11th World Congress on Computational Mechanics (WCCM XI)
5th European Conference on Computational Mechanics (ECCM V)
6th European Conference on Computational Fluid Dynamics (ECFD VI)
E. Oñate, J. Oliver and A. Huerta (Eds)

ON THE FORWARD IN TIME SOLUTION OF THE UNSTEADY ADJOINT EQUATIONS

DIMITRIOS I. PAPADIMITRIOU

Department of Mechanical Engineering, University of Thessaly
Pedion Areos, Volos 38334, Greece.
e-mail: dpapadim@uth.gr, web page: http://www.mie.uth.gr/labs/sdl/

Key words: Adjoint Methods, Optimization, Unsteady Flows

Abstract. A new algorithm for the solution of the unsteady adjoint equations is proposed in this paper aiming at overcoming the excessive computational cost and memory requirements of the conventional adjoint approach for the optimization of unsteady problems in computational mechanics. The proposed algorithm consists of two phases: the first one is based on the backward in time solution of the adjoint equations based on average in time state quantities and the second one solves the exact unsteady adjoint equations forward in time, based on the initial adjoint condition derived from the averaging technique of the first phase. The proposed algorithm is compared to other approaches in the case of the 1D unsteady Burgers equation with non-smooth source terms which resembles the modelling of unsteady turbulent flows with very fine temporal discretization modeled using LES, DES or DNS methods. The adjoint variables are computed with acceptable accuracy at a cost of four times solving the unsteady state equations with negligible additional memory requirements.

1 INTRODUCTION

The adjoint approach, either discrete [1] or continuous [2], is the most efficient means to compute the sensitivity derivatives of objective functions with respect to the control variables in optimization problems, since its cost is almost equal to that of solving the state equations, independent of the number of design variables.

The basic advantage of the adjoint approach is retained in the case of unsteady problems, since its cost is still independent of the number of control parameters. However, since the solution of the unsteady state equations marches forward in time (an initial condition is given), the solution of the derived unsteady adjoint equations should march backward in time [3]. This is obeyed from the mathematical derivation, either in the discrete or the continuous form, since a final (rather than initial condition for the adjoint equations) is derived. The actual problem arises from the fact that in this backward

propagation, the solution of the state equations should be available at each time step. Thus, the state solution should either be stored for the whole time history or recomputed from scratch. The first approach increases the memory requirements while the second one increases computational cost.

The most well known approach to circumvent the unsteady adjoint problem is the checkpointing method [4, 5]. The unsteady adjoint equations are still solved backward in time but the flow solution is stored at some of the intermediate time steps and the CPU cost and memory requirement are balanced. Although the optimization of the checkpointing technique has been achieved, the optimal CPU cost and required memory are still high and even unaffordable in the case that a lot of time steps (fine temporal discretization) is required. This is, for instance, the case of the optimization of unsteady turbulent flows especially those that require Large Eddy Simulation, Detached Eddy Simulation or Direct Numerical Simulation models.

The adjoint approach to the optimization of large scale unsteady turbulent flows presented in [6, 7] required very large storage and was still not using LES, DES or DNS. The adjoint approach to an LES model has been presented [9, 10], based on backward in time solution, and, thus, confined to simple flow problems, such as the identification of the wall model parameter in a quite simplified flow. On the other hand, the adjoint approach to the DNS model has only been applied to small scale problems [8], based on the standard backward in time propagation.

Only a few algorithmic approaches have been developed attempting to solve the unsteady adjoint equations forward in time; the local in time approach [11], where the total temporal domain is divided to subdomains and the total procedure is forward in time but at each subdomain the adjoint equations are solved backwards in time. and the Monte Carlo method [12], based on the structure of the total Jacobian matrix and using probabilistic theory.

This paper aims at overcoming the requirement of solving the unsteady adjoint equations backward in time, retaining acceptable accuracy in the adjoint field solution and computed sensitivities. The proposed approach incorporates a little bit higher computational cost, being, expected, however, to produce quite accurate adjoint fields and sensitivities. It is based on the forward solution of the exact unsteady adjoint equations; the required initial adjoint condition is provided by solving the unsteady adjoint equations backwards in time, using the average in time field of the state variables. Thus, this approach has two phases. At the first phase, the backward solution, the computational cost is almost equal to that of solving the state equations and requires the additional storage of the average in time state variable field only. An adjoint variable field corresponding to the initial time is computed and stored and this feeds the second phase, the forward solution, whose cost is almost equal to twice that of the unsteady state equations, This is due to the fact that the state variable field at each time step is not stored at the first phase and is recomputed at the second phase at each time step, together with the solution of the unsteady adjoint equations. Thus the total computational cost is almost equal to four

times the solution of the unsteady state equations which is considered quite low compared to the steady state problem, especially keeping in mind that there is negligible additional memory requirements. The proposed forward in time unsteady adjoint approach is applied in the case of 1D Burgers equation, including random source terms, in an attempt to mimic the flows obtained by models such as DES and LES.

