Available online at www.sciencedirect.com
SCIENCE DIRECT®
Mathematics and Computers in Simulation 67 (2004) 55–66
MATHEMATICS AND COMPUTERS IN SIMULATION
www.elsevier.com/locate/matcom

Analysis and optimization of inner products for mimetic finite difference methods on a triangular grid

Richard Liska [a,*], Mikhail Shashkov [b], Victor Ganzha [c]

[a] *Czech Technical University in Prague, Faculty of Nuclear Sciences and Physical Engineering, Břehová 7, 115 19 Prague 1, Czech Republic*
[b] *Los Alamos National Laboratory, Los Alamos, NM 87544, USA*
[c] *Department of Informatics, Technical University of Munich, Arcisstrasse 21, 80333 Munich, Germany*

Available online 20 July 2004

Abstract

The support operator method designs mimetic finite difference schemes by first constructing a discrete divergence operator based on the divergence theorem, and then defining the discrete gradient operator as the adjoint operator of the divergence based on the Gauss theorem connecting the divergence and gradient operators, which remains valid also in the discrete case. When evaluating the discrete gradient operator, one needs to define discrete inner products of two discrete vector fields. The local discrete inner product on a given triangle is defined by a 3×3 symmetric positive definite matrix M defined by its six independent elements–parameters. Using the Gauss theorem over our triangle, we evaluate the discrete gradient in the triangle. We require the discrete gradient to be exact for linear functions, which gives us a system of linear equations for elements of the matrix M. This system, together with inequalities which guarantee positive definiteness of the matrix M, results in a one parameter family of inner products which give exact gradients for linear functions. The traditional inner product is a member of this family. The positive free parameter can be used to improve another property of the discrete method. We show that accuracy of the method for quadratic functions improves with decreasing this parameter, however, at the same time, the condition number of the matrix M, which is the local matrix of the linear system for computing the discrete gradient, increases to infinity when the parameter goes to zero, so one needs to choose a compromise between accuracy and solvability of the local system. Our analysis has been performed by computer algebra tools which proved to be essential.

PACS: 02.30.Jr; 02.60.Lj; 02.70.Bf

Keywords: Support operator; Mimetic finite difference method; Triangular grid

* Corresponding author.
E-mail addresses: liska@siduri.fjfi.cvut.cz (R. Liska), shashkov@lanl.gov (M. Shashkov), ganzha@in.tum.de (V. Ganzha).
URLs: http://www-troja.fjfi.cvut.cz/∼liska, http://cnls.lanl.gov/∼shashkov.

0378-4754/$30.00 © 2004 Published by Elsevier B.V. on behalf of IMACS.
doi:10.1016/j.matcom.2004.05.008

1. Introduction

The *mimetic finite difference methods* [1–3] for discretizing partial differential equations take advantage of the fact that most partial differential equations of importance in mathematical physics and engineering can be formulated in terms of the invariant differential operators divergence, gradient, and curl. They provide a systematic approach to spatial differencing of partial differential equations by constructing discrete analogs of these invariant operators that *exactly* satisfy discrete analogs of important differential and integral identities satisfied by the invariant continuum operators. From the discrete identities, in direct analogy with the continuum, one can then derive *exact* discrete conservation laws and *exact* analogs of other important physical principles, which in turn assure the stability and robustness of these methods.

For example, for discretizing the Laplace equation the main steps are: choose a discretization of the scalar and vector fields; choose a discretization for the divergence (div); then choose discrete inner products for discrete scalar and vector fields; and then use a discrete analog of the *Divergence Theorem* to determine the discrete gradient (**grad**). The Divergence theorem says that:

$$\int_{\Omega} \operatorname{div} \vec{v}\, f \,\mathrm{d}V + \int_{\Omega} \vec{v}\, \mathbf{grad} f \,\mathrm{d}V = \int_{\partial\Omega} f\, \vec{v} \cdot \vec{n} \,\mathrm{d}S \tag{1}$$

where Ω is some smooth region, $\partial\Omega$ the boundary of the region, $\vec{n}$ an outward normal to the boundary, f a smooth scalar function defined on the closure of the region, and $\vec{v}$ a smooth vector field defined on the closure of the region (see [2]). So, if f and g are scalar fields and if $\vec{v}$ and $\vec{w}$ are vector fields, then relevant continuum inner products for scalars and vectors are:

$$\langle f, g\rangle = \int_{\Omega} f\, g \,\mathrm{d}V, \qquad \langle \vec{v}, \vec{w}\rangle = \int_{\Omega} \vec{v} \cdot \vec{w} \,\mathrm{d}V, \tag{2}$$

and then (1) can be written as:

$$\langle \operatorname{div} \vec{v},\, f\rangle + \langle \vec{v}, \mathbf{grad}\, f\rangle = \int_{\partial\Omega} f\, \vec{v} \cdot \vec{n} \,\mathrm{d}S \tag{3}$$

Previously, natural geometric ideas have been used to discretize these inner products, while standard finite-volumes are used to discretize the divergence. The discrete analog of the gradient is derived from the discrete analog of (3).

