Computationally Efficient Parallel Algorithms for Time-Dependent Convection-Diffusion Models

Melnik, R.V.N. and Gavrilyuk, I.P.

**In: Proceedings of the Fifth World Congress on Computational Mechanics,
WCCM V, July 7-12, Vienna, Austria,
Editors: Mang, H.A.; Rammerstorfer, F.G.; Eberhardsteiner, J.,
Publisher: Vienna University of Technology, Austria, 9 pages, 2002,
ISBN 3-9501554-0-6**

WCCM V
Fifth World Congress on
Computational Mechanics
July 7–12, 2002, Vienna, Austria
Eds.: H.A. Mang, F.G. Rammerstorfer,
J. Eberhardsteiner

Computationally Efficient Parallel Algorithms for Time-Dependent Convection-Diffusion Models

Roderick V.N. Melnik*

University of Southern Denmark, MCI, DK-6400, Denmark
e-mail: rmelnik@mci.sdu.dk

Ivan P. Gavrilyuk

Berufsakademie Thueringen, 99817 Eisenach, Germany

Key words: Parallel algorithms, evolutionary problems, Lie group models, Cayley transform

Abstract

The convection-diffusion models serve as a test-ground for the efficiency of algorithms applied in complex fluid dynamics applications, meteorology, subsurface porous medium fluid flow, aerodynamics, semiconductor device modelling, and in many other application areas.
However, most of the algorithms proposed so far for these models are not easily parallelisable in time domain. A systematic way of constructing effectively parallelisable in time algorithms is provided by the application of the Cayley transform technique. In this paper we develop such an algorithm for convection-diffusion models.
Moreover, since an intrinsic part of the proposed algorithm consists of solving approximately systems of resolvent equations at some collocation points, we show also how this procedure can be parallelised efficiently for the class of the problems we consider in this paper.

1 Introduction

Time-dependent convection-diffusion models appear frequently in many branches of computational mechanics including, but not limited to, fluid dynamics and electro-magnetic theory [13, 6, 5]. They also serve as building blocks for many complex applied problems requiring coupling several different physical fields [13]. The development of efficient discretisation methodologies for such models represent a challenging and important area of research in computational mechanics ([13, 6] and references therein).

In [6] we proposed an efficient methodology for the full time-space discretisation convection-diffusion-absorption equations based on the Cayley transform technique. Recently, a general framework based on the so-called strongly P-positive operator technique has been developed ([8] and references therein). Due to the ability of this technique to generate exponentially convergent algorithms easily amenable to efficient parallelisation strategies, this framework allows one to construct such algorithms for full time-space disretisations of evolutionary differential equations that have several important advantages over conventional algorithms. In this paper we use this general framework to develop further our ideas in the context of time-dependent convection-diffusion problems [6]. In particular, we construct exponentially convergent discretisations for these problems by using an efficient representation of the operator exponent and appropriate quadratures based on the Sinc approximations of corresponding integrals. We note that in this case the resulting quadratures, consisting of a sum of operator resolvents, can be effectively (at almost linear cost) approximated by the $\mathcal{H}$-matrix technique ([8] and references therein). The situation where the diffusion coefficient is small due to the presence of boundary layers [6] is also discussed. The algorithms presented in this paper admit parallel implementations in terms of both the computation of resolvents and the time stepping.

2 Geometric integration, Lie group models, and approximations of operator exponents

Conventional time marching algorithms used widely for mathematical models based on time-dependent PDEs are difficult to parallelise. However, if the idea of time stepping is developed on the basis of the Cayley transform technique the parallelisation in time is natural as long as one needs to calculate the solution at different moments of times. The Cayley transform methodology allows us to develop numerical discretisations of PDEs that respect the underlying geometry, and this methodology is linked closely to the concept of *geometric integration* [11]. It is well known that the linear matrix differential equation

$$y' = A(t)y \tag{1}$$

can be represented in a form of the matrix exponential of an expansion in terms of matrix commutators and integrals, and if the resulting equation is solved via Picard iterations we obtain what is known as the Magnus expansion (e.g., [12]). The question then is to approximate efficiently the integrals participating in that expansion. This and similar examples have been recently treated in a unified manner via Lie algebra actions on manifolds [11]. The main premises of such a treatment rely on the fact that any differential equation evolving on a Lie group G in t ($t \geq t_0$) can be presented via the Lie algebra of G, g, such that

$$Y' = A(t,Y)Y, \quad t \geq t_0, \quad Y(t_0) = Y_0, \quad A : [t_0, \infty) \times G \to g, \quad Y_0 \in G. \tag{2}$$