2 THE UNSTEADY ADJOINT EQUATIONS

2.1 The adjoint approach

In control theory, an optimization problem is defined as a problem in which the target is to find the optimal values of the so-called control variables $b_i,\ i=1,...,I$ that minimize an objective function F. F depends on b_i directly and indirectly through the state variables $U_m,\ m=1,...,M\ (F = F(U_k(b_i), b_i))$, that satisfy the state equations $R_m(U_k(b_i), b_i) = 0,\ m=1,...,M$. The sensitivity derivatives of F with respect to b_i are computed using the chain rule

$$\frac{dF}{db_i} = \frac{\partial F}{\partial b_i} + \frac{\partial F}{\partial U_k}\frac{dU_k}{db_i} \tag{1}$$

where the sensitivity fields $\frac{dU_k}{db_i}$ are computed by solving the equations

$$\frac{dR_m}{db_i} = \frac{\partial R_m}{\partial b_i} + \frac{\partial R_m}{\partial U_k}\frac{dU_k}{db_i} = 0 \tag{2}$$

Eqs. 1 and 2 constitute the so-called direct differentiation approach and its cost scales with the number of control variables. Alternatively, one may refer to the adjoint approach [2, 1], This is based on the solution of the adjoint equations

$$\frac{\partial F}{\partial U_k} + \Psi_m \frac{\partial R_m}{\partial U_k} = 0 \tag{3}$$

for Ψ_m and the computation of the objective function sensitivities from the expression

$$\frac{dF}{db_i} = \frac{\partial F}{\partial b_i} + \Psi_m \frac{\partial R_m}{\partial b_i} \tag{4}$$

The cost of the adjoint approach is independent of the number of design variables, being almost equal to that of solving the state equations.

2.2 The conventional unsteady adjoint approach

The advantage of the adjoint approach over direct differentiation (and finite differences) that the objective function sensitivity derivatives are computed at a cost independent of the number of control variables exists either for steady or unsteady problems. However, in the case of unsteady problems the adjoint formulation introduces extra implications as discussed below.

The unsteady state equations may be expressed using, without loss in generality, a first-order forward temporal discretization, as follows

$$\begin{aligned} R_m^1 &= U_m^1 - \bar{U}_m = 0 \\ R_m^n &= U_m^n - U_m^{n-1} + P_m^{n-1}\Delta t = 0, \quad n = 2, ..., N \end{aligned} \tag{5}$$

where P_m^{n-1} is the spatial residual at time $n-1$ and N is the number of total time steps. The state variables at time $n = 1$ are given as an initial condition and the state variables at all time steps are computed explicitly marching forward in time.

Assuming a time-dependent objective function, given by $\hat{F} = \sum_{n=1}^{N} F^n$, its derivatives with respect to the control variables, using the direct-differentiation approach, are given by

$$\frac{d\hat{F}}{db_i} = \sum_{n=1}^{N} \left(\frac{\partial F^n}{\partial b_i} + \frac{\partial F^n}{\partial U_k^n} \frac{dU_k^n}{db_i} \right) \tag{6}$$

where the sensitivities $\frac{dU_k^n}{db_i}$ are computed by solving the equations derived by differentiating eq. 5, as follows

$$\begin{aligned} &\frac{dU_m^1}{db_i} - \frac{d\bar{U}_m}{db_i} = 0 \\ &\frac{dU_m^n}{db_i} - \frac{dU_m^{n-1}}{db_i} + \left(\frac{\partial P_m^{n-1}}{\partial U_k^{n-1}} \frac{dU_k^{n-1}}{db_i} + \frac{\partial P_m^{n-1}}{\partial b_i} \right) \Delta t, \quad n = 2, ..., N \end{aligned} \tag{7}$$

Although the direct differentiation equations are solved marching forward in time, requiring, thus, no additional memory, their cost scales with the number of design variables.