There are some theoretical and numerical results on how accuracy of the mimetic methods depends on accuracy of the inner products, [4,5]. However, this question require additional investigations. In particular, an important question is how the accuracy of the gradient depends on the accuracy of the inner products and the accuracy of the divergence. Another important practical question is how definition of the inner product affects the process of solving the linear system corresponding to mimetic discretization.

The global inner product in mimetic finite difference methods is assembled from inner products for each cell. In this paper, we are considering an inner product for one triangular cell. We analyze how the inner product affects the accuracy of the gradient and the condition number of the system of linear equations for the gradient. From a formal point of view, the inner product is a symmetric positive definite bilinear form. The analysis of the inner product is performed using symbolic manipulations.

The rest of the paper is organized as follows: Section 2 introduces discretization of scalar and vector functions on an unstructured triangular grid; in Section 3 standard inner products are defined and weights

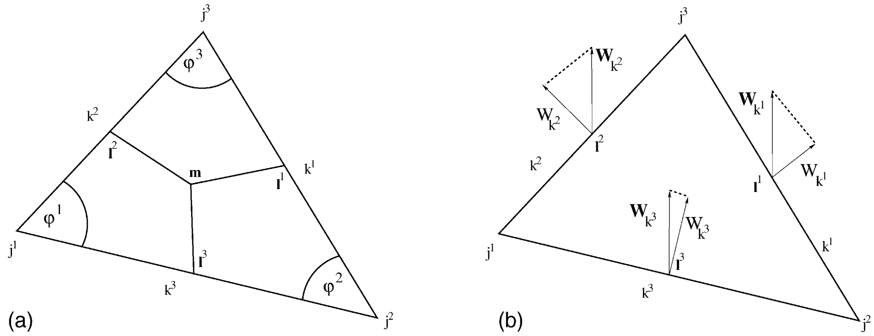

Fig. 1. (a) Triangle i and quantities related to it. (b) Components W_{k^n}, $n = 1, 2, 3$ of vector grid function $\boldsymbol{W}_{k^n}$, $n = 1, 2, 3$ are at the center of edges as projections to the edge normals.

appearing in them are derived; in Section 4 we analyze the general inner product; and in Section 5 we mention computer algebra tools used in our analysis.

2. Scalar and vector functions on a triangular grid

A triangular grid consists of N_t triangles. For numbering of the triangles, vertices and edges we will use indices i, j, k, respectively. Each triangle has three vertices j^n, $n = 1, 2, 3$, three edges k^n, $n = 1, 2, 3$, three midpoints of the edges l^n, $n = 1, 2, 3$, and at the vertices j^n the triangle has angle φ^n between two edges k^{n-2} and k^{n-1} as shown at Fig. 1(a). Wherever needed here and in the rest of the paper, cyclic extension for triangle quantities indexing is assumed, so that, e.g. index 0 means 3 or index 4 means 1. Each internal edge belongs to two triangles and each boundary edge belongs to only one triangle. The grid has N_eb boundary edges and N_e edges in total.

A scalar function is represented on a triangular grid by its value inside each triangle i and on each boundary edge k. A vector function is represented at the center of each edge by its projection on the edge normal as shown in Fig. 1(b).

3. Standard inner products

The natural inner product of scalar functions U, V on the space HC of scalar grid functions is defined by:

$$(U, V)_\mathrm{HC} = \sum_{i=1}^{N_\mathrm{t}} U_i V_i VC_i + \sum_{k=1}^{N_\mathrm{eb}} U_k V_k S_k$$

where U_i, V_i are values of scalar functions in the triangle i, U_k, V_k are values at the center of boundary edge k, VC_i is the area of triangle i and S_k is the length of the boundary edge k.

For the natural inner product of vector grid functions $\boldsymbol{A}$, $\boldsymbol{B}$ on the triangle i, we first move the normal projections of the vectors $\boldsymbol{A}$, $\boldsymbol{B}$ on two edges of the triangle into their common vertex and define a contribution from this vertex j^J to the inner product as:

$$(\boldsymbol{A},\boldsymbol{B})_{j^J} = \frac{A_{k^{J-2}}B_{k^{J-2}} + A_{k^{J-1}}B_{k^{J-1}} + (A_{k^{J-2}}B_{k^{J-1}} + A_{k^{J-1}}B_{k^{J-2}})\cos\varphi^J}{\sin^2\varphi^J}$$

The inner product at the triangle i is then given by:

$$(\boldsymbol{A},\boldsymbol{B})_i = \frac{1}{VC_i}\sum_{J=1}^{3}(\boldsymbol{A},\boldsymbol{B})_{j^J} VN_{j^J} \tag{4}$$

where VN_{j^J} are the unknown weights in the triangle i associated with the vertex j^J. One can easily verify that (4) really defines a inner product, it is symmetric, linear and positive for $\boldsymbol{A} = \boldsymbol{B} \neq \boldsymbol{0}$.

