

A New Parallel Optimization Algorithm  
for Parameter Identification  
in Ordinary Differential Equations<sup>1</sup>

J.E. Dennis, Jr.  
Karen A. Williamson

September, 1988

TR88-12

<sup>1</sup>Department of Mathematical Sciences, Rice University, P. O. Box 1892, Houston, TX 77251. This research was sponsored by AFOSR 85-0243, DOE DE-FG05-86ER25017, SDIO/IST/ARO DAAG-03-86-K-0113, and NSF grant CCR-86-19893. This paper was prepared by invitation for delivery at The 27th IEEE Conference on Decision and Control, December 1988.

# Report Documentation Page

Form Approved  
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

|  |                                    |                                     |                            |   |                                 |
|--|------------------------------------|-------------------------------------|----------------------------|---|---------------------------------|
| 1. REPORT DATE<br><b>SEP 1988</b>  |                                    | 2. REPORT TYPE                      |                            | 3. DATES COVERED<br><b>00-00-1988 to 00-00-1988</b> |                                 |
| 4. TITLE AND SUBTITLE<br><b>A New Parallel Optimization Algorithm for Parameter Identification in Ordinary Differential Equations</b>  |                                    |                                     |                            | 5a. CONTRACT NUMBER                                 |                                 |
|  |                                    |                                     |                            | 5b. GRANT NUMBER                                    |                                 |
|  |                                    |                                     |                            | 5c. PROGRAM ELEMENT NUMBER                          |                                 |
| 6. AUTHOR(S)   |                                    |                                     |                            | 5d. PROJECT NUMBER                                  |                                 |
|  |                                    |                                     |                            | 5e. TASK NUMBER                                     |                                 |
|  |                                    |                                     |                            | 5f. WORK UNIT NUMBER                                |                                 |
| 7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)<br><b>Computational and Applied Mathematics Department ,Rice University,6100 Main Street MS 134,Houston,TX,77005-1892</b> |                                    |                                     |                            | 8. PERFORMING ORGANIZATION REPORT NUMBER            |                                 |
| 9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)  |                                    |                                     |                            | 10. SPONSOR/MONITOR'S ACRONYM(S)                    |                                 |
|  |                                    |                                     |                            | 11. SPONSOR/MONITOR'S REPORT NUMBER(S)              |                                 |
| 12. DISTRIBUTION/AVAILABILITY STATEMENT<br><b>Approved for public release; distribution unlimited</b>  |                                    |                                     |                            |   |                                 |
| 13. SUPPLEMENTARY NOTES  |                                    |                                     |                            |   |                                 |
| 14. ABSTRACT   |                                    |                                     |                            |   |                                 |
| 15. SUBJECT TERMS  |                                    |                                     |                            |   |                                 |
| 16. SECURITY CLASSIFICATION OF:  |                                    |                                     | 17. LIMITATION OF ABSTRACT | 18. NUMBER OF PAGES<br><b>17</b>                    | 19a. NAME OF RESPONSIBLE PERSON |
| a. REPORT<br><b>unclassified</b>   | b. ABSTRACT<br><b>unclassified</b> | c. THIS PAGE<br><b>unclassified</b> |                            |   |                                 |

### **Abstract**

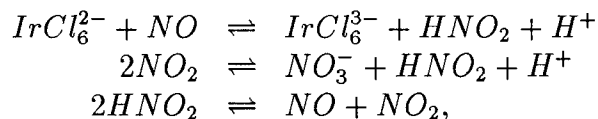
Often in mathematical modeling, it is necessary to estimate numerical values for parameters occurring in a system of ordinary differential equations from experimental measurements of the solution trajectories. We will discuss some of the difficulties involved in the solution of this problem, and we will describe a new parallel quasi-Newton algorithm for finding values of the parameters so that the numerical solution of the state equation best fits the observed data in the weighted least squares sense.



