

TIME DEPENDENT FLOW SIMULATIONS USING THE LEAST SQUARES SPECTRAL ELEMENT METHOD IN COMBINATION WITH DIRECT MINIMIZATION

Marcel Kwakkel* and Marc Gerritsma†

*Delft University of Technology, Faculty of Aerospace Engineering,
Kluyverweg 1, 2629 HS Delft, The Netherlands
e-mail: M.Kwakkel@tudelft.nl

†Delft University of Technology, Faculty of Aerospace Engineering,
Kluyverweg 1, 2629 HS Delft, The Netherlands
e-mail: M.I.Gerritsma@lr.tudelft.nl

Key words: Least-Squares, Spectral Elements, Time-Stepping

Abstract. *In this work a new approach to time dependent problems in combination with the Least-Squares Spectral Element Method (LSQSEM) will be discussed. Various time-stepping formulations will be presented. These time-stepping formulations will be compared to the full space-time formulation. It will be shown that time-stepping formulations give accurate results for comparable CPU times. Furthermore it will be shown that a smaller timestep or a higher polynomial degree not always decreases the error norm.*

1 INTRODUCTION

For computational fluid dynamics (CFD), the least-squares formulation is an attractive alternative to the standard Galerkin-type formulation. The least-squares formulation solves a system of partial differential equations by minimizing the residual with respect to a suitably chosen norm. For well-posed problems the least-squares formulation always leads to a symmetric positive definite (SPD) discrete system. This system can be solved by well-established iterative solvers, such as preconditioned Conjugate Gradient methods.

To increase the accuracy of the solution, spectral elements can be used. These elements contain higher order basis functions which are derived from so-called orthogonal polynomials. In recent studies Legendre polynomials and Chebyshev polynomials have been used ^{1 2 3 4 5}.

The resulting least-squares spectral element method (LSQSEM) combines the high accuracy of spectral elements, the efficiency of the least-squares formulation with the flexibility of the finite element method. A major concern of LSQSEM is the high condition number, which is the squared of that of the original equations. In recent studies a new technique was implemented in the conventional LSQSEM, which prevents the condition

number to be squared. This method, known as direct minimization (DM), minimizes the residual directly and circumvents a costly matrix-matrix multiplication ^{6 7 8}.

2 LEAST-SQUARES FORMULATION

Consider the following abstract boundary value problem,

$$\mathcal{L}(u) = f \quad \text{in } \Omega, \tag{1}$$

$$\mathcal{R}(u) = g \quad \text{on } \Gamma, \tag{2}$$

where \mathcal{L} is a linear first order partial differential operator and \mathcal{R} a trace operator. Both operators act on a scalar or vector unknown u . The right hand sides f and g are problem related vector-valued functions and one can take $g = 0$ without any loss of generality. If the governing equations involve second or higher order derivatives, the scalar equation or system will first be transformed into an equivalent first order system. The two main reasons to rewrite higher order PDEs into equivalent first order systems are to keep the condition number under control and to mitigate the continuity requirements between neighboring elements. The operator \mathcal{L} is chosen such that it maps elements from the function space X into the function space Y , $\mathcal{L} : X \rightarrow Y$, such that there exist two constants $C_1, C_2 > 0$ for which we have

$$C_1 \|u\|_X \leq \|\mathcal{L}(u)\|_Y \leq C_2 \|u\|_X, \quad \forall u \in X. \tag{3}$$

Both inequalities, coercivity and continuity respectively, assert that $\|\mathcal{L}(u)\|_Y$ defines a norm equivalent to $\|u\|_X$. Convergence of $\|u - u_{ex}\|$, where $u_{ex} \in X$ represents the exact solution of (1)-(2), to zero is therefore equivalent to minimizing $\|\mathcal{L}(u - u_{ex})\|_Y = \|\mathcal{L}(u) - f\|_Y$. The lower bound of (3) leads to the fundamental important relation

$$C_1 \|u - u_{ex}\|_X \leq \|\mathcal{L}(u) - f\|_Y, \quad \forall u \in X, \tag{4}$$

which shows that the if the norm of the residual approaches zero ($\|\mathcal{L}(u) - f\|_Y \rightarrow 0$), the approximate solution converges to the exact solution ($\|u - u_{ex}\|_X \rightarrow 0$). Consequently, it is equivalent to seek the minimizer of the quadratic least-squares functional

$$\mathcal{J}(u) = \frac{1}{2} \left(\|\mathcal{L}(u) - f\|_Y^2 \right). \tag{5}$$