Alternatively, one may solve the unsteady adjoint equations, given by

$$\Psi_k^n - \Psi_k^{n+1} + \Psi_m^{n+1} \frac{\partial P_m^n}{\partial U_k^n} \Delta t + \frac{\partial F^n}{\partial U_k} = 0, \quad 1 < n < N-1 \tag{8}$$

$$\Psi_m^N + \frac{\partial F^N}{\partial U_k} = 0, \quad n = N \tag{9}$$

and the sensitivity derivatives of the objective function are computed by the expression

$$\frac{\delta F_{aug}}{\delta b_i} = \sum_{n=1}^{N} \frac{\partial F^n}{\partial b_i} - \Psi_m^1 \frac{d\bar{U}_m}{db_i} + \sum_{n=1}^{N-1} \Psi_m^{n+1} \frac{\partial P_m^n}{\partial b_i} \Delta t \tag{10}$$

The major problem that arises in the unsteady adjoint approach, that does not exist in the steady adjoint, is clearly deduced by observing eqs. 8 and 9. Unlike the state equations which can be solved forward in time, starting from the initial state condition, the adjoint equations should be solved backward in time. Eq. 9 should be first solved for Ψ_m^N and then eq. 8 is solved marching backwards in time from $n = N-1$ to $n = 1$.

During this backward in time solution of the unsteady adjoint equations, the state solution at each time step should be available. Thus, since the state equations have been solved marching forward in time, the state variables at all time steps should have been stored. This fact increases the memory requirements of the unsteady adjoint approach too much, in comparison to the memory required if only the state equations are to be solved and average quantities (objective functions) are to be computed.

2.3 The unsteady adjoint in matrix form

In order to establish an algorithm for the solution of the unsteady adjoint equations forward in time, a different view of the unsteady adjoint equations should be used. The derivation of eqs. 5 with respect to the control parameters yields

$$\begin{aligned}
&\frac{dR_k^1}{db_i} = \frac{dU_k^1}{db_i} - \frac{d\bar{U}_k}{db_i} = 0 \\
&\frac{dR_k^n}{db_i} = \frac{\partial U_k^n}{\partial b_i} - \frac{\partial U_k^{n-1}}{\partial b_i} + \frac{\partial P_k^{n-1}}{\partial U_m^{n-1}} \frac{dU_m^{n-1}}{db_i} \Delta t + \frac{\partial P_k^{n-1}}{\partial b_i} \Delta t = 0, \quad n = 2, ..., N
\end{aligned} \tag{11}$$

or in matrix form (given for 5 time steps for demonstration purposes) as

$$\begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
\frac{\partial P_k^1}{\partial U_m^1}\Delta t - \delta_{km} & \delta_{km} & 0 & 0 & 0 \\
0 & \frac{\partial P_k^2}{\partial U_m^2}\Delta t - \delta_{km} & \delta_{km} & 0 & 0 \\
0 & 0 & \frac{\partial P_k^3}{\partial U_m^3}\Delta t - \delta_{km} & \delta_{km} & 0 \\
0 & 0 & 0 & \frac{\partial P_k^4}{\partial U_m^4}\Delta t - \delta_{km} & \delta_{km}
\end{bmatrix}
\begin{bmatrix}
\frac{\partial U_k^1}{\partial b_i} \\ \frac{\partial U_k^2}{\partial b_i} \\ \frac{\partial U_k^3}{\partial b_i} \\ \frac{\partial U_k^4}{\partial b_i} \\ \frac{\partial U_k^5}{\partial b_i}
\end{bmatrix} =
\begin{bmatrix}
\frac{\partial \bar{U}_k}{\partial b_i} \\ -\frac{\partial P_k^1}{\partial b_i}\Delta t \\ -\frac{\partial P_k^2}{\partial b_i}\Delta t \\ -\frac{\partial P_k^3}{\partial b_i}\Delta t \\ -\frac{\partial P_k^4}{\partial b_i}\Delta t
\end{bmatrix} \tag{12}$$

where δ_{km} is the Kronecker symbol and $\frac{\partial P_k^l}{\partial U_m^l}$ depends on the spatial discretization. The unsteady direct differentiation equations are thus given by the form $A\tilde{U} = f$

$$\begin{bmatrix}
A_{11} & 0 & 0 & 0 & 0 \\
A_{21} & A_{22} & 0 & 0 & 0 \\
0 & A_{32} & A_{33} & 0 & 0 \\
0 & 0 & A_{43} & A_{44} & 0 \\
0 & 0 & 0 & A_{54} & A_{55}
\end{bmatrix}
\begin{bmatrix}
\tilde{U}_1 \\ \tilde{U}_2 \\ \tilde{U}_3 \\ \tilde{U}_4 \\ \tilde{U}_5
\end{bmatrix} =
\begin{bmatrix}
f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5
\end{bmatrix} \tag{13}$$

offering the possibility to be solved forward in time; the system $A_{11}\tilde{U}_1 = f_1$ is firstly solved for $\tilde{U}_1$, then the system $A_{21}\tilde{U}_1 + A_{22}\tilde{U}_2 = f_2$ is solved for $\tilde{U}_2$ and so forth. Thus, the direct differentiation equations can be solved simultaneously in time with the state equations and, thus, they do not require the storage of all state variables on which the matrices A_{ij} and vectors f_i depend.