# 1 Introduction

The ability to numerically estimate parameters that appear in differential equation models of dynamic processes is crucial to the research effort in a variety of experimental science and engineering areas. Consider, for example, the process of modeling in chemical kinetics. One of the important goals in this field is to determine which elementary reactions constitute the mechanism of a complex reaction. First, a model consisting of a system of elementary chemical reactions is formulated to describe the proposed mechanism of the complex reaction system. The proposed chemical model determines the dependence of the reaction rate on the concentrations of the reactants and products. In this way, the chemical model is converted into a mathematical model in the form of a system of rate laws. This mathematical model then consists of a system of ordinary differential equations which describes the rate of change of the concentrations of the individual chemical species over time. In this model are parameters known as rate constants which relate the concentrations of the species to the rate of the reaction. Then, given some experimental measurements of the concentrations of the species at various times, we would like estimates of the rate constants. The proposed reaction mechanism can be evaluated by considering how well the solution of the dynamical system with the final estimates of the rate constants fits the experimental measurements.

For example, the following chemical reaction system;



with rate constants  $k_1, k_2, k_3, k_4, k_5$ , and  $k_6$ , has been under investigation by Ram and Stanbury [13]. The model for this reaction mechanism is a fifth-order system of ordinary differential equations. Given some measurements of the absorbance of  $IrCl_6^{2-}$  at various sample times, the objective is to estimate values for the rate constants of the reactions, i.e.  $k_1, \dots, k_6$ . Then, the proposed reaction mechanism can be evaluated by considering how well the integrated solution of the model with the final value of the parameter vector fits the data.

Important problems of this type arise in diverse areas of science and engineering. In addition to chemical kinetics, other examples of parameter

estimation problems can be found in biology, biochemistry, and robotics. In this paper we will describe some standard approaches to the solution of the parameter identification problem in systems of ordinary differential equations. Then, we will describe a new algorithm that is based on the Celis, Dennis, and Tapia trust region algorithm [7], [8] for equality constrained optimization problems. This new algorithm should be both more efficient and more stable than standard solution techniques, and it also provides a flexible framework for introducing parallelism into the parameter identification problem.

## 2 Problem Formulation

In order to discuss the new algorithm, we will first establish some notation. Let  $\mathbb{R}^n$  denote the real  $n$ -dimensional Euclidean space. Let  $p = (p_1, p_2, \dots, p_{N_p})^T \in \mathbb{R}^{N_p}$  be the vector of parameters to be estimated,  $y = (y_1, y_2, \dots, y_{N_y})^T \in \mathbb{R}^{N_y}$  be a state vector, and  $t \in \mathbb{R}$  be the independent variable, usually interpreted as time. We assume that the system we are studying is modeled by a parameterized first-order system of ordinary differential equations

$$\frac{dy}{dt} = F(t, y; p); \text{ with initial conditions } y(t_0) = y_0, \quad (1)$$

where  $F : \mathbb{R} \times \mathbb{R}^{N_y} \times \mathbb{R}^{N_p} \rightarrow \mathbb{R}^{N_y}$  satisfies some continuity conditions. Let  $y(t; p) \in \mathbb{R}^{N_y}$  denote the solution of (1) at time  $t$  with the parameter vector  $p$ , and let  $y_i(t; p)$  denote its  $i$ -th component. Then, given a set of data points

$$(\mathbf{tdata}_{ij}, \mathbf{ydata}_{ij}); \quad i = 1, \dots, N_y; \quad j = 0, \dots, N_d(i), \quad (2)$$

where  $\mathbf{ydata}_{ij}$  is an approximation to  $y_i(\mathbf{tdata}_{ij}; p_*)$  for some unknown parameter vector  $p_*$ , the parameter identification problem is to determine an estimate of  $p_*$ , such that the solution of (1) with this parameter vector “best fits” the given data. In this work, we will consider “best fit” to mean that  $p_*$  minimizes the sum of the squares of the deviations between the numerical solution of (1) and the data points,  $\mathbf{ydata}_{ij}$ , at the time points  $\mathbf{tdata}_{ij}$ ; i.e.  $p_*$  solves the following optimization problem:

$$\text{minimize } f(p) = \frac{1}{2} \sum_{i=1}^{N_y} \sum_{j=0}^{N_d(i)} (y_i(\mathbf{tdata}_{ij}; p) - \mathbf{ydata}_{ij})^2. \quad (3)$$