The minimization of (5) can be obtained by means of variational analysis, giving the Euler-Lagrange equation

$$\lim_{\epsilon \rightarrow 0} \frac{d}{d\epsilon} \mathcal{J}(u + \epsilon v) = \int_{\Omega} (\mathcal{L}(v))^T (\mathcal{L}(u) - f) d\Omega = 0, \quad \forall v \in X. \tag{6}$$

The least-squares method can therefore be stated as

$$\text{Find } u \in X \text{ such that } \mathcal{B}(u, v) = \mathcal{F}(v), \quad \forall v \in X, \tag{7}$$

where $\mathcal{B}(u, v) = (\mathcal{L}(u), \mathcal{L}(v))_Y$ and $\mathcal{F}(v) = (f, \mathcal{L}(v))_Y$. Since for a well-posed problems the operator \mathcal{L} is bounded below, (7) leads always to a SPD system ⁹.

Finally a finite dimensional subspace $X^h \subset X$ parametrized by $h \rightarrow 0$ (where h may refer to a characteristic mesh width, polynomial degrees or a combination of both) is chosen. This results to the discrete variational problem

$$\text{Find } u^h \in X^h \text{ such that } B(u^h, v^h) = F(v^h), \quad \forall v^h \in X^h. \quad (8)$$

3 SPECTRAL ELEMENTS

The domain Ω is subdivided into N non-overlapping quadrilateral sub-domains Ω^k .

$$\Omega = \bigcup_{k=1}^N \Omega^k, \quad \Omega^k \cap \Omega^l = \emptyset, k \neq l. \quad (9)$$

Each sub-domain is mapped onto a standard domain $[-1, 1]^d$, where $d = \dim(\Omega)$. Within this standard domain the unknown function is approximated by polynomials (basis functions). In this paper a spectral element method based on Legendre polynomials is employed ¹⁰. These basis functions are Lagrangian interpolants through the Gauss-Lobatto-Legendre (GLL) points and explicitly given by

$$h_i(\xi) = \frac{(\xi - 1)(\xi + 1)L'_P(\xi)}{P(P + 1)L_P(\xi_i)(\xi - \xi_i)} \quad \text{for } i = 0, \dots, P, \quad (10)$$

where $L'_P(\xi)$ denotes the derivative of the P -th Legendre polynomial and ξ_i are the $P + 1$ GLL-points ($P - 1$ zeros of the first derivative of the Legendre polynomial $L'_P(\xi)$ supplemented with the boundary nodes $\xi_0 = -1$ and $\xi_P = 1$).

For $d = 1$ the approximate solution in each sub-domain in terms of these Lagrangian basis functions is given by

$$u(\xi) = \sum_{i=0}^P \hat{u}_i h_i(\xi), \quad (11)$$

where \hat{u}_i needs to be determined by the least-squares method. Since a general higher order PDE is converted to an equivalent first order system, C^0 -continuity is sufficient to connect the individual sub-domains to the complete domain Ω . The integrals appearing in the least-squares formulation (6) are approximated numerically by Gauss-Lobatto quadrature

$$\int_{\Omega^k} f(x) dx = \int_{-1}^1 f(\xi) |J^e| d\xi \approx \sum_{i=0}^Q w_i |J^e| f(\xi_i), \quad (12)$$

where ξ_i represent the $Q + 1$ distinct GLL-points in the interval $[-1, 1]$, $|J^e|$ is the Jacobian from the transfinite mapping and w_i are the GLL weights given by

$$w_i = \frac{2}{Q(Q + 1)L_Q^2(x_i)}, \quad i = 0, \dots, Q. \quad (13)$$

Notice that the polynomial degree Q can differ from the polynomial degree P of the Lagrangian basis functions.

