AN IMPROVEMENT OF THE DUAL TIME STEPPING TECHNIQUE FOR UNSTEADY RANS COMPUTATIONS

C. Marongiu, P. L. Vitagliano, P. Catalano
Italian Center for Aerospace Research, Capua (CE), Italy

V. Tarantino, D. di Serafino
Department of Mathematics, Second University of Naples, Italy

Introduction

The aim of this work is the analysis of efficient strategies for the time integration of the unsteady Navier-Stokes equations. Among several possibilities, Jameson's Dual Time Stepping (DTS) method [5] is widely used. This method performs time integration by achieving successive steady states in a fictious dual time. As pointed out in [4], a drawback of DTS is that steady-state iterations must be fully converged to guarantee a certain time accuracy, thus requiring a large number of dual-time iterations. This can substantially increase the computational cost of the integration procedure. Unsteady simulations are usually 4-5 times more expensive than steady ones, therefore strong efforts are put in the study of efficient time accurate algorithms. In an early version of the CIRA flow solver U-ZEN [7], the DTS method was implemented by using an explicit second-order multi-stage Runge Kutta scheme, accelerated by standard techniques (multigrid, local time stepping, implicit residual averaging). In this work, some strategies to increase the efficiency of the DTS method are analyzed. These strategies are based on two concepts. In the first one the DTS method is considered as a black box, which takes in input a state $\underline{U}^n$ at time level n and returns the state $\underline{U}^{n+1}$ at time level $n+1$, with a computational cost in terms of dual iterations. The idea is to provide the black box with an input vector $\underline{U}^* \neq \underline{U}^n$ that better predicts the unknowns at the beginning of dual time integration. This strategy is analogous to the use of *predictor-corrector* techniques with multi-step methods. The second strategy operates inside the DTS box, by using a low-cost implicit scheme to approach the steady state in the dual time [4]. A hybrid approach is obtained by using the implicit scheme as a predictor for the DTS scheme. The application of these strategies inside U-ZEN is analysed; numerical results on test cases are presented.

The DTS Method

In the DTS method, the flow solution at each time step is obtained by reformulating the problem as a steady one; the steady-state solution is computed by integrating over an unphysical parameter called dual time. The relaxation operator in the dual time can be explicit or implicit. The reason for the success of DTS is that it allows to exploit in unsteady calculations all the convergence acceleration techniques (multigrid, residual averaging, local time stepping, etc.) that are typically used for unsteady calculations. Given the semidiscrete equation

$$L_t \underline{U}^{n+1} = -\underline{R}(\underline{U}^{n+1}) \qquad (1)$$

where L_t is the time derivative operator and

$$\underline{R} = \underline{R}^C + \underline{R}^A + \underline{R}^V \qquad (2)$$

is the sum of the convective, artificial and viscous fluxes, a second order backward difference formula has been applied for time discretization:

$$L_t \underline{U}^{n+1} = \frac{3\underline{U}^{n+1} - 4\underline{U}^n + \underline{U}^{n-1}}{2\Delta t} \ . \qquad (3)$$

At each time level, the DTS method considers a new residual:

$$\underline{\tilde{R}}(\underline{U}) = \underline{R}(\underline{U}) + \frac{3\underline{U} - 4\underline{U}^n + \underline{U}^{n-1}}{2\Delta t} \qquad (4)$$

and solves the following equation:

$$\frac{d\underline{U}}{d\tau} = -\underline{\tilde{R}}(\underline{U}) \, , \qquad (5)$$

where τ is the dual time. The integration of (5) to its steady state provides the solution $\underline{U}^{n+1}$ of (3) at the time level $n+1$. In this way, the nonlinear system (1) can be solved at each time step by using numerical algorithms typically applied to steady problems.