When constructing numerical discretisations to (2) it is desirable to *conserve* its Lie-group structure. Such discretisations can be obtained by approximating the operator exponent, just in the way analogous to that of (1). The point to emphasise is that not all approximations conserve the desired property, and even if they do the computational cost of evaluating the matrix exponent can be huge. This is not the case if G is a quadratic Lie group where the Cayley transform can be used efficiently not only to evaluate the matrix exponents, but also to lend itself directly in place of the exponential map (e.g., [11] and references therein). The Cayley transforms (known also as coordinatisations) exist for a wide class of Lie groups including orthogonal ($O(n)$), special orthogonal ($SO(n)$), and symplectic ($Sp(n)$). Integration techniques based on the Cayley transform allows us to construct quite efficient computational algorithms primarily because computational expenses for the Cayley transform are considerably smaller compared to those of the exponential map. In recent years, many local geometric integrator techniques, similar to RKMK methods, have been developed based on different coordinatisations of the Lie groups. By using special quadrature formulae for computing integrals participating in the Cayley expansion a high computational efficiency of this methodology can be achieved. Essentially, all classical numerical integrators are skew-symmetric preserving integrators, and therefore explicit methods can be employed to reduce the cost of the numerical procedure. One important example of this type is provided by isospectral flows described by $Y' = [B(t, Y), Y]$, $Y(0) = Y_0$, $t \quad t \geq 0$, where $Y(t)$ preserves the eigenvalues of Y_0 and is chosen to be symmetric for all t. This example covers Toda flows where $B(t, Y)$ is taken as the difference between the upper, Y_+, and the lower, Y_-, triangular parts of Y ($B(t, Y) = Y_+ - Y_-$.

When we study physical, mechanical, or biological patterns with models based on differential equations, it is natural to require that such important qualitative properties of the differential model as unitarity, symplecticness, isospectrality should be preserved in a discrete approximation constructed. Such conservative numerical methods are well known, for example, when Y_0 in (2) is a unitary matrix, and $A(t, Y)$ is a skew-Hermitian (e.g., [4, 3]). Such systems are fundamental in the computation of Lyapunov characteristic exponents (LCEs) which characterise the behaviour of nonlinear dynamical systems by giving a measure of the exponential growth (or shrinkage) of perturbations about a nominal trajectory. In particular, positive LCEs typically indicate the onset of chaotic motions [16]. The Cayley transform is at the heart of effective algorithms for computing the Lyapunov characteristic exponents since it allows to transform the unitary system into the skew-Hermitian differential system [4].

Having in mind this geometric approach to numerical integration of differential equations, we turn now our attention to convection-diffusion models known to be difficult to treat numerically for high Peclet number.

3 New efficient algorithms for time-dependent convection-diffusion models

Parallel algorithms for convection-diffusion type models and different domain-decomposition (preconditioning) methodologies for parallel FEM solutions have been developed in the literature. In particular, FEMs are developed quite well when applied on sequences of uniformly refined meshes. However, when the solution exhibits local behaviour, e.g. shocks or boundary layers, meshes should be obtained via local rather than global refinements. The difficulties involved in the numerical solution of convection-diffusion models in convection-dominated areas can be seen from the standard Galerkin approximation of the simplest steady-state variant of the model

$$-\epsilon\Delta u + u_x = f \tag{3}$$

considered on a (bounded) polygonal domain

$$\int_\Omega (\epsilon \nabla u_h \cdot \nabla v + (u_h)_x) dx = \int_\Omega f v dx \tag{4}$$