If $P = Q$, the derivative of the Lagrangian basis functions (10) evaluated at the GLL-points are given by

$$\left. \frac{dh_j}{d\xi} \right|_{\xi_i} = \begin{cases} \frac{L_Q(\xi_i)}{L_Q(\xi_j)} \frac{1}{\xi_i - \xi_j}, & i \neq j, 0 \leq i, j \leq Q, \\ 0, & 0 < i = j < Q, \\ -\frac{1}{4}Q(Q+1), & i = j = 0, \\ \frac{1}{4}Q(Q+1), & i = j = Q. \end{cases} \quad (14)$$

The extension to multidimensional problems is performed by using tensor products. Inserting the finite dimensional approximation and its derivatives in (8) and evaluating the integrals using (12) leads to an algebraic system for the unknowns \hat{u}_i .

4 DIRECT MINIMIZATION

The conventional least-squares method as described in section 2 can be written as

$$(\mathcal{L}(u), \mathcal{L}(v)) = (f, \mathcal{L}(v)) \iff \int_{\Omega} \mathcal{L}(u)\mathcal{L}(v) d\Omega = \int_{\Omega} f\mathcal{L}(v) d\Omega \quad \forall v \in x(\Omega). \quad (15)$$

The integral over the entire domain Ω can be written as a summation of the integrals over the sub-domains Ω^k

$$\sum_{k=1}^N \int_{\Omega^k} \mathcal{L}(u^k)\mathcal{L}(v^k) d\Omega^k = \sum_{k=1}^N \int_{\Omega^k} f\mathcal{L}(v^k) d\Omega^k \quad \forall v^k \in x(\Omega^k). \quad (16)$$

With the approximation of u^k as in (11) it follows

$$\sum_{k=1}^N \left[\sum_{i=0}^P \hat{u}_i^k \int_{\Omega^k} \mathcal{L}(h_i)\mathcal{L}(h_j) d\Omega^k \right] = \sum_{k=1}^N \int_{\Omega^k} f\mathcal{L}(h_j) d\Omega^k \quad \forall h_j, j = 0, \dots, P. \quad (17)$$

Inserting the Gauss-Lobato quadrature (12) finally gives, again $\forall h_j$ and $j = 0, \dots, P$

$$\sum_{k=1}^N \left[\sum_{i=0}^P \hat{u}_i^k \sum_{q=0}^Q \mathcal{L}(h_i)(\xi_q)\mathcal{L}(h_j)(\xi_q) |J^e| w_q \right] = \sum_{k=1}^N \sum_{q=0}^Q f(\xi_q)\mathcal{L}(h_j)(\xi_q) |J^e| w_q. \quad (18)$$

This discretized least-squares problem can be written as

$$\sum_{k=1}^N \left[(\mathbf{A}^k)^T \mathbf{W}^k \mathbf{A}^k \right] u^k = \sum_{k=1}^N \left[(\mathbf{A}^k) \mathbf{W}^k \mathbf{F}^k \right], \quad (19)$$

with an element matrix \mathbf{A}^k defined by $(\mathbf{A}^k)_{qi} = \mathcal{L}(h_i)(\xi_q)$, a diagonal matrix \mathbf{W}^k defined by $(\mathbf{W}^k)_{qq} = |J^e| w_q$ and where the vector \mathbf{F}^k contains the elements $(\mathbf{F}^k)_q = f(\xi_q)$.

The resulting system of algebraic equations are called the normal equations. If the polynomial degree of the Gauss-Lobatto quadrature differs from the polynomial degree of the Lagrangian basis functions, the matrix \mathbf{A}^k is non-square. The resulting normal equations still deliver a square SPD matrix.

The matrix-matrix multiplication for setting up the normal equations is expensive in terms of CPU time, it may lead to loss of information ^{6 7} and the condition number is squared with respect to the system without pre-multiplication with the transpose of the partial differential operator. An improved formulation, which avoids variational analysis, has been recently implemented in the conventional LSQSEM which minimizes the residual directly. The minimization functional is given by (5) and can be written as

Find all $u^k \in X(\Omega^k)$ which minimize the functional

$$\mathcal{J}(u^0, \dots, u^{N-1}) = \sum_{k=1}^N \|\mathcal{L}(u^k) - f\|_{Y(\Omega^k)}^2. \quad (20)$$

As in (8), the solution space is restricted to a finite dimensional subspace $X^h(\Omega^k) \subset X(\Omega^k)$, so $u^k \in X^h(\Omega^k)$, using the spectral approximation given by (10).