Prediction formulae

The simplest approach to predict the solution at the beginning of each DTS stage is based on extrapolation criteria. In a general way, these criteria can be regarded within the context of explicit formulae:

$$\underline{U}^* = f\left(\underline{U}^n, \underline{U}^{n-1}, \underline{U}^{n-2}, \ldots\right) \qquad (7)$$

The formulae considered here involve at most three time levels:

$$\underline{U}^* = \alpha \underline{U}^n + \beta \underline{U}^{n-1} + \gamma \underline{U}^{n-2} \, , \qquad (8)$$

where the coefficients α, β and γ are given in Table 1:

Table 1

	α	β	γ	
PD1	2	-1	0	O(Δt)
PD2	3	-3	1	$O(\Delta t^2)$
PD3	5/2	-2	1/2	$O(\Delta t^2)$
PD4	7/3	-5/3	2/3	$O(\Delta t^2)$

Note that the formulae of type (7) do not require the evaluation of the fluxes. Nevertheless, if the time behavior of the solution is regular enough, it is reasonable to expect some benefit. Indeed, without any prediction, i.e. setting $\underline{U}^* = \underline{U}^n$, the residual at the beginning of the dual cycle is

$$\underline{\tilde{R}}(\underline{U}^n) = \underline{R}(\underline{U}^n) + \frac{-\underline{U}^n + \underline{U}^{n-1}}{2\Delta t} . \qquad (9)$$

The integration toward the steady state is carried out until the following difference is reduced below a certain threshold:

$$\underline{\tilde{R}}(\underline{U}^{n+1}) - \underline{\tilde{R}}(\underline{U}^n) = \underline{R}(\underline{U}^{n+1}) - \underline{R}(\underline{U}^n) + \\ + \frac{3}{2\Delta t}\left(\underline{U}^{n+1} - \underline{U}^n\right) + \mathrm{O}\left(\Delta t^2\right). \qquad (10)$$

Hence, by adopting a prediction formula, and assuming that

$$\left|\underline{U}^{n+1} - \underline{U}^*\right| \le \left|\underline{U}^{n+1} - \underline{U}^n\right| \qquad (11)$$

the residual difference might be expected to decrease:

$$\left|\underline{\tilde{R}}\left(\underline{U}^{n+1}\right) - \underline{\tilde{R}}\left(\underline{U}^*\right)\right| \le \left|\underline{\tilde{R}}\left(\underline{U}^{n+1}\right) - \underline{\tilde{R}}\left(\underline{U}^n\right)\right| \qquad (12)$$

ADI schemes

Another way of computing the solution at the time level $n+1$ is to use implicit methods. These methods are generally much more stable than explicit ones and allow large time steps. The main drawback is that large linear systems must be solved at each time step, leading to computationally expensive procedures. In order to reduce the cost of linear system solution, the Alternating Direction Implicit (ADI) splitting scheme can be used, which approximates the linear system matrix with the product of matrices that can be efficiently inverted with direct methods. Indicating with k the dual iteration, a first order backward difference is applied to Equation (5):

$$\underline{U}^{k+1} - \underline{U}^{k} = -\Delta\tau \underline{\tilde{R}}^{k+1} \tag{13}$$

and the following linearization is performed to evaluate the residuals:

$$\underline{\tilde{R}}^{k+1} = \underline{\tilde{R}}^{k} + \frac{\partial \underline{\tilde{R}}^{k}}{\partial \underline{U}} \left(\underline{U}^{k+1} - \underline{U}^{k} \right). \tag{14}$$

Substituting (14) into (13), the following linear system is obtained:

$$\Delta \underline{U}^{k+1} = -\Delta\tau \left[\underline{\tilde{R}}^{k} + \frac{\partial \underline{\tilde{R}}^{k}}{\partial \underline{U}} \Delta \underline{U}^{k+1} \right], \tag{15}$$

which can be written as

$$\underline{\underline{A}} \cdot \Delta \underline{U}^{k+1} = -\Delta\tau \underline{\tilde{R}}^{k}, \tag{16}$$

where

$$\underline{\underline{A}} = \left[\underline{\underline{I}} + \Delta\tau \frac{\partial \underline{\tilde{R}}^{k}}{\partial \underline{U}} \right]. \tag{17}$$