3.1. General cell

Before proceeding to weights VN_{j^J}, let's describe a general triangle on which we will be working. We consider a general triangle with vertices (x_l, y_l), $l = 1, 2, 3$ with the following triangle quantities:

- weights VN_l, $l = 1, 2, 3$ of vertex l appearing in the inner product (4); we require the sum of these weights to be the volume of the triangle $V = VN_1 + VN_2 + VN_3$;
- edge lengths s_l, $l = 1, 2, 3$ are given by $s_l = \sqrt{(x_{l+2} - x_{l+1})^2 + (y_{l+2} - y_{l+1})^2}$ (with cyclic extension $x_4 = x_1$, etc.);
- angles φ_l, $l = 1, 2, 3$ for which

$$\cos\varphi_l = \frac{(x_{l+1} - x_l)(x_l - x_{l+2}) + (y_{l+1} - y_l)(y_l - y_{l+2})}{s_{l+1}s_{l+2}};$$

- outer normals to the edges n_l, $l = 1, 2, 3$, $n_l = (n_{lx}, n_{ly})$

$$n_{lx} = \frac{y_{l+2} - y_{l+1}}{s_l},\ n_{ly} = \frac{x_{l+1} - x_{l+2}}{s_l};$$

- function values u_l, $l = 1, 2, 3$ at centers of edges;
- function value inside the triangle $u_{\mathrm{c}} = (u_1 + u_2 + u_3)/3$.

3.2. Weights evaluation

The divergence theorem applied to our triangle:

$$\int_V \operatorname{div} \boldsymbol{w}\,\mathrm{d}V = \oint_{\partial V} (\boldsymbol{w}, \boldsymbol{n})\,\mathrm{d}S$$

gives us the operator DIV of discrete divergence:

$$\mathrm{DIV}\,\boldsymbol{w} = \frac{1}{V}\sum_{l=1}^{3} w_l s_l$$

The Gauss theorem applied to our triangle:

$$\int_V u \operatorname{div} \boldsymbol{w}\mathrm{d}V + \int_V (\boldsymbol{w}, \mathbf{grad}\, u)\mathrm{d}V = \oint u(\boldsymbol{w}, \boldsymbol{n})\,\mathrm{d}S \tag{5}$$

with traditional inner product (4) results in:

$$
\begin{aligned}
&u_{\rm c}(w_1s_1+w_2s_2+w_3s_3)+\frac{VN_1}{\sin^2\varphi_1}[G_2w_2+G_3w_3+(G_2w_3+G_3w_2)\ \cos\varphi_1]\\
&\frac{VN_2}{\sin^2\varphi_2}[G_1w_1+G_3w_3+(G_1w_3+G_3w_1)\ \cos\varphi_2]\\
&\frac{VN_3}{\sin^2\varphi_3}[G_1w_1+G_2w_2+(G_1w_2+G_2w_1)\ \cos\varphi_3]=u_1w_1s_1+u_2w_2s_2+u_3w_3s_3
\end{aligned}
\tag{6}
$$

where $\boldsymbol{G}_l = (\mathbf{grad}\, u, \boldsymbol{n}_l)$, $w_l = (\boldsymbol{w}, \boldsymbol{n}_l)$, $l = 1, 2, 3$ are projections of $\mathbf{grad}\, u$ and $\boldsymbol{w}$ to the outer normals of the edges of the triangle. This Eq. (6) holds for any vector $\boldsymbol{w}$, for any w_l, $l = 1, 2, 3$ so the coefficient of each w_l in (6) has to be zero:

$$
\begin{aligned}
&(u_{\rm c}-u_1)s_1+\frac{VN_2}{\sin^2\varphi_2}(G_1+G_3\ \cos\varphi_2)+\frac{VN_3}{\sin^2\varphi_3}(G_1+G_2\ \cos\varphi_3)=0\\
&(u_{\rm c}-u_2)s_2+\frac{VN_1}{\sin^2\varphi_1}(G_2+G_3\ \cos\varphi_1)+\frac{VN_3}{\sin^2\varphi_3}(G_2+G_1\ \cos\varphi_3)=0\\
&(u_{\rm c}-u_3)s_3+\frac{VN_1}{\sin^2\varphi_1}(G_3+G_2\ \cos\varphi_1)+\frac{VN_2}{\sin^2\varphi_2}(G_3+G_1\ \cos\varphi_2)=0
\end{aligned}
\tag{7}
$$