The integrals which constitute the L^2 -norm can numerically be written with (12) as

Find all $u^k \in X^h(\Omega^k)$ which minimize the functional

$$\mathcal{J}(u^0, \dots, u^{N-1}) \approx \sum_{q=0}^Q \sum_{k=1}^N \left(\mathcal{L}(u^k) - f \right)^2 \Big|_{x_p} w_p. \quad (21)$$

In matrix notation, as presented before, this can be written as

Find all $u^k \in X^h(\Omega^k)$ which minimize the functional

$$\sum_{k=1}^N \left(\mathbf{A}^k u^k - \mathbf{F}^k \right) \mathbf{W}^k \left(\mathbf{A}^k u^k - \mathbf{F}^k \right) = \sum_{k=1}^N \left\| \sqrt{\mathbf{W}^k} \left(\mathbf{A}^k u^k - \mathbf{F}^k \right) \right\|^2. \quad (22)$$

The solution which minimizes the residual norm of (22) is given by

$$\sum_{k=1}^N \sqrt{\mathbf{W}^k} \mathbf{A}^k u^k = \sum_{k=1}^N \sqrt{\mathbf{W}^k} \mathbf{F}^k. \quad (23)$$

It can be proven ^{6 7 8} that this is equivalent to (19), the normal equations, but this system is much better conditioned, avoids matrix-matrix multiplication and prevents fill-in. This is, however, an overdetermined algebraic system which has to be solved in the least-squares sense.

5 TIME INTEGRATION

The time integration can be implemented in several ways. One approach is to treat space and time uniformly ¹¹. This is a space-time formulation which increases the physical dimension with one. Therefore, for a one-dimensional time-dependent testcase, two-dimensional spectral elements are required. The advantage is the balanced accuracy in

both space and time and with that an increase in stability. The main disadvantage is the larger system that has to be solved and with that the increase of required memory. By dividing the space-time domain into space-time strips, the required memory can be reduced.

Another approach is a time-stepping formulation. The main advantage is that the system is much smaller and with that the required memory. To maintain a stable solution, an higher order time-stepping formulation has to be used. The first time-stepping formulation that is used in this work is the second order Backward Difference Formula (BDF2). This scheme is unconditionally asymptotically stable (A-stable) and stiffly stable (L-stable)¹². Applying this integration scheme to the model equation

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{F}(\mathbf{u}) = 0, \quad (24)$$

results into

$$\frac{3\mathbf{u}^{n+1} - 4\mathbf{u}^n + \mathbf{u}^{n-1}}{2\Delta t} + \mathbf{F}(\mathbf{u}) = 0, \quad (25)$$

where the superscript indicates the timelevel and Δt is the timestep size. Since this scheme is not self-starting, it needs an implicit Euler step to start. After this first timestep the BDF2 scheme is used.

Another used time-stepping formulation is the third order BDF (BDF3) scheme. Applying this scheme to the model equation (24) results to

$$\frac{11\mathbf{u}^{n+1} - 18\mathbf{u}^n + 9\mathbf{u}^{n-1} - 2\mathbf{u}^{n-2}}{6\Delta t} + \mathbf{F}(\mathbf{u}) = 0, \quad (26)$$

which is only $L(\alpha)$ -stable with $\alpha = 86.03^\circ$ ¹². This scheme is also not self-starting, so the first time step is also done by an implicit Euler step. The second step is done by the BDF2 scheme, after which the BDF3 scheme is used.