In a 3D formulation, the ADI method approximates the matrix $\underline{\underline{A}}$ with a product of three matrices. The solution $\Delta \underline{U}^{k+1}$ is obtained in three steps

$$\begin{aligned} \underline{\underline{A}}_{\xi} \cdot \Delta \underline{U}^{1} &= -\Delta\tau \underline{\tilde{R}}^{k}, \\ \underline{\underline{A}}_{\eta} \cdot \Delta \underline{U}^{2} &= \Delta \underline{U}^{1}, \\ \underline{\underline{A}}_{\zeta} \cdot \Delta \underline{U}^{k+1} &= \Delta \underline{U}^{2}, \end{aligned} \tag{18}$$

where (ξ, η, ζ) are generalized curvilinear coordinates. This approximation introduces an error proportional to the CFL number. Additionally, the treatment of the boundary conditions affects the stability of the scheme. The linearization of the residual performed in (14) is a further source of error. The terms $\frac{\partial \underline{\tilde{R}}^{k}}{\partial \underline{U}}$ represent the Jacobian matrices of the residuals. Their evaluation depends on the spatial discretization scheme used in the solver and can be decomposed into the following contributions:

$$\underline{\tilde{R}}(\underline{U}) = \frac{3\underline{U} - 4\underline{U}^{n} + \underline{U}^{n-1}}{2\Delta t} + \underline{R}^{C} + \underline{R}^{A} + \underline{R}^{V}. \tag{19}$$

Hence, the Jacobian matrices assume the following form:

$$\frac{\partial \underline{\tilde{R}}}{\partial \underline{U}} = \frac{3}{2\Delta t} \underline{\underline{I}} + \frac{\partial}{\partial \underline{U}} \left(\underline{R}^{C} + \underline{R}^{A} + \underline{R}^{V} \right) \tag{20}$$

and each matrix $\underline{\underline{A}}_{\delta}$ $(\delta = \xi, \eta, \zeta)$ has the following expression:

$$\underline{\underline{A}}_{\delta} = \underline{\underline{I}} + \frac{3}{2\Delta t} \underline{\underline{I}} + \frac{\partial}{\partial \underline{U}} \left(\underline{R}^{C}_{\delta} + \underline{R}^{A}_{\delta} + \underline{R}^{V}_{\delta} \right). \tag{21}$$

Note that the unsteady term increases the diagonal dominance, thus improving the stability. The implicit formulation with the backward difference formula (3) is

$$\left(\underline{\underline{I}}+\frac{2\Delta t}{3}\frac{\partial \underline{R}}{\partial \underline{U}}\right)\cdot\Delta\underline{U}^{n+1}=\frac{\Delta\underline{U}^{n}}{3}-\frac{2\Delta t}{3}\underline{R}^{n}. \quad (22)$$

The value $\underline{U}^{n+1}$ obtained from Equation (22) is second order in time and can be used as predictor for the DTS cycle. This is an example of using an implicit prediction formula of the form

$$\underline{U}^{*}=f\left(\underline{U}^{n+1},\underline{U}^{n},\underline{U}^{n-1},\underline{U}^{n-2},\ldots\right). \quad (23)$$

Results

The NACA0012 profile at Mach = 0.6 in conditions of oscillatory pitching motion has been considered. The experimental data are referred to the AGARD-CT1 test case [6]. The reduced frequency is k=0.0808, and the incidence varies with the following law:

$$\alpha(t)=2.89^{\circ}+2.41^{\circ}\sin(\omega t) \quad (24)$$

Here, an inviscid flow model over a structured 8 block grid of 32768 cells has been considered. The airfoil motion has been simulated through transpiration velocities applied as a boundary condition on the body [13]. The benefits deriving from the use of prediction formulae or ADI methods can be discussed in terms of CPU time saving and accuracy. The motion period T has been subdivided with five different time steps from T/25 to T/720. Regarding the accuracy, between the simulations with 360 and 720 steps/period, no significant differences can be seen in the solutions. The simulation with 720 steps/period without any prediction has been taken as the reference solution for all the cases considered with prediction. In Figure 1, the CPU time needed to simulate 4T has been plotted for the different time steps. Without any prediction, the increase of the CPU time obtained by halving the time step is about 30÷40% and, by applying the predictions, the CPU time increases by 6÷7%. The results for the prediction formulae PD3 and PD4 shown in Figure 1 indicate slightly improved performance compared to the other cases. Irrespective of various formulae used, the saving remains at the same order of magnitude. In Figure 2 the C_L-α plot is presented with solutions at 25 steps/period. Without prediction, the discrepancies are evident. The NP T/25 curve does not match the reference curve near the extremities. During the α increasing phase, the C_L is under predicted. By adopting a prediction formula, (PD2 is shown), the improvement in the accuracy is evident, and comparable with the T/720 solution. This demonstrates that the use of these techniques affects the convergence and the accuracy simultaneously. The instantaneous pressure coefficient in the incidence increasing phase, at α=5°, during the shock wave formation is shown in Figure 3. The implicit prediction with the Runge Kutta relaxation is compared with the reference solution. Globally, no significant differences in the computed C_P profiles are noted. In Figure 4, the curve without prediction (NP) at 25 steps/period shows discrepancies compared to that of the reference solution. By employing the ADI prediction the results are comparable to the reference solution at 720 time steps/period.