for all v from an appropriately defined finite element space (e.g., [2]). This standard methodology works well as long as the Peclet number is low, and it breaks down as soon as the Peclet number is high enough. In other words, as soon as $\epsilon/h << 1$, this procedure is unstable giving large (maximum-principle-violating) oscillations in the numerical solution. In this situation we are within singular perturbation models presenting non-trivial numerical difficulties. As a quick fix, different upwind schemes can be used compromising in accuracy of the numerical solution, either due to oscillations or due to excessive diffusion. In this sense, a better approach is to use some more efficient stabilisation techniques such as the Streamline-Upwind/Petrov-Galerkin (with Galerkin/least-squares methodologies as a special case), flux-corrected transport methodologies, the residual-free bubble approaches, coupled FV/FE procedures, the subgrid viscosity methods, and the like. In the case SUPG, for example, we base the stabilisation procedure on a modification of the classical variational formulation (4) in a consistent way by adding elementwise stabilisation terms which measure the local (strong) residual of the equation in an L^2 type norms [2]

$$\int_\Omega (\epsilon \nabla u_h \cdot \nabla v + (u_h)_x) dx - \sum_\Delta \tau_\Delta \int_\Delta (-\epsilon \Delta u_h + (u_h)_x - f)(-\epsilon \Delta v - v_x) dx = \int_\Omega f v dx, \tag{5}$$

where the sum and the integral are taken over all triangles of the given discretisation. The need for such a stabilisation stems from the fact that for convection-dominated problems the coerciveness constant is substantially smaller compared to the continuity constant, so that at a discrete level the problem may become noncoercive. A general framework has been recently developed based on stabilising terms considered in negative norms which can be computed via multiscale decomposition techniques (e.g., wavelets [1]). The choice of the "time-scale" parameter τ_Δ is critical and has much in common with other just mentioned procedures of stabilisation (e.g., the subgrid viscosity method). Note also that the SUPG can be derived from the residual-free bubble approach or from the local Green's function approach (see [2] and references therein). In all these approaches the choice of τ_Δ is critical to the success of the whole underlying procedure. Certainly, model (3) is part of a bigger picture

$$(\epsilon A + C) u = f, \tag{6}$$

where A is symmetric second-order elliptic operator, and C is a skew-symmetric first-order differential operator, $\epsilon > 0$. Most of the methodology described briefly above are well suited for elliptic problems, but when they are applied to time-dependent problems, in particular hyperbolic, the well-known non-trivial additional difficulties arise. The main purpose of this paper is, therefore, to develop an alternative numerical methodology well-suited for time-dependent convection-diffusion problems, and to show that the develop procedure allows several levels of parallelisation.

Note that the nonstationary counterpart of (6) with non-self-adjoint spatial operator is a natural candidate for applying the Cayley transform technique. Recall the model deduced from the general convection-diffusion-reaction equation in [6]

$$\frac{\partial u}{\partial t} = \mathcal{L} u + f, \quad \mathcal{L} \equiv \tilde{\beta} \frac{\partial u}{\partial x} + \mu \frac{\partial^2 u}{\partial x^2} + \tilde{\alpha} u \tag{7}$$

with given coefficients $\tilde{\alpha}$, $\tilde{\beta}$, and μ, and with initial and boundary conditions

$$u(x, 0) = u_0(x), \quad u(d_0, t) = u(d, t) = 0. \tag{8}$$

For nonhomogeneous boundary conditions, e.g. $u(a, t) = u_1(t)$, $u(b, t) = u_2(t)$ the problem is reduced to the previous situation by the change of variables $u \to u(x, t) - \dfrac{x - a}{b - a} u_2(t) - \dfrac{x - b}{a - b} u_1(t)$.

