Analysis and Optimization of Inner Products for Mimetic Finite Difference Methods on 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,* `liska@siduri.fjfi.cvut.cz`, `http://www-troja.fjfi.cvut.cz/~liska`

[b] *Los Alamos National Laboratory, Los Alamos, New Mexico 87544, USA,* `shashkov@lanl.gov`, `http://cnls.lanl.gov/~shashkov`

[c] *Department of Informatics, Technical University of Munich, Arcisstrasse 21, 80333 Munich, Germany,* `ganzha@in.tum.de`

Abstract

The support operator method designs mimetic finite difference schemes by first constructing discrete divergence operator based on divergence theorem and then defining the discrete gradient operator as the adjoint operator of the divergence based on the Gauss theorem connecting operators divergence and gradient which remains valid also in the discrete case. During evaluating the discrete gradient operator one needs to define discrete inner products of two discrete vector fields. Local discrete inner product on a given triangle is defined by 3×3 symmetric positive definite matrix M defined by its six independent elements - parameters. Using Gauss theorem over our triangle we evaluate discrete gradient in the triangle. We require the discrete gradient being exact for linear functions, which gives us system of linear equations for elements of matrix M. This system together with inequalities which quarantee positive definiteness of matrix M results in one parameter family of inner products which give exact gradients for linear functions and traditional inner product is a member of this family. This 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 matrix M, which is the local matrix of 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.

Key words: support operator, mimetic finite difference method, triangular grid
PACS: 02.30.Jr, 02.60.Lj, 02.70.Bf

---

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 the invariant differential operators divergence, gradient, and curl. It provides 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 assures the stability and robustness of the method.

For example, for discretization of 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

$$\int_{\Omega} \mathrm{div}\vec{v}\, f\, dV + \int_{\Omega} \vec{v}\, \mathbf{grad} f\, dV = \int_{\partial\Omega} f\, \vec{v}\cdot\vec{n}\, dS\,, \tag{1}$$

where Ω is some smooth region, $\partial\Omega$ is the boundary of the region, $\vec{n}$ is an outward normal to the boundary, f is a smooth scalar function defined on the closure of the region, and $\vec{v}$ is 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

$$\langle f, g\rangle = \int_{\Omega} f\, g\, dV\,, \quad \langle \vec{v}, \vec{w}\rangle = \int_{\Omega} \vec{v}\cdot\vec{w}\, dV\,, \tag{2}$$

and then (1) can be written

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

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 the (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, 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 affect process of the solution of linear system corresponding to mimetic discretization.

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

The rest of the paper is organized as follows: the section 2 introduces discretization of scalar and vector functions on unstructured triangular grid, in the section 3 standard inner products are defined and weights appearing in them are derived, in the section 4 we analyze the general inner product and in the section 5 we mention computer algebra tools used in our analysis.

2 Scalar and Vector Functions on Triangular Grid

Triangular grid consists of N_t triangles. For numbering of triangles, edges and vertices we will use indices i, k, j respectively. Each triangle i has three vertices $j_i^n, n = 1, 2, 3$, three edges $k_i^n, n = 1, 2, 3$, three centers of the edges $l_i^n, n = 1, 2, 3$ and at the vertices j_i^n the triangle has angle $\varphi_{k_i^{n-1}}^{k_i^n}$ between two edges k_i^n and k_i^{n-1} as shown at Fig. 1(a). 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

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

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

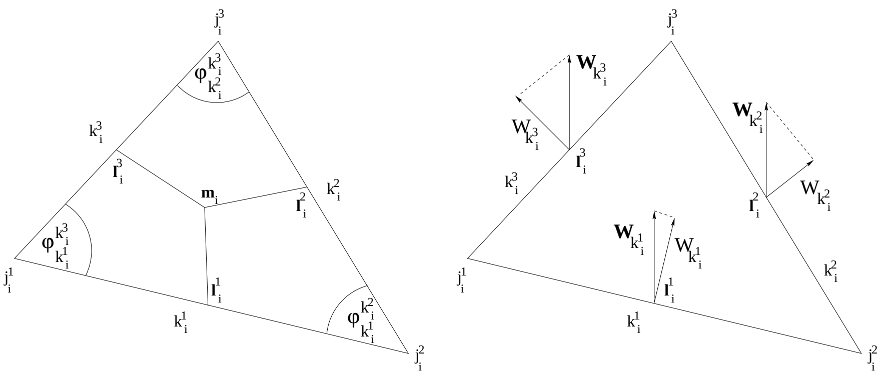
where U_i, V_i are values of scalar functions in the triangle i, U_k, V_k are values at the center of boundary edge k, VC_i is the volume of triangle i and S_k is the length of the boundary edge k.