A 2D U-RANS calculation has been carried out for the flow around the square cylinder at Re=22000, [12], using the k-ε turbulence model [10]. The computational grid has 12 blocks with 61912 cells. At the time of this work, the treatment of the viscous fluxes is still explicit. The turbulence equations are not forced by any prediction method and they are integrated in time in the same way of the Navier Stokes equations. The dimensionless time step is 0.05 which samples the vortex shedding period T in about 140 parts. One level of multigrid has been used during the dual time iterations. A solution obtained without predictions and the number of subiterations fixed to 125 has been previously computed, providing satisfactory results [7]. Nevertheless, a fully converged solution, without limitations on dual time iterations,

and without predictions, has again been computed at a considerable computational cost. The effects on the level of convergence can be seen in Figure 5, where the C_L versus C_D is plotted over the vortex shedding period. The tests show different max C_D values during the periodic cycle. The reference computation has the smallest C_{Dmax}, which also corresponds to the smallest mean C_D. The trend of the computed C_{Dmax} is opposite with respect to the experimental value, as well as the Strouhal number, see Table 2. The solutions obtained by the simulation with prediction formulae reproduce the reference solution with a satisfactory accuracy, permitting a CPU time saving up to 70 %. Anyway, the differences of the averaged fields are negligible, as it can be seen in Figure 6.

Conclusions

The DTS method is a very efficient strategy for generic unsteady problems. The time step size can be chosen on the basis of accuracy considerations, without stability limitations. Nevertheless, the results are sensitive to the level of convergence reached in the dual time cycle. In this paper, explicit and implicit methodologies to improve the convergence have been considered. The combination of prediction formulae, based on explicit or implicit methods, with the relaxation operator in the dual time cycle has demonstrated advantages in both convergence and accuracy.

Future works will concern the completion of the ADI formulation as relaxation operator in dual time as well as predictor for the real time integration.

References

[1] Beam R. M., Warming R. F., *An Implicit Factored Scheme for the Compressible Navier-Stokes Equations*, AIAA Journal, Vol. 16, n.4, 1978, pp393-402

[2] Douglas J. J., Gunn J. E., A General Formulation of Alternating Direction Methods, Numerische Mathematik 6, 428-453, 1964

[3] Fletcher C. A. J., *Computational Techniques for Fluid Dynamics*, Springer-Verlag, 1990

[4] Hsu J. M., Jameson A., *An Implicit–Explicit Hybrid Scheme for Calculating Complex Unsteady Flows*, 40th AIAA Aerospace Sciences Meeting and Exhibit, January 2002, Reno, 2002-0714

[5] Jameson A., *Time-Dependent Calculations Using Multrigrid, with Application to Unsteady Flows past Airfoils and Wings*, AIAA paper 91-1596, 1991

[6] Landon R. H., *NACA 0012. Oscillating and Transient Pitching, Compendium of Unsteady Aerodynamics Measurements*, AGARD R 702, Aug 1982.

[7] Marongiu C., Catalano P., Amato M., Iaccarino G., *U-ZEN: A Computational Tool Solving U-RANS Equations For Industrial Unsteady Applications*, AIAA Paper 2004-2345, 2004

[8] Melson N. D., Sanetrik. M. D., Atkins. H. L., *Time-Accurate Navier-Stokes Calculations with Multigrid Acceleration*, in N.D. Melson, T. A., Manteuffel, S. F., McCormick.(eds.), Proceedings of the Sixth Copper Mountain conference on multigrid methods, 1993