On the other hand, the adjoint equations are given by the expression $A^T\Psi = g$:

$$\begin{bmatrix} A_{11}^T & A_{21}^T & 0 & 0 & 0 \\ 0 & A_{22}^T & A_{32}^T & 0 & 0 \\ 0 & 0 & A_{33}^T & A_{43}^T & 0 \\ 0 & 0 & 0 & A_{44}^T & A_{54}^T \\ 0 & 0 & 0 & 0 & A_{55}^T \end{bmatrix} \begin{bmatrix} \Psi_1 \\ \Psi_2 \\ \Psi_3 \\ \Psi_4 \\ \Psi_5 \end{bmatrix} = \begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \\ g_5 \end{bmatrix} \tag{14}$$

Eqs. 14 cannot be solved forward in time. One should start by solving the system $A_{55}^T\Psi_5 = g_5$ for Ψ_5, then solve the system $A_{44}^T\Psi_4 = A_{54}^T\Psi_5 = g_4$ for Ψ_4 and so forth. Thus, since the state equations are solved forward in time, one should either store all state variables, recompute them at each backward step or, optimally, choose and store some intermediate state variables at some checkpoints to compensate between computational cost and memory requirement.

3 FORWARD IN TIME ADJOINT

The proposed algorithm for the solution of the unsteady adjoint equations consists of two phases. During the first phase, the adjoint equations are solved backwards in time. However, instead of solving the exact adjoint equations (eq. 14), the approximate equations $\bar{A}^T\Psi = \bar{g}$ are solved, written as follows

$$\begin{bmatrix} \bar{A}_{11}^T & \bar{A}_{21}^T & 0 & 0 & 0 \\ 0 & \bar{A}_{22}^T & \bar{A}_{32}^T & 0 & 0 \\ 0 & 0 & \bar{A}_{33}^T & \bar{A}_{43}^T & 0 \\ 0 & 0 & 0 & \bar{A}_{44}^T & \bar{A}_{54}^T \\ 0 & 0 & 0 & 0 & \bar{A}_{55}^T \end{bmatrix} \begin{bmatrix} \Psi_1^* \\ \Psi_2^* \\ \Psi_3^* \\ \Psi_4^* \\ \Psi_5^* \end{bmatrix} = \begin{bmatrix} \bar{g}_1 \\ \bar{g}_2 \\ \bar{g}_3 \\ \bar{g}_4 \\ \bar{g}_5 \end{bmatrix} \tag{15}$$

where the matrices $\bar{A}_{ij}^T$ and the vectors $\bar{g}_i$ are the quantities A_{ij}^T and g_i computed using the average in time values of the state variables. By using the average quantities instead of the exact in time ones, the storage of the state solutions at each and every time step is avoided; only an on-line averaging of the state solutions in time take places and, thus, the storage of an average state field is only required, whose memory requirement is almost that of an equivalent steady state solution. The outcome of this first phase of the proposed approach, which is based on that simple averaging of the state variables, is the computation of the approximate adjoint solution Ψ_1^* at the first time step.

Now that an initial condition for adjoint variables is available, a forward in time sweeping and solution of the unsteady adjoint equations is viable. The exact unsteady adjoint equations are now solved forward in time using the approximate initial solution Ψ_1^*

$$\begin{bmatrix} A_{11}^T & A_{21}^T & 0 & 0 & 0 \\ 0 & A_{22}^T & A_{32}^T & 0 & 0 \\ 0 & 0 & A_{33}^T & A_{43}^T & 0 \\ 0 & 0 & 0 & A_{44}^T & A_{54}^T \\ 0 & 0 & 0 & 0 & A_{55}^T \end{bmatrix} \begin{bmatrix} \Psi_1^* \\ \Psi_2^{**} \\ \Psi_3^{**} \\ \Psi_4^{**} \\ \Psi_5^{**} \end{bmatrix} = \begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \\ g_5 \end{bmatrix} \tag{16}$$

for Ψ_2^{**}, Ψ_3^{**}, Ψ_4^{**} and Ψ_5^{**}, solving the first equation of the system 16 for Ψ_2^{**}, the second for Ψ_3^{**} and so forth.

