Computers and Chemical Engineering 78 (2015) 39–50

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/compchemeng

Initialization strategies for optimization of dynamic systems

Seyed Mostafa Safdarnejad[a], John D. Hedengren[a,*], Nicholas R. Lewis[a], Eric L. Haseltine[b]

[a] Department of Chemical Engineering, Brigham Young University, Provo, UT 84602, United States
[b] Vertex Pharmaceuticals, 50 Northern Avenue, Boston, MA 02210, United States

ARTICLE INFO

Article history:
Received 2 February 2015
Received in revised form 10 April 2015
Accepted 15 April 2015
Available online 24 April 2015

Keywords:
Initialization
Decomposition
Differential algebraic equations
Dynamic optimization
Large-scale
Smart grid energy system

ABSTRACT

For dynamic optimization applications, real-time solution reliability is improved if there is an initialized prior solution that is sufficiently close to the intended solution. This paper details several initialization strategies that are useful for obtaining an initial solution. Methods include warm start from a prior solution, linearization, structural decomposition, and an incremental unbounding of decision variables that leads up to solving the originally intended problem. Even when initialization is not required to solve a dynamic optimization problem, a staged initialization approach sometimes leads to an overall faster solution time when compared to a single optimization attempt. Several challenging optimization problems are detailed that include a high-index differential and algebraic equation pendulum model, a standard reactor model used in many benchmark tests, a tethered aerial vehicle, and smart grid energy storage. These applications are representative of a larger class of applications resulting from the simultaneous approach to optimization of dynamic systems.

1. Introduction

Differential and algebraic equations (DAEs) are natural expressions of many physical systems found in business, mathematics, systems biology, engineering, and science. In business, the supply chain can be optimized by modeling the storage, production, and consumption throughout a network (Manenti et al., 2010). In mathematics, ordinary (ODEs) or partial differential equations (PDEs) are used to describe certain classes of boundary value problems. In engineering, these equations result from material, energy, momentum, and force balances (Buzzi-Ferraris and Manenti, 2014). In science, laws of motion are naturally described by differential equations that relate position, velocity, and acceleration (Sun et al., 2014; Kumar and Daoutidis, 1999).

Just as differential equations naturally describe many systems, these same equations can also be used to optimize among many potential designs or feasible solutions. One difference between static or steady-state models and dynamic models is that optimal solutions must not only observe constraints at one time point, but also along a future time window. Part of what makes a dynamic solution challenging is that design variables at one time instant affect both current and future objective values and constraints in the time horizon. This is generally challenging from an optimization standpoint because of many degrees of freedom that are adjustable at each time step, strong nonlinear relationships, and a wide range of sensitivities between the adjustable parameters and multiple objectives.

1.1. Simulation and optimization of DAE systems

There are many solution approaches for sets of ODEs or DAEs and a review of all possible methods is beyond the scope of this work. Dynamic systems can be solved as ODEs or DAEs through the simultaneous approach (Biegler, 2007; Biegler et al., 2012; Carey and Finlayson, 1975; Renfro et al., 1987; Liebman et al., 1992; Albuquerque and Biegler, 1995; Cervantes and Biegler, 1998) to dynamic optimization as opposed to a semi-sequential (Hong et al., 2006) or sequential approach (Míguez, 2010; Binder et al., 2001; Diehl et al., 2002; Assassa and Marquardt, 2014; Leppävuori et al., 1020g). The sequential method is where the model equations and objective function are calculated in successive evaluations. In a sequential approach, the DAEs are solved independently of the objective function. Each evaluation of the objective function involves fixing the independent variables at current iteration values and solving the dynamic equations forward in time with a shooting approach. It is referred to as a shooting method because trial solutions are propagated forward in time and the resulting dynamic trajectory is used to calculate the objective function. Successive evaluations of the objective function are used to compute gradients of the objective with respect to the decision variables

* Corresponding author. Tel.: +1 801 477 7341; fax: +1 801 422 0151.
E-mail address: john.hedengren@byu.edu (J.D. Hedengren).

http://dx.doi.org/10.1016/j.compchemeng.2015.04.016
0098-1354/

and drive towards an optimal solution. Terminating the optimization progress before convergence typically produces a feasible yet sub-optimal result. Sequential or shooting methods use forward integrating solvers for differential equations with variable time steps to maintain the integration accuracy. A number of solvers or modeling platforms exist for solving ODE or DAE problems with either sequential or simultaneous methods (Brenan et al., 1996; Ascher and Petzold, 1998) such as DASSL (Petzold, 1982), SUNDIALS (Hindmarsh et al., 2005), and many others (Cizniar et al., 2005; Houska et al., 2011; Piela et al., 1991; Tummescheit et al., 2010; Simon et al., 2009; Nagy et al., 2007).

