A Boundary Condition Capturing Method for Poisson's Equation on Irregular Domains

Xu-Dong Liu *
Ronald P. Fedkiw †
Myungjoo Kang ‡

February 17, 2000

Abstract

Interfaces have a variety of boundary conditions (or jump conditions) that need to be enforced. In [3], the Ghost Fluid Method (GFM) was developed to capture the boundary conditions at a contact discontinuity in the inviscid Euler equations. This method was extended to treat more general discontinuities such as shocks, detonations, and deflagrations in [2] and compressible viscous flows in [4]. In this paper, a similar boundary condition capturing approach is used to develop a new numerical method for the variable coefficient Poisson equation in the presence of interfaces where both the variable coefficients and the solution itself may be discontinuous. This new method is robust and easy to implement even in three spatial dimensions. Furthermore, the coefficient matrix of the associated linear system is the standard symmetric matrix for the variable coefficient Poisson equation in the absence of interfaces allowing for straightforward application of standard "black box" solvers.

*Research supported in part by NSF DMS-9805546
†Research supported in part by ONR N00014-97-1-0027
‡Research supported in part by NSF and DARPA grant NSF-DMS961854, for Virtual Integrated Prototyping (VIP)

1 Introduction

The "immersed boundary" method [20] uses a δ-function formulation to smear out the solution of the variable coefficient Poisson equation on a thin finite band about the interface. See [21] for details. In [23], the "immersed boundary" method was combined with the level set method resulting in a first order numerical algorithm that is simple to implement even in multiple spatial dimensions. However, the numerical smearing at the interface has an adverse effect on the solution forcing continuity at the interface regardless of the appropriate interface boundary conditions. That is, the numerical solution is continuous at the interface even if the actual boundary conditions imply that the solution should be discontinuous.

The "immersed interface" method [10] is a second order numerical method designed to preserve the jump conditions at the interface in contrast to the numerical smearing introduced by the δ-function formulation of the "immersed boundary" method. The "immersed interface" method incorporates the interface boundary conditions into the finite difference stencil in a non-trivial way that preserves jumps in both the function and its derivatives. However, this algorithm is fairly complex and has only been extended to three spatial dimensions for the simple case of a stationary interface [12], i.e. the method has not yet been extended to treat three dimensional moving interfaces. Furthermore, the corresponding linear system that needs to be solved is not symmetric, dramatically reducing the number of standard fast linear solvers that can be utilized with this method, although it should be noted that one fast solution technique was used in conjunction with this method in [11]. In contrast, the δ-function formulation of the "immersed boundary" method has a corresponding linear system whose matrix is symmetric allowing a wide range of standard fast linear solvers to be utilized.

Another notable technique, presented in [7], is a second order accurate numerical method that preserves jumps at the interface with a resolution comparable to that of the "immersed interface" method. A clever premise underlying this method is the ability to smoothly extend the solution outside the physical domain into a fictitious domain and to use these extended values in the numerical method. While this method suffers from a non-symmetric linear system and the usual difficulties that this introduces, the authors did show that the method was compatible with both multigrid and adaptive mesh techniques. However, [7] addressed only Dirichlet boundary conditions and did not extend the method to treat interface jump conditions.

It should be noted that the idea of using extended values and fictitious domains is not new, e.g. [15], [14] and [17] used similar ideas to solve the Laplace equation on irregular domains with the help of integral equations. That is, a system of integral equations is solved, and then the results are used in the discretization of the Laplacian. In [18], a fast version of this algorithm was presented which depends in part on the on fast algorithms for computing the integrals [16]. The interested reader is also referred to [5] for more details on fast methods for the integral equations.

In [3], the Ghost Fluid Method (GFM) was developed to properly treat the boundary conditions in [19], removing the spurious oscillations shown in [9]. The GFM was originally designed to treat contact discontinuities in the inviscid Euler equations, but it was generalized to treat shocks, detonations, and deflagrations in [2] and compressible viscous flows in [4]. The generalized GFM captures the appropriate Rankine-Hugoniot jump conditions at an interface without explicitly enforcing these jump conditions. Instead, the GFM creates an artificial fluid which implicitly induces the proper conditions at the interface. In the flavor of the level set function which gives an implicit representation of the interface, the GFM gives an implicit representation of the Rankine-Hugoniot jump conditions at the interface. Since the jump conditions are handled implicitly by the construction of a ghost fluid, the overall scheme becomes easy to implement in multidimensions.

In this paper, a similar boundary condition capturing approach is used to develop a new numerical method for the variable coefficient Poisson equation in the presence of interfaces where both the variable coefficients and the solution itself may be discontinuous. This new method is implemented using a standard finite difference discretization on a Cartesian grid making it simple to apply in as many as three spatial dimensions. Furthermore, the coefficient matrix of the associated linear system is the standard symmetric matrix for the variable coefficient Poisson equation in the absence of interfaces allowing for straightforward application of standard "black box" solvers. Most importantly, this new numerical method does not suffer from the numerical smearing prevalent in the δ-function formulation of the "immersed boundary" method. In fact, the new method preserves jumps at the interface with a resolution comparable to that of the "immersed interface" method. See [13] for a theoretical justification of this new method.

Before proceeding, a few comments on the need for yet another new method may be in order, especially since this new method is only first order accurate. First and foremost, note that the second order accurate methods have not yet achieved widespread success. For example, the "immersed interface" method is more of a strained attempt to satisfy truncation error for static two dimensional interfaces, than a robust second order method. That is, the "immersed interface" method has not yet been applied to three dimensional problems with moving interfaces or to the multiphase Navier Stokes equations in any dimension. Furthermore, when it was applied to the Hele-Shaw problem in [6], the solutions quickly degenerated to first order accuracy even though new ad hoc fixes were used in the discretization. Because of these complications, the first order "immersed boundary" method is the only scheme that is currently used for complex numerical simulations such as the three dimensional multiphase Navier Stokes equations. However, the "immersed boundary" method has problems of its own. For example, the "immersed boundary" method cannot produce discontinuous solutions and thus is unable to properly model the jump in pressure due to surface tension forces in the Navier Stokes equations. Various authors have avoided this problem by treating the pressure as a continuous function and adding new source terms to the momentum equations, see for example see [1], [24], and [23]. Unlike the "immersed boundary" method, our method can be used to obtain discontinuous solution profiles as will be shown in this paper. Furthermore, we note that our new method can be used to model the Navier Stokes equations directly, i.e. without the addition of source terms to model the effects of surface tension [8].

2 Equations

Consider a Cartesian computational domain, Ω, with exterior boundary, $\partial\Omega$, and a lower dimensional interface, Γ, that divides the computational domain into disjoint pieces, Ω^- and Ω^+. The variable coefficient Poisson equation is given by

$$\begin{array}{ll} \nabla \cdot (\beta(\vec{x})\nabla u(\vec{x})) = f(\vec{x}), & \vec{x} \in \Omega \\ u(\vec{x}) = g(\vec{x}), & \vec{x} \in \partial\Omega \end{array} \tag{1}$$

where $\vec{x} = (x, y, z)$ are the spatial dimensions, $\nabla = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z})$ is the divergence operator, and $\beta(\vec{x})$ is presumed to be continuous on each disjoint subdomain, Ω^- and Ω^+, but may be discontinuous across the interface Γ. Furthermore, $\beta(\vec{x})$ is assumed to be positive and bounded below by some $\epsilon > 0$.

The jump conditions or internal boundary conditions are specified along the interface Γ as

$$\begin{array}{ll} [u]_\Gamma = a(\vec{x}), & \vec{x} \in \Gamma \\ [\beta u_n]_\Gamma = b(\vec{x}), & \vec{x} \in \Gamma \end{array} \tag{2}$$

where

$$\begin{array}{r} [u]_\Gamma = u^+(\vec{x}) - u^-(\vec{x}) \\ [\beta u_n]_\Gamma = \beta^+(\vec{x})u_n^+(\vec{x}) - \beta^-(\vec{x})u_n^-(\vec{x}) \end{array} \tag{3}$$

specifies the direction of the jump with the "$\pm$" subscripts referring to $\Omega^\pm$. Note that $u_n = \nabla u \cdot \vec{N}$ is the normal derivative of u where $\vec{N}$ is the local unit normal to the interface.

Equation 1 uses Dirichlet boundary conditions for illustration purposes only as the boundary conditions on $\partial\Omega$ are not crucial to our numerical method. In fact, it takes little effort to replace the Dirichlet boundary conditions with Neumann boundary conditions reformulating the Poisson equation as

$$\begin{array}{ll} \nabla \cdot (\beta(\vec{x})\nabla u(\vec{x})) = f(\vec{x}), & \vec{x} \in \Omega \\ u_n(\vec{x}) = g(\vec{x}), & \vec{x} \in \partial\Omega \end{array} \tag{4}$$

throughout the text.

3 Numerical Method

Since the interface can have a fairly complex shape, the interface location is represented by the zero level of a signed distance function, i.e. a level set representation of the interface is used [22].

3.1 One Dimension

Consider the unit domain $\Omega = [0, 1]$ where the interface is a single point $\Gamma = .5$ with a level set representation of $\phi = x - .5$ so that the interface location is recovered when $\phi = 0$. Since ϕ is the signed distance function, the set of all points where $\phi < 0$ and the set of all points where $\phi > 0$ represent two disjoint subdomains, Ω^- and Ω^+, respectively. For the numerical algorithm, one needs to identify whether a given point is located in Ω^- or Ω^+ which is determined by considering the local sign of ϕ, unless $\phi = 0$ implying that the point is located directly on the interface itself. Since the interface is a lower dimensional set, this situation can be rectified by defining Ω^- as the set of all points where $\phi \leq 0$ and Ω^+ as the set of all points where $\phi > 0$ so that no points lie directly on the interface.