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

$$(\mathbf{A},\mathbf{B})_{j_i^J} = \frac{A_{k_i^J} B_{k_i^J} + A_{k_i^{J-1}} B_{k_i^{J-1}} + (A_{k_i^J} B_{k_i^{J-1}} + A_{k_i^{J-1}} B_{k_i^J}) \cos \phi_{k_i^J}^{k_i^{J-1}}}{\sin^2 \phi_{k_i^J}^{k_i^{J-1}}}$$

The inner product at the triangle i is then given by

$$(\mathbf{A},\mathbf{B})_i = \frac{1}{VC_i} \sum_{J=1}^{3} (\mathbf{A},\mathbf{B})_{j_i^J} V_{j_i^J} \qquad (4)$$



(a) (b)

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

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

3.1 General Cell

Before proceeding to weights $V_{j_i^J}$ lets 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 following triangle quantities:

- weights $V_l, l = 1, 2, 3$ of vertex l appearing in the inner product (4); we require the sum of these weigths begin the volume of the triangle $V = V_1 + V_2 + V_3$
- 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 $\varphi_l, l = 1, 2, 3$ 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 $u_c = (u_1 + u_2 + u_3)/3$

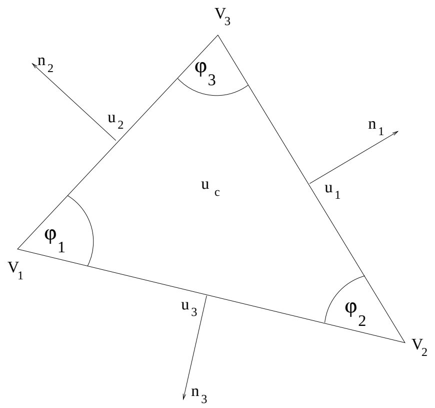


Fig. 2. General triangle and its local quantities

3.2 Weights Evaluation

The divergence theorem applied to our triangle

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

gives us the operator DIV of discrete divergence

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

The Gauss theorem applied to our triangle

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

with traditional inner product (4) results in

$$\begin{aligned}
& u_c(w_1 s_1 + w_2 s_2 + w_3 s_3) + \frac{V_1}{\sin^2 \varphi_1}[G_2 w_2 + G_3 w_3 + (G_2 w_3 + G_3 w_2)\cos\varphi_1] \\
& \quad \frac{V_2}{\sin^2 \varphi_2}[G_1 w_1 + G_3 w_3 + (G_1 w_3 + G_3 w_1)\cos\varphi_2] \\
& \quad \frac{V_3}{\sin^2 \varphi_3}[G_1 w_1 + G_2 w_2 + (G_1 w_2 + G_2 w_1)\cos\varphi_3] \\
& \qquad = u_1 w_1 s_1 + u_2 w_2 s_2 + u_3 w_3 s_3
\end{aligned} \tag{6}$$

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

$$\begin{aligned}
(u_c - u_1)s_1 + \frac{V_2}{\sin^2\varphi_2}(G_1 + G_3\cos\varphi_2) + \frac{V_3}{\sin^2\varphi_3}(G_1 + G_2\cos\varphi_3) &= 0 \\
(u_c - u_2)s_2 + \frac{V_1}{\sin^2\varphi_1}(G_2 + G_3\cos\varphi_1) + \frac{V_3}{\sin^2\varphi_3}(G_2 + G_1\cos\varphi_3) &= 0 \\
(u_c - u_3)s_3 + \frac{V_1}{\sin^2\varphi_1}(G_3 + G_2\cos\varphi_1) + \frac{V_2}{\sin^2\varphi_2}(G_3 + G_1\cos\varphi_2) &= 0
\end{aligned} \tag{7}$$

We require to get exact gradients for linear functions u by our method. So we try to find unknown weights $V_l, l = 1, 2, 3$ so that equations (7) are fullfilled for two particular linear funcitons $u = X$ and $u = Y$ (we use capital (X, Y) to denote our coordinate system to distinquish from parameters x, y used later). For 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 normals $\mathbf{n}_l$, edge length s_l and angles φ_l into (7) we obtain a system of three linear equations for three weights V_l with only six parameters $x_l, y_l, l = 1, 2, 3$. Doing the same for function $u = Y$ with gradient projections $G_l = n_{ly}, l = 1, 2, 3$ gives us another three linear equations for the weights V_l. So in total we have a system of six linear equations for three unknown weights $V_l, l = 1, 2, 3$. Fortunatelly these equations are dependent, leaving after elimination three linearly independent equations with unique solution

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

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 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 of vector functions with the weights computed in the previous section which gives 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 functions and has better properties than the standard SO method. So lets 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 $\mathbf{A}, \mathbf{B}$ on a triangle can be written as

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

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 the equation (6) involving matrix M elements m_{ln} instead of coefficients involving the angles φ_l. And again this equation is linear combination of three components of the arbitrary vector $\mathbf{w}$, it has to hold for any values of these components, so three coefficients of these components must be zero what give us 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 projection of the gradient $\mathbf{grad}u$ to the normals of the edges of the triangle $G_l = (\mathbf{grad}\, u, \mathbf{n}_l), l = 1,2,3$.