During this forward sweeping, the exact values of the state variables are used to compute A_{ij}^T and g_i and not the averaged ones used in the first phase. Attention should be payed here, because these variables were not stored during the first forward solution of the state equations, in contrast to the conventional approach where the state variables are stored at each time step (or at some time steps in the case of a checkpointing approach). Thus, in this proposed algorithm, the unsteady state equations are solved again, simultaneously to the forward in time solution of eq. 16, providing in real time the required quantities A_{ij}^T and g_i.

The total computational cost is almost that of 4 equivalent unsteady flow solutions, i.e. four times as much as the cost for solving the unsteady state equations, which corresponds to a twice as much as the cost of the solution of the exact steady state and adjoint equations, eliminating completely (in contrast to checkpointing techniques) the requirement for additional storage.

4 APPLICATION TO THE 1D BURGERS EQUATION

In this section, the proposed algorithm is applied to the one-dimensional Burgers equation

$$R(u) = \frac{\partial u}{\partial t} + \frac{1}{2}\frac{\partial (u^2)}{\partial x} = s \tag{17}$$

with the initial condition

$$u = \bar{u} sin(\pi x) \tag{18}$$

The forward Euler temporal discretization and first-order upwind spatial discretization of eqs. 17 and 18 gives

$$\begin{aligned} R_k^1 &= U_k^1 - \bar{U} sin(\pi x_k) = 0, \quad k = 1, ..., K \\ R_k^n &= \frac{U_k^n - U_k^{n-1}}{\Delta t} + \frac{(U_k^{n-1})^2 - (U_{k-1}^{n-1})^2}{2\Delta x} = S_k^{n-1}, \quad n = 2, ..., N, \quad k = 1, ..., K \end{aligned} \tag{19}$$

taking the form of eq. 5 for $P_k^{n-1} = \frac{(U_k^{n-1})^2 - (U_{k-1}^{n-1})^2}{2\Delta x} - S_k^{n-1}$

Two cases are investigated: the first one assumes that there is not a source term in the Burgers equations, i.e. $s = 0$ and the second one uses a source term given by

$$s = \bar{s} sin(\pi x) rand() \tag{20}$$

where $rand()$ is a random function chosen to take values between -0.4 and 0.6.

For the first case, the solution of the unsteady state equations is illustrated in fig. 1 where the distribution of the state variables over space and time is plotted. A linear distribution of the state variables over time may be observed.

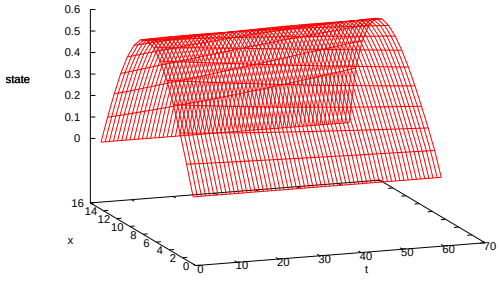

Figure 1: Case 1: Distribution over space and time of the state variable.

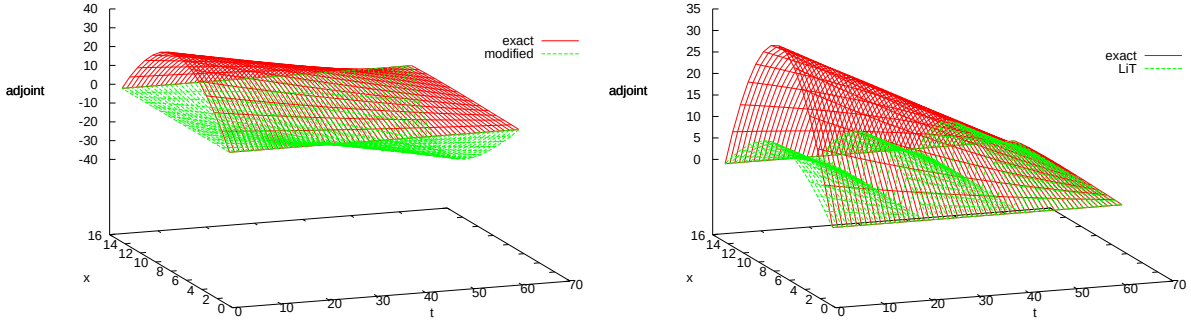

Figure 2: Case 1: Distribution over space and time of the adjoint variable computed using the solution of the unsteady adjoint equations based on unconditionally modified adjoint equations (left) and local in time solution (right) of the unsteady adjoint equations.