Dynamic models can be translated into sets of algebraic constraints that can be solved with standard gradient-based optimization techniques. The differential terms can be translated into algebraic equations through orthogonal collocation on finite elements. Orthogonal collocation on finite elements allows a simultaneous solution where objective function and equations are solved together instead of sequentially. Orthogonal collocation is simply a technique that relates differential terms to state values in a discretized time horizon. This translation of DAEs into a set of algebraic equations also allows capable linear programming (LP), quadratic programming (QP), nonlinear programming (NLP), or mixed-integer nonlinear programming (MINLP) solvers to optimize these dynamic systems with a simultaneous approach instead of shooting methods that rely on forward integrating simulators. Similar approaches are used for ODEs, DAEs, PDEs, and Partial DAEs. Large-scale problems such as PDEs or PDAEs with few decision variables may be best suited for analysis by a sequential or shooting method. Small or medium scale problems with many decision variables or unstable systems are best suited for analysis with the simultaneous approach. Dynamic problems can include continuous or discrete variables that can be solved with MINLP solvers, have multiple competing objectives, and require robust or stochastic optimization methods to deal with uncertainty. Unlike sequential approaches, terminating the optimization progress does not give a feasible sub-optimal result. It is only at final convergence that the equations are satisfied with the objective function at an optimal value. The solvers and modeling platform used in this study are embedded in the APMonitor Modeling Language and Optimization Suite (Hedengren, 2015).

1.2. Standard DAE form

Dynamic modeling of physical systems involves several phases starting with the selection of a model form. Dynamic model forms may be empirical where the form of the model is determined from data, fundamental where the model parameters and equations are derived from first principles, or hybrid with a mix of empirical and fundamental relationships. One advantage of using empirical models is that only inputs and outputs must be collected for the model development and less must be known about the process in order to develop a model. Fundamental models are often difficult to develop because particular relationships can either be unknown or impossible to isolate. In each case, the differential equations relate certain process inputs (u) to differential states (x) or algebraic states (y).

The method taken in this work is to solve hybrid dynamic process models in open-equation form with either differential or algebraic equations while minimizing an objective function. Differential equations are simply those that contain at least one differential term and algebraic equations are those that do not. Equations may also consist of equality (=) or inequality (< or $\leq$) constraints as shown in Eq. (1):

$$\min_u h(x, y, u) \tag{1a}$$

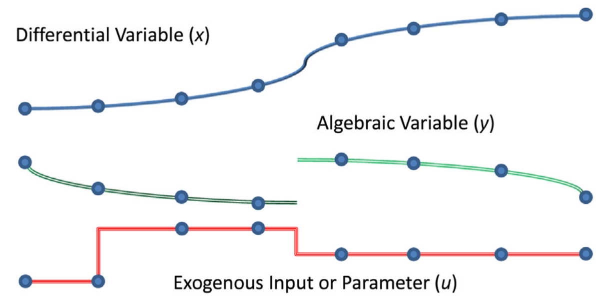

Fig. 1. DAE model equations are discretized and solved over a time horizon.

$$0 = f\left(\frac{dx}{dt}, x, y, u\right) \tag{1b}$$

$$0 \leq g\left(\frac{dx}{dt}, x, y, u\right) \tag{1c}$$

where Eq. (1b) is the set of DAE equality constraints and Eq. (1c) is the set of DAE inequality constraints. For solvers that require only equality constraints and simple inequality bounds on variables, the inequality constraints are converted to an equality constraint with the addition of a slack variable (Vu and Li, 2010). Equations need not contain differential states, states variables, inputs, and outputs. However, each equation must contain at least one differential or algebraic state or output variable.

The inputs may consist of parameters that are either known from fundamental relationships or measured directly. There may also be unknown parameters that can either be inferred from other measurements or unknown parameters that are unobservable given the available measurements. Other types of inputs may be disturbances that affect the system that are either measured or unmeasured. Finally, inputs also include those that can be changed to optimize or control the system. These are referred to as design variables or manipulated variables depending on whether it is a design or control application. These parameters, disturbances, or manipulated variables constitute the set of exogenous inputs that change independently of the system dynamics and act on the system to change the dynamic response.

Differential states are those variables that are calculated based on differential equations while algebraic states are those variables that do not appear as differential terms. Algebraic states may be either continuous or discontinuous while differential states are typically considered as continuous as shown in Fig. 1. For dynamic simulation models there must be a unique equality constraint or binding inequality constraint for each model state. If there are more variables than equations ($n_{var} \geq n_{eqn}$), the system has degrees of freedom that can be arbitrarily adjusted to best meet one or more objectives. If there are more equations than variables ($n_{eqn} \geq n_{var}$), the system may be over-specified and there is likely no set of variables that can simultaneously satisfy all constraints.

1.3. DAE models with higher order derivatives

Equations that contain higher order derivatives can also be fit into the standard form as shown in Eq. (1) by creating additional variables for every higher order derivative. For example, acceleration is equal to the second derivative of position as in $a = d^2x/dt^2$. By adding the additional variable of velocity and an additional equation, the second order system becomes a set of two first order differential equations as in $a = dv/dt$ and $v = dx/dt$ where a is acceleration, v is velocity, and x is position. A similar approach can be used for any higher order derivatives. Initialization of higher order derivative models requires an initial condition that is specified for each differential variable.