We require producing exact gradients for linear functions u by our method. So, we try to find unknown weights V_l, $l = 1, 2, 3$ so that Eq. (7) are fulfilled for two particular linear functions $u = X$ and $u = Y$ (we use capital (X, Y) to denote our coordinate system to distinguish from parameters x, y used later). For the function $u = X$, the exact values of the gradient projected on the outer normal are $G_l = n_{lx}$, $l = 1, 2, 3$; the values of u at the centers of edges are $u_l = (x_{l+1} + x_{l+2})/2$. After substitutions of these and the above mentioned expressions for the normals $\boldsymbol{n}_l$, edge lengths s_l and angles φ_l into (7), we obtain a system of three linear equations for the three weights VN_l with only six parameters x_l, y_l, $l = 1, 2, 3$. Doing the same for the function $u = Y$ with gradient projections $G_l = n_{ly}$, $l = 1, 2, 3$ gives us another three linear equations for the weights VN_l. So, in total we have a system of six linear equations for the three unknown weights VN_l, $l = 1, 2, 3$. Fortunately these equations are dependent, leaving after elimination three linearly independent equations with unique solution:

$$
VN_l=\frac{1}{6}(x_1(y_2-y_3)+x_2(y_3-y_1)+x_3(y_1-y_2))=\frac{V}{3}, l=1,2,3
$$

So, the value of all the weights is the same, namely one-third of the volume of the triangle. At the same time, the above requirement that the volume of the triangle is equal to the sum of the weights is fulfilled automatically. Deriving the system of six linear equations and solving it is quite an easy task for a computer algebra system.

4. General inner product of vector functions

We have the standard inner product for vector functions with the weights computed in the previous section yielding the standard support operator (SO) method which is exact for linear functions. There might, however, exist another inner product resulting in another SO method which is also exact for linear

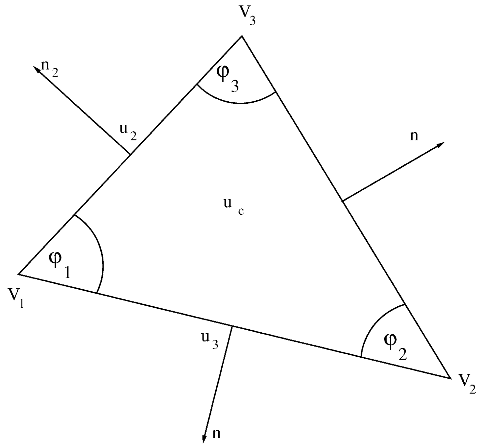

Fig. 2. General triangle and its local quantities.

functions and has better properties than the standard SO method. So, let's try to derive a general inner product which gives a SO method which is exact on linear functions.

A general discrete inner product of vector grid functions $\boldsymbol{A}$, $\boldsymbol{B}$ on a triangle can be written as:

$$(\boldsymbol{A}, \boldsymbol{B})_M = (M \cdot \boldsymbol{A}) \cdot \boldsymbol{B} \tag{8}$$

where M is a symmetric positive definite matrix:

$$M = \begin{pmatrix} m_{11} & m_{12} & m_{13} \\ m_{12} & m_{22} & m_{23} \\ m_{13} & m_{23} & m_{33} \end{pmatrix} \tag{9}$$

Applying the Gauss theorem (5) to our triangle as in the previous section, we obtain an analog of equation (6) involving the matrix M elements m_{ln} instead of coefficients involving the angles φ_l. As again this equation is a linear combination of three components of the arbitrary vector $\boldsymbol{w}$, it has to hold for any values of these components, so the three coefficients of these components must be zero giving us the three equations:

$$\begin{aligned} (u_c - u_1)s_1 + V(m_{11}G_1 + m_{12}G_2 + m_{13}G_3) = 0, \\ (u_c - u_2)s_2 + V(m_{12}G_1 + m_{22}G_2 + m_{23}G_3) = 0, \\ (u_c - u_3)s_3 + V(m_{13}G_1 + m_{23}G_2 + m_{33}G_3) = 0, \end{aligned} \tag{10}$$

where again $G_l, l = 1, 2, 3$ are projections of the gradient $\mathbf{grad}\, u$ on the normals to the edges of the triangle: $G_l = (\mathbf{grad}\, u, \boldsymbol{n}_l), l = 1, 2, 3$. In the following, we will use a special Cartesian coordinate system (X, Y) to describe our general triangle shown in Fig. 2 as simply as possible. First we choose edge 3 as the longest edge of the triangle and put the X coordinate axis on this edge, so that the vertex 3 is above the X-axis, i.e. it has positive Y coordinate. Now we put the left vertex 1 on the X-axis at the origin. In such a moved and rotated coordinate system, the three vertices of the triangle have coordinates $(0, 0)$, $(X_2, 0)$, (X_3, Y_3). For the remaining three parameters defining the triangle, we denote

$X_3 = x$, $Y_3 = y$, $X_2 = z$, so that the triangle has vertices $(0, 0)$, $(z, 0)$, (x, y). From our construction, it follows that $0 \le x \le z$, $0 < y \le z$ (z is the length of the longest edge). The volume of this triangle is $V = yz/2$, the edge lengths are $s_1 = \sqrt{(x-z)^2 + y^2}$, $s_2 = \sqrt{x^2 + y^2}$, $s_3 = z$ and the outer normals are $\boldsymbol{n}_1 = (y/s_1, (z-x)/s_1)$, $\boldsymbol{n}_2 = (-y/s_2, x/s_2)$, $\boldsymbol{n}_3 = (0, -1)$.