If $\mathcal{L}^+$ had "nice" properties such as symmetry, strong elipticity and compact invertibility, then a traditional way to represent the solution to such a problem would be via the analytic semigroup T given by the operator $\mathcal{L}^+$ (e.g., [14, 6])

$$u(t) = T(t)u_0 + \int_0^t T(t-s)f(s)ds. \tag{9}$$

Both the first term in (9) and the integrand can be computed by solving a homogeneous problem (7) (written in terms of a new variable associated with $T(t)$ which we denote with the same letter for simplicity)

$$\frac{\partial u}{\partial t} = \mathcal{L}^+ u, \tag{10}$$

solution to which (even in a more general situation) can be given in the form of the Dunford-Cauchy integral [8]

$$u(t) = \frac{1}{2\pi i}\int_\Gamma e^{-zt}(zI - \mathcal{L}^+)^{-1}u_0 dz = \frac{1}{2\pi i}\int_\Gamma e^{-zt}\hat{u}(z)dz, \tag{11}$$

where $\hat{u}$ denotes the resolvent of the operator, that is the solution to

$$(zI - \mathcal{L})\hat{u}(z) = u_0. \tag{12}$$

In the case of symmetric, positive definite operator $\mathcal{L}^+$ the above procedure has been recently applied in [15] to develop a parallel algorithm for time stepping. The case treated here is more general, and the rate of convergence of the algorithm discussed here is exponential rather than polynomial as in the above-mentioned paper. The key of this substantial improvement lies with the fact that the choice of the integration path depends decisively on the geometric properties of the spectrum and the behaviour of the resolvent, and this fact can be used effectively in the algorithm construction [8].

In the case of the convection-diffusion models we have to deal with non-self-adjoint operators such as those in (6) and (7). Moreover, for convection-diffusion models the operator under consideration is a non-normal operator [13]. Therefore, although the above procedure can still be applied (with some modifications, see [8]), the choice of the integration path Γ in (11) should respect the geometry of the problem in a sense that it should be chosen depending on properties of the spectrum as well as behaviour of the resolvent [6, 13, 8].

A relaxation of the assumptions discussed above to allow the treatment of a more general class of models has been developed recently in [7] where the concept of P-positive operators was introduced. If $\mathcal{L}^+$ is a strongly P-positive densely defined (on a domain $D(L) \subset E$ where E is a Banach space) closed operator such that its spectrum is within a parabola Γ, then representation (11) is valid for the solution to the convection-diffusion model (7), (8) as well. Therefore, the algorithmic procedure for the approximation of (7), (8) follows immediately if we choose N "collocation" points on the parabola and find the solution to the auxiliary problem of finding resolvents at these points by using (12). Then, the "semi-discrete" approximation to (7), (8) can be found as

$$u_N(t) = \sum_{k=-N}^{N} \alpha_k e^{-z_k t}\hat{u}(z_k). \tag{13}$$

The choice of "collocation" points and the coefficients α_i in (13) has been recently described in detail in [8, 9], and here we only note that the choice is made based on the approximation of the integral for operator exponent $\exp(-t\mathcal{L}^+)$ by a Sinc-based quadrature formulae applied to a parametrised representation of the integral. This choice relies substantially on the geometric shape of the parabola participating in the definition of strong positivity of the operator. In particular, integral (11) can be represented in the following form (e.g., [8])

$$u(t) = \frac{1}{2\pi i}\int_{-\infty}^{\infty} F(\eta, t) d\eta, \tag{14}$$

where $F(\eta, t) = e^{-zt}\hat{u}(z)\dfrac{dz}{d\eta}$ with $z = \tilde{a}\eta^2 + b - i\eta$ and $\tilde{a}$ and b as parameters, and therefore the Sinc approximation can be used on the whole axis $(-\infty, \infty)$. We note that this representation is deduced from the standard Dunford-Cauchy formula (see (11)) which is parametrisable with respect to the contour of integration, so in its essence we are dealing with efficient approximations of operator exponents via their representations in terms of operator resolvents

$$\exp(-t\mathcal{L}) = \frac{1}{2\pi i}\int_{\Gamma} e^{-zt}(zI - \mathcal{L}^{-1}dz = \frac{1}{2\pi i}\int_{-\infty}^{\infty} F(\eta, t) d\eta. \tag{15}$$