6 NUMERICAL RESULTS

6.1 1D linear advection equation

The first test case consists of a one-dimensional linear advection problem given by

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = 0, \quad 0 \leq x \leq L, \quad t \geq 0, \quad (27)$$

with

$$u(0, x) = u_0(x) \text{ and } u(t, 0) = u(t, L). \quad (28)$$

The exact solution of this problem is given by

$$u(t, x) = u_0(x - at). \quad (29)$$

In this work the initial condition is a single cosine hill on $0 \leq x \leq 1$, and zero on the rest of the domain, which has a length $L = 4$.

$$u_0(x) = \begin{cases} \frac{1}{2}(1 - \cos(2\pi x)) & \text{for } 0 \leq x \leq 1, \\ 0 & \text{for } 1 < x \leq L = 4. \end{cases} \quad (30)$$

It can be proven that the solution $u(x, T) \in H^{2\frac{1}{2}-\epsilon}(x)$, $\forall \epsilon > 0$, with T the final time level.

To perform a convergence study, the error in the L^2 -norm will be compared. This error is defined as

$$\|\epsilon\|_{L^2} = \|\epsilon\|_0 = \left(\int_0^L (u^h(x) - u_e(x))^2 dx \right)^{\frac{1}{2}}, \quad (31)$$

where u_e is the exact solution and $u^h(x)$ is the approximate solution at the final time level T .

Two types of convergence can be studied when spectral elements are applied. The number of cells can be increased (h-refinement) or the polynomial degree of the elements can be increased (p-enrichment). Only the first type is done in this work. It can be proven that for h-refinement, when the exact solution is $u \in H^s(\Omega)$, the error in the L^2 -norm is given by ¹

$$\|\epsilon\|_0 \leq Ch^l |u|_l, \quad \text{with } l = \min(P + 1, s), \quad (32)$$

where P is the polynomial degree of the spectral elements and h represents a grid parameter. In this work $h = \Delta t$, the size of the time step. The seminorm $|u|_l$ and C are constant for a given problem.

Figure 1 shows the convergence for the single cosine hill. The second and third order BDF schemes and three space-time formulations ($P_t = 1$, $P_t = 2$ and $P_t = 3$) are compared with a constant polynomial degree in x-direction ($P_x = 8$) and a constant $\Delta x = 0.1$. It is expected that the convergence rate for the second order BDF scheme is equal to two, but the measured value is slightly lower. Because this scheme is A-stable and L-stable no problems occur for larger Δt . This is different with the third order BDF scheme, which is only L(α)-stable with $\alpha = 86.03^\circ$. This scheme did not converge for Δt values larger than 0.002. For decreasing Δt , the error slightly increases. A cause of this effect could be explained with ¹³

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} = \frac{d\mathbf{u}}{dt} + \mathcal{O}\left(\Delta t^k + \frac{\mathcal{O}(\Delta x^{P+1})}{\Delta t}\right), \quad (33)$$

where k is the order of the integration method and P the order of the polynomial basis. The first part of the overall error is due to the backward integration and the second part is due to the interpolation. This equation shows that the overall error is not monotonic with respect to Δt . When the polynomial order P is small, the interpolation error dominates; as Δt decreases, the overall error increases. If $\mathcal{O}(\Delta t^k)$ is subdominant, further increase of