Using the above values in (10) with u_c being the average of the three values u_l, $l = 1, 2, 3$ on the edges, we obtain for the two linear cases $u = X$ and $u = Y$ the system:

$$
\begin{aligned}
&-(x+z)s_1 + 3yz(m_{11}n_{1x} + m_{12}n_{2x}) = 0\\
&(-x+2z)s_2 + 3yz(m_{12}n_{1x} + m_{22}n_{2x}) = 0\\
&(2x-z) + 3y(m_{13}n_{1x} + m_{23}n_{2x}) = 0\\
&-ys_1 + 3z(m_{11}n_{1x}(z-x) - m_{12}n_{2x}x - m_{13}y) = 0\\
&-ys_2 + 3z(m_{12}n_{1x}(z-x) - m_{22}n_{2x}x - m_{23}y) = 0\\
&2y + 3(m_{13}n_{1x}(z-x) - m_{23}n_{2x}x - m_{33}y) = 0
\end{aligned}
$$

with additional constraints defining parameters s_1, s_2, n_{1x}, n_{2x}:

$$
\begin{aligned}
&s_1^2 - (x-z)^2 - y^2 = 0\\
&s_2^2 - x^2 - y^2 = 0\\
&n_{1x}^2(y^2 + (x-z)^2) - y^2 = 0\\
&n_{2x}^2(y^2 + x^2) - y^2 = 0
\end{aligned}
$$

So, we have a system of six linear equations for the six unknowns $m_{11}, m_{12}, m_{13}, m_{22}, m_{23}, m_{33}$. One of the equations in this system is linearly dependent on the others and the system has the general solution:

$$
\begin{aligned}
m_{11} &= \frac{((x-2z)x + y^2 + z^2)(x+z)s_2 + 3m_{12}s_1y^2z}{3s_2y^2z}\\
m_{13} &= \frac{-((y+z)(y-z) + x^2)s_1s_2 + 3m_{12}y^2z^2}{3s_2y^2z}\\
m_{22} &= \frac{-(x^2+y^2)(x-2z)s_1 + 3m_{12}s_2y^2z}{3s_1y^2z}\\
m_{23} &= \frac{-(x^2 - 2xz + y^2)s_1s_2 + 3m_{12}y^2z^2}{3s_1y^2z}\\
m_{33} &= \frac{((x-z)x + y^2 + z^2)s_1s_2 + 3m_{12}y^2z^2}{3s_1s_2y^2}
\end{aligned}
\qquad (11)
$$

with m_{12} remaining as a free parameter. Again, deriving and solving this system is an easy task for computer algebra. One does need to be careful using additional constraints, and making the final results compact is a bit tricky.

The matrix M (9) has to be positive definite, otherwise (8) would not be an inner product. The Sylvester criterion for positive definiteness of the matrix M produces the inequalities:

$$
\begin{aligned}
&m_{11} > 0\\
&m_{11}m_{22} - m_{12}^2 > 0\\
&m_{11}m_{22}m_{33} + 2m_{12}m_{23}m_{13} - m_{22}m_{13}^2 - m_{11}m_{23}^2 - m_{33}m_{12}^2 > 0
\end{aligned}
$$

We substitute the above general solution into these three inequalities, which after simplification involving the definitions of s_1 and s_2, factorize. Leaving out positive factors, the simplified inequalities are:

$$x + z + 3M_{12}z > 0 \tag{12}$$

$$-(x + z)(x - 2z) + 9M_{12}z^2 > 0 \tag{13}$$

$$-y^2 - (x + z)(x - 2z) + 9M_{12}z^2 > 0 \tag{14}$$

where $M_{12} = m_{12}y^2/(s_1s_2)$. It is obvious that the third inequality implies the second one (14) ⇒ (13). To prove that the second inequality implies the first one (13) ⇒ (12), we have used quantifier elimination[1] by QEPCAD [6,7] which proved that:

$$\forall M_{12}\forall x\forall z[(13) \wedge (0 \le x \le z) \wedge z > 0] \Rightarrow (12)$$

is true. We now have (14) ⇒ (13) ⇒ (12), so all three inequalities hold if and only if the third inequality (14) holds. The inequality (14) defines a minimal value $m_{12}^{\min}$ of the parameter m_{12}, so we have ($M_{12} = m_{12}y^2/(s_1s_2)$):

$$m_{12} > m_{12}^{\min} = \frac{s_1s_2}{9z^2}\left(1 + \frac{(x + z)(x - 2z)}{y^2}\right)$$