The computational domain is discretized into cells of size $\triangle x$ where the cell centers are referred to as grid points or grid nodes with the i-th grid node located at x_i. The cell edges are referred to as fluxes so that the two fluxes bounding the i-th computational cell are located at $x_{i\pm\frac{1}{2}}$. The solution to the Poisson equation is computed at the grid nodes and is written as $u_i = u(x_i)$. An analogous definition holds for f_i, g_i, and the level set function ϕ_i. In general, interfaces move throughout the grid as ϕ is evolved in time, and a reinitialization procedure is needed to maintain ϕ as an approximate distance function [23]. Since ϕ is known only at the grid nodes x_i, the value of ϕ at fluxes is defined by the linear average of the nodal values, e.g.

$$\phi_{i+\frac{1}{2}} = \frac{\phi_i + \phi_{i+1}}{2} \tag{5}$$

is a second order accurate approximation to ϕ at the flux located between the i-th and $(i+1)$-st cells.

The level set function is used to define the unit normal as,

$$\vec{N} = \frac{\nabla \phi}{|\nabla \phi|} \tag{6}$$

where the normal is computed at each grid node using central differencing. For example,

$$\vec{N}_i = \frac{\left(\frac{\phi_{i+1}-\phi_{i-1}}{\triangle x}\right)}{\left|\frac{\phi_{i+1}-\phi_{i-1}}{\triangle x}\right|} = \frac{\phi_{i+1}-\phi_{i-1}}{|\phi_{i+1}-\phi_{i-1}|} \tag{7}$$

in one spatial dimension. Note that the denominator in equation 7 could be identically zero in certain rare situations, but the numerical method does not make use of the normal in these situations. When $\phi = x - .5$, equation 6 implies that $\vec{N} = 1$ everywhere so that $u_n = u_x$.

3.1.1 The Laplace Equation

Consider the one dimensional Laplace equation with $\beta = 1$ and $f(x) = 0$ given by

$$u_{xx} = 0 \tag{8}$$

with fixed Dirichlet boundary conditions on $\partial\Omega$. Ignoring the interface, or equivalently setting $[u]_\Gamma = [u_x]_\Gamma = 0$, the exact solution is merely a straight line connecting the two fixed points on $\partial\Omega$. For example, if the boundary conditions are $u(0) = 0$ and $u(1) = 1$ then the solution is $u = x$ on $[0, 1]$. The standard second order discretization

$$\frac{\left(\frac{u_{i+1}-u_i}{\triangle x}\right) - \left(\frac{u_i-u_{i-1}}{\triangle x}\right)}{\triangle x} = 0 \tag{9}$$

can be used to solve this problem. For each unknown, u_i, equation 9 is used to fill in one row of a matrix creating a linear system of equations. Since the resulting matrix is symmetric, a wide number of fast linear solvers can be used. For linear solvers that require an initial guess, setting all u_i identically zero is usually sufficient.

Next consider $[u]_\Gamma = 1$ and $[u_x]_\Gamma = 0$ with Dirichlet boundary conditions $u(0) = 0$ and $u(1) = 2$ yielding an exact solution of $u = x$ in $[0, .5]$ and $u = x + 1$ in $(.5, 1]$ where $x = .5$ is included in Ω^- as previously discussed. Consider the exact solution. If x_k and x_{k+1} are the nodes adjacent to the interface, one can see that $\frac{u_{k+1}-u_k}{\triangle x}$ is $O(\frac{1}{\triangle x})$, while all the other terms of the form $\frac{u_{i+1}-u_i}{\triangle x}$ are $O(1)$ and approximate the local derivative. Since the derivative is not defined *across* the interface, the $\frac{u_{k+1}-u_k}{\triangle x}$ term is not well defined. Obviously, this term needs to be modified to give a reasonable approximation of the derivative near the interface.

The jump condition, $[u]_\Gamma = 1$, implies that $u^+(x) - u^-(x) = 1$ at the interface Γ. Since the underlying idea of the Ghost Fluid Method is to apply boundary conditions near the interface as opposed to applying them at the exact interface location, the jump conditions are rewritten as $u_i^+ - u_i^- = 1$ at every computational grid node. Then for every value of u_i^- in Ω^-, one can define $u_i^+ = u_i^- + 1$ including the boundary where $u^+(0) = u^-(0) + 1 = 1$. Likewise, for every value of u_i^+ in Ω^-, one can define $u_i^- = u_i^+ - 1$ including the boundary where $u^-(1) = u^+(1) - 1 = 1$. At this point, every grid node has two values for the solution, u_i^- and u_i^+, and one can see from the boundary conditions that the exact solutions are $u^- = x$ on $[0,1]$ and $u^+ = x + 1$ on $[0,1]$. Furthermore, the jump condition, $u_i^+ - u_i^- = 1$, is satisfied at every grid point and the boundary.

Two equations from the linear system contain the $\frac{u_{k+1} - u_k}{\triangle x}$ term. It is the first term in

$$\frac{\left(\frac{u_{k+1}^+ - u_k^-}{\triangle x}\right) - \left(\frac{u_k^- - u_{k-1}^-}{\triangle x}\right)}{\triangle x} = 0 \tag{10}$$

and the second term in

$$\frac{\left(\frac{u_{k+2}^+ - u_{k+1}^+}{\triangle x}\right) - \left(\frac{u_{k+1}^+ - u_k^-}{\triangle x}\right)}{\triangle x} = 0 \tag{11}$$

where the "$\pm$" superscripts have been added to emphasize the domain as $\Omega^\pm$ for each u_i. Both these equations suffer from the mixing of terms from different domains and are poor candidates for obtaining the exact solution. The previous paragraph illustrates that equations 10 and 11 should be replaced by

$$\frac{\left(\frac{u_{k+1}^- - u_k^-}{\triangle x}\right) - \left(\frac{u_k^- - u_{k-1}^-}{\triangle x}\right)}{\triangle x} = 0 \tag{12}$$

and

$$\frac{\left(\frac{u_{k+2}^+ - u_{k+1}^+}{\triangle x}\right) - \left(\frac{u_{k+1}^+ - u_k^+}{\triangle x}\right)}{\triangle x} = 0 \tag{13}$$

to remove the mixing of the "±" values. The nodal jump conditions imply that $u_{k+1}^- = u_{k+1}^+ - 1$ and $u_k^+ = u_k^- + 1$ giving rise to

$$\frac{\left(\frac{(u_{k+1}^+ - 1) - u_k^-}{\triangle x}\right) - \left(\frac{u_k^- - u_{k-1}^-}{\triangle x}\right)}{\triangle x} = 0 \tag{14}$$

and

$$\frac{\left(\frac{u_{k+2}^+ - u_{k+1}^+}{\triangle x}\right) - \left(\frac{u_{k+1}^+ - (u_k^- + 1)}{\triangle x}\right)}{\triangle x} = 0 \tag{15}$$

from equations 12 and 13 as replacements for equations 10 and 11 in the linear system. In equations 14 and 15 the $\frac{u_{k+1} - u_k}{\triangle x}$ terms have been modified in a way that makes them $O(1)$ instead of $O(\frac{1}{\triangle x})$.

In general, $[u]_\Gamma = a(x_\Gamma)$ where x_Γ is the interface location. The jumps at the grid nodes, $a_k = a(x_k)$ and $a_{k+1} = a(x_{k+1})$, and the local values of ϕ can be used to interpolate the jump at the interface as

$$a_\Gamma = \frac{a_k|\phi_{k+1}| + a_{k+1}|\phi_k|}{|\phi_k| + |\phi_{k+1}|} \tag{16}$$

writing

$$\frac{\left(\frac{(u_{k+1} - a_\Gamma) - u_k}{\triangle x}\right) - \left(\frac{u_k - u_{k-1}}{\triangle x}\right)}{\triangle x} = 0 \tag{17}$$

and

$$\frac{\left(\frac{u_{k+2} - u_{k+1}}{\triangle x}\right) - \left(\frac{u_{k+1} - (u_k + a_\Gamma)}{\triangle x}\right)}{\triangle x} = 0 \tag{18}$$

for use in the linear solver. It is interesting to note that one must use the same value of a_Γ in both equation 17 and equation 18. For example, if one were to use a_k in equation 17 and a_{k+1} in equation 18, then the linear system does not have a solution since the jump at the interface is not well defined.

Next, consider the derivative jump condition and rewrite the standard second order discretization (equation 9) as

$$\frac{(u_x)_{i+\frac{1}{2}} - (u_x)_{i-\frac{1}{2}}}{\triangle x} = 0 \tag{19}$$

assuming that the derivatives, u_x, are known at fluxes and thus subscripted with $i \pm \frac{1}{2}$. Once again assume that the interface is located between x_k and x_{k+1} and use the sign of $\phi_{k+\frac{1}{2}}$ to determine whether $(u_x)_{k+\frac{1}{2}}$ lies in Ω^- or Ω^+. For the sake of exposition, assume that $(u_x)_{k+\frac{1}{2}}$ lies in Ω^+. The case where $(u_x)_{k+\frac{1}{2}}$ lies in Ω^- is similar and will be considered shortly.

Consider $[u]_\Gamma = 0$ and $[u_x]_\Gamma = 1$ with Dirichlet boundary conditions $u(0) = 0$ and $u(1) = 1.5$ yielding an exact solution of $u = x$ in $[0, .5]$ and $u = 2x$ in $(.5, 1]$. The exact solution dictates that $(u_x)_{k-\frac{1}{2}} = 1$ and $(u_x)_{k+\frac{1}{2}} = 2$ at the fluxes adjacent to the interface producing an $O(\frac{1}{\triangle x})$ value of $\frac{(u_x)_{k+\frac{1}{2}} - (u_x)_{k-\frac{1}{2}}}{\triangle x}$ in equation 19, while all the other terms of the form $\frac{(u_x)_{i+\frac{1}{2}} - (u_x)_{i-\frac{1}{2}}}{\triangle x}$ are identically zero and approximate the local second derivative. In a more general situation, the terms of the form $\frac{(u_x)_{i+\frac{1}{2}} - (u_x)_{i-\frac{1}{2}}}{\triangle x}$ will be $O(1)$ and approximate the nonzero second derivative, while the $\frac{(u_x)_{k+\frac{1}{2}} - (u_x)_{k-\frac{1}{2}}}{\triangle x}$ term is still $O(\frac{1}{\triangle x})$. Since the second derivative is not defined *across* the interface, the $\frac{(u_x)_{k+\frac{1}{2}} - (u_x)_{k-\frac{1}{2}}}{\triangle x}$ term is not well defined.

