Application of Targeted Automatic Differentiation to Large Scale Dynamic Optimization

Derya B. Özyurt and Paul I. Barton
Department of Chemical Engineering
Massachusetts Institute of Technology
Cambridge, MA 02139
derya@mit.edu, pib@mit.edu

Automatic Differentiation (AD) has an increasingly important and enabling role in the large-scale computations of interest to the chemical and biological processing industries. The solution of the large-scale dynamic optimization described by ODEs or DAEs and many optimization parameters is one of these computations. In this paper, an efficient method employing a targeted AD approach is described to solve the following problem

$$\min_{p} \quad J(p) = \int_{t_0}^{t_f} g(t, x(t,p), p)\; dt + h(x(t_f, p), p)$$

subject to

$$\begin{aligned} f(t, x, \dot{x}, p) &= 0 \\ f(t_0, x(t_0), \dot{x}(t_0), p) &= 0 \\ p^L \leq p &\leq p^U. \end{aligned} \tag{1}$$

In general, the solution procedure for this problem requires the computation of the first and second order derivatives for the objective function. These derivatives can be obtained by solving first and second order sensitivities of the state variables. However, this procedure of calculating exact derivatives is computationally expensive unless the number of optimization parameters are relatively small. By solving the adjoint system for a given direction in the parameter space this dependence on the number of the parameters can be reduced significantly.

A targeted AD approach converts the problem consisting of state equations into the directional Second Order Adjoint (dSOA) system consisting of the directional first order sensitivity equations, first order adjoint of the sensitivities and directional second order adjoint system [5]. The solution of the constructed system yields the directional second order derivative information efficiently. The usage of automatic differentiation provides a subtle way to obtain directional first order sensitivity system, and first and second order adjoint system equations in addition to some other matrix-vector, vector-matrix and vector-matrix-vector products required, efficiently and without round-off errors. Moreover, for stiff ODEs and DAEs this targeted AD approach enables a better utilization of the state-of-the-art integrators.

The second order information provided by dSOA is used in updating the current estimate of the solution for (1) by approximately solving the Newton equations

$$\nabla^2 J(p^{(k)}) u = -\nabla J(p^{(k)}) \tag{2}$$

using a Truncated Newton method [4, 2].

The Truncated Newton algorithm adapted to solve these large-scale dynamic optimization problems can be summarized as follows:

1. Set iteration number, $k = 0$ and obtain an initial approximation $p^{(0)}$ to the optimal solution p^*.

2. If the approximation $p^{(k)}$ is the local minimizer of J within a given tolerance, Stop.

3. Solve Eqn.(2) approximately by a modified-Lanczos algorithm [2] using preconditioning [3] to obtain a search direction, u. Gradient evaluations are performed by a first order adjoint method [1]. At each iteration of this inner loop the Hessian-vector products are calculated by dSOA method.

4. Apply a line search to find $\alpha > 0$ such that $J(p^{(k)} + \alpha u) < J(p^{(k)})$.

5. Set $p^{(k+1)} = p^{(k)} + \alpha u$, $k = k + 1$ and go to Step 2.

This algorithm is implemented using Nash's Truncated Newton code for optimization problems with bound on variables [2]. The original code is modified to employ dSOA method in Step 3.

The implementation of "dSOA powered" Truncated Newton method is tested on several dynamic optimization problems successfully. These examples show that the proposed approach employing targeted AD and adjoint system theory along with a Truncated Newton method results in an accurate and cost efficient procedure for the solution of large scale dynamic optimization problems.

References

[1] Y. Cao, S. Li, L. Petzold, and R. Serban. Adjoint sensitivity analysis for differential-algebraic equations: The adjoint DAE system and its numerical solution. *SIAM Journal of Scientific Computing*, 24(3):1076–1089, 2003.

[2] Stephen G. Nash. Newton-type minimization via the Lanczos method. *SIAM Journal on Numerical Analysis*, 21(4):770–788, 1984.

[3] Stephen G. Nash. Preconditioning of Truncated-Newton methods. *SIAM Journal on Scientific and Statistical Computing*, 6(3):559–616, 1985.

[4] Stephen G. Nash. A survey of Truncated-Newton methods. *Journal of Computational and Applied Mathematics*, 124:45–59, 2000.

[5] Derya B. Özyurt and Paul I. Barton. Cheap second order directional derivatives of stiff ODE embedded functionals. Submitted to *SIAM Journal on Scientific Computing*, 2003.