Estimation of a State of the Model Variable Field for the Burgers Equation

Yasuyoshi Horibata

Faculty of Science and Enginnering, Hosei University, Japan

1 Introduction

Numerical simulation of an unsteady model is an initial/boundary value problem: given an initial condition of the model, and boundary conditions, the model simulates the evolution of the model variables. Obviously, the more accurate the estimate of the initial conditions, the better the quality of the forecasts. Estimation is the process through which all the available information is used in order to estimate as accurately as possible the state of all the model variables (an initial condition). The available information consists of the observation proper, and of the physical laws that govern the evolution of the model variables. The latter are available in practice under the form of a numerical model [1].

In this paper, the Estimation is formulated as a nonlinear optimization problem with tens of thousands of variables. Problems of this size can be solved efficiently only if the storage and computational costs of the optimization algorithm can be kept at a tolerable level. Hence, a large-scale optimization method is employed to minimize the objective function. At each iteration, it requires the gradient of the objective function. A method is developed for computing the gradient efficiently. Finally, numerical experiment is presented.

2 Formulation

Consider the nonlinear Burgers equation:

$$\frac{\partial u}{\partial t} = -u\frac{\partial u}{\partial x} + \frac{1}{R}\frac{\partial^2 u}{\partial x^2} \tag{1}$$

Once it has been disctretized in space using finite differences to n independent variables, the model can be written as a set of n nonlinear coupled ordinary differential equations:

$$\frac{d\mathbf{x}}{dt} \quad = \quad \mathbf{F}(\mathbf{x}) \tag{2}$$

where

$$\mathbf{x} = \begin{pmatrix} u_1 \\ \vdots \\ u_n \end{pmatrix} \qquad \mathbf{F} = \begin{pmatrix} F_1 \\ \vdots \\ F_n \end{pmatrix} \tag{3}$$

This is the model in differential form. Once a time-difference scheme is chosen, it becomes a set of nonlinear-coupled difference equations; this is the forward model.

A numerical solution of the model (2) starting from an initial time t_0 can be readily obtained by integrating the model numerically using the forward model between t_0 and a final time t. Let $\mathbf{x}(t_0)$ denote the initial condition from which the model is numerically integrated. The initial condition defines a unique solution $\mathbf{x}(t)$ to the model.

The distance function is taken as

$$J = \sum_{n_2}^{n_1} H[\mathbf{x}(t_i)]\, dt \tag{4}$$

where $H[\mathbf{x}(t_i)]$ is a scalar measuring the distance between $\mathbf{x}(t_i)$ and its observations available at time $t_i = i\Delta t$. The available observations are assumed to be distributed over a limited time interval $[t_2, t_1] \quad (t_2 = n_2\Delta t, \quad t_1 = n_1\Delta t, \quad t_0 < t_2 < t_1)$. The constraint is the model equation. For a given initial condition and for the corresponding solution $\mathbf{x}(t_i)$ of the forward model, the distance function is evaluated Thus, the distance function is regarded as a function of $\mathbf{x}(t_0)$.

Hence, estimation of $\mathbf{x}(t_0)$ is formulated as an optimization problem:

$$\min\ J(\mathbf{x}(t_0))$$

An optimization method is used to find the value $\mathbf{x}(t_0)$ which minimizes J, starting from an initial guess of $\mathbf{x}(t_0)$.

3 Numerical Optimization

This optimization problem has thousands or millions of variables. Hence, L-BFGS-B [2] are used to solve this large-scale problem. Since the method does not require second derivatives of the objective function, it can be applied when the Hessian is not practical to compute. It uses the limited memory BFGS approximation to the Hessian, and so creates a quadratic model function using gradient information in such a way that the storage required in linear in the number of the variables.

4 Forward Model

The forward model integrates the Buugers equation using the finite difference scheme; the central difference scheme is used for the diffusion term whereas the second order upwind difference scheme is used for the advection term:

$$\left(\frac{\partial u}{\partial x}\right)_i = \begin{cases} \frac{u_{i-2} - 4u_{i-1} + 3u_i}{2\Delta x} & (u_i > 0) \\ \frac{-3u_i + 4u_{i+1} - u_{i+2}}{2\Delta x} & (u_i < 0) \end{cases} \tag{5}$$

where $x_i = i\Delta x$.

5 Gradient Computation

Each iteration of numerical optimization requires the gradient of the objective function. If a finite-difference derivative approximation is used to calculate the gradient, then it requires executing as many integrations of the model equation as the number of the variables. Hence, huge CPU time is consumed.

We derive the tangent linear model that propagates the perturbation from t_0 to t from the forward model, and then the corresponding adjoint model [3]. The gradient is computed from the adjoint model.

6 Numerical Experiment

The Burgers equation is numerically integrated between the $t_0 = 0$ and $t_1 = 1000\Delta t$. Figure 1 - Figure 6 show the evolution of the model variable field.