This time only one term of the form of equation 19 involves differencing across the interface. It can be written as

$$\frac{(u_x)^+_{k+\frac{1}{2}} - (u_x)^-_{k-\frac{1}{2}}}{\triangle x} = 0 \tag{20}$$

with the "$\pm$" superscripts emphasizing the domain. This equation suffers from the mixing of terms from different domains and should be replaced by

$$\frac{(u_x)^-_{k+\frac{1}{2}} - (u_x)^-_{k-\frac{1}{2}}}{\triangle x} = 0 \tag{21}$$

where the "-" signs are used for Ω^- instead of the "+" signs for Ω^+, since this is the equation for u_k which has previously been assumed to be in Ω^-. Writing the derivative jump condition at fluxes implies that $(u_x)^-_{k+\frac{1}{2}} = (u_x)^+_{k+\frac{1}{2}} - 1$ leading to

$$\frac{\left((u_x)^+_{k+\frac{1}{2}} - 1\right) - (u_x)^-_{k-\frac{1}{2}}}{\triangle x} = 0 \tag{22}$$

where the modified $\frac{(u_x)_{k+\frac{1}{2}}-(u_x)_{k-\frac{1}{2}}}{\triangle x}$ term is $O(1)$ instead of $O(\frac{1}{\triangle x})$.

In general, $[u_x]_\Gamma = b(x_\Gamma)$ where x_Γ is the interface location and $b_k = b(x_k)$ and $b_{k+1} = b(x_{k+1})$ can be used to define

$$b_\Gamma = \frac{b_k|\phi_{k+1}| + b_{k+1}|\phi_k|}{|\phi_k| + |\phi_{k+1}|} \tag{23}$$

as the jump at the interface with

$$\frac{\left((u_x)^+_{k+\frac{1}{2}} - b_\Gamma\right) - (u_x)^-_{k-\frac{1}{2}}}{\triangle x} = 0 \tag{24}$$

replacing equation 22.

Finally consider both jump conditions, $[u]_\Gamma = a(x_\Gamma)$ and $[u_x]_\Gamma = b(x_\Gamma)$. Combining equations 17, 18, and 24 gives

$$\frac{\left(\frac{(u_{k+1}-a_\Gamma)-u_k}{\triangle x} - b_\Gamma\right) - \left(\frac{u_k-u_{k-1}}{\triangle x}\right)}{\triangle x} = f_k \tag{25}$$

and

$$\frac{\left(\frac{u_{k+2}-u_{k+1}}{\triangle x}\right) - \left(\frac{u_{k+1}-(u_k+a_\Gamma)}{\triangle x}\right)}{\triangle x} = f_{k+1} \tag{26}$$

where $f(x)$ is no longer set to zero. Note that this particular discretization is based on the fact that the interface lies between x_k and x_{k+1} and that $\phi_{k+\frac{1}{2}}$ implies that the flux located at $x_{k+\frac{1}{2}}$ lies in Ω^-. If instead $\phi_{k+\frac{1}{2}}$ implies that the flux located at $x_{k+\frac{1}{2}}$ lies in Ω^+ then equations 25 and 26 are replaced with

$$\frac{\left(\frac{(u_{k+1}-a_\Gamma)-u_k}{\triangle x}\right) - \left(\frac{u_k-u_{k-1}}{\triangle x}\right)}{\triangle x} = f_k \tag{27}$$

and

$$\frac{\left(\frac{u_{k+2}-u_{k+1}}{\triangle x}\right) - \left(\frac{u_{k+1}-(u_k+a_\Gamma)}{\triangle x} + b_\Gamma\right)}{\triangle x} = f_{k+1} \tag{28}$$

for use in the linear system.

Note that equations 25 and 26 can be rewritten as

$$\frac{\left(\frac{u_{k+1}-u_k}{\triangle x}\right)-\left(\frac{u_k-u_{k-1}}{\triangle x}\right)}{\triangle x}=f_k+\frac{a_\Gamma}{(\triangle x)^2}+\frac{b_\Gamma}{\triangle x} \tag{29}$$

and

$$\frac{\left(\frac{u_{k+2}-u_{k+1}}{\triangle x}\right)-\left(\frac{u_{k+1}-u_k}{\triangle x}\right)}{\triangle x}=f_{k+1}-\frac{a_\Gamma}{(\triangle x)^2} \tag{30}$$

while equations 27 and 28 can be rewritten as

$$\frac{\left(\frac{u_{k+1}-u_k}{\triangle x}\right)-\left(\frac{u_k-u_{k-1}}{\triangle x}\right)}{\triangle x}=f_k+\frac{a_\Gamma}{(\triangle x)^2} \tag{31}$$

and

$$\frac{\left(\frac{u_{k+2}-u_{k+1}}{\triangle x}\right)-\left(\frac{u_{k+1}-u_k}{\triangle x}\right)}{\triangle x}=f_{k+1}-\frac{a_\Gamma}{(\triangle x)^2}+\frac{b_\Gamma}{\triangle x} \tag{32}$$

to emphasize that this numerical method yields a linear system with no modifications to the coefficient matrix. That is, all modifications occur on the right hand side of the linear system and the coefficients of the unknowns remain equal to those of the standard Laplace equation on a uniform domain. This allows standard "black box" solvers to be used on the associated linear system.

3.1.2 Subcell Resolution

Taking the subcell location of the interface into account allows us to discretize the derivative jump condition more accurately. Assume that the interface lies in between x_k and x_{k+1}, and that $[u]_\Gamma = 0$ and $[u_x]_\Gamma = b_\Gamma$. Then

$$\theta=\frac{|\phi_k|}{|\phi_k|+|\phi_{k+1}|} \tag{33}$$

can be used to estimate the subcell interface location. That is, the interface splits this cell into two pieces of size $\theta\triangle x$ on the left and size $(1-\theta)\triangle x$ on the right. Denoting the value of u at this subcell interface location by u_I, and discretizing the jump condition, $[u_x]_\Gamma = b_\Gamma$, as

$$\left(\frac{u_{k+1}-u_I}{(1-\theta)\triangle x}\right)-\left(\frac{u_I-u_k}{\theta\triangle x}\right)=b_\Gamma \tag{34}$$

allows one to solve for u_I as

$$u_I = u_{k+1}\theta + u_k(1-\theta) - b_\Gamma\theta(1-\theta)\triangle x \tag{35}$$

so that approximations to the derivatives on the left and right sides of the interface can be written as

$$\frac{u_I - u_k}{\theta\triangle x} = \frac{u_{k+1} - u_k}{\triangle x} - b_\Gamma(1-\theta) \tag{36}$$

and

$$\frac{u_{k+1} - u_I}{(1-\theta)\triangle x} = \frac{u_{k+1} - u_k}{\triangle x} + b_\Gamma\theta \tag{37}$$

respectively.

These new approximations allow one to write

$$\frac{\left(\frac{(u_{k+1}-a_\Gamma)-u_k}{\triangle x} - b_\Gamma(1-\theta)\right) - \left(\frac{u_k-u_{k-1}}{\triangle x}\right)}{\triangle x} = f_k \tag{38}$$

and

$$\frac{\left(\frac{u_{k+2}-u_{k+1}}{\triangle x}\right) - \left(\frac{u_{k+1}-(u_k+a_\Gamma)}{\triangle x} + b_\Gamma\theta\right)}{\triangle x} = f_{k+1} \tag{39}$$

in place of equations 25, 26, 27, and 28 where the position of the interface with regard to the subcell flux is no longer a consideration. Likewise,

$$\frac{\left(\frac{u_{k+1}-u_k}{\triangle x}\right) - \left(\frac{u_k-u_{k-1}}{\triangle x}\right)}{\triangle x} = f_k + \frac{a_\Gamma}{(\triangle x)^2} + \frac{b_\Gamma(1-\theta)}{\triangle x} \tag{40}$$

and

$$\frac{\left(\frac{u_{k+2}-u_{k+1}}{\triangle x}\right) - \left(\frac{u_{k+1}-u_k}{\triangle x}\right)}{\triangle x} = f_{k+1} - \frac{a_\Gamma}{(\triangle x)^2} + \frac{b_\Gamma\theta}{\triangle x} \tag{41}$$

replace equations 29, 30, 31, and 32.

3.1.3 The Poisson Equation

Consider the one dimensional variable coefficient Poisson equation

$$(\beta u_x)_x = f(x) \tag{42}$$

with fixed Dirichlet boundary conditions on $\partial\Omega$ and a standard second order discretization of

$$\frac{\beta_{i+\frac{1}{2}}\left(\frac{u_{i+1}-u_i}{\triangle x}\right) - \beta_{i-\frac{1}{2}}\left(\frac{u_i-u_{i-1}}{\triangle x}\right)}{\triangle x} = f_i \tag{43}$$

for each unknown u_i. At the fluxes, $\beta_{i\pm\frac{1}{2}} = \beta(x_{i\pm\frac{1}{2}})$ are defined in accordance with the side of the interface the flux is located on as determined by $\phi_{i\pm\frac{1}{2}}$. Note that this produces a β with no numerical smearing or averaging, i.e. β may be discontinuous across the interface. Next consider the jump conditions $[u]_\Gamma = a(x_\Gamma)$ and $[\beta u_x]_\Gamma = b(x_\Gamma)$. If ϕ_k and ϕ_{k+1} indicate that the interface is located between x_k and x_{k+1}, then equations 16, 23, and 33 can be used to define a_Γ, b_Γ, and θ.

