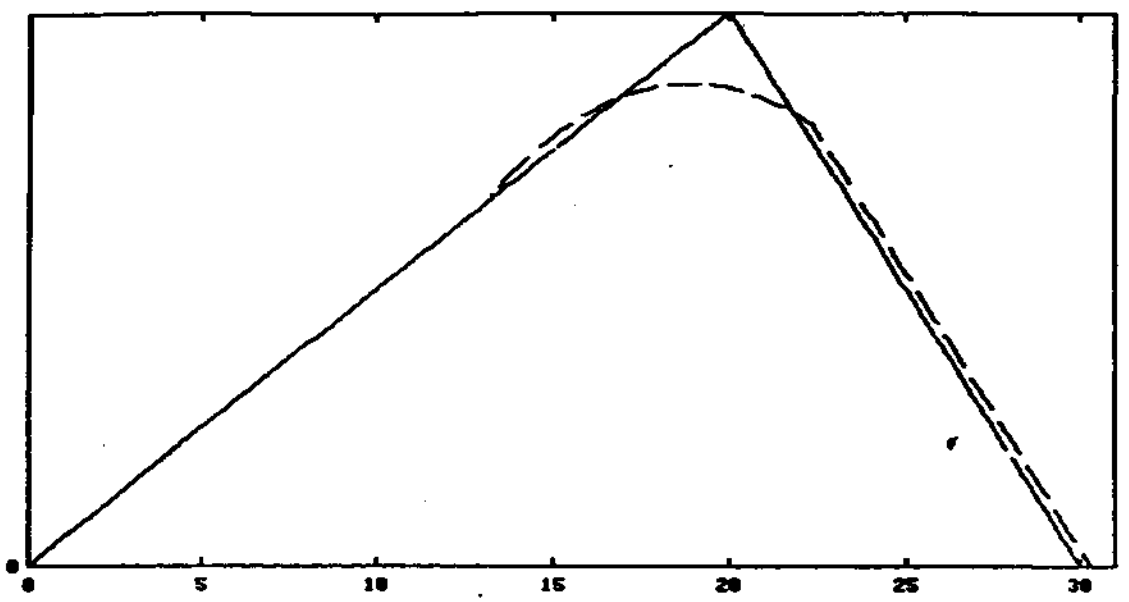

Figure 3-6: State Profile $Z_2(t)$ & $(z_{KM}(t))_{m\cdot 2}$ with NE=3

Clearly NLP4 fails to solve PI with neither the correct t_r t_f nor the profiles being found accurately. The most obvious reason is that the bang-bang control profile is approximated poorly in both cases (although it appears to have improved for NE=3). It is instructive at this point to investigate some factors which did not contribute to this failure. First, the order of the polynomial approximation is sufficient (quadratics were used). Second, the number of finite elements used is also high enough (compared to the analytical solution). Here the two finite element case, at. first glance, should have been acceptable since there is only one poini of discontinuity. Third, jhe degree of polynomial continuity enforced is adequate. Recall thai states are required only to be continuous (and here both analytical state profiles are non-differentiable at Q_f, and that no control profile continuity at the knots is enforced.

It is clear from these observations that the solution to the knot placement equations has precluded the locations of the knots from being at positions of control profile discontinuity. Therefore an additional set of "elements" (or "knots") is needed, which is independent of the knot placement equations. This motivates our discussion of super-elements which is presented in the next section.

3.2. SUPER-ELEMENTS - PROBLEM PI SOLVED

We now define a new level of elements, termed super-elements, defined by $A^{\wedge}_{ije}$, ise=l,...,NSE with the help of Figure 3-7 (for K=2, NE=3, NSE=2; NSE is the number of super-elements).

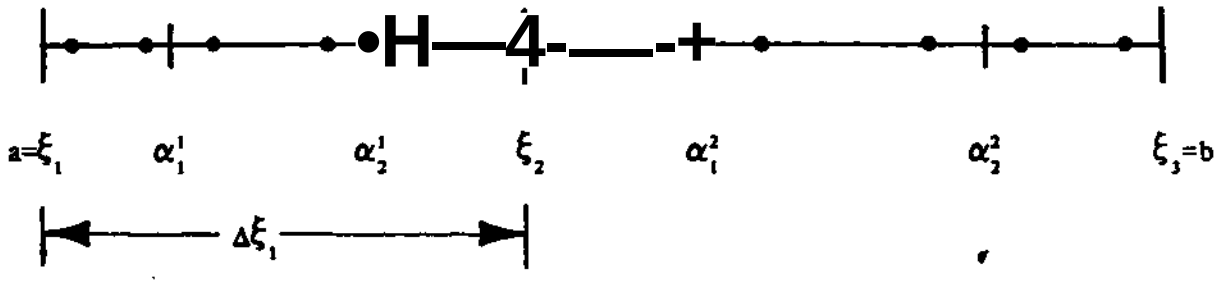

Figure 3-7: Super-Element Structure

A super-element, $A^{\wedge}_{i\$e}$, is composed of a set of NE finite elements and is bounded by two breakpoints $\pounds_{Ue}$, $\pounds_{ise+1}$ with $A\pounds_{ise} \equiv \pounds_{ise*l} - \pounds_{ise}$. For the remainder of this paper the term breakpoint will refer to a super-element boundary and is to be distinguished from the term knot (which refers to finite element boundaries). In this strategy breakpoints are left completely free, to be chosen to correspond to the optimal locations of control profile discontinuity. Within each super-element the knot placement equations (20) are applied to obtain accurate state approximations. Also, with the addition of this extra level of elements. state profile continuity must be enforced at all interior breakpoints. The control profiles however can be discontinuous at the breakpoints.