Notice that we have included the components corresponding to initial values at  $t_0$  in the objective function, and we have included initial values,  $y(t_0)$ , in the statement of the model, given in (1). Ideally, we would have exact initial values for all of the components of  $y$  in the form of the data points corresponding to  $t_0$ , and thus,

$$y_i(t_0) \equiv \text{ydata}_{i0}; \quad i = 1, \dots, N_y .$$

However, it is clear that there will be cases in which some or all of the initial values are known only approximately. In these situations, some or all of the components of  $y(t_0)$  should be treated as parameters. We use a vector,  $\text{ivpar} \in \mathbb{R}^{N_y}$ , to keep track of which components of the initial values are being included as parameters, i.e.:

$$\text{ivpar}_i = \begin{cases} 0 & \text{if } y_i(t_0) \text{ is not included as a parameter} \\ 1 & \text{if } y_i(t_0) \text{ is included as a parameter,} \end{cases} \quad (4)$$

and let  $N_{iv} \equiv \sum_{i=1}^{N_y} \text{ivpar}_i$  denote the number of initial values that are included as parameters. The dimension of the parameter vector, still denoted by  $N_p$ , now includes both the number of parameters contained in  $F(t, y; p)$  and the number of initial values that are included as parameters. Therefore, we will use

$$y_i(t_0) \equiv \begin{cases} \text{ydata}_{i0} & \text{if } \text{ivpar}_i = 0 \\ p_{N_p - N_{iv} + i} & \text{if } \text{ivpar}_i = 1 \end{cases} \quad (5)$$

as the initial conditions for the initial value problem given in (1). We will define the residual vector,  $R(p) = (R_1, R_2, \dots, R_{N_R})^T \in \mathbb{R}^{N_R}$ , such that for  $i = 1, \dots, N_y$ ,  $j = 1, \dots, N_d(i) + \text{ivpar}_i$ , and  $k = j + \sum_{m=1}^{i-1} N_d(m) + \text{ivpar}_m$ ,

$$R_k(p) = y_i(\text{tdata}_{ij}; p) - \text{ydata}_{ij}, \quad (6)$$

and the dimension of the residual is given by

$$N_R = N_{iv} + \sum_{i=1}^{N_y} N_d(i).$$

Thus, the objective function  $f(p)$  given by (3) is equivalent to

$$f(p) = \frac{1}{2} R(p)^T R(p) = \frac{1}{2} \|R(p)\|_2^2, \quad (7)$$

and it has the special structure of the nonlinear least squares objective function. We should point out that in some problems there is motivation to use a weighted least squares objective function, but for simplicity, we will not include the weighting factors in this discussion.

At this point, some remarks about the form we have assumed for the problem are appropriate. We have chosen this complex structure for the data set because its flexibility allows us to consider some real applications that would otherwise be excluded. The general data structure given in (2) allows us to include cases in which no data points are available for some of the components of  $y$  and cases in which the data points for different components of  $y$  do not occur at the same time points. To accomplish this,  $N_d(i)$  denotes the number of data points available for the  $i$ -th component of the solution vector and  $\mathbf{tdata}_{i;j}$  denotes the time corresponding to the  $j$ -th data point of the  $i$ -th component of  $y$ , i.e.  $\mathbf{ydata}_{i;j}$ .

In addition, we have assumed that the model is a first-order normal system of differential equations. Since any general  $n$ -th order differential equation which is resolved with respect to the  $n$ -th derivative can be transformed into an equivalent first-order system, this assumption is not restrictive.