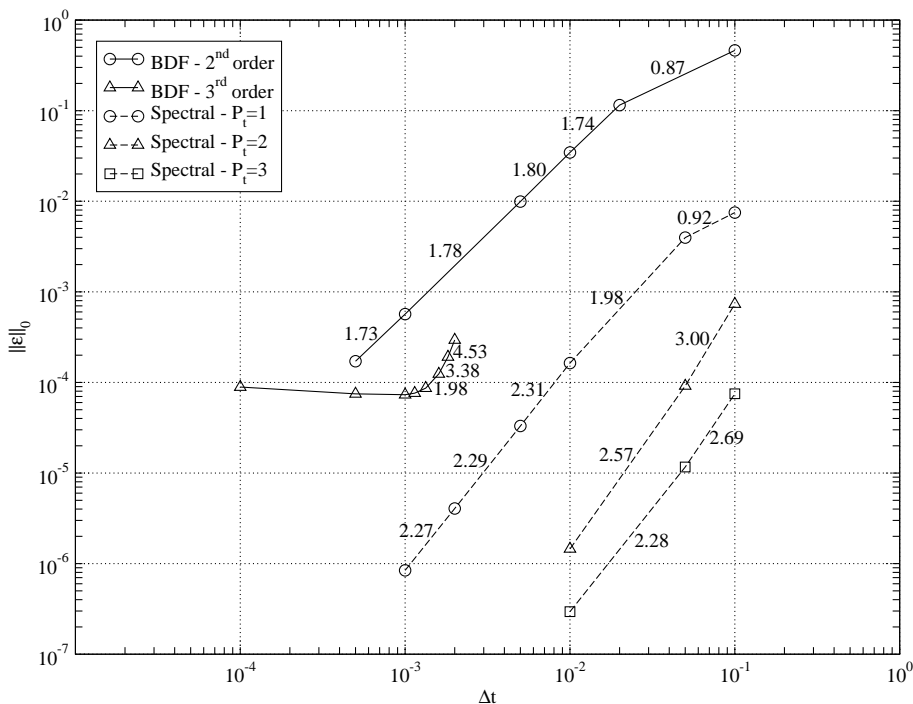


Figure 1: Convergence for the 1D linear advection of the single cosine hill at a fixed time level $T = 10$ for 2 time-stepping and 3 space-time formulations, in all cases $P_x = 8$ and $\Delta x = 0.1$. The value next to a line is the convergence rate of that line.

k will not improve the overall accuracy. If on the other hand $\mathcal{O}(\Delta x^{P+1})$ is subdominant, the overall error is dominated by the time stepping. To improve the results it is therefore important to know which error dominates the overall error.

The convergence rate of the space-time version is determined by (32). For the space-time approach with $P_t = 1$ a convergence rate of 2 is expected. The numerical results show a slightly higher convergence rate. For the $P_t = 2$ and $P_t = 3$ cases the expected convergence rate is 2.5. For both cases this value is not reached, but simulation with a smaller time step could show the correct convergence rate.

Figure 2 shows the CPU times in seconds against the overall error norm for the 5 cases. What stands out is that the second order BDF scheme and the space-time scheme with $P_t = 1$, which both have a convergence rate of 2, have almost equal CPU times for a certain error. The third order BDF scheme shows similar results for CPU times around 100 seconds, but it must be mentioned that both space-time schemes with $P_t = 2$ and $P_t = 3$ have only a convergence rate of 2.5. To get a better insight it is interesting to investigate an infinitely smooth function (smooth cosine function) and less smooth function (discontinuous square wave).

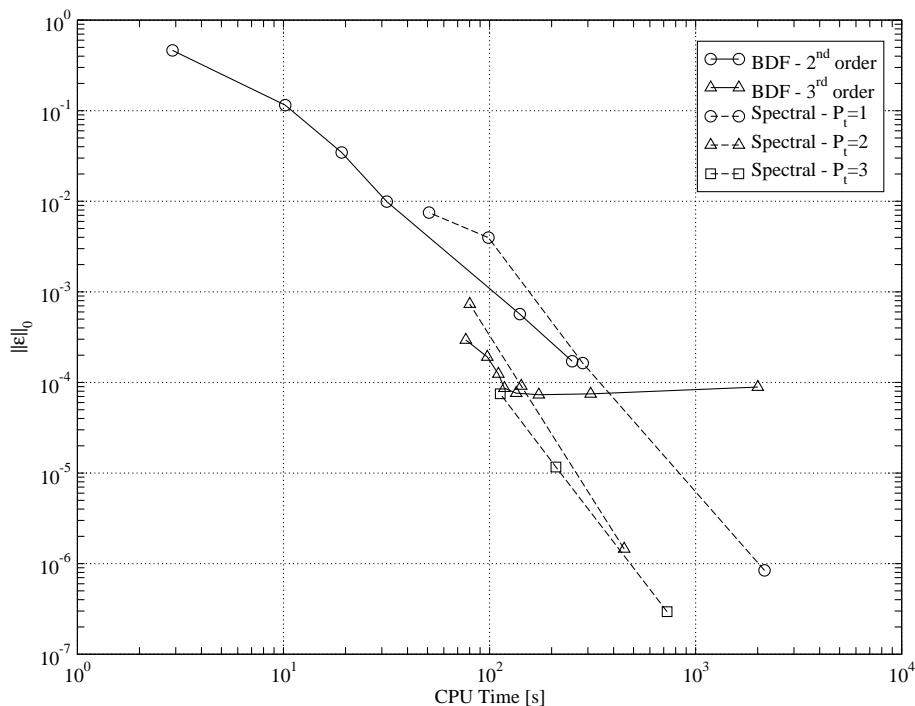


Figure 2: CPU times for the 1D linear advection of the single cosine hill at a fixed time level $T = 10$, in all cases $P_x = 8$ and $\Delta x = 0.1$.