This modification makes it possible to solve problems which have both control profile discontinuities and state profiles which are hard to approximate. The ability to do ih 15. however, comes at the expense of having to solve a large optimization problem. The number of equations and variables representing the approximation grows large as NE is increased and almost doubles as NSE is increased from 1 to 1 Although it is difficult to imagine a problem requiring NSE>3 one still deab with potentially very large OlOO variables and equations) NLPs.

To demonstrate the need to use super-elements, consider again problem PL Recall that quadratic (K=2) approximations of the states will be exact if the control profile is found accurately. The use of finite elements for this problem is therefore unnecessary; their only function is to ensure accuracy of the state profiles. As a result PI was resolved using K=2.

NSE=2 & 3, and no finite elements. The results are presented in Table 3-2 (with £$_1$=0 and noting that $Z_{v\%t}+i^{sl}?^m$

NSE	starting point	optimum	state errors	
	f£, ise=2,....NSE+l	^ ise=2....NSE+1	$\|Z_1^{err}(t)\|$	$\|Z_2^{err}(t)\|$
2	5.10	20,30	<1(T^{24}	<10''^{26}
3	30,40,50	12.66,20,30	<1<T^{17}	<1(T^{18}

Table 3-2: Results for PI using Super-Elements

The results point out clearly that, given enough flexibility, the method can solve problems which have discontinuous profiles. Here the correct t_f and $t_($ were obtained, and $u_K(t)$ was approximated exactly. And, as a result, accurate state approximations were obtained. Note that 2 super-elements were sufficient to solve this problem exactly. A few runs were tried for NSE=3 from various starting points, £^, although only one solution is presented here. Interestingly, but not too surprisingly, the location of the "extra" breakpoint is not unique. It has no effect on a problem that has only one discontinuity. Therefore it appears that to solve problems which may contain discontinuities, NSE needs to be sufficiently large so that all points of discontinuity can be found. This parallels the well known fact that the goodness of an approximation obtained with finite elements increases up to a point where little more accuracy is achievable.

4, OPTIMIZATION OF A REACTOR PROBLEM WITH ADAPTIVE KNOT PLACEMENT

In this section we present the optimization of a reactor problem using NLP4. This problem contains no discontinuities and hence requires no super-elements. A number of finite elements, however, are necessary to approximate a temperature profile which contains a hot spot Also shown, through this example, will be that imposition of bounds on continuous profiles can be done easily by using bounds on the coefficients.

4.1. STATEMENT OF PROBLEM P2

The reactor optimization problem is illustrated in Figure 4-1. Here a 3:1 ratio of reactants B/A is first preheated by reactor effluent. This stream is then fed to a packed bed reactor where the reaction A+3B→C+3D occurs. The reactor is jacketed to allow the heat of reaction to be used to raise steam for the rest of the process.

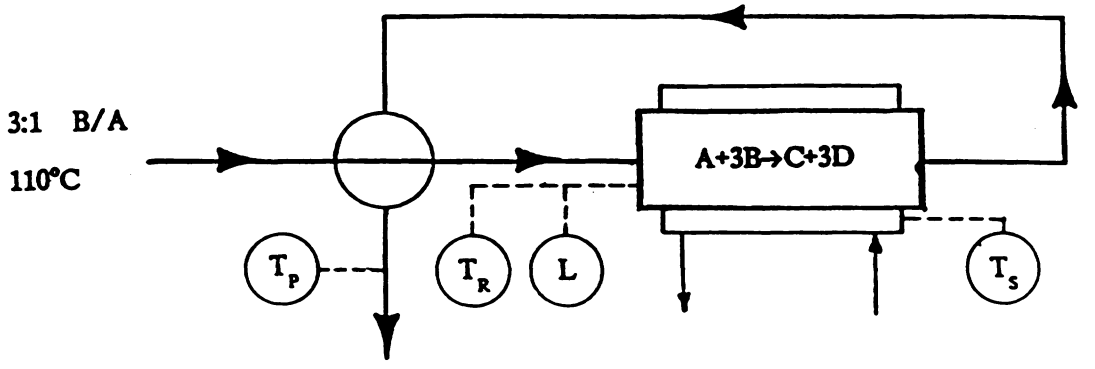

Figure 4-1: Flowsheet for Problem P2

The differential-algebraic optimization problem for this reactor design is stated below in P2.

$$\underset{T_P,T_R,L,T_S,q(t),\bar{T}(t)}{\text{Min}} \quad \Phi = L - \int_0^L \left(\bar{T}(t) - T_S/T_R \right) dt \tag{P2}$$

s.t.

$$\frac{dq(t)}{dt} = 0.3(1-q(t))\exp[20-20/\bar{T}(t)] \qquad q(0)=0$$

$$\frac{d\bar{T}(t)}{dt} = -1.5(\bar{T}(t)-T_S/T_R) + 2/3\,\frac{dq(t)}{dt} \qquad \bar{T}(0)=1$$

$$t \in [0,L]$$

$$C_{preheater} = \Delta H_{feed}(T_R, 110°C) - \Delta H_{product}(T_P, T(L)) = 0$$