Once again, taking the subcell location of the interface into account and assuming that $[u]_\Gamma = 0$ allows one to discretize the jump condition, $[\beta u_x]_\Gamma = b_\Gamma$, as

$$\beta^+\left(\frac{u_{k+1}-u_I}{(1-\theta)\triangle x}\right) - \beta^-\left(\frac{u_I-u_k}{\theta\triangle x}\right) = b_\Gamma \tag{44}$$

and solve for u_I as

$$u_I = \frac{\beta^+ u_{k+1}\theta + \beta^- u_k(1-\theta) - b_\Gamma\theta(1-\theta)\triangle x}{\beta^+\theta + \beta^-(1-\theta)} \tag{45}$$

so that approximations to the derivatives on the left and right sides of the interface can be written as

$$\beta^-\left(\frac{u_I-u_k}{\theta\triangle x}\right) = \hat{\beta}\left(\frac{u_{k+1}-u_k}{\triangle x}\right) - \frac{\hat{\beta} b_\Gamma(1-\theta)}{\beta^+} \tag{46}$$

and

$$\beta^+\left(\frac{u_{k+1}-u_I}{(1-\theta)\triangle x}\right) = \hat{\beta}\left(\frac{u_{k+1}-u_k}{\triangle x}\right) + \frac{\hat{\beta} b_\Gamma\theta}{\beta^-} \tag{47}$$

where

$$\hat{\beta} = \frac{\beta^+\beta^-}{\beta^+\theta + \beta^-(1-\theta)} \tag{48}$$

defines an effective β.

The discussion above leads naturally to

$$\frac{\hat{\beta}\left(\frac{(u_{k+1}-a_\Gamma)-u_k}{\triangle x} - \frac{b_\Gamma(1-\theta)}{\beta^+}\right) - \beta_{k-\frac{1}{2}}\left(\frac{u_k-u_{k-1}}{\triangle x}\right)}{\triangle x} = f_k \tag{49}$$

and

$$\frac{\beta_{k+\frac{3}{2}}\left(\frac{u_{k+2}-u_{k+1}}{\triangle x}\right) - \hat{\beta}\left(\frac{u_{k+1}-(u_k+a_\Gamma)}{\triangle x} + \frac{b_\Gamma\theta}{\beta^-}\right)}{\triangle x} = f_{k+1} \tag{50}$$

as the equations for the unknowns u_k and u_{k+1} respectively. Of course, these can be rewritten as

$$\frac{\hat{\beta}\left(\frac{u_{k+1}-u_k}{\triangle x}\right) - \beta_{k-\frac{1}{2}}\left(\frac{u_k-u_{k-1}}{\triangle x}\right)}{\triangle x} = f_k + \frac{\hat{\beta}a_\Gamma}{(\triangle x)^2} + \frac{\hat{\beta}b_\Gamma(1-\theta)}{\beta^+\triangle x} \tag{51}$$

and

$$\frac{\beta_{k+\frac{3}{2}}\left(\frac{u_{k+2}-u_{k+1}}{\triangle x}\right) - \hat{\beta}\left(\frac{u_{k+1}-u_k}{\triangle x}\right)}{\triangle x} = f_{k+1} - \frac{\hat{\beta}a_\Gamma}{(\triangle x)^2} + \frac{\hat{\beta}b_\Gamma\theta}{\beta^-\triangle x} \tag{52}$$

to emphasize that this numerical method yields a symmetric linear system with $\beta_{k+\frac{1}{2}} = \hat{\beta}$.

3.1.4 Summary

Consider the variable coefficient Poisson equation

$$(\beta u_x)_x = f(x) \tag{53}$$

with interface jump conditions, $[u]_\Gamma = a(x_\Gamma)$ and $[\beta u_n]_\Gamma = b(x_\Gamma)$. In addition, assume that $\phi \le 0$ in Ω^- and $\phi > 0$ in Ω^+ so that the unit normal points from Ω^- into Ω^+. Since $\vec{N} = n^1 = \pm 1$, one can write $u_n = u_x n^1$ and $[\beta u_n]_\Gamma = [\beta u_x]_\Gamma n^1 = b(x_\Gamma)$. Moreover, $[\beta u_x]_\Gamma = b(x_\Gamma)n^1$.

For each grid point i, one can write a linear equation of the form

$$\frac{\beta_{i+\frac{1}{2}}\left(\frac{u_{i+1}-u_i}{\triangle x}\right)-\beta_{i-\frac{1}{2}}\left(\frac{u_i-u_{i-1}}{\triangle x}\right)}{\triangle x}=f_i+F^L+F^R \tag{54}$$

and assemble the system of linear equations into matrix form. Each $\beta_{k+\frac{1}{2}}$ is evaluated based on the side of the interface that x_k and x_{k+1} lie on. If x_k and x_{k+1} lie on opposite sides of the interface, then $\beta_{k+\frac{1}{2}}$ is defined along the lines of equation 48 and equation 33. First define ϕ^- and ϕ^+ equal to the values of ϕ_k and ϕ_{k+1} in the obvious fashion, then $\beta_{k+\frac{1}{2}}$ is defined as

$$\frac{\beta^+\beta^-\left(|\phi^-|+|\phi^+|\right)}{\beta^+|\phi^-|+\beta^-|\phi^+|} \tag{55}$$

Consider the left arm of the stencil, i.e. the line segment connecting x_i and x_{i-1}. If both $\phi_i \le 0$ and $\phi_{i-1} \le 0$ or if both $\phi_i > 0$ and $\phi_{i-1} > 0$, then $F^L = 0$. Otherwise, define

$$\theta=\frac{|\phi_{i-1}|}{|\phi_i|+|\phi_{i-1}|} \tag{56}$$

$$a_\Gamma=\frac{a_i|\phi_{i-1}|+a_{i-1}|\phi_i|}{|\phi_i|+|\phi_{i-1}|} \tag{57}$$

and

$$b_\Gamma=\frac{b_i n_i^1|\phi_{i-1}|+b_{i-1}n_{i-1}^1|\phi_i|}{|\phi_i|+|\phi_{i-1}|} \tag{58}$$

where the normal is calculated at each grid node using central differencing. If $\phi_i \le 0$ and $\phi_{i-1} > 0$, then

$$F^L=\frac{\beta_{i-\frac{1}{2}}a_\Gamma}{(\triangle x)^2}-\frac{\beta_{i-\frac{1}{2}}b_\Gamma\theta}{\beta^+\triangle x} \tag{59}$$

otherwise if $\phi_i > 0$ and $\phi_{i-1} \le 0$, then

$$F^L=-\frac{\beta_{i-\frac{1}{2}}a_\Gamma}{(\triangle x)^2}+\frac{\beta_{i-\frac{1}{2}}b_\Gamma\theta}{\beta^-\triangle x} \tag{60}$$

Next consider the right arm of the stencil, i.e. the line segment connecting x_i and x_{i+1}. If both $\phi_i \le 0$ and $\phi_{i+1} \le 0$ or if both $\phi_i > 0$ and $\phi_{i+1} > 0$, then $F^R = 0$. Otherwise, define

$$\theta = \frac{|\phi_{i+1}|}{|\phi_i| + |\phi_{i+1}|} \tag{61}$$

$$a_\Gamma = \frac{a_i|\phi_{i+1}| + a_{i+1}|\phi_i|}{|\phi_i| + |\phi_{i+1}|} \tag{62}$$

and

$$b_\Gamma = \frac{b_i n_i^1|\phi_{i+1}| + b_{i+1}n_{i+1}^1|\phi_i|}{|\phi_i| + |\phi_{i+1}|} \tag{63}$$

where the normal is calculated at each grid node using central differencing. If $\phi_i \le 0$ and $\phi_{i+1} > 0$, then

$$F^R = \frac{\beta_{i+\frac{1}{2}} a_\Gamma}{(\triangle x)^2} + \frac{\beta_{i+\frac{1}{2}} b_\Gamma \theta}{\beta^+ \triangle x} \tag{64}$$

otherwise if $\phi_i > 0$ and $\phi_{i+1} \le 0$,then

$$F^R = -\frac{\beta_{i+\frac{1}{2}} a_\Gamma}{(\triangle x)^2} - \frac{\beta_{i+\frac{1}{2}} b_\Gamma \theta}{\beta^- \triangle x} \tag{65}$$

3.2 Two Dimensions

Consider the two dimensional Poisson equation

$$(\beta u_x)_x + (\beta u_y)_y = f(\vec{x}) \tag{66}$$

with interface jump conditions, $[u]_\Gamma = a(\vec{x}_\Gamma)$ and $[\beta u_n]_\Gamma = b(\vec{x}_\Gamma)$. The unit normal is $\vec{N} = (n^1, n^2)$ with $\phi \le 0$ in Ω^- and $\phi > 0$ in Ω^+ implying that the unit normal points from Ω^- into Ω^+.

The normal and tangential derivatives can be defined in terms of u_x, u_y and $\vec{N}$ as

$$u_n = u_x n^1 + u_y n^2 \tag{67}$$

and

$$u_t = u_x n^2 - u_y n^1 \tag{68}$$

respectively. Then

$$u_x = u_n n^1 + u_t n^2 \tag{69}$$

and

$$u_y = u_n n^2 - u_t n^1 \tag{70}$$

follow directly from equations 67 and 68. Multiplying equations 67 and 68 by β and taking the jump across the interface leads to

$$[\beta u_n]_\Gamma = [\beta u_x]_\Gamma n^1 + [\beta u_y]_\Gamma n^2 \tag{71}$$

and

$$[\beta u_t]_\Gamma = [\beta u_x]_\Gamma n^2 - [\beta u_y]_\Gamma n^1 \tag{72}$$

noting that $\vec{N}$ is continuous across the interface. In the same fashion,

$$[\beta u_x]_\Gamma = [\beta u_n]_\Gamma n^1 + [\beta u_t]_\Gamma n^2 \tag{73}$$

and

$$[\beta u_y]_\Gamma = [\beta u_n]_\Gamma n^2 - [\beta u_t]_\Gamma n^1 \tag{74}$$

can be obtained from equations 69 and 70.