We introduce a positive parameter:

$$d = m_{12} - m_{12}^{\min} > 0$$

yielding a family of inner products depending on the parameter $d > 0$ which all produce an exact gradient for linear functions. We can try to use this free parameter to improve some other property of the support operator method defined by this inner product. The standard support operator (SSO) inner product described in the Section 3 belongs to this family with:

$$m_{12}^{\mathrm{SSO}} = \frac{s_1s_2}{3z^2}\left(1 + \frac{x(x - z)}{y^2}\right)$$

which gives us the corresponding value of the parameter d:

$$d^{\mathrm{SSO}} = \frac{2s_1s_2}{9z^2}\left(1 + \frac{x^2 - xz + z^2}{y^2}\right)$$

[1] Quantifier elimination (QE) is a procedure which transforms the formula:

$$Q_1x_1 \in R,\ Q_2x_2 \in R, \ldots, Q_kx_k \in R, \quad F(x_1, \ldots, x_m),$$

where $m \ge k$, $Q_i, i = 1, \ldots, k$ are quantifiers ∀ (for all) or ∃ (there exists) and F is an arbitrary logical combination of polynomial equations and inequalities in the real variables $x_1, \cdots, x_m$, into the equivalent formula which does not contain any quantifier and contains only non-quantified variables $x_{k+1}, \cdots, x_m$ and is again a logical combination of polynomial equations and inequalities.

4.1. Better accuracy

The first property which we would like to improve by changing the parameter d is the accuracy, and one improvement in the accuracy would be the exact resolution of gradients of higher order terms starting with quadratic functions. We have tried to repeat our analysis with the matrix elements (11) for two quadratic functions $u = X^2$, $u = Y^2$ and a bilinear function $u = XY$, however, we found that there is no inner product in our family which would be exact for quadratic or bilinear functions. So there is no inner product which would allow the SO method to exactly resolve the gradients of both linear and quadratic functions.

On the other hand, we can try to improve the approximation error of the SO method. We know the exact gradient $\boldsymbol{G}^{\mathrm{e}}$ and we can symbolically evaluate the approximate gradient $\boldsymbol{G}^{\mathrm{a}}$ from equations derived from the Gauss theorem (10) with matrix M elements given by (11). The gradient error is $\boldsymbol{R} = \boldsymbol{G}^{\mathrm{e}} - \boldsymbol{G}^{\mathrm{a}}$, and we evaluate its norm in the inner product (8) given by matrix M with elements (10) as:

$$||\boldsymbol{R}||_M^2 = (\boldsymbol{R}, \boldsymbol{R})_M = (M \cdot \boldsymbol{R}) \cdot \boldsymbol{R}$$

To simplify the formulas, we will here and in the following consider a special case of our triangle with $z = 1$, so the triangle vertices are $(0, 0)$, $(1, 0)$, $(\hat{x}, \hat{y})$ with the longest edge on the X-axis (i.e., we scale the triangle $(0, 0)$, $(z, 0)$, (x, y) by $1/z$, so that $\hat{x} = x/z$, $\hat{y} = y/z$) and in the following we will skip the hats over x, y, so that our triangle now is $(0, 0)$, $(1, 0)$, (x, y) with $0 \le x \le 1$, $0 < y \le 1$. Any triangle can be transformed to this triangle by translation, rotation and scaling.

For the quadratic function $u = X^2$, the norm of the gradient error, after performing all the necessary symbolic computation, is:

$$\begin{aligned}||\boldsymbol{R}||_M^2 = {} & \frac{1}{36 s_1^2 s_2 y^2}\{36 d s_1 y^4 + s_2[3(3(9y^2 + 2 + 2s_2^2)s_2^2 - (57y^2 + 23)y^2)s_1^2 \\ & -(s_2^4 - 2s_2^2 + 4y^2 + 1)(9s_2^2 - 85y^2)]\}\end{aligned}$$

The gradient error is linear in the parameter d with a positive coefficient for this parameter, so it is linearly increasing with d. For the quadratic function $u = Y^2$, we get a similar result with the gradient error norm $||\boldsymbol{R}||_M^2$ increasing linearly with d. For the bilinear function $u = XY$, the gradient error is:

$$||\boldsymbol{R}||_M^2 = \frac{(s_2^2 - 3y^2)(s_1^2 - 3y^2) + 6y^2}{4y^2}$$

and does not depend on the parameter d. In general, the parameter $d > 0$ is positive and to achieve better accuracy on quadratic functions (i.e., better approximation of second derivatives), we should choose very small d.