### 3 Initial Value Approaches

The most straightforward approach to solving (3) is to use a quasi-Newton method from unconstrained optimization for the minimization and an initial value method for the required numerical integration. In order to fully exploit the structure of the problem, we can use a nonlinear least squares algorithm such as an augmented Gauss-Newton trust region method [10] to minimize the objective function (7). Let  $J(p) \in \mathbb{R}^{N_R \times N_p}$  denote the Jacobian matrix, i. e. the first derivative matrix of  $R(p)$ , where  $J(p)_{ij} = \frac{\partial R_i(p)}{\partial p_j}$ . Then, the first derivative of  $f(p) = \frac{1}{2}R(p)^T R(p)$  is

$$\nabla f(p) = \sum_{i=1}^{N_R} R_i(p) \cdot \nabla R_i(p) = J(p)^T R(p).$$

Similarly, the second derivative of  $f(p)$  is

$$\nabla^2 f(p) = \sum_{i=1}^{N_R} (\nabla R_i(p) \cdot \nabla R_i(p)^T + R_i(p) \cdot \nabla^2 R_i(p))$$



$$\begin{aligned}\nabla^2 f(p) &= J(p)^T J(p) + S(p) \\ \text{where } S(p) &= \sum_{i=1}^{N_R} R_i(p) \cdot \nabla^2 R_i(p)\end{aligned}$$

is the second-order information in the Hessian of  $f(p)$ . Then, at each iteration, a step  $s$  is chosen to solve the following trust region subproblem:

$$\begin{aligned}\text{minimize } m_c(p_c + s) &= \frac{1}{2}R(p_c)^T R(p_c) + (J(p_c)^T R(p_c))^T s \\ &\quad + s^T (J(p_c)^T J(p_c) + S(p_c))s \\ \text{subject to } \|s\|_2 &\leq \delta_c,\end{aligned}$$

where  $m_c$  is the local quadratic model of  $f$  at the current parameter vector  $p_c$ . The trust region radius  $\delta_c$  provides a region in which we can “trust” the local model  $m_c$ .

At each iteration we must solve the initial value problem (1) in order to compute  $R(p_c)$ . We can use finite differences to compute the required Jacobian of  $R$ , or we can use the sensitivity equations. In the sensitivity equations approach, the following system of equations:

$$\frac{d}{dt} \left( \frac{\partial y_i}{\partial p_j} \right) = \frac{\partial F_i}{\partial p_j} + \sum_{k=1}^{N_y} \frac{\partial F_i}{\partial y_k} \cdot \frac{\partial y_k}{\partial p_j}; \quad i = 1, \dots, N_y; \quad j = 1, \dots, N_p$$

with appropriate initial conditions is numerically integrated to obtain an approximation to the Jacobian. Notice that this approach requires the simultaneous integration of the initial value problem (1), and it requires analytical expressions for the derivatives  $\frac{\partial F}{\partial p}$  and  $\frac{\partial F}{\partial y}$ . Finally, we use a secant approximation to the second-order portion of the Hessian [10].

Numerical testing indicates that these nonlinear least squares algorithms with the Jacobian calculated using finite differences or using the sensitivity equations work well on a wide variety of test problems. However, these algorithms have room for improvement in two major areas. Our experience indicates that a good initial guess for the parameter vector is important, but not because of the optimization method. When the initial guess is far from the solution, the difficulty is that the algorithm may reach a parameter vector for which the resulting initial value problem is unstable and where the residual can not be calculated.

In addition, the calculation of the residual and of the columns of the Jacobian are computationally intensive. In fact, experience indicates that typically ninety percent or more of the cpu time is spent on these calculations.

One way to handle this difficulty is through the use of parallel computing to exploit the independence of the calculation of the residual and each of the columns of the Jacobian. This parallelism occurs very naturally in the finite difference nonlinear least squares method. Similarly, in the sensitivity equations approach, each column of the Jacobian requires the numerical integration of an independent block of equations. However, the parallelism in this method is complicated by the fact that  $y' = F(t, y; p)$  must be integrated along with each block of equations in order to evaluate the partial derivative matrices. In addition, further speed-up will be obtained by exploiting the parallelism in the linear algebra needed for the calculation of the optimization step, [5], [9].