The approximation discussed above is generalised to the nonhomogeneous case of (7) in a way described in [8] where the acceleration of the convergence rate for Sinc quadratures have been also discussed based on the solution of problem (7), (8) with a modified right hand side taken in the form $f^{(n)} = f_0 + tf_1 + ... + t^n f_n$, and the recursive computing of the integrals $I_k = \int_0^t e^{-\mathcal{L}^+(t-\psi)}\psi^k d\psi$ as

$$I_k = A^{-1}[t^k - kI_{k-1}], \quad k = 0, 1..., \quad I_0 = A^{-1}[I - e^{-At}]. \tag{16}$$

4 Approximating resolvents can be effectively parallelised

The step left to be considered in our algorithm is connected with the solution of (12). This step can be effectively parallelised.

Indeed, one way to do that is to use the idea of a projective methodology and solve the discrete problem:

$$(z_j - \mathcal{L}^+)\hat{u}_{\Delta x}(z_j) = P_{\Delta x} u_0, \tag{17}$$

where by $P_{\Delta x}$ we denote a projection operator. This leads to the fully discrete approximation to (7) (with an additional term in the non-homogeneous case):

$$u_{N,\Delta x}(t) = \sum_{j=-N}^{N} \alpha_j \hat{u}_{\Delta x}(z_j) e^{-z_j t}. \tag{18}$$

Note further that the resolvents can be approximated efficiently with the $\mathcal{H}$-matrix arithmetics. In particular, if M_j is the $\mathcal{H}$-matrix approximation of the resolvent $(z_j I - \mathcal{L}^+)$, we have the following representation [9]

$$u_{N,\mathcal{H}} = \sum_{j=-N}^{N} \alpha_j e^{-z_j t} M_j (u_0 + \int_0^t e^{z_j s} f(s) ds). \tag{19}$$

The computational cost of such $\mathcal{H}$-matrix approximations, effective in data-sparse cases, is close to linear (with the linear-logarithmic theoretical estimate [10]). The rest of the procedure remains the same as described above.

5 The Cayley transform and conservative numerical discretisations

The procedure described in the previous sections has the theoretical exponential rate of convergence [8], and several test examples have been used to investigate the convergence rate numerically. The Cayley transform can also be used to derive algorithms with no iterations for time-dependent problems employing the idea of *dynamic* adaptation of the computed solution to the smoothness of input data. This feature is quite important for the models considered in this paper. We recall that in its essence the procedure reported by us in [6] is closely connected with the procedure just described in a sense that both are based on inverting the operator $(zI - \mathcal{L}^+)$. More precisely, first of all under the assumption of strong P-positivity of operator $\mathcal{L}^+$ (see the complete set of assumption in Section 2 of [6]) we can apply the representation of the solution in terms of the semigroup T as in (9). Then, a natural way to preserve qualitative properties of the solution of the differential problem into the numerical discretisation is to use a one-to-one correspondence between T and T_γ^n. This would lead to a conservative numerical discretisation in a sense described in Section 2. Recall that we have

$$T(t) = \sum_{k=0}^{\infty} (-1)^k \varphi_k(2\gamma t) T_\gamma^k, \quad T_\gamma^k = (-1)^k \left[\int_0^\infty \psi_k(t) T(t/(2\gamma)) dt + I \right] \tag{20}$$

with functions $\varphi_k(t)$ and $\psi_k(t)$ specified in [6]. This connection is established on the basis of the solution of the discrete problem

$$y_\gamma^{n+1} = T_\gamma^n y_\gamma^n, \quad n = 0, 1, ..., \quad y_\gamma^0 = u_0, \tag{21}$$

which allows us to determine the operator action $T^N(t)u_0$ which, in its turn, allows us to find the solution to our convection-diffusion problem as

$$u^N(t) = T^N(t)u_0 + \sum_{k=0}^{N} (-1)^k T_\gamma^k (I + T_\gamma) \int_0^t e^{-\gamma(t-s)} L_k^{(0)}[2\gamma(t-s)] f(s) ds, \tag{22}$$

where

$$T^N(t)u_0 = e^{-\gamma t} \sum_{k=0}^{N} (-1)^k L_k^{(0)}[2\gamma t](y_\gamma^k + y_\gamma^{k+1}), \tag{23}$$

with $L_k^{(\alpha)}$ being the corresponding Laguerre polynomials.

