58

Ordering techniques for convection dominated problems on unstructured three-dimensional grids

S. LE BORNE[1]

INTRODUCTION

For convection dominated problems in two spatial dimensions, multigrid methods using a simple Gauß-Seidel method as a smoother can lead to excellent results if the unknowns are ordered appropriately. However, the difficulties with constructing robust multigrid solvers in three dimensions are much greater: Many ordering strategies that have been proposed for two-dimensional problems make explicit use of the planarity of the underlying graphs and are not directly applicable to three-dimensional problems.

In this paper we introduce the concept of interior surfaces in tetrahedral grids which we use to construct an ordering for the unknowns. The proposed ordering algorithm produces a decomposition of the unknowns into disjoint blocks so that block iterative methods may be applied.

We study problems where the matrix A arises from the finite element discretisation of the convection-diffusion equation

$$-\Delta u + (b \cdot \nabla)\, u \quad = \quad f \qquad \text{in } \Omega \subset \mathbb{R}^n. \tag{1}$$

Here n denotes the dimension (2 or 3) and suitable boundary conditions are assumed. Let $\mathcal{T}$ be the underlying triangulation of the domain Ω. We mention that in the numerical solution of

1 Praktische Mathematik, University Kiel, Hermann-Rodewald-Str. 3, 24098 Kiel. email: sg@numerik.uni-kiel.de
Eleventh International Conference on Domain Decomposition Methods
Editors Choi-Hong Lai, Petter E. Bjørstad, Mark Cross and Olof B. Widlund

systems of equations, such as the Navier-Stokes equations, solving a set of scalar convection-diffusion equations can be required as a subproblem (see e.g., [BWY90], [BS97] or [HGMM98] for smoothers for the Stokes problem that are based on a block ILU decomposition and hence depend on the ordering of the unknowns).

Here we will use the graph of a matrix to develop and illustrate ordering techniques and thus recall the definition of the graph of a matrix: The matrix graph $G = G(A)$ consists of the vertex set $V = V(G)$ and the edges $E = E(G) = \{(i,j) \in V \times V : \ a_{ij} \neq 0\}$. For continuous piecewise linear basis functions with the standard nodal basis, the vertices and edges of the triangulation correspond, respectively, to the vertices and edges of the matrix graph.

Typically, the size $|a_{ij}|$ of the matrix entries varies considerably over the grid due to the convective term. For the proposed ordering strategies we neglect the small matrix entries and define the graph of dominant entries as the graph with the reduced edge set $E_0 = \{(i,j) \in E : |a_{ij}| \text{ 'is large'}\}$. Typically, at most one of the entries $|a_{ij}|$ and $|a_{ji}|$ is large, thus making the reduced graph a directed graph. Henceforth, all graphs are assumed to be directed graphs. There are several reasonable classifications for an entry to be large, one of which defines $|a_{ij}|$ as 'large' if $|a_{ij}| > \kappa\ |a_{ii}|$ with a parameter $\kappa \in (0,1)$.

The task is to find a permutation P such that PAP^T is easier to solve. The ideal case would have PAP^T an upper (or lower) triangular matrix for which a simple backward (or forward) substitution process can be employed. With respect to the matrix graph, this situation corresponds to an acyclic graph.

However, many applications involve cyclic flows that may result in cyclic graphs, cf. [GDN95]. In this case, several approaches have been suggested for two-dimensional problems where the graph is planar. These ordering algorithms provide a numbering strategy that produces a block structure which can be used for block iterative methods. Numerical results for the two-dimensional convection-diffusion equation as well as for the Stokes equations with a convective term have been provided in [GP97], [HGMM98] and [HP97]. Additionally, ordering techniques to improve multigrid convergence have also been developed and applied by other authors, e.g., in [RR96], [Tur97].

In three spatial dimensions we also have a structure consisting of vertices and edges but additionally, we can involve the faces of the tetrahedra. A collection of neighbouring faces describes an interior surface in the tetrahedron. In turn, an ordering of these surfaces in addition to an ordering within each surface defines an ordering for the vertices in the tetrahedral grid.

Following, we provide the details needed to accomplish the task of constructing interior surfaces. In section 58 we briefly review (prerequisites for) ordering strategies that have been proposed for planar graphs and point out the difficulties in generalising the concepts to graphs that are not planar. Section 58 deals with ordering concepts for the three-dimensional case.