## 4 Nonlinear Programming Approaches

In order to improve the stability and efficiency of our algorithms, we want to investigate techniques for calculating the residual based on boundary value methods [11], [14]. Specifically, we are considering a multiple shooting technique in place of the initial value methods for the numerical integration of the system given in (1). Multiple shooting can be considered a form of domain decomposition for ordinary differential equations. Thus, in addition to adding stability, multiple shooting provides a technique for introducing additional parallelism into the parameter identification problem. Furthermore, multiple shooting leads to a constrained formulation of the optimization problem that will allow us to use methods which we believe will be more efficient than the initial value approaches presented in the previous section.

Let  $t_f \equiv \max\{t_{data_{ij}}; i = 1, \dots, N_y; j = 0, \dots, N_d(i)\}$ . In the multiple shooting method, shooting parameters,  $z_k(p) \in \mathbb{R}^{N_y}$ ;  $k = 1, \dots, M$ , are introduced into the problem at time points  $T_k$ ;  $k = 1, \dots, M$ , satisfying

$$t_0 < T_1 < T_2 < T_3 \dots T_M < t_f.$$

The shooting parameters,  $z_k$ ;  $k = 1, \dots, M$ ; represent the solution of the initial value problem (1) at the times  $T_k$ ;  $k = 1, \dots, M$ . Let  $y(t, p; T_k, z_k)$  denote the solution of

$$y' = F(t, y; p); \text{ with initial values } y(T_k; p) = z_k \quad (8)$$

on the interval  $[T_k, T_{k+1}]$ . Then, the problem consists of determining the vectors  $z_k$ ;  $k = 1, \dots, M$ ; in such a way that the function

$$y(t; p) = \begin{cases} y(t, p; t_0, y(t_0)) & \text{for } t \in [t_0, T_1) \\ y(t, p; T_k, z_k) & \text{for } t \in [T_k, T_{k+1}); k = 1, \dots, M - 1 \\ y(t, p; T_M, z_M) & \text{for } t \in [T_M, t_f] \end{cases}$$

is continuous over the entire interval, and thus a solution of the initial value problem (1). This yields continuity conditions at each  $T_k$ ;  $k = 1, \dots, M$  for the unknowns  $z_k$ ;  $k = 1, \dots, M$ . These conditions constitute a nonlinear system of equations of dimension  $M \cdot N_y$  of the form:

$$h(p, z) = \begin{bmatrix} y(T_1, p; t_0, y(t_0)) - z_1 \\ y(T_2, p; T_1, z_1) - z_2 \\ \vdots \\ y(T_M, p; T_{M-1}, z_{M-1}) - z_M \end{bmatrix} = 0 \quad (9)$$

Thus, the numerical integration of the system of differential equations (1) now requires the solution of the nonlinear system given by (9).

The most obvious way to add the stability of the multiple shooting approach to the algorithms described in the previous section is to calculate the residual using this multiple shooting technique instead of an initial value method. Unfortunately, each residual calculation now has the additional expense of the iterative solution of the nonlinear system of equations (9). Even though we could use a structured variation of Broyden's method to calculate the Jacobian of  $h$  at each step of the iterative solution of this nonlinear system of equations, we must numerically integrate each of the initial value problems of the form (8) in order to evaluate  $h(p_c, z(p_c))$ . Therefore, it is clear that merely substituting multiple shooting for the initial value methods in the algorithms discussed in Section 3 will lead to more stable, but slower algorithms. However, we will now show that multiple shooting leads to a different formulation of the optimization problem which will allow us to develop a more efficient and stable algorithm.

Recall that we are interested in solving the following optimization problem:

$$\text{minimize } f(p) = \frac{1}{2} R(p)^T R(p),$$

and that calculating the residual for the current parameter vector using multiple shooting is equivalent to solving the nonlinear system,  $h(p_c, z(p_c)) = 0$ .