$$T_P \geq 120°C, \quad T(L) \geq T_R + 10°C$$

with $\bar{T}(t) = T(t)/T_R$

where

T_P = specified product temperature

T_R = reactor inlet and reference temperature

L = reactor length

T_S = steam sink temperature

$q(t)$ = reactor conversion profile for species A

$\bar{T}(t)$ = normalized reactor temperature profile

This optimization involves maximizing the production of steam (utility profit) from a reactor while minimizing its length (capital cost). For the examples presented in this paper an equal normalized unit price was used to weight each term in the objective function. Similar results, however, were obtained with other weightings. The optimization of problem P2 is done with respect to four decision variables, T_p. T_R. L and $T_\$$ as well as the two continuous profiles q(t) and $\bar{T}(t)$. The reactor model consists of two ODE state equations, one for conversion of A and one for normalized temperature. More details about the reactor model can be found in Finlayson (1972) and Hlavacek (1970). Under the conditions given in Tables 4-1 & 4-2 for cases la and Ib the reactor model exhibits a hot spot while for case II no hot spot exists. Normally, when minimizing capital cost a hot spot is undesirable since beyond the hot spot the reaction rate is essentially zero. (The existence of a hot spot is really a manifestation of an overly long reactor.) However, since steam is raised, better designs can exist at longer reactor lengths. An additional process constraint appears in the form of a feed preheater heat balance. This constraint merely equates the enthalpy changes between the reactor feed and reactor effluent over the indicated temperature ranges. Two other temperature constraints are written to prevent temperature crossover in the preheater.

These two cases (with and without a hot spot) will be discussed in more detail in the next two sections. Initialization of the continuous profiles, however, is an important consideration for these problems since it can affect significantly the performance of the optimizer and the knot placement strategy. We treat this topic in the following section.

4.2. INITIALIZATION PROCEDURE FOR REACTOR OPTIMIZATION

In order to illustrate the knot placement strategy and NLP4, we consider the following cases as optimization problems:

Case la - reactor optimization problem P2 with a hot spot appearing in the temperature profile.

Case Ib - same as case la except with the additional state profile constraint $\bar{T}(t) \leq 1.45$ imposed.

Case II - reactor optimization problem P2 without a hot spot. Although this case used the same reactor model, the hot spot was avoided by using a different starting point and imposing different constraints (e.g. a shorter length).

The decision variable and profile bounds used for these cases can be found in Table 4-1.

	Case Ia		Case Ib		Case II	
	lower bound	upper bound	lower bound	upper bound	lower bound	upper bound
T_R(°C)	400	500	400	500	400	500
L	0.5	1.25	0.5	1.25	0.5	1.0
T_S(°C)	400	500	400	500	400	500
T_P(°C)	100	1000	100	1000	100	1000
q(t)	0	1.5	0	1.5	0	1.0
$\bar{T}$(t)	0	3.0	0	1.45	1.0	3.0

Table 4-1: Initial Bounds for Problem P2

Recall from section 2.3.3 that in order to obtain good knot placement the approximations $z^i_{K+1}(t)$ $\forall$ i must be reasonably close to Z(t). This requires that good guesses be used for the initial approximations to q(t) and $\bar{T}$(t). For cases I and II the starting profiles were different and therefore we discuss them separately.

In both case Ia and Ib steep profiles exist and care must be taken in initializing the optimization problem. For the initial decision variable sets given in Table 4-2, the model was first integrated with the ODE solver LSODE (see Hindmarsh (1979)). Next an initial set of knots α_i i=2,...,NE, with α_1=0 and α_{NE+1}=L were fixed and the collocation points calculated using (7). State coefficients at the collocation points were then obtained from the integrated profiles with the coefficients at the knots obtained by solving the linear system of continuity equations (14). Note that since the linear system of continuity equations is solved initially any subsequent Newton or SQP procedure always remains in the feasible subspace of these constraints (assuming no infeasible QP subproblems exist). Finally the residual, knot continuity, and knot placement equations were completely converged prior to starting the optimization.

This was done with Newton's method starting from an evenly spaced set of knots and resulted in the knot distribution listed in Table 4-4. The resulting continuous approximations are superimposed upon the integrated profiles in Figs. 4-2 & 4-3.

	Case la & Ib	Case II
T_R	462.23	434.80
L	1.0	0.67
T_S	425.25	400
T_P	250	400
Φ	-120.702	-45.544

r

Table 4-2: Initial Decision Variables and Objective Function for P2

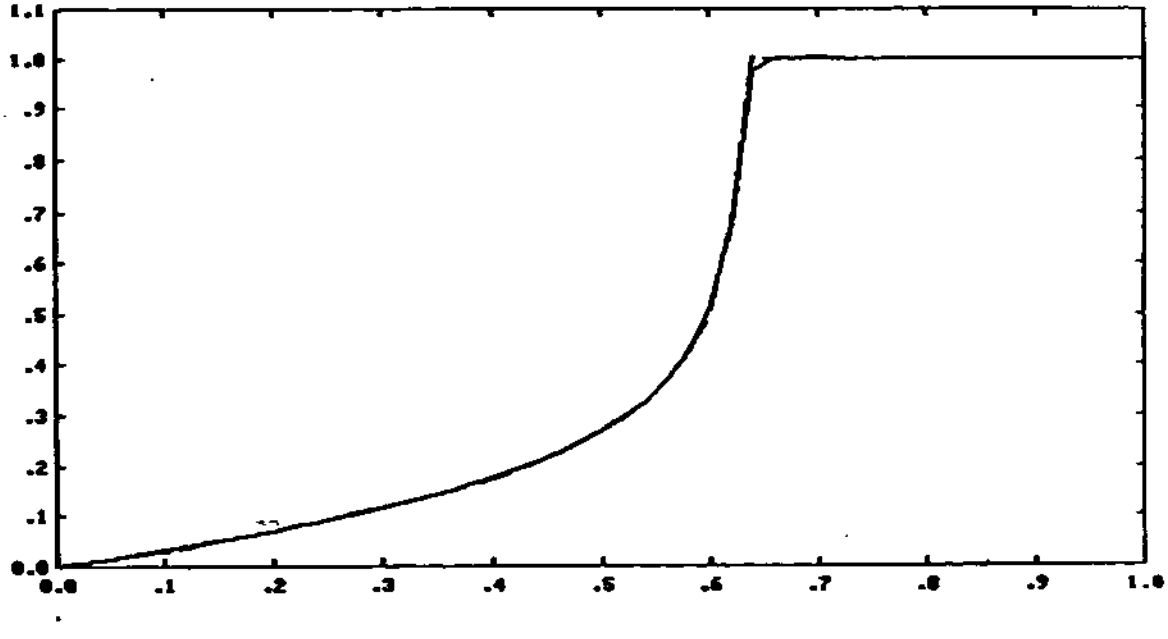

Figure 4-2: Starting Conversion Profile for Problem P2 Case la and Ib

For these approximations the maximum deviation from the integrated profiles is 0.02707 for conversion and 0.01745 for normalized temperature. This additional effort was required since in case la and Ib existence of the hot spot makes doing the approximation more difficult