THE TWO-DIMENSIONAL CASE

In preparation for the three-dimensional case we now review some of the ideas that have been previously considered for two dimensions. The motivation will not change: find a permutation P such that PAP^T has a structure that is better suited for applying linear equation algorithms.

Given a planar graph $G(V,E)$, the edges associated with a vertex can be ordered periodically with respect to their defining angles. A useful property of graphs that arise from the discretisation of the convection-diffusion equation is the *one-flow-direction condition*: for all vertices $p \in V$ there is a (unique) sequence $\{e_1, \cdots, e_m\}$ of outward edges and a sequence $\{e_{m+1}, \cdots, e_k\}$ of inward edges enumerated in the counter-clockwise direction.

For the two-dimensional case, several ordering techniques such as *ordering with concentric cycles*, *feedback vertex set ordering* or *parallel ordering* have been developed for planar graphs that fulfil the one-flow-direction condition.

For further details on these ordering algorithms as well as numerical results for the convection-diffusion equation and the Stokes equations with a convective term, the interested reader is referred to [GP97], [Hac97], [HGMM98], [HP97].

However, for the three-dimensional case it unfortunately remains unclear how to order the edges adjacent to a vertex or how to characterise minimal or concentric cycles. For this reason the concepts that apply to orderings for two-dimensional graphs cannot be generalised to three-dimensional graphs in a straightforward way.

In the following section new concepts are proposed for the three-dimensional case. The goal is to identify subgraphs that possess properties of planar graphs so that ordering strategies for planar graphs can be applied.

THE THREE-DIMENSIONAL CASE

In the three-dimensional case, looking for graph cycles may not be the best concept: As seen in section 58, minimal and concentric cycles produce favourable orderings. In three dimensions, it is not clear how to characterise cycles as minimal or concentric. Alternatively, we propose to decompose the graph into a set of disjoint planar subgraphs that might contain cycles, and then apply ordering techniques for planar graphs on each subgraph separately.

In order to determine suitable subgraphs, we must first introduce the concept of interior surfaces. For example, if the underlying triangulation consists of tetrahedra, an interior surface can be described through a combination of neighbouring faces (of the tetrahedra). If such a surface is constructed carefully its vertices and edges define a planar graph in which we can apply the ordering algorithms of section 58. The cylinder depicted in Figure 1 illustrates this concept. Interior to the cylinder we have detailed the cyclic flow, however, the tetrahedral triangularisation has been left out. Slicing through the cylinder in a horizontal direction (i.e. parallel to the top and bottom but orthogonal to the cylindrical walls,) produces circular surfaces in which the flow can be seen to be concentric cycles. On these surfaces, the ordering algorithms for planar graphs are applicable.

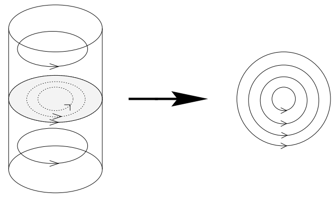

Figure 1 Cyclic flow in a cylinder

Before an algorithm can be developed that will construct such interior surfaces we first must establish terminology for the description of edges, faces and surfaces.

Surfaces and surface graphs

In the three-dimensional case we are dealing with tetrahedral grids, $\mathcal{T}(V, E, F)$, consisting of a set of vertices, $V(\mathcal{T})$, and a set of edges, $E(\mathcal{T})$ and, additionally, we have a set of faces, $F(\mathcal{T})$. We call two faces *neighbours* if they have a common edge (note that a common vertex is not sufficient).

Definition 1 *A subset $\mathcal{S}$ of $\mathcal{T}$ is called a* surface *if it possesses the following three properties:*

(i) *The* degree *of every edge in $\mathcal{S}$, i.e., the number of faces in $\mathcal{S}$ sharing this edge, is at most two.*
(ii) *$\mathcal{S}$ is* connected, *i.e., for any $f, g \in F(\mathcal{S})$ there is a path $f = f_0, f_1, \cdots, f_k = g$ of faces $f_i \in F(\mathcal{S})$ where f_i and f_{i+1} are neighbours for all $i = 1, \cdots, k-1$.*
(iii) *For any $f, g \in F(\mathcal{S})$ with a common vertex p there is a path $f = f_0, \cdots, f_k = g$ of faces $f_i \in F(\mathcal{S})$ each having p as a vertex where f_i and f_{i+1} are neighbours for all $i = 1, \cdots, k-1$.*

Figure 2 shows an example for a surface and three examples that violate one or more of the required properties.