The solution of the unsteady adjoint equations for this case, using various different methods exposed in the previous sections is shown in the following figures and compared to the exact adjoint solution that is obtained by solving backwards in time the unsteady adjoint equations.

In fig. 2, left, the adjoint solution based on an unconditionally modified adjoint approach (elimination of the first column of the matrix of eq. 14) is compared to the exact adjoint approach. The comparison shows that even in this simple unsteady example, in which the state solution is linear with time, this approach fails to provide acceptable adjoint variable distributions. This is also the case for the second examined adjoint approach that is based on the local in time solution of the unsteady adjoint equations, fig. 2, right, using in this case three steps and zeroing the adjoint variables at each sub-interval.

The application of the proposed algorithm is illustrated in fig. 3. In the left, only the

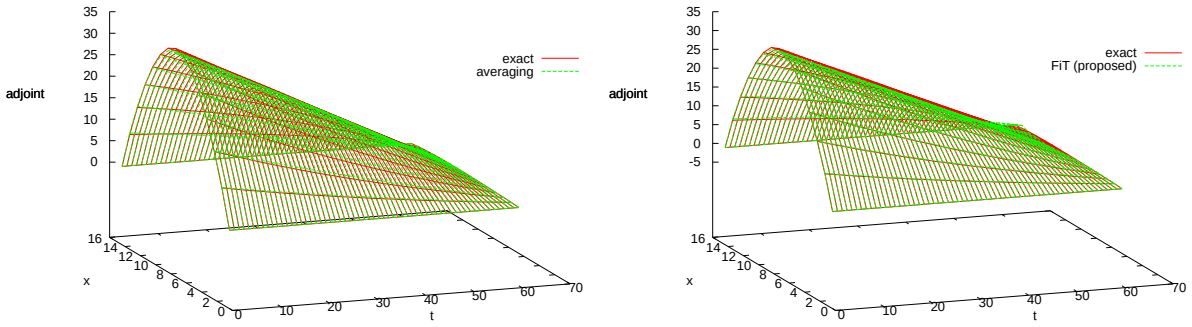

Figure 3: Case 1: Distribution over space and time of the adjoint variable computed using the solution of the unsteady adjoint equations based on averaging of the flow solution (left) and and the forward in time solution of the unsteady adjoint equations (right).

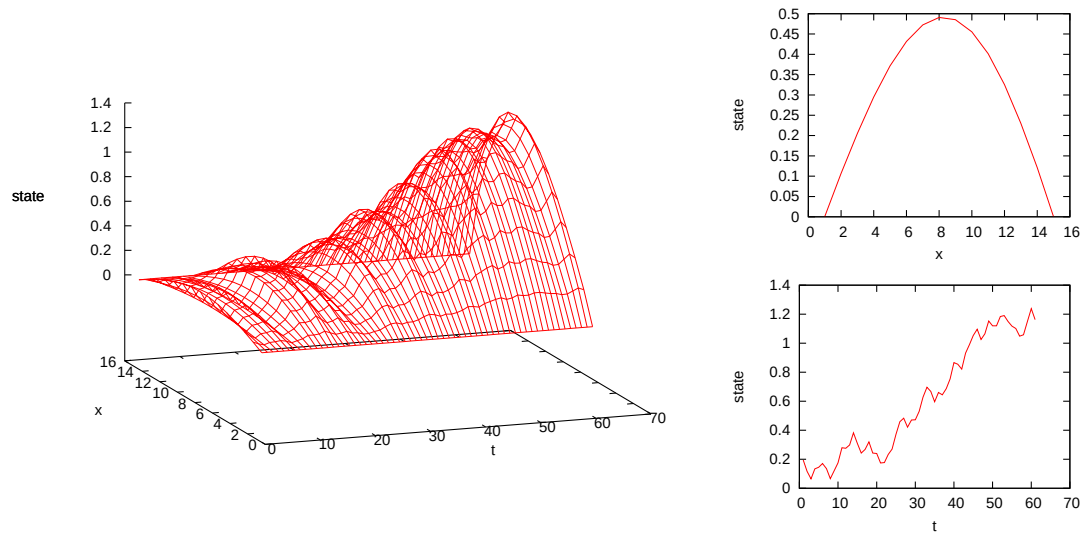

Figure 4: Case 2: Distribution over space and time of the state variable (left), distribution over space of the state variable for $n = T/2$ (top-right) and distribution over time of the state variable for $k = K/2$ (bottom-right).

first part of the proposed algorithm is applied, which is based on the backward solution of the unsteady adjoint equations based on averaged values of the state variables. It seems that in this linear in time case, the averaging technique is enough to compute satisfactory adjoint variable distribution and the gain from the second step of this algorithm in the accuracy of the adjoint variable values is small.