Suppose that

$$[\beta u_x]_\Gamma = [\beta u_n]_\Gamma n^1 \tag{75}$$

and

$$[\beta u_y]_\Gamma = [\beta u_n]_\Gamma n^2 \tag{76}$$

are used in place of equations 73 and 74. While equations 75 and 76 are false in general, they still lead to an identity when plugged into equation 71. However, they lead to $[\beta u_t]_\Gamma = 0$ when plugged into equation 72. That is, equations 75 and 76 allow one to correctly capture the jump in the normal derivative while smearing out the jump in the tangential derivative. More importantly, equations 75 and 76 allow the derivative jump condition, $[\beta u_n]_\Gamma = b(\vec{x}_\Gamma)$, to be rewritten as two separate jump conditions, $[\beta u_x]_\Gamma = b(\vec{x}_\Gamma)n^1$ and $[\beta u_y]_\Gamma = b(\vec{x}_\Gamma)n^2$, allowing a dimension by dimension application of the numerical method.

In two dimensions, each grid point (i,j) is discretized as

$$\frac{\beta_{i+\frac{1}{2},j}\left(\frac{u_{i+1,j}-u_{i,j}}{\triangle x}\right)-\beta_{i-\frac{1}{2},j}\left(\frac{u_{i,j}-u_{i-1,j}}{\triangle x}\right)}{\triangle x}+ \\ \frac{\beta_{i,j+\frac{1}{2}}\left(\frac{u_{i,j+1}-u_{i,j}}{\triangle y}\right)-\beta_{i,j-\frac{1}{2}}\left(\frac{u_{i,j}-u_{i,j-1}}{\triangle y}\right)}{\triangle x} = f_{i,j}+F^x+F^y \tag{77}$$

and included in the linear system of equations. Each $\beta_{k+\frac{1}{2},j}$ is evaluated based on the side of the interface that $x_{k,j}$ and $x_{k+1,j}$ lie on. If $x_{k,j}$ and $x_{k+1,j}$ lie on opposite sides of the interface, then ϕ^- and ϕ^+ are set equal to the values of $\phi_{k,j}$ and $\phi_{k+1,j}$ in the obvious fashion, then $\beta_{k+\frac{1}{2},j}$ is defined according to equation 55. Similarly, $\beta_{i,k+\frac{1}{2}}$ is defined according to $x_{i,k}$ and $x_{i,k+1}$. In addition, note that $F^x = F^L + F^R$ and $F^y = F^B + F^T$.

Consider the left arm of the stencil, i.e. the line segment connecting $x_{i,j}$ and $x_{i-1,j}$. If both $\phi_{i,j} \le 0$ and $\phi_{i-1,j} \le 0$ or if both $\phi_{i,j} > 0$ and $\phi_{i-1,j} > 0$, then $F^L = 0$. Otherwise, define

$$\theta = \frac{|\phi_{i-1,j}|}{|\phi_{i,j}| + |\phi_{i-1,j}|} \tag{78}$$

$$a_\Gamma = \frac{a_{i,j}|\phi_{i-1,j}| + a_{i-1,j}|\phi_{i,j}|}{|\phi_{i,j}| + |\phi_{i-1,j}|} \tag{79}$$

and

$$b_\Gamma = \frac{b_{i,j}n^1_{i,j}|\phi_{i-1,j}| + b_{i-1,j}n^1_{i-1,j}|\phi_{i,j}|}{|\phi_{i,j}| + |\phi_{i-1,j}|} \tag{80}$$

where the components of the normal are calculated at each grid point with central differencing. If $\phi_{i,j} \le 0$ and $\phi_{i-1,j} > 0$, then

$$F^L = \frac{\beta_{i-\frac{1}{2},j}a_\Gamma}{(\triangle x)^2} - \frac{\beta_{i-\frac{1}{2},j}b_\Gamma\theta}{\beta^+\triangle x} \tag{81}$$

otherwise if $\phi_{i,j} > 0$ and $\phi_{i-1,j} \le 0$, then

$$F^L = -\frac{\beta_{i-\frac{1}{2},j}a_\Gamma}{(\triangle x)^2} + \frac{\beta_{i-\frac{1}{2},j}b_\Gamma\theta}{\beta^-\triangle x} \tag{82}$$

Consider the right arm of the stencil, i.e. the line segment connecting $x_{i,j}$ and $x_{i+1,j}$. If both $\phi_{i,j} \le 0$ and $\phi_{i+1,j} \le 0$ or if both $\phi_{i,j} > 0$ and $\phi_{i+1,j} > 0$, then $F^R = 0$. Otherwise, define

$$\theta = \frac{|\phi_{i+1,j}|}{|\phi_{i,j}| + |\phi_{i+1,j}|} \tag{83}$$

$$a_\Gamma = \frac{a_{i,j}|\phi_{i+1,j}| + a_{i+1,j}|\phi_{i,j}|}{|\phi_{i,j}| + |\phi_{i+1,j}|} \tag{84}$$

and

$$b_\Gamma = \frac{b_{i,j}n^1_{i,j}|\phi_{i+1,j}| + b_{i+1,j}n^1_{i+1,j}|\phi_{i,j}|}{|\phi_{i,j}| + |\phi_{i+1,j}|} \tag{85}$$

where the components of the normal are calculated at each grid point with central differencing. If $\phi_{i,j} \leq 0$ and $\phi_{i+1,j} > 0$, then

$$F^R = \frac{\beta_{i+\frac{1}{2},j}a_\Gamma}{(\triangle x)^2} + \frac{\beta_{i+\frac{1}{2},j}b_\Gamma\theta}{\beta^+\triangle x} \tag{86}$$

otherwise if $\phi_{i,j} > 0$ and $\phi_{i+1,j} \leq 0$, then

$$F^L = -\frac{\beta_{i+\frac{1}{2},j}a_\Gamma}{(\triangle x)^2} - \frac{\beta_{i+\frac{1}{2},j}b_\Gamma\theta}{\beta^-\triangle x} \tag{87}$$

Consider the bottom arm of the stencil, i.e. the line segment connecting $x_{i,j}$ and $x_{i,j-1}$. If both $\phi_{i,j} \leq 0$ and $\phi_{i,j-1} \leq 0$ or if both $\phi_{i,j} > 0$ and $\phi_{i,j-1} > 0$, then $F^B = 0$. Otherwise, define

$$\theta = \frac{|\phi_{i,j-1}|}{|\phi_{i,j}| + |\phi_{i,j-1}|} \tag{88}$$

$$a_\Gamma = \frac{a_{i,j}|\phi_{i,j-1}| + a_{i,j-1}|\phi_{i,j}|}{|\phi_{i,j}| + |\phi_{i,j-1}|} \tag{89}$$

and

$$b_\Gamma = \frac{b_{i,j}n^2_{i,j}|\phi_{i,j-1}| + b_{i,j-1}n^2_{i,j-1}|\phi_{i,j}|}{|\phi_{i,j}| + |\phi_{i,j-1}|} \tag{90}$$

where the components of the normal are calculated at each grid point with central differencing. If $\phi_{i,j} \leq 0$ and $\phi_{i,j-1} > 0$, then

$$F^B = \frac{\beta_{i,j-\frac{1}{2}}a_\Gamma}{(\triangle y)^2} - \frac{\beta_{i,j-\frac{1}{2}}b_\Gamma\theta}{\beta^+\triangle y} \tag{91}$$

otherwise if $\phi_{i,j} > 0$ and $\phi_{i,j-1} \leq 0$, then

$$F^B = -\frac{\beta_{i,j-\frac{1}{2}}a_\Gamma}{(\triangle y)^2} + \frac{\beta_{i,j-\frac{1}{2}}b_\Gamma\theta}{\beta^-\triangle y} \tag{92}$$

Consider the top arm of the stencil, i.e. the line segment connecting $x_{i,j}$ and $x_{i,j+1}$. If both $\phi_{i,j} \le 0$ and $\phi_{i,j+1} \le 0$ or if both $\phi_{i,j} > 0$ and $\phi_{i,j+1} > 0$, then $F^T = 0$. Otherwise, define

$$\theta = \frac{|\phi_{i,j+1}|}{|\phi_{i,j}| + |\phi_{i,j+1}|} \tag{93}$$

$$a_\Gamma = \frac{a_{i,j}|\phi_{i,j+1}| + a_{i,j+1}|\phi_{i,j}|}{|\phi_{i,j}| + |\phi_{i,j+1}|} \tag{94}$$

and

$$b_\Gamma = \frac{b_{i,j}n^2_{i,j}|\phi_{i,j+1}| + b_{i,j+1}n^2_{i,j+1}|\phi_{i,j}|}{|\phi_{i,j}| + |\phi_{i,j+1}|} \tag{95}$$

where the components of the normal are calculated at each grid point with central differencing. If $\phi_{i,j} \le 0$ and $\phi_{i,j+1} > 0$, then

$$F^T = \frac{\beta_{i,j+\frac{1}{2}} a_\Gamma}{(\triangle y)^2} + \frac{\beta_{i,j+\frac{1}{2}} b_\Gamma \theta}{\beta^+ \triangle y} \tag{96}$$

otherwise if $\phi_{i,j} > 0$ and $\phi_{i,j+1} \le 0$, then

$$F^T = -\frac{\beta_{i,j+\frac{1}{2}} a_\Gamma}{(\triangle y)^2} - \frac{\beta_{i,j+\frac{1}{2}} b_\Gamma \theta}{\beta^- \triangle y} \tag{97}$$

3.3 Three Dimensions

Consider the three dimensional Poisson equation

$$(\beta u_x)_x + (\beta u_y)_y + (\beta u_z)_z = f(\vec{x}) \tag{98}$$

with interface jump conditions, $[u]_\Gamma = a(\vec{x}_\Gamma)$ and $[\beta u_n]_\Gamma = b(\vec{x}_\Gamma)$. The unit normal is $\vec{N} = (n^1, n^2, n^3)$ with $\phi \le 0$ in Ω^- and $\phi > 0$ in Ω^+ implying that the unit normal points from Ω^- into Ω^+.