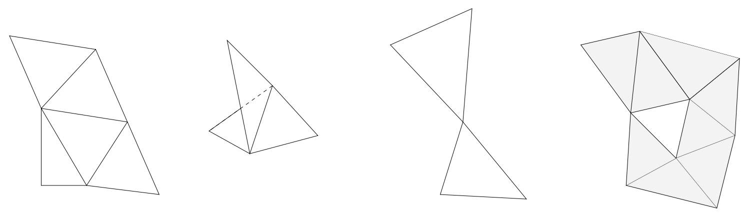

Figure 2 A surface and three violated situations

The *surface graph* $G(\mathcal{S}) = (V(\mathcal{S}), E(\mathcal{S}))$ is the graph consisting of vertices and edges on the surface. In the following we only consider orientable surfaces.

As in the two-dimensional case, for all vertices $v \in V(\mathcal{S})$ of an orientable surface $\mathcal{S}$, we can order the edges adjacent to v counter-clockwise with respect to the orientation. We note that this would not be the case without property *(iii)* above. Hence, we can determine whether a one-flow-direction condition holds for the vertices of the surface graph $G(\mathcal{S})$ which establishes an important prerequisite for applying the ordering techniques for planar graphs.

Flow surface

As illustrated in the introductory example in Figure 1, the goal is to construct surfaces that lie parallel to the flow while retaining the information about the flow, in particular concerning cyclic dependencies, is contained in the surface. To obtain such surfaces one begins with an arbitrary face and then iteratively attaches faces in the flow direction across the edges. In order to formalise this procedure we will define a function $\varphi : F(\mathcal{T}) \times \tilde{E}(\mathcal{T}) \to F(\mathcal{T})$ ($\tilde{E}(\mathcal{T}) \subseteq E(\mathcal{T})$) that assign a face to a given face f across one of its edges e in the following way:

We assume that the convection direction b of the partial differential equation (1) is available. For an edge e with endpoints $v, w \in V(\mathcal{T})$, let $m = \frac{1}{2}(v + w)$ denote the edge midpoint. Define $\tilde{E}(\mathcal{T})$ to be the set of edges with degree greater than one that have a convection direction $b(m)$ with $\|b(m)\|_2 > \sigma > 0$ in the edge midpoint for a parameter $\sigma > 0$. Let $\tilde{E}_1(\mathcal{T}) \subseteq \tilde{E}(\mathcal{T})$ be the subset of edges for which the convection in the edge midpoint is not parallel to the edge itself, and let $\tilde{E}_2(\mathcal{T}) = \tilde{E}(\mathcal{T}) \setminus \tilde{E}_1(\mathcal{T})$. Then functions φ_f, $\varphi_b : E_1(\mathcal{T}) \to F(\mathcal{T})$ (the index 'f' refers to *forward*, 'b' to *backward*) can be defined as follows: Let $\gamma \in (0, \pi/2)$ be a given angle. For an edge $e \in \tilde{E}_1(\mathcal{T})$, let $\varphi_f(e)$ be the face adjacent to e that has the smallest angle to the plane spanned by the convection direction in the edge midpoint and the edge e, provided that the angle between $b(m)$ and this face is smaller than γ. Analogously, we define $\varphi_b(e)$ as the face adjacent to e that has the smallest angle to the plane spanned by the *negative* convection direction in the edge midpoint and the edge e, again provided that the angle between $-b(m)$ and this face is smaller than γ.

Note that in these definitions the orientation of the convection direction is of importance when making the decision whether an angle is α or $\pi - \alpha$. For example, in Figure 3, the angle between $b(m)$ and the face f is 0, whereas the angle between $b(m)$ and the face g is π. Hence we have $\varphi_f(e) = f$ and $\varphi_b(e) = g$. The angle γ was introduced to ensure that the face $\varphi_f(e)$ ($\varphi_b(e)$) is nearly parallel (up to an error smaller than γ) to the flow (negative flow).