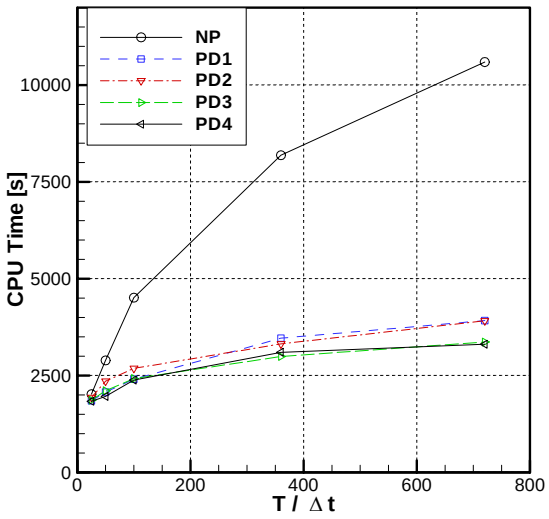
[9] Men'shov I., Nakamura Y., *Hybrid Explicit-Implicit Unconditionally Stable Scheme for Unsteady Compressible Flows*, AIAA Journal, 42(3), 551, March 2004

[10] Myong H., Kasagi N., *A new approach to the improvement of the k-ε turbulence model for wall bounded shear flows,* JSME Intern. J., Ser.2 33(1) 63-72.

[11] Tarantino V., *La strategia del Dual Time Stepping nella Simulazione di flussi non stazionari*, Degree Thesis, 2004

[12] Lyn D. A., Einav S., Rodi W., Park J. H., *A laser-Doppler velocimetry study of ensemble averaged characteristics of the turbulent near wake of a square cylinder*, J. Fluid Mech. 304, 205-232

[13] Ruo S. Y., Sankar L. N., *Euler Calculations for Wing-Alone Configuration*, J. Aircraft, Vol. 25(5), 436-441



(a)

Figure 1. AGARD CT1 – Simulations at various time steps. Explicit prediction with Runge Kutta relaxation. (NP – no prediction, PD – explicit predictions, see Table 1). Simulated time = 4T. CPU time on NEC SX-6

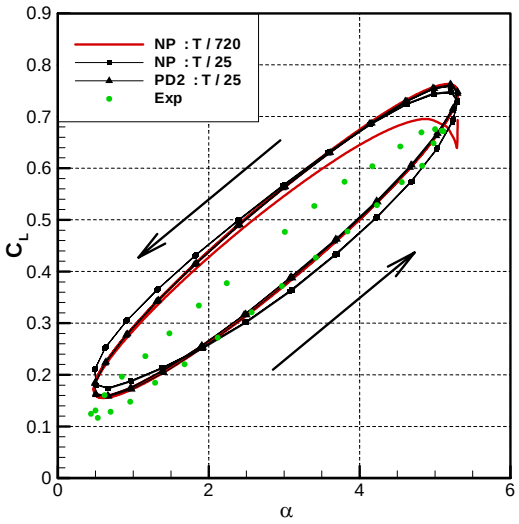


Figure 2. AGARD CT1 – C_L-α Diagram. Runge Kutta relaxation. NP, without prediction. PD2, explicit prediction (see Table 1).

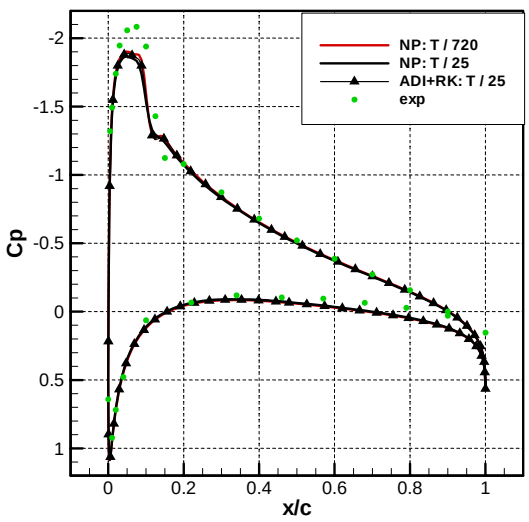


Figure 3. AGARD CT1 – Pressure coefficient. α(t)=5°. Increasing phase. NP, without predictions, ADI+RK implicit prediction with Runge Kutta relaxation