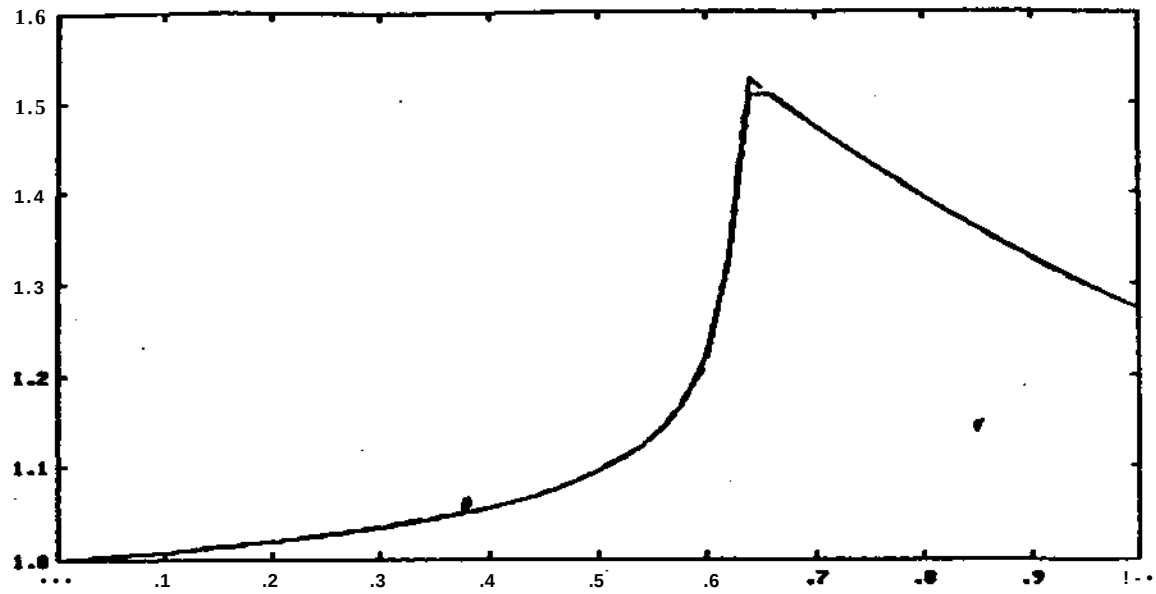

Figure 4-3: Starting Temperature Profile for Problem P2 Case la and Ib

For case II the profiles do not contain non-differentiabilities and are thus easier to approximate. Consequently the initialization for this problem did not require the additional initial convergence of the knot placement, continuity, and collocation equations with Newton's method. Here it was sufficient to intialize the coefficients at the collocation points from an integrated profile and solve the continuity equations to obtain the coefficients at the knots.

4.3. RESULTS AND DISCUSSION OF PROBLEM P2

In this section we present the results obtained from solving P2 using formulation NLP4 and the SQP algorithm described in Biegler & Cuthrell (1985). Table 4-3 presents the optimal values of the decision variables, and some diagnostic information. The error norms represent the max difference over t6[0.L] between a profile constructed using Lagrange polynomials (cqn (9)) and one obtained *a po&tejUoA*:* through numerical integration. Also, # of iterations refers to the number of QP subproblems solved using SQP, and K-T error represents the final Kuhn-Tucker error. In Table 4-4 a comparison is made between the initial and final knot distributions.

Case la and Ib represent formulations for which the solution has a hot spot From the solutions the length and inlet reactor temperature clearly indicate that the utility term in the objective function dominates. Since both variables are at their upper bounds the reactor is producing as much steam as possible. This condition is preferred over one with a shorter (and less expensive) reactor. Note also from Figs. 4-2 to 4-7 how the hot spot shifts in size and location from the initial profile as a result of the optimization. Again, in these figures the approximated and integrated profiles are superimposed. Good approximations are obtained for the continuous profiles in both cases la and Ib with the max norms for conversion and temperature errors being around 0.01 or less. Table 4-4 clearly demonstrates the effectiveness of the knot placement equations. For cases la and Ib the final knots are grouped in the region of the hot spot where both state profiles are very steep. *