In the following we will use a special cartesian coordinate system (X, Y) to describe our general triangle shown on Fig. 2 as simple as possible. First we choose the 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. has positive Y coordinate. Now we put the left vertex 1 on the axis X into the origin. In such 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 use denotation $X_3 = x, Y_3 = y, X_2 = z$, so that the triangle has vertices $(0,0), (z,0), (x,y)$. From our construction follows that $0 \le x \le z, 0 < y \le z$. Volume of this triangle is $V = yz/2$, edge lengths are $s_1 = \sqrt{(x-z)^2 + y^2}, s_2 = \sqrt{x^2+y^2}, s_3 = z$ and outer normals are $\mathbf{n}_1 = (y/s_1, (z-x)/s_1), \mathbf{n}_2 = (-y/s_2, x/s_2), \mathbf{n}_3 = (0, -1)$.

Using these values in (10) with u_c being average of three values $u_l, l = 1,2,3$ on the edges we obtain for 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 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 a general solution

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

with m_{12} remaining as a free parameter. Again deriving and solving this system is an easy task for computer algebra only one needs to be carefull with using additional constraints and making final results above compact is a bit tricky.

The matrix M (9) has to be positive definite, otherwise (8) would not be inner product. Sylvester criterion for positive definiteness of matrix M gives 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 definitions of s_1 and s_2 factorize. Leaving out positive factors the simplified inequalities are

$$x + z + 3M_{12}z > 0 \qquad (13)$$

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

$$-y^2 - (x+z)(x-2z) + 9M_{12}z^2 > 0 \qquad (15)$$

where $M_{12} = m_{12}y^2/(s_1s_2)$. It is obvious that the third inequality implies the second one (15) $\Rightarrow$ (14). To prove that the second inequality implies the first one (14) $\Rightarrow$ (13) we have used quantifier elimination [1] by QEPCAD [6,7] which proved that

$$\forall M_{12} \forall x \forall z [(14) \wedge 0 \le x \le z \wedge z > 0] \Rightarrow (13)$$

is true. We now have (15) $\Rightarrow$ (14) $\Rightarrow$ (13), so the all three inequalitied hold if and only if the third inequality (15) holds. The inequality (15) defines minimal value m_{12}^{min} of the parameter m_{12} and we have

$$m_{12} > m_{12}^{min} = \frac{s_1 s_2}{9z^2}\left(1 + \frac{(x+z)(x-2z)}{y^2}\right).$$

We introduce a positive parameter

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

and we have a family of inner products depending on the parameter $d > 0$ which all produce 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. Standard support operator (SSO) inner product described in the section 3 belongs to this family with

$$m_{12}^{SSO} = \frac{s_1 s_2}{3z^2}\left(1 + \frac{x(x-z)}{y^2}\right)$$

which gives us corresponding value of the parameter d

$$d^{SSO} = \frac{2 s_1 s_2}{9z^2}\left(1 + \frac{x^2 - xz + z^2}{y^2}\right).$$

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

$$Q_1 x_1 \in R,\ Q_2 x_2 \in R, \cdots, Q_k x_k \in R, \quad F(x_1, \cdots, x_m),$$

where $m \ge k$, $Q_i, i = 1, \cdots, k$ are quantifiers $\forall$ (for all) or $\exists$ (there exists) and F is an arbitrary logical combination of polynomial equations or 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.

First property which we would like to improve by changing the parameter d is the accuracy and an improvement in the accuracy would be exact resolving of gradients of higher order terms, first quadratic functions. We have tried to repeat our analysis with 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 give SO method resolving exactly gradients of both linear and quadratic functions.

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

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

To simplify the formulas we will here and in the following consider 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 moving, rotation and scaling. For the quadratic function $u = X^2$ the norm of the gradient error is after performing all the necessary symbolic computation

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

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

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

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

4.2 Condition Number of 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 how easy or hard it should be to numerically solve a linear system with matrix M. The condition number of the matrix M is defined as

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

The condition number of any matrix is greater or equal one and the condition number is big for ill-conditioned matrix. For evaluation of condition number we use the Frobenius norm of matrix