Following Bock, [1], [2], we consider instead the following constrained parameter identification problem:

Problem CPID:

$$\begin{aligned} & \text{minimize} && f(p, z) = \frac{1}{2}R(p, z)^T R(p, z) \\ & \text{subject to} && h(p, z) = 0, \end{aligned}$$

where  $p$  and  $z$  are regarded as independent variables. In this context, the initial value approaches described in Section 3 can be considered as methods which require each parameter iterate to be *feasible*, i.e.  $y(t; p)$  solves the initial value problem (1) at each iteration, by regarding  $z$  as dependent on  $p$ . However, this constrained formulation will allow us to consider algorithms designed for equality constrained optimization: algorithms which will allow *infeasible* iterates. Nonlinear programming experience has shown that algorithms which allow infeasible iterates are generally more efficient than algorithms which require each iterate to be feasible.

## 4.1 SQP Formulation

One of the most popular methods for solving the equality constrained optimization problem is the successive quadratic programming (SQP) method. At each iteration, the SQP method solves a quadratic programming problem of the form

$$\begin{aligned} & \text{Problem QP:} && (10) \\ & \text{minimize} && \nabla f(p, z)^T s + \frac{1}{2}s^T B s \\ & \text{subject to} && h(p, z) + \nabla h(p, z)^T s = 0 \end{aligned}$$

where  $B$  is an approximation to the Hessian of the Lagrangian  $l(p, z, \lambda)$  for the step  $s_{QP} \equiv (\Delta p, \Delta z)^T$  and the Lagrange multipliers  $\lambda^{QP}$ . The Lagrangian function associated with Problem CPID is the function

$$l(p, z, \lambda) = f(p, z) + \lambda^T h(p, z) \quad (11)$$

where  $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_{N_p \cdot M})^T$  are the Lagrange multipliers. We say that the point  $(p_+, z_+)^T = (p_c, z_c)^T + (\Delta p, \Delta z)^T$  is *linearly feasible* if it satisfies the linearized constraint

$$h(p_c, z_c) + \nabla h(p_c, z_c)^T s = 0$$

where  $s \equiv (\Delta p, \Delta z)^T$ .

The fast local convergence properties of the SQP method have been fairly well established, but the issue of a satisfactory globalization strategy still remains open. A number of line search techniques have been proposed by various authors, but none of them have proven to be particularly successful. One of the reasons for this is that there is no natural descent function for the constrained optimization problem due to the conflict between decreasing the objective function and moving towards feasibility.

H. G. Bock and his coworkers have implemented a Han-Powell type SQP method for the parameter identification problem in their code PARFIT [1], [2], [3]. Unfortunately, they experience all of the difficulties that make SQP implementations hard. Older implementations of SQP encounter difficulties in handling rank degeneracy in the Jacobian of the constraints, lack of second-order sufficiency in the QP subproblem, and the situation when the solution to the QP subproblem is not a good descent direction for the standard merit functions. We believe that our new algorithm will overcome these difficulties.

## 4.2 CDT Formulation

In this section we will discuss our new algorithm which is based on a trust region globalization strategy for equality constrained optimization that has been developed by Celis, Dennis, and Tapia, [7], [8]. The most obvious trust region approach is to simply add a trust region constraint, i.e. a constraint which limits the size of the step, to Problem QP. Unfortunately, the resulting trust region subproblem may not have a solution since there may not be a linearly feasible point inside the trust region. Therefore, we want to relax the constraint  $h(p_c, z_c) + \nabla h(p_c, z_c)^T s = 0$  so that the trust region subproblem has a solution without sacrificing convergence to a feasible point.