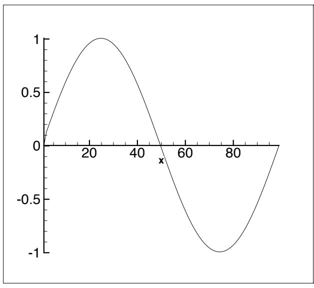

Fig. 1. Time = 0

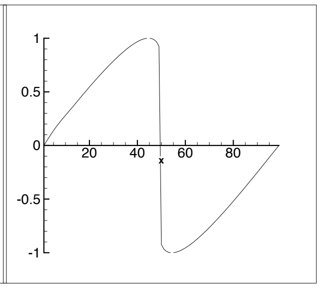

Fig. 2. Time = 200 Δt

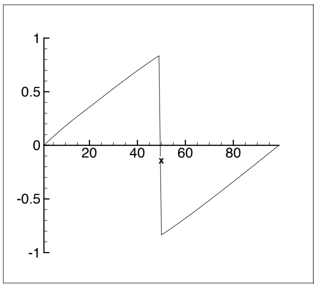

Fig. 3. Time = 400 Δt

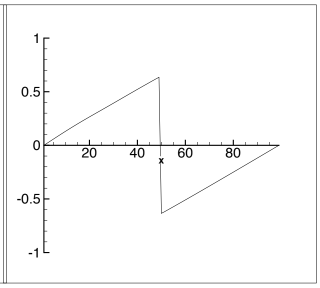

Fig. 4. Time = 600 Δt

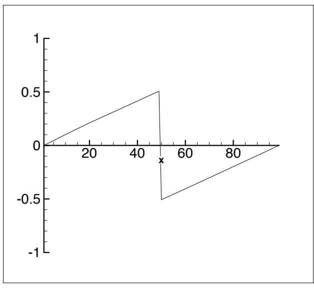

Fig. 5. Time = 800 Δt

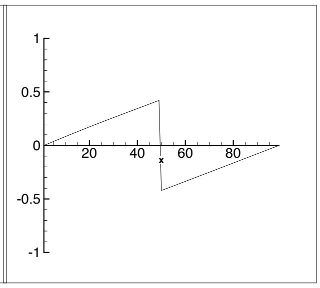

Fig. 6. Time = 1000 Δt

This solution is assumed to be observations. Using the observations between $t_2 = 201\Delta t$ and $t_1 = 1000\Delta t$, we estimate the state of the field at $t_0 = 0$ used for the numerical integration. L-BFGS-B solves the large-scale nonlinear optimization problem starting an initial guess for the state. Figure 7 shows the initial guess.

Figure 8 shows the convergence history. The objective function decreases by a factor of about 10^6 after 500 iterations.

Figures 9 compares the estimated state and its truth.

7 Conclusions

The estimated state agrees reasonably with the initial condition used in the numerical integration.

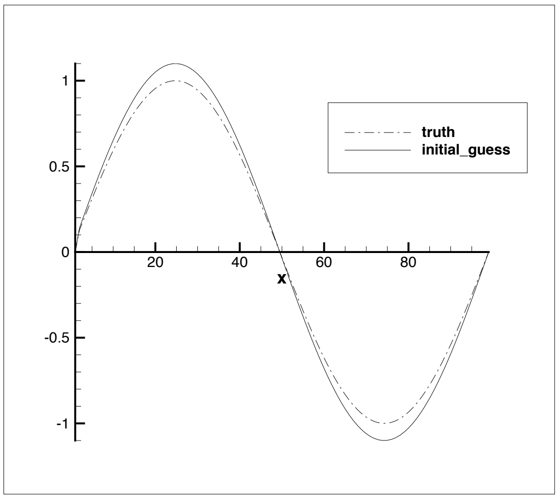

Fig. 7. Initial guess

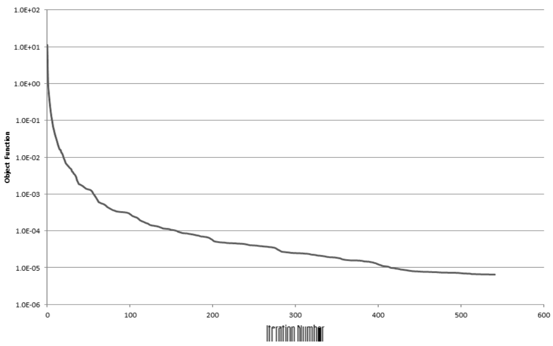

Fig. 8. Convergence history of the optimization

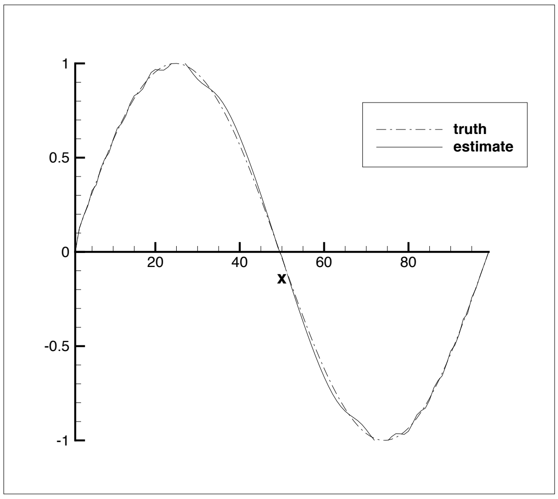

Fig. 9. Estimated state and its truth

References

[1] O. Talagrand, Assimilation of Observations, An Introduction, *J. Met. Soc. Japan*, Special Issue **75**, 1B, pp. 191-209 (1997)

[2] C. Zhu, R. H. Byrd, P. Lu, and J. Nocedal, Algorithm 778: L-BFGS-B, FORTRAN subroutines for large scale bound constrained optimization, *ACM Transactions on Mathematical Software*, **23**, pp. 550-560 (1997)

[3] E. Kalnay, Atmospheric Modeling, Dats Assimilation and Predictability, *Cambridge University Press*, Cambridge, (2003)