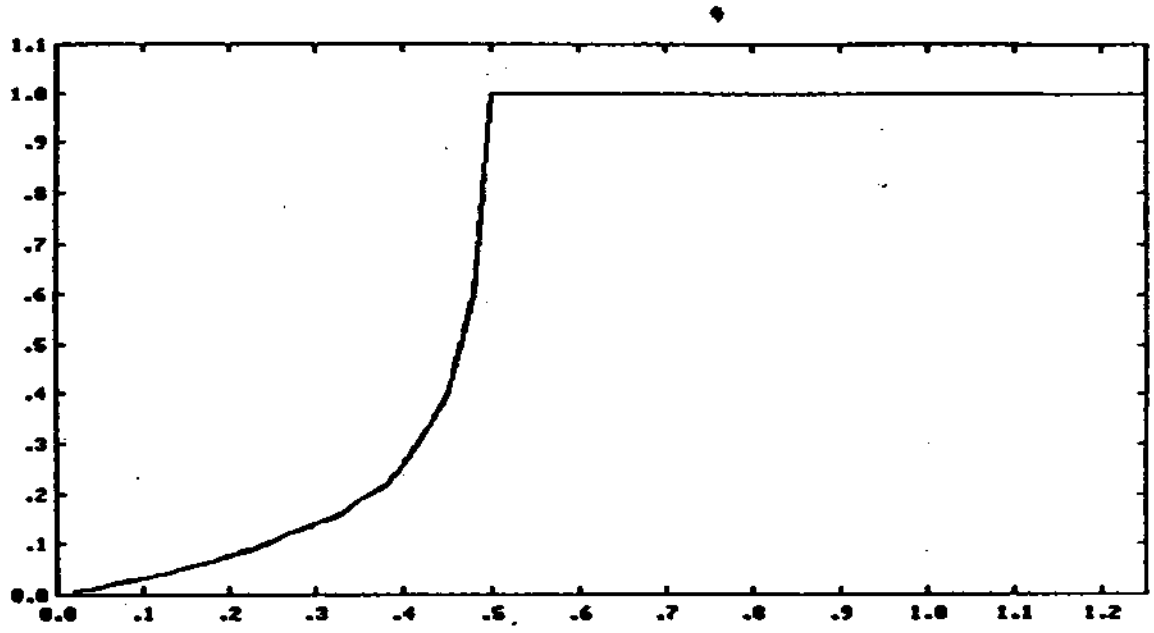

Figure 4-4: Final Conversion Profiles for Problem P2 Case la

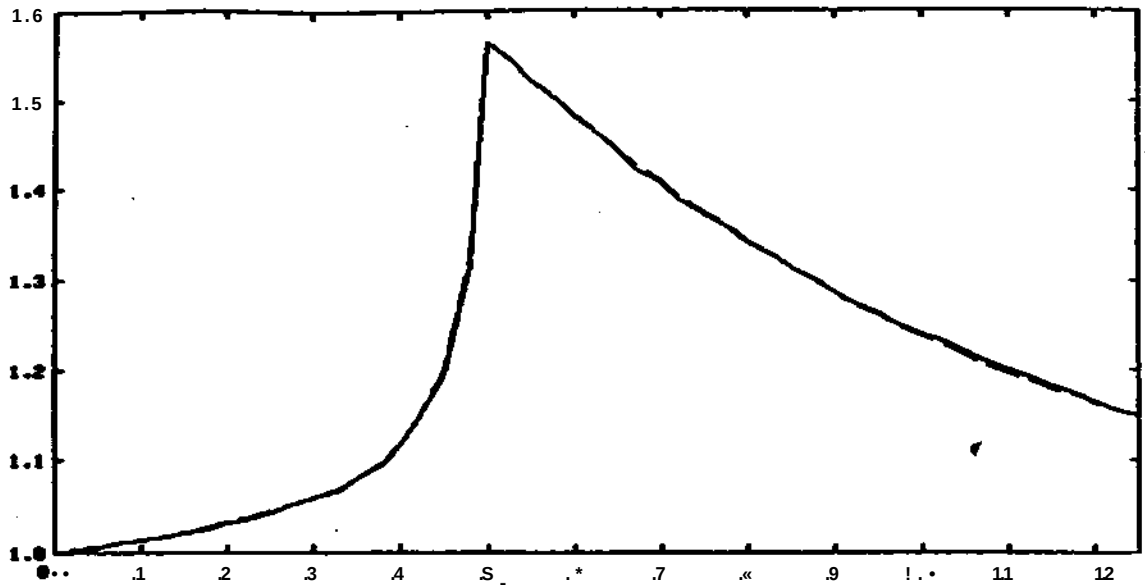

Figure 4-5: Final Temperature Profiles for Problem P2 Case la

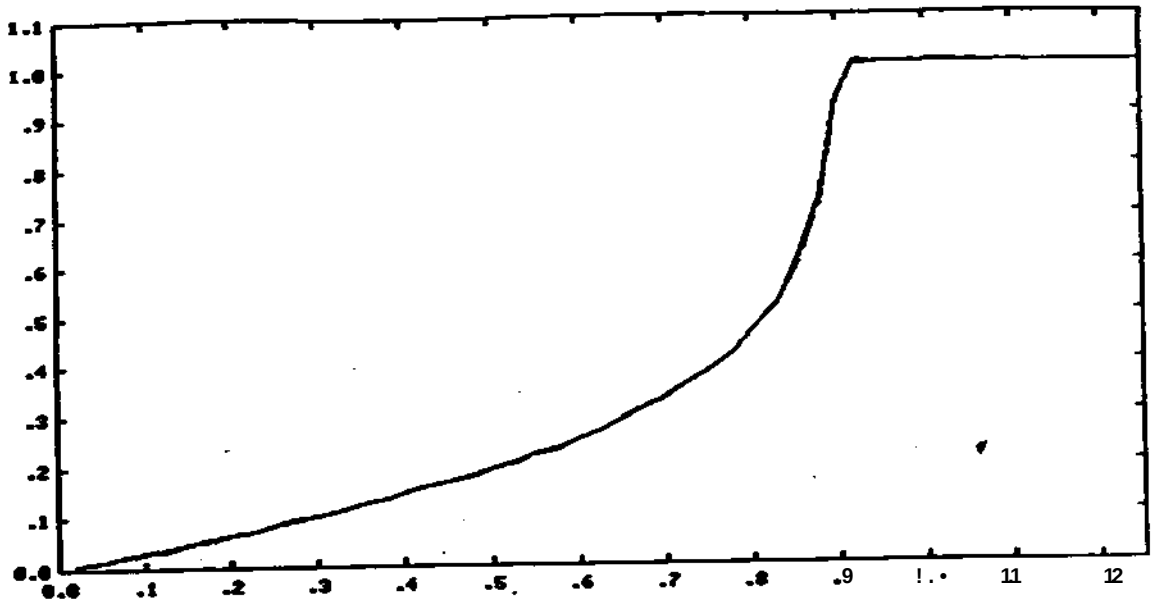

Figure 4-6: Final Conversion Profiles for Problem P2 Case Ib

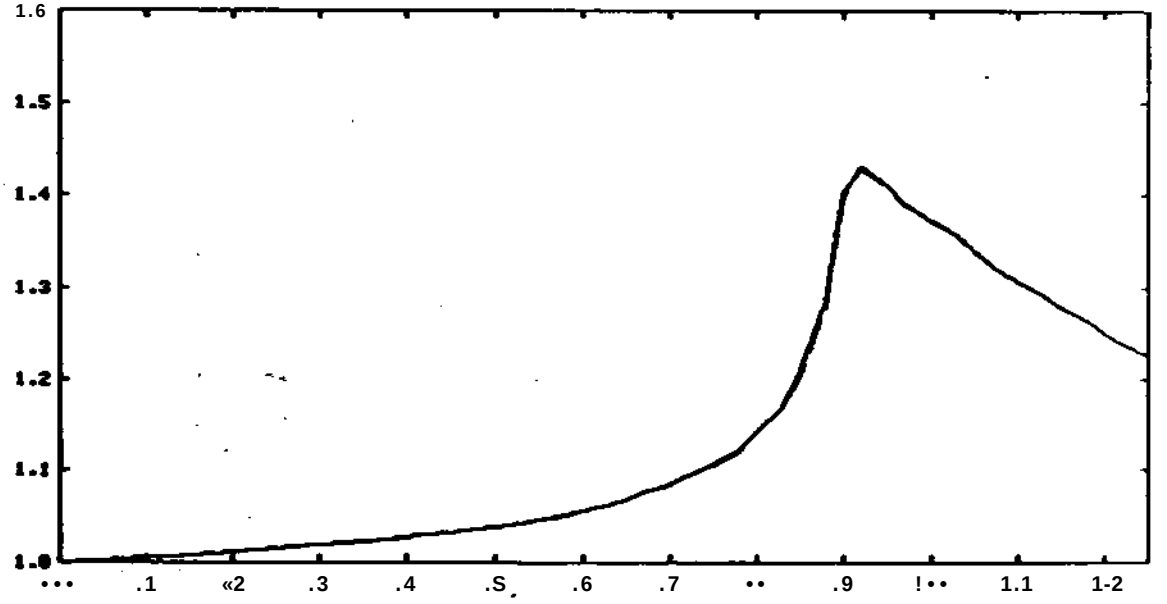

Figure 4-7: Final Temperature Profiles for Problem P2 Case Ib

	Case Ia	Case Ib	Case II
T_R	500	500	500
L	1.25	1.25	1.0
T_s	473.8	449.9	448.4
T_p	192.3	237.1	344.9
$\|q^{err}(t)\|$	0.01103	0.01129	0.02116
$\|\bar{T}^{err}(t)\|$	0.007113	0.006999	0.01437
K-T error	10^{-14}	10^{-14}	10^{-14}
Φ	-171.438	-145.644	-82.7024
# SQP iterations	28	18	24
CPU time (sec)	262	162	228

Table 4-3: Results for P2 Solved with NLP6

	Case Ia	Case Ib	Case II
α_i initial	0,0.365,0.534,0.591	same as Ia	0,0.201,0.268,0.335
i=1,...,NE+1	0.636,0.692,1.0		0.402,0.469,0.67
α_i final	0,0.287,0.416,0.456	0,0.486,0.728,0.828	0,0.369,0.624,0.771
i=1,...,NE+1	0.489,0.528,1.25	0.908,1.00,1.25	0.852,0.924,1.0

Table 4-4: Initial and Final Knot Distributions for P2

In case Ib imposition of the state profile bound $\bar{T}(t) \leq 1.45$ lowered the peak of the temperature profile and moved it to a later position in the reactor. This is partially due to a lower sink temperature (449.9 for case Ib vs. 473.8 for case Ia) which reduced the rate of heat removal from the reactor. More importantly, however, the optimal knot locations are near the region $t \in [0.7,1.0]$ of steep profiles and thus good approximations again result.

In case II the hot spot does not exist (see Figs. 4-8 & 4-9). Nevertheless good

approximations and good knot placement were again obtained Here the knots are grouped in the region t€[0.6,1.0] where both profiles are steep. Since a large part of the reactor for case I represents unnecessary capital expense it was instructive to investigate a design without a hot spot The solution for this case, however, again has both inlet temperature and length at their upper bounds and indicates that the desire to raise steam outweighs any capital cost influence. Comparison of objective functions, <f>=-171.438 vs. 4>=-82.7024, therefore shows case la is preferred over case II.