As motivation for how much linear feasibility we will require at each iteration, consider solving  $h(p, z) = 0$  using an unconstrained trust region approach. At each iteration we would minimize a quadratic model of  $\frac{1}{2}\|h(p, z)\|_2^2$  subject to a trust region constraint on the length of the step, i. e. ,  $\|s\|_2 \leq \delta_c$ . Let  $s_{CP} \equiv -\alpha_c \nabla h(p_c, z_c)^T h(p_c, z_c)$  denote the step to the Cauchy point, i.e., the minimizer in the trust region  $\{s : \|s\|_2 \leq \delta_c\}$  of  $\|h(p_c, z_c) + \nabla h(p_c, z_c)^T s\|_2^2$  along the direction of its negative gradient. If the algorithm took the step to the Cauchy point  $s_{CP}$  on the model  $\|h(p_c, z_c) + \nabla h(p_c, z_c)^T s\|_2^2$ , then under mild assumptions, first order convergence can be established, [6]. With

this motivation, Celis, Dennis, and Tapia consider the following trust region subproblem:

Problem NDTR:

$$\begin{aligned} & \text{minimize} && \nabla f(p_c, z_c)^T s + \frac{1}{2} s^T B s \\ & \text{subject to} && \|h(p_c, z_c) + \nabla h(p_c, z_c)^T s\|_2 \leq \theta_c \\ & && \|s\|_2 \leq \delta_c. \end{aligned}$$

where  $\theta_c$  is chosen to be  $\|h(p_c, z_c) + \nabla h(p_c, z_c)^T \hat{s}\|_2$  for some  $\hat{s}$  that satisfies  $\|\hat{s}\|_2 \leq \delta_c$ . For example,  $\hat{s}$  could be taken to be the step to the Cauchy point,  $s_{CP}$ . Unfortunately, in the case where both of the quadratic constraints are binding, Problem NDTR becomes difficult and expensive to solve.

Motivated by the work of Byrd, Schnabel and Shultz [4] on trust region methods for unconstrained optimization, Celis, Dennis and Tapia [8] have developed a more convenient trust region subproblem by restricting the trust region subproblem to a two-dimensional subspace spanned by  $s_{QP}$  and  $\hat{s}$  as follows:

Problem 2DTR:

$$\begin{aligned} & \text{minimize} && \nabla f(p_c, z_c)^T s + \frac{1}{2} s^T B s \\ & \text{subject to} && \|h(p_c, z_c) + \nabla h(p_c, z_c)^T s\|_2 \leq \theta_c \\ & && \|s\|_2 \leq \delta_c \\ & && s \in \text{span}\{s_{QP}, \hat{s}\} \end{aligned}$$

where  $s_{QP}$  is the solution to Problem QP. Numerical testing on standard nonlinear programming problems indicates that this two-dimensional trust region algorithm is generally more efficient than other SQP implementations. Furthermore, we will be assured of global convergence, under reasonable conditions, by the results given in the thesis of El-Alem [12].

Thus, our new algorithm retains the good stability properties of the multiple shooting approach, and in addition, we believe that it will prove to be more efficient and economical than the algorithms based on initial value methods. First, our new algorithm should be more efficient than the initial value approaches since it will allow infeasible iterates. The initial value approaches require each iterate to be feasible. In addition, we expect our algorithm to be more efficient than other SQP implementations because we do

not require the iterates to be linearly feasible. This feature of the algorithm also overcomes the difficulty when there is no linearly feasible point.

In addition to its stability and efficiency, our new algorithm provides a technique for introducing additional parallelism into the solution of the parameter identification problem. This is due to the crucial observation that the multiple shooting approach can be viewed as a powerful domain decomposition technique for this problem. Furthermore, our new algorithm retains the parallelism inherent in the initial value approaches since the residual and the columns of the Jacobian of the residual are still independent calculations. The Jacobian can be calculated using either finite differences or the sensitivity equations, and the obvious parallelism in both of these methods has been discussed previously. As a result of the multiple shooting approach, the numerical integration needed for the computation of  $h(p, z)$  and the necessary Jacobian matrices has been divided into many shorter, independent numerical integrations on the subintervals  $[T_i, T_{i+1}]$ .