6.2 2D linear advection equation

The second test case consists of a two-dimensional linear advection equation given by

$$\frac{\partial u}{\partial t} + a_x \frac{\partial u}{\partial x} + a_y \frac{\partial u}{\partial y} = 0, \quad -1 \leq x \leq 1, \quad t \geq 0, \quad (34)$$

where $a_x = 2\pi y$ and $a_y = -2\pi x$ to create a rotating transport field. The initial condition is given by the Gaussian cone

$$u(x, y, 0) = u_0(x, y) = e^{-\frac{[(x-x_0)^2 + (y-y_0)^2]}{2\lambda^2}}. \quad (35)$$

The exact solution to this problem is

$$u(x, y, t) = e^{-\frac{[\hat{x}^2 + \hat{y}^2]}{2\lambda^2}}. \quad (36)$$

where

$$\hat{x} = x - x_0 \cos 2\pi t - y_0 \sin 2\pi t, \quad \hat{y} = y + x_0 \sin 2\pi t - y_0 \cos 2\pi t \quad (37)$$

Fixing the constants ¹⁴ as $\lambda = \frac{1}{8}$ and $(x_0, y_0) = (-\frac{1}{2}, 0)$. The region $-1 \leq x, y \leq 1$ is discretised with a mesh consisting of 4×4 quadrilateral elements. The solution is time integrated for one revolution, corresponding to $t = 1$.

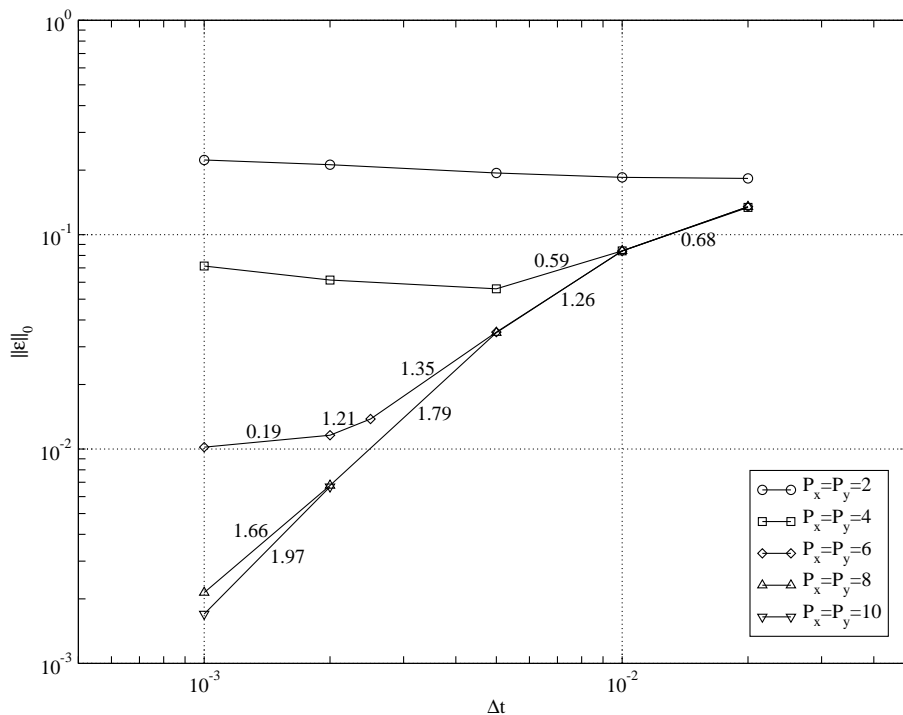


Figure 3: Convergence for the 2D linear advection of the Gaussian cone after one revolution, corresponding with $T = 1$. In all cases $\Delta x = 0.5$ and time integration by the BDF2 scheme, again the value next to a line is the convergence rate of that line.