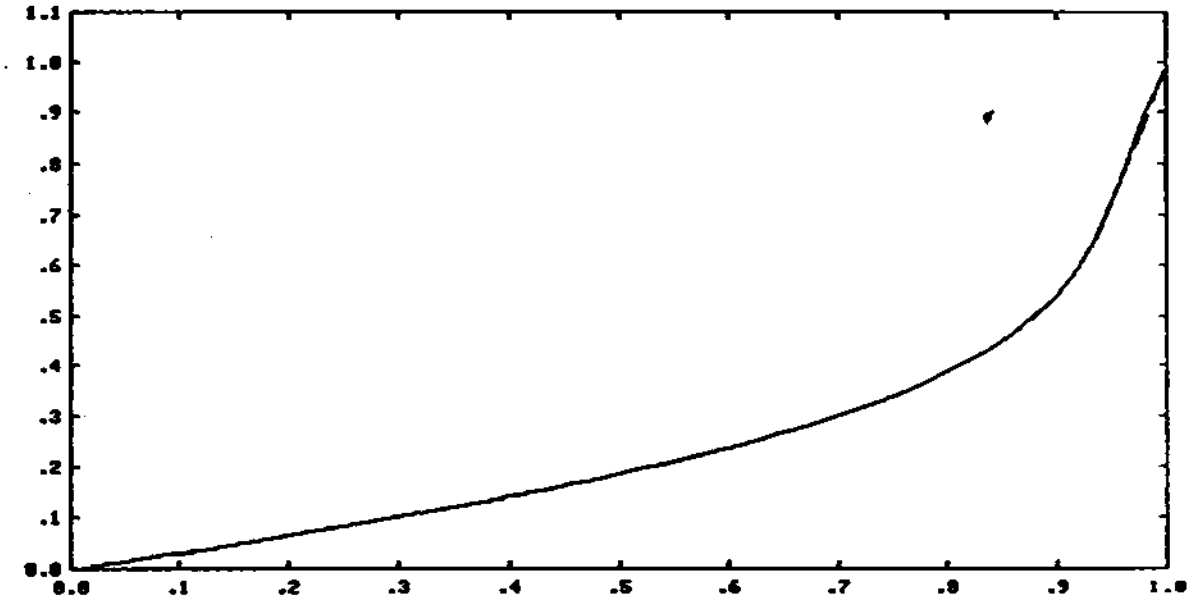

Figure 4-8: Final Conversion Profiles for Problem P2 Case II

5. SUMMARY AND CONCLUSIONS

This paper presents a general and accurate method for optimizing differential-algebraic systems of equations. The differential equations are discretized by using Lagrangc form polynomials and orthogonal collocation. The resulting set of algebraic approximations arc then written as constraints in a nonlinear program. Accuracy of these approximations is guaranteed by additionally solving within the NLP the sufficient conditions for error minimization. These conditions are written as a set of knot placement constraints which allow the knot locations to be chosen adaptively. An extra level of elements, super-elements, has also been developed in order to solve problems containing control profile discontinuities. Location of the super-element breakpoints is left as a decision in the optimization problem.

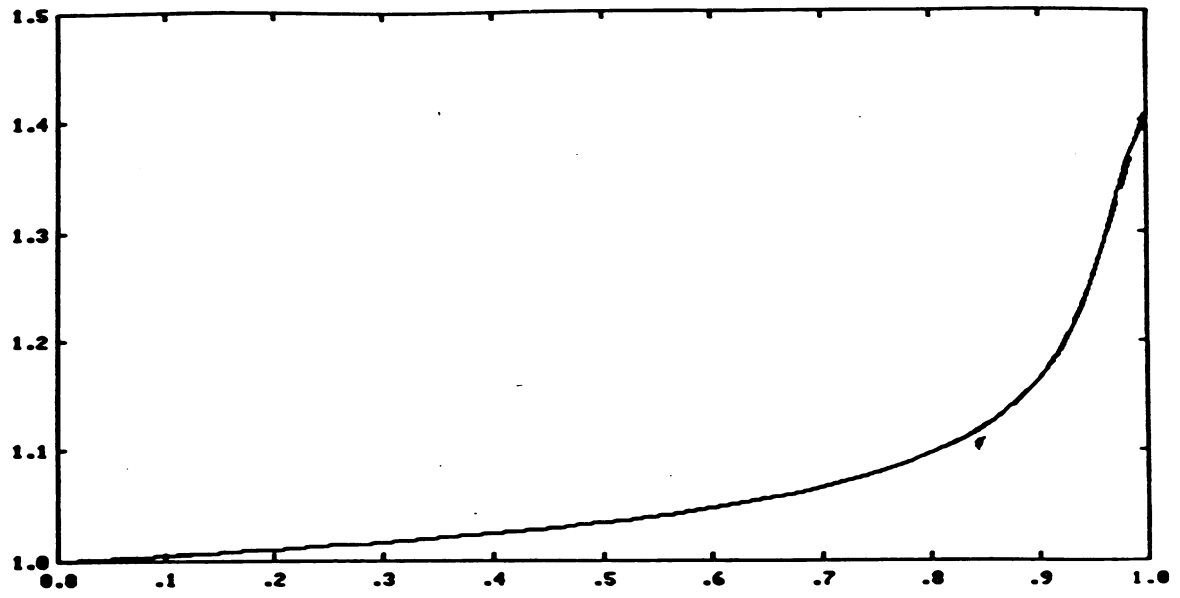

Figure 4-9: Final Temperature Profiles for Problem P2 Case II

Two examples are presented to illustrate the usefulness of the method. First, a small optimal control problem was solved using super-elements and quadratic state profile approximations. Both the discontinuous (bang-bang) control profile and the state profiles were approximated extremely well. For this problem the need to use finite elements was avoided since the analytical solutions were known to be linear and quadratic polynomials. Second, a reactor optimization problem was presented to illustrate that accurate approximations can be obtained within the context of optimization by using finite element collocation and an error minimization strategy. This problem also illustrates that bounding of continuous profiles can be done with no additional difficulty.