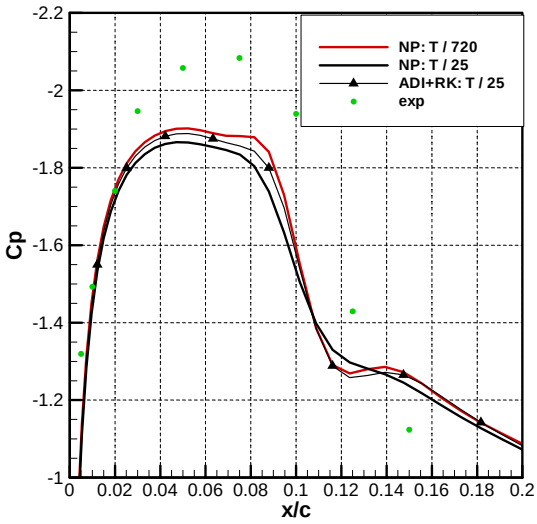


Figure 4. AGARD CT1 – Enlarged view of Figure 3.

Label	CPU time *	dual it*	$\overline{C_D}$	St
Exp	-	-	2.1	0.132
REF [a]	66817	187928	2.080	0.145
PD+RK [b]	25616	69300	2.082	0.145
ADI+RK[c]	13946	38920	2.084	0.144
NC [d]	6613	17500	2.094	0.143

a. without predictions

b. linear prediction and Runge Kutta relaxation

c. ADI prediction and Runge Kutta relaxation

d. no predictions, fixed number of dual iterations (125)

* *for one vortex shedding period*

Table 2

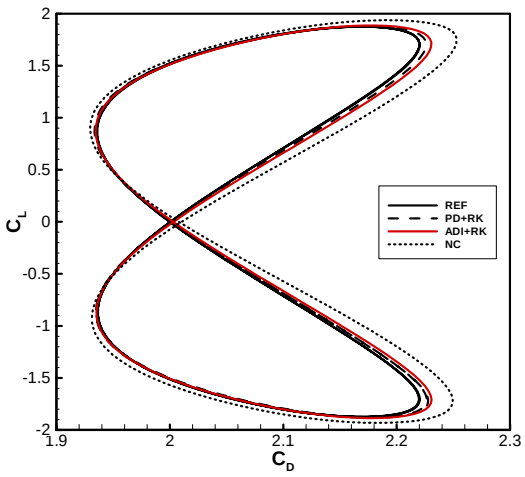


Figure 5. SQUARE CYLINDER – C_L-C_D curve.

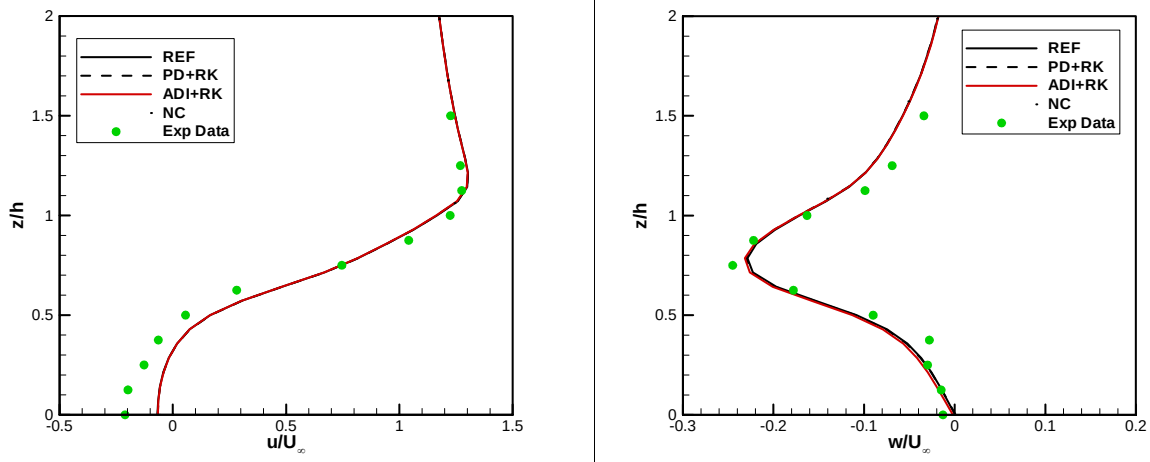


Figure 6. SQUARE CYLINDER – Averaged velocity profiles in the wake, at 0.375h downstream the square. z is normal to the streamwise direction.