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

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

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

where instead of the free parameter d we have introduced another free parameter $f = d/(s_1 s_2)$, the coefficients $a_n, b_n, n = 1, 2, 3$ are rational functions depending only on x and y (we still use the triangle with vertices $(0, 0), (1, 0), (x, y)$). Important is that the coefficients a_2 and b_2 are positive, so that for very small and very big f the condition number grows without any bounds. As an example we present here a special case of 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 condition number $C^2(M)$ on the free parameter d with the value of the parameter d for standard inner product for this triangle $d^{SSO} = 4\sqrt{2}/9 \approx 0.6285$. We see that for traditional

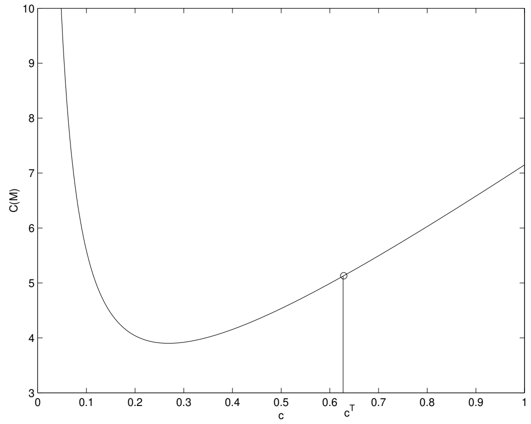


Fig. 3. Dependence of contition number $C(M)^2$ on the free parameter d for the triangle with vertices $(0,0),(1,0),(0,1)$ with the value of d^{SSO} for standard inner product.

inner product $C^2(M)|_{d=d^{SO}} \approx 5.1$ while $C^2(M)$ has mininum $C^(M)|_{d=d_0} \approx 3.9$ at $d_0 \approx 0.27$. So the matrix (11) is better conditioned for d_0 than for d^{SSO} and further according to the accuracy analysis in the previous section it should also give more accurate solution.

Numerical solution of the linear system with matrix M behaves best for minimal condition number $C(M)$. To minimize $C(M)$ we solve

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

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

$$\frac{d^2\, C^2(M)}{d\, c^2} = \frac{4c_0 + 12s1s2c_1d + 17496y^4c_4d^4}{2187c^4s_1^4s_2^4y^4} \tag{16}$$

with coefficients c_0, c_1 and c_4 being reasonable dense polynomials in x and y only with total degrees 20, 16 and 8 respectively which cannot be factored out. 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 $c_0 \geq 0, c_1 \geq 0, c_4 \geq 0$ which means as $d > 0$ that the second derivative (16) is always non-negative. So the first derivative is monotonicaly 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 is its global minimum for $d > 0$ and numerical minimization has to 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 needed, 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 mentioned earlier. We have used this approach in our analysis and quantifier elimination has provided us sufficient grounds for treating inequalities. For QE we have used program QEPCAD (Quantifier Elimination by Partial Cylindrical Algebraic Decomposition) by Hong and others [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 number of variable) which limits its usage to problems involving only few variables. However often we succeed to transform our QE problems to QE problems with less variables which have been 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 have to be equal to one third of the volume of the triangle, otherwise the support operator method would not produce exact gradients for linear functions. The general inner product of vector grid functions has been proposed and analyzed. Requirement of getting exact gradients for linear functions by the support operator method restricts the general inner product to an 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 accuracy of the method for quadratic functions improves with decreasing this parameter, however at the same time the condition number of matrix M, which is the local matrix of linear system for computing the discrete gradient, increases to infinity when the parameter goes to zero. On the other hand there exists unique minimum of the condition number for which the matrix M should be conditioned best from the family.

It is shown that by choosing appropriate inner product the accuracy and the condition number of local system for gradient can be improved when compared with standard inner product. However 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 global system of linear equations. Another question for us is what is the best way to create a vector inner product for quadrilateral in 2D and logiacal bricks in three dimension. These questions will be addressed in our future work by using combination of numerical and analytical tools.

Acknowledgement

This research has been partially supported by International Bureau of the BMBF and Czech Ministry of Education in the framework of German-Czech join 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-36 and 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.

References

[1] J. M. Hyman, M. Shashkov, Natural discretizations for the divergence, gradient, and curl on logically rectangular grids, An International Journal Computers & Mathematics with Applications 33 (4) (1997) 81–104.

[2] J. M. Hyman, M. Shashkov, The adjoint operators for the natural discretizations for the divergence, gradient, and curl on logically rectangular grids, IMACS J. of Applied Numerical Mathematics 25 (1997) 413–442.

[3] V. Ganzha, R. Liska, M. Shashkov, C. Zenger, Support operator method for Laplace equation on unstructured triangular grid, Selcuk Journal of Applied Mathematics 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 Journal on Numerical Mathematics 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, Computers & Mathematics with Applications 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. Comp. 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.), Special issue of J. Symbolic Computation on Applications of Quantifier Elimination, Vol. 24, 1997, nr. 2.