4.2. Condition number of the local matrix

Solving the equations (10) for the gradient components G_l, $l = 1, 2, 3$, we need to invert the matrix M (11). Let us look at the condition number of the matrix M which should give us some measure of how easy or hard it should be to numerically solve a linear system involving M. The condition number of M is defined as:

$$C(M) = ||M|| \cdot ||M^{-1}||$$

The condition number of any matrix is greater than or equal to one and will be big for an ill-conditioned matrix. To evaluate the condition number, we use the matrix Frobenius norm:

$$||M||^2 = \sum_{i,j} m_{ij}^2$$

Now the issue is how the condition number $C(M)$ of the matrix (11) depends on the free parameters m_{12} and d? After performing all the necessary symbolic computation, we find:

$$||M||^2 = a_2 d^2 + a_1 d + a_0$$
$$||M^{-1}||^2 = \frac{b_2}{d^2} + \frac{b_1}{d} + b_0$$
$$C^2(M) = (a_2 d^2 + a_1 d + a_0)\left(\frac{b_2}{d^2} + \frac{b_1}{d} + b_0\right)$$

where the coefficients $a_n, b_n, n = 1, 2, 3$ are rational functions depending only on x, y, s_1 and s_2 (we are still using the triangle with vertices (0, 0), (1, 0), (x, y)). It is important to note that the coefficients a_2 and b_2 are positive, so that for very small and very large d the condition number grows without bound. As an example, we present the special case for the triangle with $x = 0$, $y = 1$, $z = 1$ for which the condition number is:

$$C^2(M) = \frac{2}{81}(324d^2 + 9\sqrt{2}d + 53)\frac{810d^2 + 18\sqrt{2}d + 25}{162d^2}$$

Fig. 3 shows for this triangle the dependence of the condition number $C^2(M)$ on the free parameter d, with the value of the parameter d for standard inner product of this triangle $d^{\mathrm{SSO}} = 4\sqrt{2}/9 \approx 0.6285$. We see that for the traditional inner product $C^2(M)|_{d=d^{\mathrm{SSO}}} \approx 5.1$ while $C^2(M)$ has minimum value $C^2(M)|_{d=d_0} \approx 3.9$ at $d_0 \approx 0.27$. So the matrix (11) is better conditioned for d_0 than for d^{SSO} and according to the accuracy analysis in the previous section it should also produce a more accurate solution.

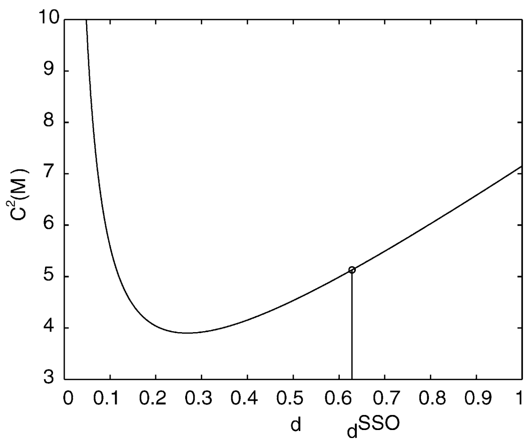

Fig. 3. Dependence of the condition number $C^2(M)$ on the free parameter d for the triangle with vertices (0, 0), (1, 0), (0, 1) with the value of d^{SSO} for the standard inner product marked.

Numerical solution of the linear system involving M behaves best for a minimal condition number $C(M)$. To minimize $C(M)$, we can try to solve:

$$\frac{\mathrm{d}\, C^2(M)}{\mathrm{d}d} = 0$$

for the free parameter d, however, it does not have a symbolic solution. The second derivative of $C^2(M)$ is:

$$\frac{\mathrm{d}^2 C^2(M)}{\mathrm{d}\, d^2} = \frac{4c_0 + 12 s_1 s_2 c_1 d + 17496 y^4 c_4 d^4}{2187 d^4 s_1^4 s_2^4 y^4} \tag{15}$$

with coefficients c_0, c_1 and c_4 being reasonably dense polynomials in x and y only, with total degrees 20, 16 and 8, respectively, which cannot be factored. Using quantifier elimination by QEPCAD, we have proved that for any x, y all three coefficients c_0, c_1, c_4 are non-negative, and as $d > 0$, also the second derivative (15) is always non-negative. Therefore, the first derivative is monotonically increasing from negative to positive values and has one root d_0 for $d > 0$. This means that the condition number $C^2(M)$ has only one local minimum for $d > 0$ which will be its global minimum (for $d > 0$) and numerical minimization of $C^2(M)$ must converge to this global minimum.

5. Computer algebra aspects

Computer algebra systems have allowed us to perform tedious algebra during our analysis. Simple tasks which we performed, such as simplification, substitution or solution of a system of linear equations are supported by any general computer algebra system (we have used Reduce [8] and Maple [9]).