As we showed in [6] the problem of finding y_γ^k is reducible to a class of projector operator equations, and the particular choice we use is

$$\tilde{y}_\gamma^{k+1} = (1 - \psi^2) \sum_{s=0}^{n-2} a_s^{(s)} \psi^s. \tag{24}$$

Under this choice boundary conditions will be satisfied by default if we include points $\psi_0 = -1$ and $\psi_n = 1$ into a set of collocation points which are chosen in this approach as the nodes of the Gauss-Lobatto quadrature formulae. These are the roots of the corresponding Gegenbauer ultraspherical polynomials $C_n^\lambda(\psi) = \dfrac{(2\lambda)_n}{(\lambda + 1/2)_n} P_n^{(\lambda-1/2,\lambda-1/2)}(\psi)$ with $(\alpha)_n = \Gamma(\alpha+n)/\Gamma(\alpha) = \alpha(\alpha+1)...(\alpha+n-1)$ and $P_n^{(\alpha,\beta)} = \dfrac{(-1)^n}{2^n n!}(1-\psi)^{-\alpha}(1+\psi)^{-\beta} \dfrac{d^n}{d\psi^n}[(1-\psi)^{n+\alpha}(1+\psi)^{n+\beta}]$, and they are determined in a recursive manner for the case $\lambda = 2$.

Then the following procedure is parallelisable in a straightforward manner

- Set k=0 and compute

$$F(\psi, \tilde{y}_\gamma^0) = -\mu(\tilde{y}_\gamma^0)'' - \tilde{\beta})(\tilde{y}_\gamma^0)' - [\tilde{\alpha} + \gamma]\tilde{y}_\gamma^0; \tag{25}$$

- Find the values $F(\psi_j, \tilde{y}_\gamma^0)$, $j = 1, ..., N$ (recall that we have $\tilde{y}_\gamma^k(\psi_0) = \tilde{y}_\gamma^k(\psi_n) = 0$);

- Using polynomial representations of $\tilde{y}_\gamma^1$ find the general form of $(\tilde{y}_\gamma^1(\psi))'$ and $(\tilde{y}_\gamma^1(\psi))''$ in terms of ψ;

- Solve the following system of linear equations

$$\mu(\tilde{y}_\gamma^1)''(\psi_j) + \tilde{\beta}(\tilde{y}_\gamma^1)'(\psi_j) + [\tilde{\alpha} + \gamma]\tilde{y}_\gamma^1(\psi_j) = F(\psi_j, \tilde{y}_\gamma^0), \quad j = 1, ..., N \tag{26}$$

  with respect to (n-1) unknowns $(a_0^{(n)}, a_1^{(n)}, ..., a_{n-2}^{(n)})$;

- Increase k: $k \to k + 1$, etc

This process has been implemented in MATLAB, and a series of numerical experiments have been conducted to investigate the theoretical convergence rate reported by us in [6], in particular in the view that the error of the algorithm can be estimated as

$$\sup_{t \in [0, T_0)} \|u^N(\cdot, t) - u(\cdot, t)\|_{D^\Theta} \leq cN^{-(\sigma - \Theta - \delta)} \left[\|(\mathcal{L}^+)^\sigma u_0\| + \|(\mathcal{L}^+)^\sigma f(t)\|_{L^1}\right] \tag{27}$$

which holds for all $\delta \in (0, \sigma - \Theta)$ with $\sigma > \Theta \geq 0$. Note that it can be shown that locally for each interval $[t_0, t_1] \subset (0, \infty)$ the power of N decreases for 0.25 with $\delta \in (0, \sigma - \Theta + 0.25)$ (see details in [6]). The numerical convergence depends also on parameter γ and our test examples show that an appropriate choice of γ can accelerate the convergence.