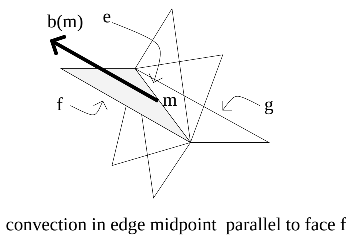

Figure 3 Illustration for the mappings φ_f and φ_b.

For an edge $e \in \tilde{E}_2(\mathcal{T})$ of a face f, the convection direction is by definition of $\tilde{E}_2(\mathcal{T})$ nearly parallel to the edge so that we cannot define faces in the flow direction as above. Instead, we define $\varphi_p : F(\mathcal{T}) \times \tilde{E}_2(\mathcal{T}) \to F(\mathcal{T})$ (index 'p' refers to *parallel*) where $\varphi_p(f, e)$ is defined to be the face with edge e that is closest to being *parallel* to f, i.e., that has an angle to f as close to π as possible, provided that the angle between this face and f is at least $\pi - \gamma$. Using the example given in figure 3, only changing the convection direction $b(m)$ to be parallel to e so that $e \in \tilde{E}_2(\mathcal{T})$, we have $\varphi_p(g, e) = f$.

We define the sets

$$\begin{aligned}
\tilde{F}_f(\mathcal{T}) &= \varphi_f(\tilde{E}_1(\mathcal{T})), \\
\tilde{F}_b(\mathcal{T}) &= \varphi_b(\tilde{E}_1(\mathcal{T})), \\
\tilde{F}_p(\mathcal{T}) &= \varphi_p(\tilde{F}_f(\mathcal{T}) \times \tilde{E}_2(\mathcal{T})) \cup \varphi_p(\tilde{F}_b(\mathcal{T}) \times \tilde{E}_2(\mathcal{T})), \\
\tilde{F}(\mathcal{T}) &= \tilde{F}_f(\mathcal{T}) \cup \tilde{F}_b(\mathcal{T}) \cup \tilde{F}_p(\mathcal{T}).
\end{aligned}$$

The set $\tilde{F}(\mathcal{T})$ can be considered as the set of faces lying in the flow. The collection of faces in $\tilde{F}(\mathcal{T})$, however, does not typically represent a surface since an edge might have a degree higher than two. We next define a flow surface:

Definition 2 *Let $\alpha \in (0, \pi)$ be a given angle. An oriented surface $\mathcal{S}$ consisting of a set of faces $F(\mathcal{S}) \subseteq \tilde{F}(\mathcal{T})$ is called a* flow surface of minimal angle α *if, for every edge $e \in E(\mathcal{S})$ with adjacent faces f and g, the angle between these faces is greater than α, i.e., $\angle(f, g) \geq \alpha$.*

Surface algorithm

Presented is an algorithm that constructs flow surfaces. The intuitive description of the algorithm is as follows: Start with an arbitrary edge of $\tilde{E}_1(\mathcal{T})$ and determine the face $\varphi_f(e)$. Next append faces across the edges using the functions φ_f, φ_b or φ_p. The functions φ_f, φ_b or φ_p are combined into a single function by defining

$$\begin{aligned} \varphi &: F(\mathcal{T}) \times \tilde{E}(\mathcal{T}) \to F(\mathcal{T}), \\ \varphi(f,e) &= \begin{cases} \varphi_p(f,e) & : e \in \tilde{E}_2(\mathcal{T}) \\ \varphi_f(e) & : e \in \tilde{E}_1(\mathcal{T}) \text{ and } \angle(\varphi_f(e), f) > \angle(\varphi_b(e), f) \\ \varphi_b(e) & : \textit{otherwise}. \end{cases} \end{aligned}$$

The following Pascal-like procedure is the main ingredient to an algorithm to construct a flow surface $\mathcal{S}$.

```
procedure surface( e, f );
begin
    g := φ(f, e);
    if g ∉ S and S ∪ {g} is a flow surface then
    begin
        S := S ∪ {g};
        surface( ẽ, g ) for all edges ẽ ≠ e of g
end end;
```

Now an algorithm can be introduced that constructs disjoint flow surfaces in $\mathcal{T}$:

```
procedure decompose( T );
begin
    while F̃(T) not empty
    begin
        for an arbitrary f ∈ F̃(T) initialise S := {f};
        surface( ẽ, f ) for all edges ẽ of f;
        delete all faces that have at least one vertex in common with S
end end;
```

Remark 1 *The constructed surfaces depend on the starting edge. For the example involving the cylinder depicted in Figure 1 we could either generate vertical or horizontal surfaces.*

An example

We applied the surface algorithm to a cyclic flow in the unit cube. The flow direction and the resulting flow surfaces for a regular grid as well as an irregular grid are depicted in Figure 4. In the case of the regular grid, a single vertical surface and several horizontal surfaces have been constructed. On each of these surfaces we can now apply ordering techniques that have been developed for the two-dimensional case.