The results are different in the second examined case where a source term is added to the Burgers equation. A random in time source term is added in order to resemble

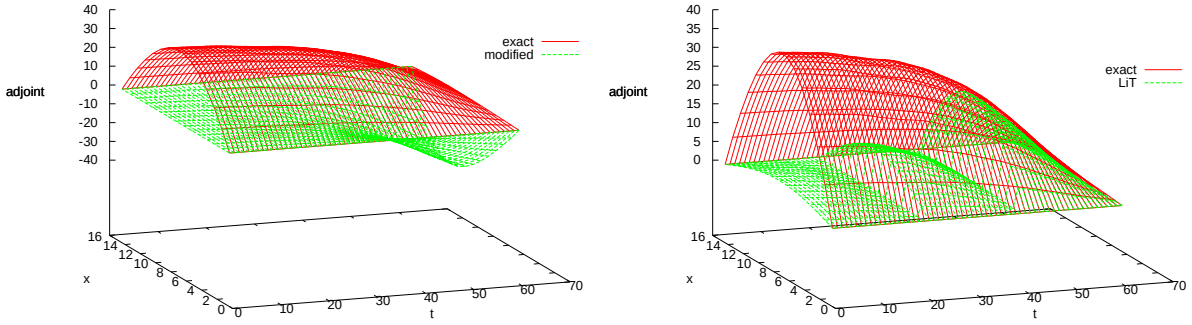

Figure 5: Case 2: Distribution over space and time of the adjoint variable computed using the solution of the unsteady adjoint equations based on unconditionally modified adjoint equations (left) and local in time solution of the unsteady adjoint equations (right).

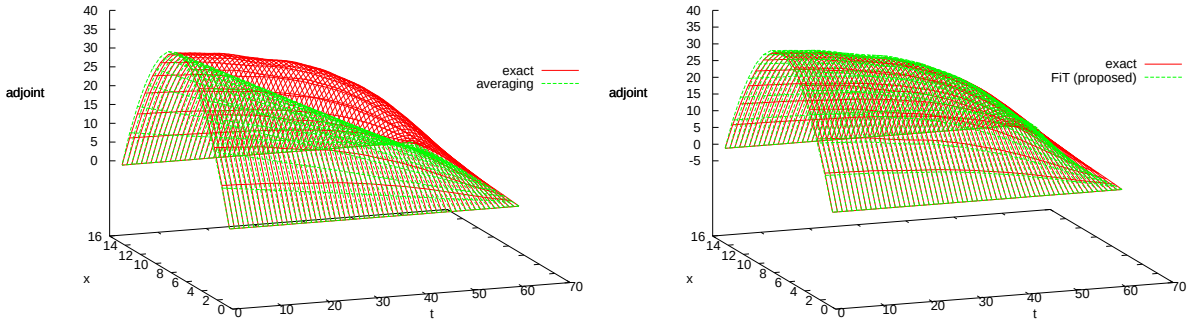

Figure 6: Case 2: Distribution over space and time of the adjoint variable computed using the solution of the unsteady adjoint equations based on averaging of the flow solution (left) and forward in time solution of the unsteady adjoint equations (right).

(as much as possible for a 1D example) the URANS or LES/DES models for unsteady turbulent flow analysis. This time, it can be seen from fig. 4 that the state solution is not linear, not only in space but in time as well.

The first conclusion drawn from the comparison of the different algorithms for the solution of the adjoint approach approves that the first two approaches, namely the unconditionally modified adjoint approach and the local in time adjoint approach fail to compute accurate adjoint variable distributions, fig. 5. This statement is in total agreement with the previous simpler case.

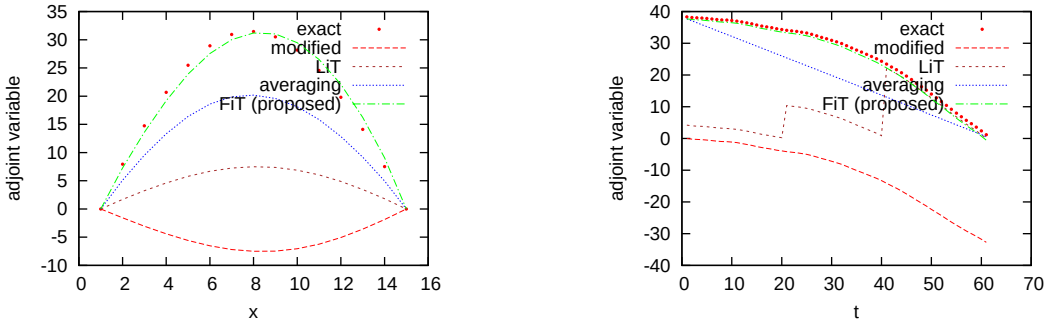

Figure 7: Case 2: Distribution over space of the adjoint variable for $n = T/2$ (left) and distribution over time of the adjoint variable for $k = K/2$ (right), computed using the aforementioned methods.