The normal derivative is defined by,

$$u_n = u_x n^1 + u_y n^2 + u_z n^3 \tag{99}$$

leading to

$$[\beta u_n]_\Gamma = [\beta u_x]_\Gamma n^1 + [\beta u_y]_\Gamma n^2 + [\beta u_z]_\Gamma n^3 \tag{100}$$

since $\vec{N}$ is continuous across the interface. Note that $[\beta u_x]_\Gamma = [\beta u_n]_\Gamma n^1$, $[\beta u_y]_\Gamma = [\beta u_n]_\Gamma n^2$, and $[\beta u_z]_\Gamma = [\beta u_n]_\Gamma n^3$ are not identities, but they do lead to the correct jump condition on the normal derivative when plugged into equation 100, although the tangential derivatives are smeared out. Rewriting $[\beta u_n]_\Gamma = b(\vec{x}_\Gamma)$ as three separate jump conditions, $[\beta u_x]_\Gamma = b(\vec{x}_\Gamma)n^1$, $[\beta u_y]_\Gamma = b(\vec{x}_\Gamma)n^2$, and $[\beta u_z]_\Gamma = b(\vec{x}_\Gamma)n^3$ allows the three dimensional numerical method to be applied in a dimension by dimension fashion with a discretization of

$$\begin{aligned}
&\frac{\beta_{i+\frac{1}{2},j}\left(\frac{u_{i+1,j}-u_{i,j}}{\triangle x}\right)-\beta_{i-\frac{1}{2},j}\left(\frac{u_{i,j}-u_{i-1,j}}{\triangle x}\right)}{\triangle x}+ \\
&\frac{\beta_{i,j+\frac{1}{2}}\left(\frac{u_{i,j+1}-u_{i,j}}{\triangle y}\right)-\beta_{i,j-\frac{1}{2}}\left(\frac{u_{i,j}-u_{i,j-1}}{\triangle y}\right)}{\triangle x}+ \\
&\frac{\beta_{i,j+\frac{1}{2}}\left(\frac{u_{i,j+1}-u_{i,j}}{\triangle y}\right)-\beta_{i,j-\frac{1}{2}}\left(\frac{u_{i,j}-u_{i,j-1}}{\triangle y}\right)}{\triangle x} = f_{i,j} + F^x + F^y + F^z
\end{aligned} \tag{101}$$

for the grid point (i, j, k).

The remaining details are left to the reader, noting that they are a straightforward extension of the two dimensional discretization.

4 Examples

4.1 Example 1

Here we consider two examples that were discussed earlier in the text. Consider $u_{xx} = 0$ on $[0, 1]$ with $u(0) = 0$ and $u(1) = 2$. The interface is located at $x = .5$ with $[u] = 1$ and $[u_x] = 0$. Figure 1 shows the solution computed with 100 grid points plotted on top of the exact solution of $u = x$ to the left of $x = .5$ and $u = x + 1$ to the right of $x = .5$. In figure 2, the boundary conditions and jump conditions are changed to $u(0) = 0$, $u(1) = 1.5$, $[u] = 0$, and $[u_x] = 1$. Once again the solution was computed with 100 grid points and plotted on top of the exact solution of $u = x$ to the left of $x = .5$ and $u = 2x - .5$ to the right of $x = .5$. Note the crisp representation of the interface characteristic of the GFM with no numerical smearing.

4.2 Example 2

Consider $(\beta u_x)_x = f(x)$ on $[0, 1]$ with $u(0) = 0$ and $u(1) = 0$. The interior region is defined by $|x - .45| \leq .15$ with the unit normal pointing from the interior region to the exterior region. On the interior region $\beta = 2$ and $f(x) = (8x^2 - 4)e^{-x^2}$ while on the exterior region $\beta = 1$ and $f(x) = 0$. At $x = .3$, $[u] = -e^{-.09}$ and $[\beta u_n] = -1.2e^{-.09}$ while at $x = .6$, $[u] = -e^{-.36}$ and $[\beta u_n] = 2.4e^{-.36}$. Figure 3 shows the solution computed with 100 grid points plotted on top of the exact solution of $u(x) = e^{-x^2}$ on the interior region and $u(x) = 0$ on the exterior region.

4.3 Example 3

Consider $\nabla \cdot (\beta \nabla u) = f(x, y)$ in two spatial dimensions on $[0, 1] \times [0, 1]$ with the interface defined by the circle $(x - 0.5)^2 + (y - 0.5)^2 = .25^2$ with an outward pointing normal vector, $\vec{N} = (4x - 2, 4y - 2)$. As an exact solution, $u(x, y) = e^{-x^2-y^2}$ on the interior of the circle and $u(x, y) = 0$ on the exterior of the circle with the appropriate Dirichlet boundary conditions. $\beta = 2$ with $f(x, y) = 8(x^2 + y^2 - 1)e^{-x^2-y^2}$ on the interior of the circle and $\beta = 1$ with $f(x, y) = 0$ on the exterior of the circle. The jump conditions are $[u] = -e^{-x^2-y^2}$ and $[\beta u_n] = 8(2x^2 + 2y^2 - x - y)e^{-x^2-y^2}$. Figure 4 shows the numerical solution with 61 grid points in each direction and Table 1 shows the results of numerical accuracy tests.

4.4 Example 4

Consider $\nabla \cdot (\beta \nabla u) = f(x, y, z)$ in three spatial dimensions on $[0,1] \times [0,1] \times [0,1]$ with the interface defined by the sphere $(x - 0.5)^2 + (y - 0.5)^2 + (z - 0.5)^2 = .25^2$ with an outward pointing normal vector, $\vec{N} = (4x - 2, 4y - 2, 4z - 2)$. As an exact solution, $u(x, y, z) = e^{-x^2-y^2-z^2}$ on the interior of the sphere and $u(x, y, z) = 0$ on the exterior of the sphere with the appropriate Dirichlet boundary conditions. $\beta = 2$ with $f(x, y, z) = 8(x^2 + y^2 + z^2 - \frac{3}{2})e^{-x^2-y^2-z^2}$ on the interior of the sphere and $\beta = 1$ with $f(x, y, z) = 0$ on the exterior of the sphere. The jump conditions are $[u] = -e^{-x^2-y^2-z^2}$ and $[\beta u_n] = 8(2x^2 + 2y^2 + 2z^2 - x - y - z)e^{-x^2-y^2-z^2}$. Figure 5 shows the $u(x, y, z = .4)$ cross-section of the the numerical solution with 61 grid points in each direction and Table 2 shows the results of numerical accuracy tests for the two dimensional slice of data.

4.5 Example 5

This example was taken from [10]. Consider $\triangle u = 0$ in two spatial dimensions on $[-1,1] \times [-1,1]$ with the interface defined by the circle $x^2 + y^2 = .5^2$ with an outward pointing normal vector, $\vec{N} = (2x, 2y)$. As an exact solution, $u(x, y) = 1$ on the interior of the circle and $u(x, y) = 1 + \ln(2\sqrt{x^2 + y^2})$ on the exterior of the circle with the appropriate Dirichlet boundary conditions. The jump conditions are $[u] = 0$ and $[u_n] = 2$. Figure 6 shows the numerical solution with 61 grid points in each direction and Table 3 shows the results of numerical accuracy tests.

4.6 Example 6

This example was taken from [10]. Consider $\triangle u = 0$ in two spatial dimensions on $[-1,1] \times [-1,1]$ with the interface defined by the circle $x^2 + y^2 = .5^2$ with an outward pointing normal vector, $\vec{N} = (2x, 2y)$. As an exact solution, $u(x, y) = e^x \cos(y)$ on the interior of the circle and $u(x, y) = 0$ on the exterior of the circle with the appropriate Dirichlet boundary conditions. The jump conditions are $[u] = -e^x \cos(y)$ and $[u_n] = 2e^x(y \sin(y) - x \cos(y))$. Figure 7 shows the numerical solution with 61 grid points in each direction and Table 4 shows the results of numerical accuracy tests.

4.7 Example 7

This example was taken from [10]. Consider $\triangle u = 0$ in two spatial dimensions on $[-1,1]\times[-1,1]$ with the interface defined by the circle $x^2+y^2 = .5^2$ with an outward pointing normal vector, $\vec{N} = (2x, 2y)$. As an exact solution, $u(x,y) = x^2 - y^2$ on the interior of the circle and $u(x,y) = 0$ on the exterior of the circle with the appropriate Dirichlet boundary conditions. The jump conditions are $[u] = y^2 - x^2$ and $[u_n] = 4(y^2 - x^2)$. Figure 8 shows the numerical solution with 61 grid points in each direction and Table 5 shows the results of numerical accuracy tests.

4.8 Example 8

This example was taken from [11]. Consider $\nabla \cdot (\beta \nabla u) = f(x,y)$ in two spatial dimensions on $[-1,1] \times [-1,1]$. The interface is defined by the collection of points $(x(\theta), y(\theta))$ where $x(\theta) = .02\sqrt{5} + (.5 + .2\sin(5\theta))\cos(\theta)$, $y(\theta) = .02\sqrt{5} + (.5 + .2\sin(5\theta))\sin(\theta)$, $\theta \in [0, 2\pi)$, and the unit normal, $\vec{N} = (n_1, n_2)$, is assumed to point from the interior region to the exterior region. As an exact solution, $u(x,y) = x^2 + y^2$ on the interior region and $u(x,y) = .1(x^2+y^2)^2 - .01\ln(2\sqrt{x^2+y^2})$ on the exterior region with the appropriate Dirichlet boundary conditions. $\beta = 1$ with $f(x,y) = 4$ on the interior region and $\beta = 10$ with $f(x,y) = 16(x^2+y^2)$ on the exterior region. The jump conditions are $[u] = .1(x^2+y^2)^2 - .01\ln(2\sqrt{x^2+y^2}) - (x^2+y^2)$ and $[\beta u_n] = (4(x^2+y^2) - .1(x^2+y^2)^{-1} - 2)(xn_1 + yn_2)$. Figure 9 shows the numerical solution with 61 grid points in each direction and Table 6 shows the results of numerical accuracy tests.