Figure 3 shows the convergence for the Gaussian cone. The infinitely smooth Gaussian cone is time integrated by the BDF2 scheme. According to (32) the convergence rate should be equal to 2. For a polynomial degree of 10 in x and y direction ($P_x = P_y = 10$) and a small timestep this value is reached. For lower polynomial degrees the convergence rates are much lower. For a small timestep combined with a low polynomial degree the error even grows. This is an equivalent behaviour as before with the 1D advection equation. This could therefore also be explained by (33).

7 CONCLUSIONS

All presented results were obtained without any problems with stability caused by the least-squares formulation. No upwinding or artificial diffusion is needed as for the Galerkin formulation, which is a big advantage of the least-squares method. Because no problems with accuracy were expected, the conventional least-squares method was used in this work. The only stability problems that occurred are related with the time-stepping method. The third order BDF scheme is not unconditionally stable, so the choice of the stepsize and grid spacing is important. For the time-stepping method it is also important to know which error term dominates the overall error. A smaller time step or a higher polynomial degree is not always better for the overall error. The best solution, in terms of

CPU time, is obtained when both error terms are equally important for the overall error.

REFERENCES

- [1] M.M.J. Proot. *The least-squares spectral element method - Theory, Implementation and Application to Incompressible Flows*, PhD thesis, Delft University of Technology, (2003).
- [2] M.M.J. Proot and M.I. Gerritsma. *A least-squares spectral element formulation for the Stokes problem*, J. Sci. Comput. 17(1-3) 311-322, (2002).
- [3] M.M.J. Proot and M.I. Gerritsma. *Least-squares spectral elements applied to the Stokes problem*, J. Comput. Phys. 181(2) 454-477, (2002).
- [4] M.M.J. Proot and M.I. Gerritsma. *Application of the least-squares spectral element method using Chebyshev polynomials to solve the incompressible Navier-Stokes equations*, Num. Alg. 38, 155-172, (2005).
- [5] B. De Maerschalck and M.I. Gerritsma. *The use of Chebyshev Approximation in the Space-Time Least-Squares Spectral Element Method*, Num. Alg. 38, 173-196, (2005).
- [6] R. de Groot. *Direct Minimization of Equation Residuals in Least-Squares hp-Finite Element Methods - Numerical study of a low Re number flow around a circular cylinder*, Thesis report, Delft University of Technology, (2004).
- [7] W. Hoitinga. *Direct Minimization of Equation Residuals in Least-Squares hp-Finite Element Methods - A direct and iterative solution method*, Thesis report, Delft University of Technology, (2004).
- [8] W. Hoitinga, R. de Groot and M.I. Gerritsma. *Direct Minimization of the Least-Squares Spectral Element Method - Part I: Direct Solver*, in preparation.
- [9] B.N. Jiang. *On the Least-Squares Method*, Applied Mathematics and Computation 116 (3): 297-305, (Dec 2000).
- [10] G.E. Karniadakis and S. Sherwin. *Spectral/hp Element Methods for Computational Fluid Dynamics*, Second Edition, Oxford University Press, (2005).
- [11] B. De Maerschalck. *Space-Time Least-Squares Spectral Element Method for Unsteady Flows - Application and Evaluation for Linear and Non-linear Hyperbolic equations*, Thesis report, Delft University of Technology, (2003).
- [12] H. Bijl, M.H. Carpenter and V.N. Vatsa. *Time Integration Schemes for the Unsteady Navier-Stokes Equations*, AIAA-2001-2612, (2001).

- [13] M. Falcone and R. Ferretti. *Convergence analysis for a class of high-order semi-Lagrangian advection schemes*, SIAM J. Numer. Anal., 35, 909, (1998).
- [14] F.X. Giraldo. *The Lagrange-Galerkin Spectral Element Method on Unstructured Grids*, Journal of Computational Physics, (Aug 1998).