However, the observation of fig. 6 leads to a different conclusion. The application of only the first step of the proposed adjoint approach is not adequate to provide accurate enough adjoint variable solution, a lot different than the exact backward in time adjoint solution, fig. 6, left. Moreover, the application of the second step of the proposed algorithm where the adjoint variables are recomputed forward in time based on the initial solution provided by the first step, leads to quite accurate prediction of the adjoint field, fig. 6, right, (and fig. 7 for better view of the comparison).

This outcome proves that in such cases the full proposed algorithm should be applied and "pay" the additional computational cost (twice the cost of the conventional approach), to avoid completely the huge memory requirements of the conventional approach, with acceptable accuracy of the computed adjoint fields.

5 CONCLUSIONS

A new iterative algorithm was proposed for the solution of the unsteady adjoint equations forward in time. The algorithm consists of two phases; a backwards sweeping phase based on averaged flow quantities and a forward sweeping phase, based on the solution found from the first phase. The algorithm proved to compute very accurate adjoint fields in the case of the Burgers equation, comparing to other approaches, especially in the case of a random source term that mimics the turbulent flows modeled by high fidelity (DES, LES) CFD models, yielding promises for the optimization based on those large-scale models.

ACKNOWLEDGMENTS

The research project is implemented within the framework of the Action "Supporting Postdoctoral Researchers" of the Operational Program "Education and Lifelong Learning" (Actions Beneficiary: General Secretariat for Research and Technology), and is co-financed by the European Social Fund (ESF) and the Greek State.

REFERENCES

[1] Burgreen, G.W. and Baysal, O. Three-dimensional aerodynamic shape optimization using discrete sensitivity analysis. *AIAA Journal* (1996) **34**(9):1761-1770.

[2] Jameson A. Aerodynamic design via control theory. *Journal of Scientific Computing* (1988) **3**:233–260.

[3] Nadarajah, S.K. and Jameson, A. Optimum shape design for unsteady flows with time-accurate continuous and discrete adjoint method. *AIAA Journal* (2007) **45**(7).

[4] Griewank, A. and Walther, A. Algorithm 799: revolve: an implementation of checkpointing for the reverse or adjoint mode of computational differentiation. *ACM Transactions on Mathematical Software (TOMS)* (2000) **27**:19–45.

[5] Wang, Q., Moin, P. and Iaccarino, G. Minimal repetition dynamic checkpointing algorithm for unsteady adjoint calculation. *SIAM Journal on Scientific Computing* (2009) **31**:2549–2567.

[6] Nielsen, E., Diskin, B. and Yamaleev, N. Discrete adjoint-based design optimization of unsteady turbulent flows on dynamic unstructured grids. *AIAA Journal* (2010) **48**(6):1195–1206.

[7] Nielsen, E.J. and Diskin, B. Discrete adjoint-based design for unsteady turbulent flows on dynamic overset unstructured grids. *AIAA Journal* (2013) **51**(6):1355–1373.

[8] Bewley T., P. Moin, R. Temam, DNS-based predictive control of turbulence: an optimal benchmark for feedback algorithms. *Journal of Fluid Mechanics* (2001), **447**:179–225.

[9] Nicoud, F., Baggett, J.S., Moin, P. and Cabot, W. Large eddy simulation wall-modeling based on suboptimal control theory and linear stochastic estimation. *Physics of Fluids* (2001) **13**(10):2968–2984.

[10] Templeton, J.A., Wang, M. and Moin P. Towards LES wall models using optimization techniques. Center for Turbulence Research *Annual Research Briefs* (2002):189-200.

[11] Yamaleev, N., Diskin, B., and Nielsen, E., Local-in-time adjoint-based method for design optimization of unsteady flows. *Journal of Computational Physics*, (2010) **229**(14):5394–5407.

[12] Wang, Q., Gleich, D., Saberi, A., Etemadi, N. and Moin, P. A Monte Carlo method for solving unsteady adjoint equations. *Journal of Computational Physics* (2008) **227**(12):6184–6205.