More complicated is the treatment of inequalities which are not supported well in general computer algebra systems yet. Many problems involving inequalities as equivalence of inequalities or proving inequalities can be easily stated as quantifier elimination (QE) problems as mentioned earlier. We have used this approach in our analysis and quantifier elimination has provided us sufficient capabilities for treating inequalities. For QE, we have used the program Quantifier Elimination by Partial Cylindrical Algebraic Decomposition (QEPCAD), written by Hong and co-workers [6,7], which is the best complete QE program implemented up to now. The only problem with using this general QE program is its very high complexity (it is double exponential in the number of variables) which limits its usage to problems involving only few variables. However, we often succeeded in transforming our QE problems into QE problems with less variables which were solvable by QEPCAD. More information on QE can be found in [10] and on its applications in [11].

6. Conclusion

We have shown that the weights in the standard inner product of vector grid functions need to be equal to one-third of the volume of the triangle, otherwise the support operator method will not produce exact gradients for linear functions. The general inner product of vector grid functions has been proposed and analyzed. The requirement of producing exact gradients for linear functions by the support operator method restricts the general inner product to a one parameter family of inner products. The positive free parameter can be used to improve another property of the discrete method. We have shown that the accuracy of the method for quadratic functions improves with a decrease in this parameter, however, at the same time, the condition number of the matrix M, the local matrix of the linear system for computing the discrete gradient, increases to infinity when the parameter goes to zero. On the other hand, there exists a unique minimum of the condition number for which the matrix M will be conditioned best from the family of inner products.

We have shown that by choosing an appropriate inner product, the accuracy and the condition number of the local system of equations for the gradient can be improved when compared with the standard inner product. However, a more important issue for us is not how much the gradient improves, but how much the inner product can improve the accuracy of the solution of Laplace's equation and how it will affect properties of the global system of linear equations. Another question for us is what is the best way to create a vector inner product for a quadrilateral in 2D and logical bricks in three dimensions. These questions will be addressed in our future work by using a combination of numerical and analytical tools.

Acknowledgements

This research has been partially supported by International Bureau of the BMBF and Czech Ministry of Education in the framework of German-Czech Joint Research Project nr. CZE-00-010 and by the Project nr. ME 436/2001 from the program Kontakt of Czech Ministry of Education. The work of M. Shashkov and partially of R. Liska was performed under auspices of the US Department of Energy under Contract W-7405-ENG-36and DOE/BES Program in the Applied Mathematical Sciences Conract KC-07-01-01. R. Liska would like to thank the Institute for Geophysics and Planetary Physics (IGPP) of the Los Alamos National Laboratory for hosting his visits at Los Alamos National Laboratory. We thank anonymous referees for useful comments.

References

[1] J.M. Hyman, M. Shashkov, Natural discretizations for the divergence, gradient, and curl on logically rectangular grids, Int. J. Comput. Math. Appl. 33 (4) (1997) 81–104.
[2] J.M. Hyman, M. Shashkov, The adjoint operators or the natural discretizations for the divergence, gradient, and curl on logically rectangular grids, IMACS J. Appl. Numer. Math. 25 (1997) 413–442.
[3] V. Ganzha, R. Liska, M. Shashkov, C. Zenger, Support operator method for Laplace equation on unstructured triangular grid, Selcuk J. Appl. Math. 3 (1) (2002) 21–48.
[4] M. Berndt, K. Lipnikov, J.D. Moulton, M. Shashkov, Convergence of mimetic finite difference discretizations of the diffusion equation, East–West J. Numer. Math. 9 (4) (2001) 253–316.
[5] J. Hyman, M. Shashkov, S. Steinberg, The effect of inner products for discrete vector fields on the accuracy of mimetic finite difference methods, Comput. Math. Appl. 42 (2001) 1527–1547.
[6] H. Hong, Improvements in CAD-based quantifier elimination, Ph.D. Thesis, The Ohio State University, 1990.
[7] G.E. Collins, H. Hong, Partial cylindrical algebraic decomposition for quantifier elimination, J. Symb. Comput. 12 (3) (1991) 299–328.
[8] A.C. Hearn, REDUCE User's Manual, version 3.7, Tech. Rep., RAND, Santa Monica, 1999.
[9] B. Char, K. Geddes, G. Gonnet, B. Leong, M. Monagan, S. Watt, First Leaves: A Tutorial Introduction to Maple V, Springer-Verlag, New York, 1992.
[10] B. Caviness, J. Johnson (Eds.), Quantifier Elimination and Cylindrical Algebraic Decomposition, Springer-Verlag, Berlin, 1996.
[11] H. Hong, R. Liska (Eds.), J. Symb. Comput. Appl. Quantif. Eliminat. (Special Issue) 24 (2) (1997).