Thus, this domain decomposition technique is a powerful tool that will allow us to overcome several difficulties. First, we will choose the shooting points, i.e.  $T_i$ , that are necessary to obtain stability of the initial value problem. In addition, this multiple shooting approach provides us with a flexible framework for load balancing, for we can now choose additional shooting points to further divide the time interval into smaller intervals to make effective use of the number of processors that are available. Finally, this algorithm also has opportunities for further speed-up within the trust region subproblem since this subproblem can be divided into several independent computations.

## 5 Conclusion

Through the use of multiple shooting, we have developed a new algorithm for the parameter identification problem that promises to be more efficient than the initial value approach and other SQP implementations. In addition, multiple shooting provides a domain decomposition technique which allows us to judiciously introduce parallelism into this optimization problem. Furthermore, this approach already puts us in the proper framework to treat important practical side constraints on the parameter vector, both direct constraints, like  $p \geq 0$ , or more interesting state constraints like  $G(t, y; p) \geq 0$ .

This paper provides an overview of a new parallel optimization algorithm for the solution of the parameter estimation problem in systems of ordinary differential equations. Further details may be found in [15].

The authors wish to thank Richard Tapia and David Dobson for their helpful comments.

## References

- [1] H. G. Bock. Recent advances in parameteridentification techniques for O.D.E. In *Numerical Treatment of Inverse Problems in Differential and Integral Equations*, 1983.
- [2] H. G. Bock. *Randwertproblemmethoden zur Parameteridentifizierung in Systemen Nichtlinearer Differentialgleichungen*. PhD thesis, Universität Bonn, 1985.
- [3] H. G. Bock, E. Eich, and J. P. Schlöder. Numerical solution of constrained least squares boundary value problems in differential-algebraic equations. Technical report, Universität Heidelberg, 1987.
- [4] R. H. Byrd, R. B. Schnabel, and G. A. Shultz. Approximate solution of the trust region problem by minimization over two-dimensional subspaces. Technical report, University of Colorado, 1986.
- [5] R. H. Byrd, R. B. Schnabel, and G. A. Shultz. Parallel quasi-Newton methods for unconstrained optimization. Technical Report CU-CS-396-88, University of Colorado, 1988.
- [6] M. R. Celis. *A Trust Region Strategy for Nonlinear Equality Constrained Optimization*. PhD thesis, Rice University, 1985.
- [7] M. R. Celis, J. E. Dennis, and R. A. Tapia. A trust region strategy for nonlinear equality constrained optimization. In P. Boggs, R. Byrd, and R. Schnabel, editors, *Numerical Optimization*, pages 71–82. SIAM, 1985.
- [8] M. R. Celis, J. E. Dennis, and R. A. Tapia. An algorithm based on a convenient trust-region subproblem for nonlinear programming. Technical report, Rice University, 1988.



- [9] T. F. Coleman and P. E. Plassmann. Solution of nonlinear least-squares problems on a multiprocessor. Technical Report TR 88-923, Cornell University, 1986.
- [10] J. E. Dennis, Jr., D. M. Gay, and R. E. Welsch. An adaptive nonlinear least squares algorithm. *TOMS*, 7, 1981.
- [11] P. Deuffhard and G. Bader. Multiple shooting techniques revisited. In *Numerical Treatment of Inverse Problems in Differential and Integral Equations*, 1983.
- [12] M. M. El-Alem. *A Global Convergence Theory for the Celis-Dennis-Tapia Trust Region Algorithm for Constrained Optimization*. PhD thesis, Rice University, 1988.
- [13] M. S. Ram and D. M. Stanbury. Kinetics and equilibria for the hexachloriridate(IV/III) redox couple in nitrous acid. *Inorg. Chem.*, 1984.
- [14] J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis*. Springer-Verlag, 1980.
- [15] K. A. Williamson. *Parameter Identification in Systems of Ordinary Differential Equations*. PhD thesis, Rice University, Department of Mathematical Sciences, 1989. In preparation.