4.9 Example 9

This example was taken from [11]. Consider $\nabla \cdot (\beta \nabla u) = f(x,y)$ in two spatial dimensions on $[-1,1] \times [0,3]$. The interface is defined by the collection of points $(x(\theta), y(\theta))$ where $x(\theta) = .6\cos(\theta) - .3\cos(3\theta)$, $y(\theta) = 1.5 + .7\sin(\theta) - .07\sin(3\theta) + .2\sin(7\theta)$, $\theta \in [0, 2\pi)$, and the unit normal, $\vec{N} = (n_1, n_2)$, is assumed to point from the interior region to the exterior region. As an exact solution, $u(x,y) = e^x(x^2\sin(y) + y^2)$ on the interior region and $u(x,y) = -(x^2+y^2)$ on the exterior region with the appropriate Dirichlet boundary conditions. $\beta = 1$ with $f(x,y) = e^x(2+y^2+2\sin(y)+4x\sin(y))$ on the interior region and $\beta = 10$ with $f(x,y) = -40$ on the exterior region. The jump conditions are $[u] = -(x^2+y^2) - e^x(x^2\sin(y) + y^2)$ and $[\beta u_n] = (-20x - e^x((x^2+2x)\sin(y)+y^2))n_1 + (-20y - e^x(x^2\cos(y)+2y))n_2$.

Figure 10 shows the numerical solution with 61 grid points in the x-driection and 91 grid points in the y-direction. Table 7 shows the results of the numerical accuracy tests.

Δx	L^∞-error in U	order	L^2-error in U	order	L^2-error in $\nabla \cdot U$	order
$\frac{1}{20}$	0.0088		0.0027		0.0675	
$\frac{1}{40}$	0.0041	1.07	0.0010	1.43	0.0348	0.95
$\frac{1}{80}$	0.0020	1.04	0.0003	1.73	0.0186	0.90
$\frac{1}{160}$	0.0011	0.86	0.0001	1.58	0.0108	0.78

Table 1

Δx	L^∞-error in U	order	L^2-error in U	order	L^2-error in $\nabla \cdot U$	order
$\frac{1}{20}$	0.0064		7.49e-4		0.0335	
$\frac{1}{40}$	0.0035	0.87	2.37e-4	1.66	0.0142	1.24
$\frac{1}{80}$	0.0023	0.54	1.01e-4	1.23	0.0074	0.94
$\frac{1}{160}$	0.0015	0.68	4.66e-5	1.12	0.0049	0.59

Table 2

Δx	L^∞-error in U	order	L^2-error in U	order	L^2-error in $\nabla \cdot U$	order
$\frac{1}{10}$	0.0326		0.0299		0.1240	
$\frac{1}{20}$	0.0130	1.33	0.0111	1.43	0.0463	1.42
$\frac{1}{40}$	0.0050	1.38	0.0040	1.47	0.0167	1.47
$\frac{1}{80}$	0.0019	1.40	0.0014	1.51	0.0060	1.48

Table 3

Δx	L^∞-error in U	order	L^2-error in U	order	L^2-error in $\nabla \cdot U$	order
$\frac{1}{10}$	0.0153		0.0054		0.1380	
$\frac{1}{20}$	0.0081	0.92	0.0022	1.30	0.0703	0.97
$\frac{1}{40}$	0.0044	0.88	0.0009	1.29	0.0370	0.93
$\frac{1}{80}$	0.0023	0.94	0.0003	1.59	0.0209	0.82

Table 4

Δx	L^∞-error in U	order	L^2-error in U	order	L^2-error in $\nabla \cdot U$	order
$\frac{1}{10}$	0.0068		0.0033		0.0770	
$\frac{1}{20}$	0.0033	1.04	0.0014	1.24	0.0337	1.19
$\frac{1}{40}$	0.0014	1.24	0.0005	1.49	0.0169	1.00
$\frac{1}{80}$	0.0008	0.81	0.0002	1.32	0.0108	0.65

Table 5

Δx	L^∞-error in U	order	L^2-error in U	order	L^2-error in $\nabla \cdot U$	order
$\frac{1}{10}$	4.90e-4		3.69e-4		0.0610	
$\frac{1}{20}$	3.20e-4	0.61	1.53e-4	1.27	0.0221	1.46
$\frac{1}{40}$	1.67e-4	0.94	8.49e-5	0.85	0.0079	1.48
$\frac{1}{80}$	7.35e-5	1.18	3.64e-5	1.22	0.0028	1.50

Table 6

Δx	L^∞-error in U	order	L^2-error in U	order	L^2-error in $\nabla \cdot U$	order
$\frac{1}{10}$	0.2437		0.1136		1.3141	
$\frac{1}{20}$	0.1534	0.67	0.0666	0.77	0.7969	0.72
$\frac{1}{40}$	0.0498	1.62	0.0189	1.82	0.2798	1.51
$\frac{1}{80}$	0.0358	0.48	0.0082	1.20	0.1992	0.49