6 Parallelisation in time based on the Cayley-transform methodology and concluding remarks

The Cayley transform is a systematic way in developing efficient time-stepping algorithms. It is well suited for discretising such differential equations that evolve in some Lie groups, as we discussed in Section 2. The technique is part of geometric integration tools that provide a highly efficient and precise way of obtaining numerical solution, while retaining qualitative properties of the differential systems. Many well-known procedures follow as a special case from this general consideration (e.g., the standard diagonal Pade approximant is a product of Cayley transforms).

Since both approaches discussed in this paper lead to formulae (see (19) and (22)) that do not require iterations in time a parallelisation becomes natural as soon as the time marching is required.

Finally, note that the terms in the Cayley expansion can be associated with a set of planar rooted trees [11], the fact which opens further opportunities for exploiting the algorithmic parallelism of the procedure proposed here on the basis of the Cayley transform.

References

[1] S. Bertoluzza, C. Canuto, A. Tabacco, *Negative norm stabilization of convection-diffusion problems*, Appl. Math. Let., 13, (2000), 121-127.

[2] F. Brezzi, D. Marini, E. Suli, *Residual-free bubbles for advection-diffusion problems: the general error analysis*, Numer. Math., 85, (2000), 31–47.

[3] N.D. Buono, L. Lopez, *Numerical methods for Hermitian unitary differential systems*, J. Comput. Appl. Math., 111, (1999), 133–145.

[4] F. Diele, L. Lopez, T. Politi, *One step semi-explicit methods based on the Cayley transform for solving isospectral flows*, J. Comput. Appl. Math., 89, (1998), 219–223.

[5] R.E. Ewing, H.Wang, *A summary of numerical methods for time-dependent advection-dominated PDEs*, J. Comput. Appl. Math., 128, (2001), 423–445.

[6] I.P. Gavrilyuk, R.V.N. Melnik, *Constructive approximations of the convection-diffusion-absorption equation based on the Cayley transform technique*, in S.R. Idelsohn, E. Onate, E.N. Dvorkin, eds., *Computational Mechanics: New Trends and Applications. Proc. 4th World Congress on Computational Mechanics, 1998*, CIMNE, Barcelona (1998), pp.1–14.

[7] I.P. Gavrilyuk, *Strongly P-positive operators and explicit representatations of the solutions of initial value problems for second-order differential equations in Banach spaces*, J. Math. Analysis and Appl., 236, (1999), 327–349.

[8] I.P. Gavrilyuk, V.L. Makarov, *Exponentially convergent parallel discretization methods for the first order evolution equations*, Comput. Methods in Appl. Math., 1(4), (2001), 333-355.

[9] I.P. Gavrilyuk, W. Hackbusch, B.N. Koromskij, *H-matrix approximation for the operator exponential withj applications*, Numerische Math., (2002), to appear.

[10] W. Hackbusch, B.N. Khoromskij, *A sparse $\mathcal{H}$-matrix arithmetic: general complexity estimates*, J. Comp. Appl. Math., 125, (2000), 479–501.

[11] A. Iserles, *On Cayley-transform methods for the discretization of Lie-group equations*, Found. Comput. Math., 1, (2001), 129–160.

[12] A. Marthinsen, B. Owren, *Quadrature methods based on the Cayley transform*, Appl. Numer. Math., 39, (2001), 403–413.

[13] R.V.N. Melnik, *Topological analysis of eigenvalues in engineering computations*, Engineering Computations, 17, (2000), 386–416.

[14] A. Pazy, *Semigroup of linear operators and applications to partial differential equations*, Springer, N.Y. (1983).

[15] D. Sheen, I.H. Sloan, V. Thomee, *A parallel method for time-discretisation of parabolic problems based on contour integral representation and quadrature*, Math. Comp., 69, 2000, 177–195.

[16] F.E. Udwadia, H.F. von Bremen, *Computation of Lyapunov characteristic exponents for continuous dynamical systems*, ZAMP, 53, (2001), 123–146.