In future work more efficient methods need to be developed to solve the large NLPs which result. The advantage of the approach outlined in this paper over one using repeated numerical integration was demonstrated in Cuthrell & Biegler (1985). However, for problems solved with NLP4 which require both a large number of finite elements and super-elements the task of solving the NLP is not trivial. Investigation of decomposition techniques applicable to SQP methods remains one of the future directions of this research.

REFERENCES

Ascher. U., J. Christiansen, and R.D. Russell "A Collocation Solver for Mixed Order Systems of Boundary Value Problems." *Math. Comp.*, 33(146). pp. 659-679. (1979).

Bapat, S.. and J.C Heydweiller. "Adaptive Polynomial Approximations for Process Design," paper 18b.. AIChE National Meeting. Chicago. (Nov. 1985).

Biegler, L.T.. and J.E. Cuthrell. "Improved Infeasible Path Optimization for Sequential Modular Simulators-II: The Optimization Algorithm." *Comp. and ChE.*, 9(3). pp. 257-267. (1985).

Bryson. A.E.. and Y.-C Ho. Applied Optimal Control. Hemisphere Publishing. Washington D.C. (1975).

Cuthrell. J.E.. and L.T. Biegler. "Simultaneous Solution and Optimization of Process Flowsheets with Differential Equation Models." /. *Chem.E Symposium Series #92,* p. 13, (1985).

deBoor, C. and R Swartz. "Collocation at Gaussian Points." *SIAM J. Numer. Ana/..* 10(4), pp. 582-606, (1973)

deBoor. C. "Good Approximation by Splines with Variable Knots. II" *Dundee Conf. on Numerical Solution of Differential Equations,* in Lecture Notes in Math #363, (1974)

deBoor. C. A Practical Guide to Splines. Springer-Verlag. (1978).

Fiacco. A.V., "Sensitivity Analysis for Nonlinear Programming Using Penalty Methods," *Math Prog,* 10. pp. 287-311. (1976).

Finlayson. B.A.. "Packed Bed Reactor Analysis By Orthogonal Collocation." *Chem. Eng. Set..* 26. pp. 1081-1091, (1971).

Finlayson. RA., The Method of Weighted Residuals and Variational Principles, Academic Press. (1972).

Finlayson. RA., Nonlinear Analysis in Chemical Engineering. McGraw-Hill (1980).

Gardini, L., A- Servida. and M. Morbidelli. and S. Carra, "Use of Orthogonal Collocation on Finite Elements with Moving Boundaries For Fixed Bed Catalytic Reactor Simulation." *Comp. & ChE..* 9(1). pp. 1-17. (1985).

Han. S.-P.. "A Globally Convergent Method for Nonlinear Programming." *JOTA* 22(3). p. 297. (1977).

Hindmarsh, A.G, "Lsode and LsodI, Two new initial value ordinary differential equation solvers," ACM-SIGNUM Newsletter, 15(4), pp. 10-11, (1980).

Hlavacek, V., "Aspects in Design of Packed Catalytic Reactors," *Ind. & Eng. Chem.*, 62(7), (1970).

Murtagh, RA., and M.A. Saunders, "MINOS/Augmented Supplementary User's Manual" Report OR/78/6 (1978), The University of New South Wales, Australia.

Powell, M.J.D., "A Fast Algorithm for Nonlinearly Constrained Optimization Calculations," *Paper presented at the Dundee Conf. on Numerical Analysis,* Dundee, Scotland (1977).

Pereyra, V., and E.G. Sewell, "Mesh Selection for Discrete Solution of Boundary Problems in Ordinary Differential Equations," *Numer. Math.*, 23. pp. 261-268, (1975).

Rivlin, T.J., An Introduction to the Approximation of Functions, Dover, (1969).

Russell, R.D., and J. Christiansen, "Adaptive Mesh Selection Strategies for Solving Boundary Value Problems," *SIAM J. Numerical Anal.*, 15(1), (197§).

Sargent, R.W.H., and G.R. Sullivan, The Development of an Efficient Optimal Control Package,*" *Proceedings of the 8th IFIP Conference on Optimization Techniques Pt. 2.*, (1977).

Villadsen, J., and M.L. Michelsen, Solution of Differential Equation Models by Polynomial Approximation, Prentice-Hall Inc., (1978).

Villadsen, J., and W.E. Stewart. "Solution of Boundary Value Problems by Orthogonal Collocation," *Chem. Eng. Sri.*, 22. p. 1483, (1967).

ACKNOWLEDGEMENT

Partial support for this research is made possible through an educational gram to Chemical Engineering from the Shell Education Foundation and from NSF grant #ENG-8451058.

APPENDIX

Equation (9) can be written in general as

$$z^{i}_{K+i}(t) = \sum_{j=o}^{K} z_{(i-1)(K+1)+j}\ \phi_{(i-1)(K+1)+j}(t)$$

$$A_{Mi\text{-}lXK\text{-}MHj}(t) = \prod_{\sim}^{K} \frac{U \text{ " } VIXK+IM/}{I\sim 7} T$$

for i=l....NE

which becomes for NE=3 and K=2:

$$z^{1}_{j}(t) = \sum_{j=o}^{2} z_j\ \phi_j(t) \qquad \phi_J(t) = \prod_{k=oj}^{2} \frac{(t - t_k)}{(t_j - t_k)}$$

$$z^{2}_{3}(t) = \sum_{j=0}^{2} \wedge\wedge \quad \wedge - \prod_{k=0,j}^{2} \frac{(t - t_{3+k})}{(t_{3+j} - t_{3+k})}$$

$$z^{1}_{:}(t) = \sum_{\wedge J=0}^{2} z\ \hat{0}\ (t) \qquad 4\ (t! = \prod_{k=0j}^{2} \frac{6\wedge k'}{6\wedge k \cdot}$$