Conclusions and future work

We have provided a general framework for constructing (disjoint) surfaces in a tetrahedral grid. To this end, we have constructed surfaces that retain important information about

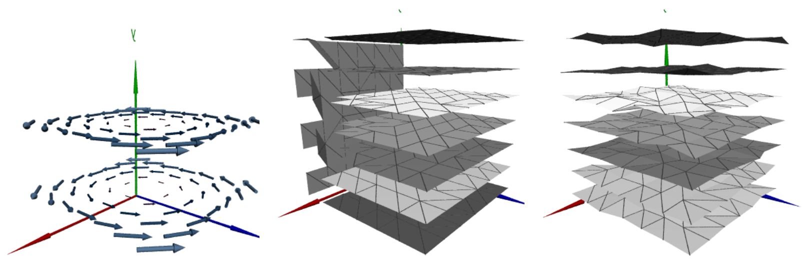

Figure 4 Cyclic flow and resulting flow surfaces on a regular and irregular grid.

the flow direction. Therefore existing ordering techniques that have proved successful in the two-dimensional case may be applied to the three-dimensional case.

Even though the investigation of iterative schemes for solving the linear system of equations was not the goal of this paper, any analysis of the proposed reordering scheme (in terms of improved convergence rates) requires the application of the algorithm to flow problems. This work is presently being performed and the results will be forthcoming. Additional future activities include the choice of parameters in the algorithm, e.g. the parameter κ in the definition of strong edges as well as the parameters α and γ in the definition of the function φ and of flow surfaces. By extending the theory that has been applied to the two-dimensional case to the three-dimensional case, the proposed ordering scheme serves to efficiently decompose the set of unknowns of a convection dominated problem into disjoint blocks for which well-known ordering algorithms are available. Once the disjoint blocks are found, the well-studied block iterative methods may then be applied.

Acknowledgements. The author wishes to thank Thomas Probst and Ronald Kriemann (both University of Kiel) for helpful discussions as well as for implementing the proposed algorithms.

REFERENCES

[BS97] Braess D. and Sarazin R. (1997) An efficient smoother for the Stokes problem. *Applied Numerical Mathematics* 23: 3–19.

[BWY90] Bank R. E., Welfert B. D., and Yserentant H. (1990) A class of iterative methods for solving saddle point problems. *Numerische Mathematik* 56: 645–666.

[GDN95] Griebel M., Dornseifer T., and Neunhoeffer T. (1995) *Numerische Simulation in der Strömungsmechanik - Eine praxisorientierte Einführung*. Vieweg.

[GP97] Gutsch S. and Probst T. (1997) Cyclic and feedback vertex set ordering for the 2d convection-diffusion equation. Technical Report 97-22, Christian-Albrechts-Universität Kiel.

[Hac97] Hackbusch W. (1997) On the feedback vertex set problem for a planar graph. *Computing* 58: 129–155.

[HGMM98] Hackbusch W., Gutsch S., Maitre J.-F., and Musy F. (1998) The appropriate numbering for the multigrid solution of convection dominated problems. In *Notes on Numerical Fluid Mechanics*, volume 66, pages 75–88.

Vieweg.

[HP97] Hackbusch W. and Probst T. (1997) Downwind Gauß-Seidel Smoothing for Convection Dominated Problems. *Numerical Linear Algebra with Applications* 4: 85–102.

[RR96] Rentz-Reichert H. (1996) *Robuste Mehrgitterverfahren zur Lösung der inkompressiblen Navier-Stokes Gleichung: Ein Vergleich.* PhD thesis, Institut für Computeranwendungen der Universität Stuttgart.

[Tur97] Turek S. (1997) On ordering strategies in a multigrid algorithm. In *Notes on Numerical Fluid Mechanics*, volume 41. Vieweg. Proc. 8th GAMM-Seminar, Kiel.