Table 7

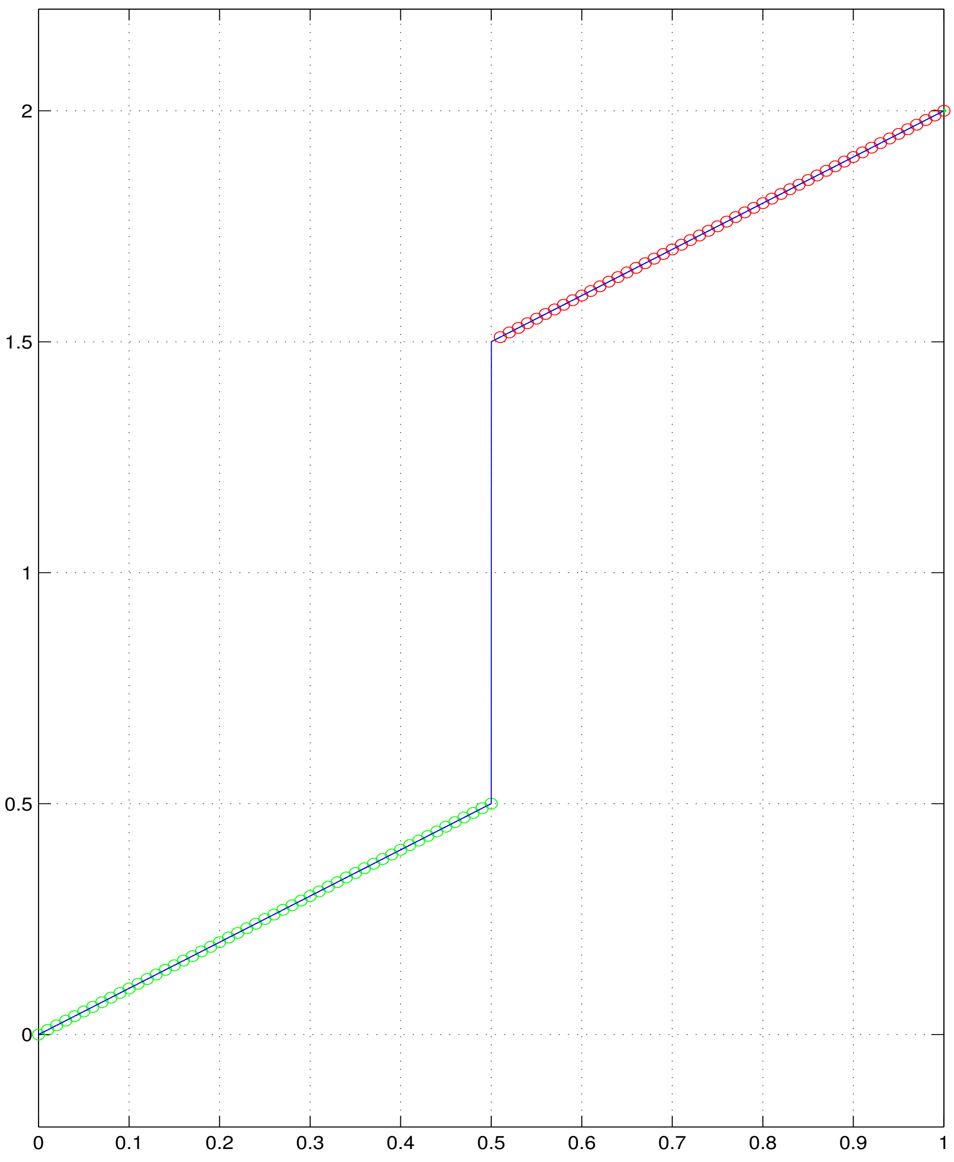

Figure 1: $u_{xx} = 0$, $[u] = 1$, $[u_x] = 0$

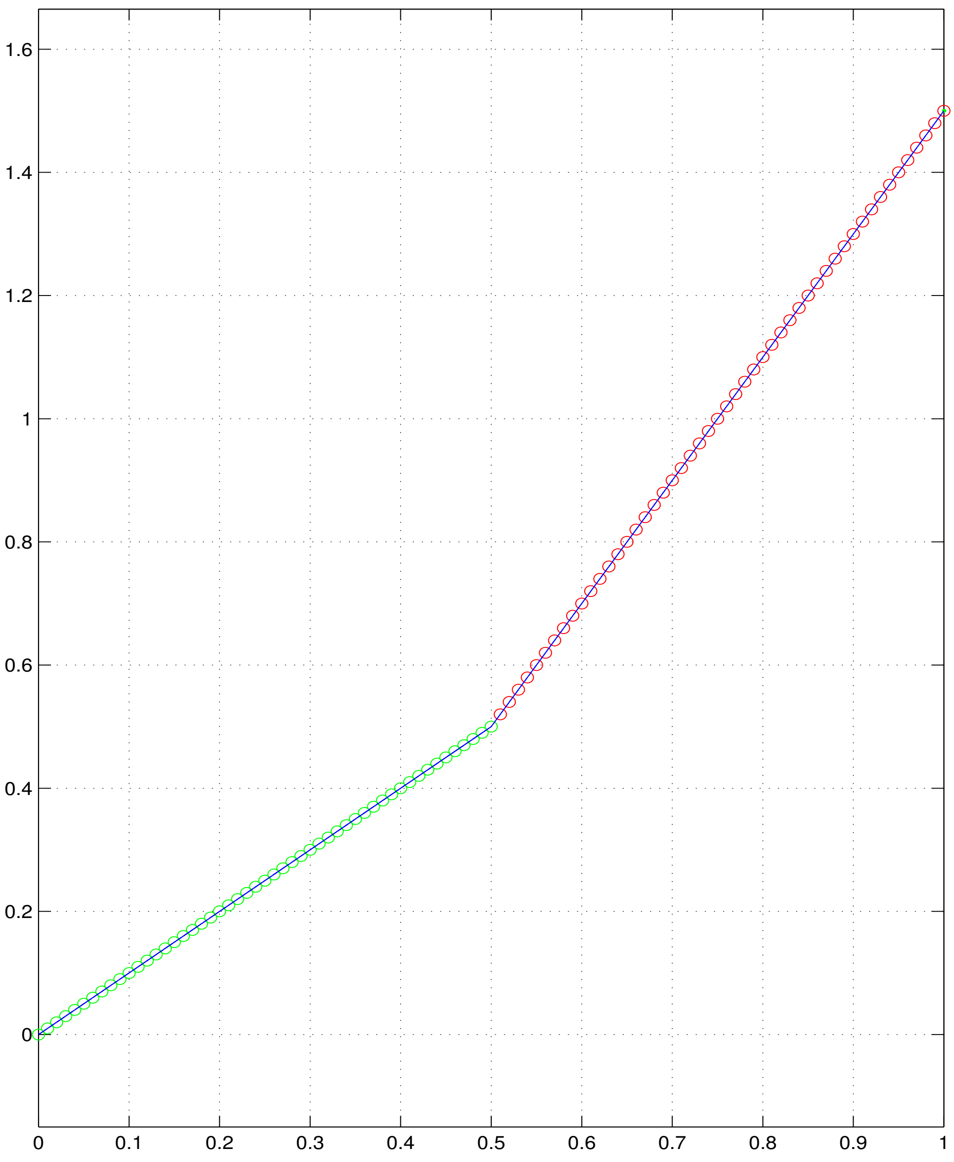

Figure 2: $u_{xx} = 0$, $[u] = 0$, $[u_x] = 1$

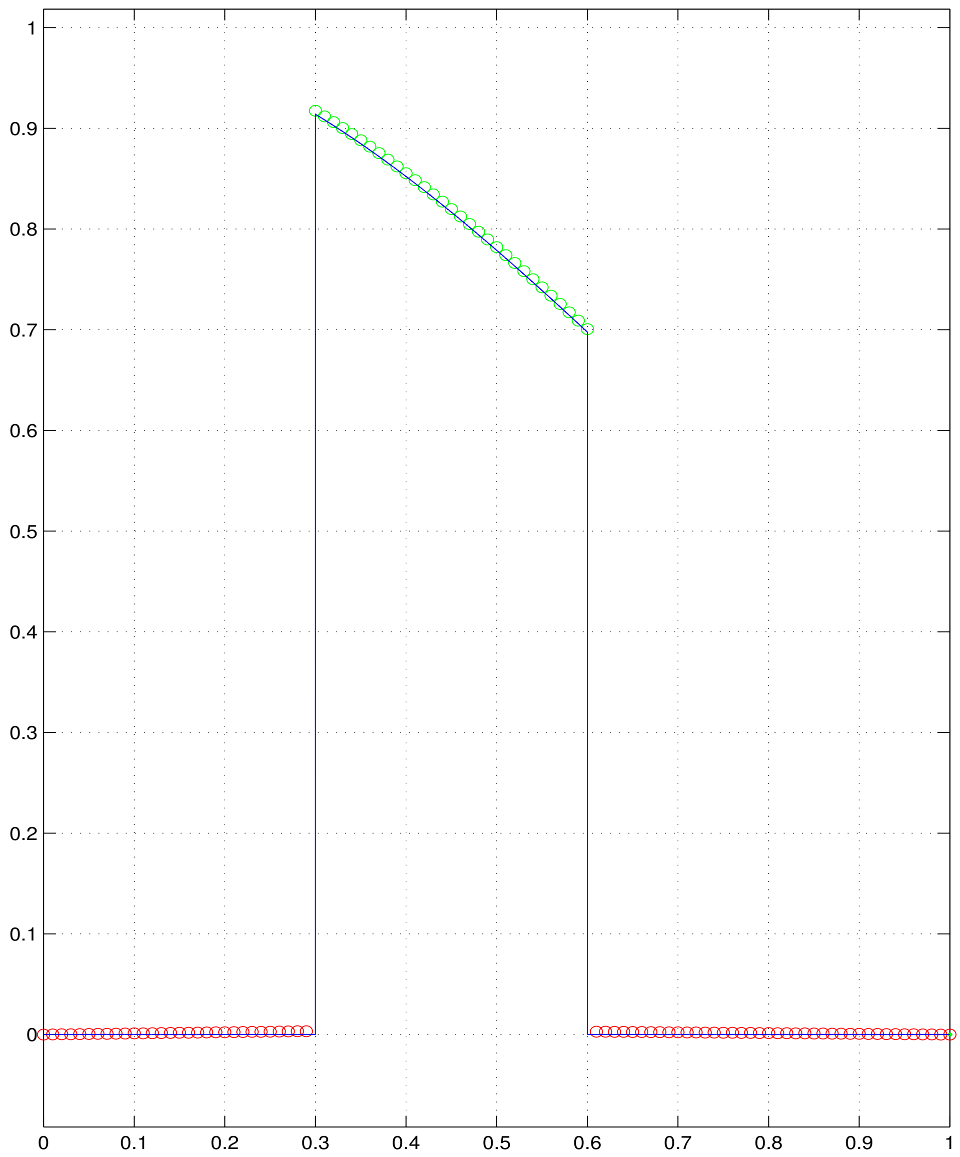

Figure 3: One Spatial Dimension, $\nabla \cdot (\beta \nabla u) = f(x)$, $[u] \neq 0$, $[\beta u_n] \neq 0$

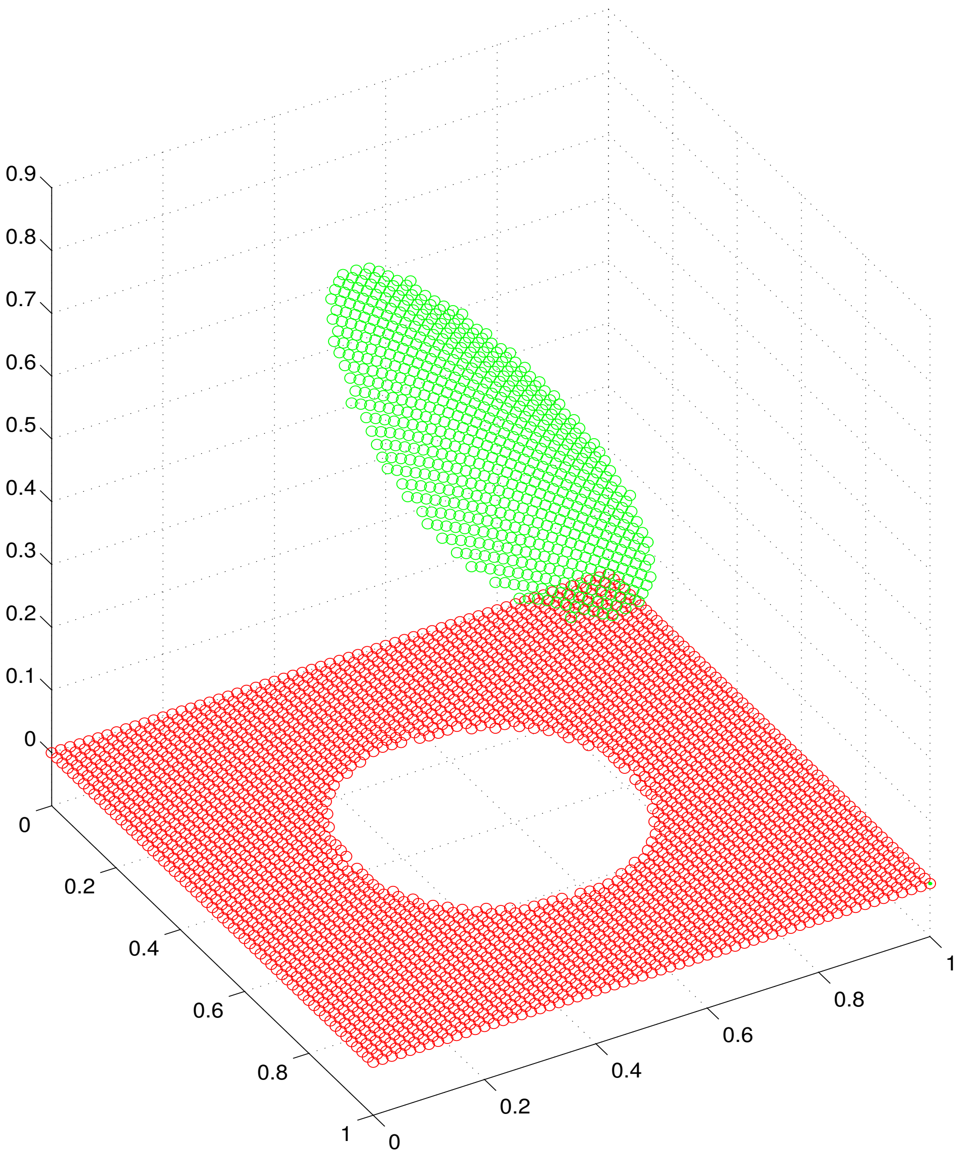

Figure 4: Two Spatial Dimensions, $\nabla \cdot (\beta \nabla u) = f(x, y)$, $[u] \neq 0$, $[\beta u_n] \neq 0$

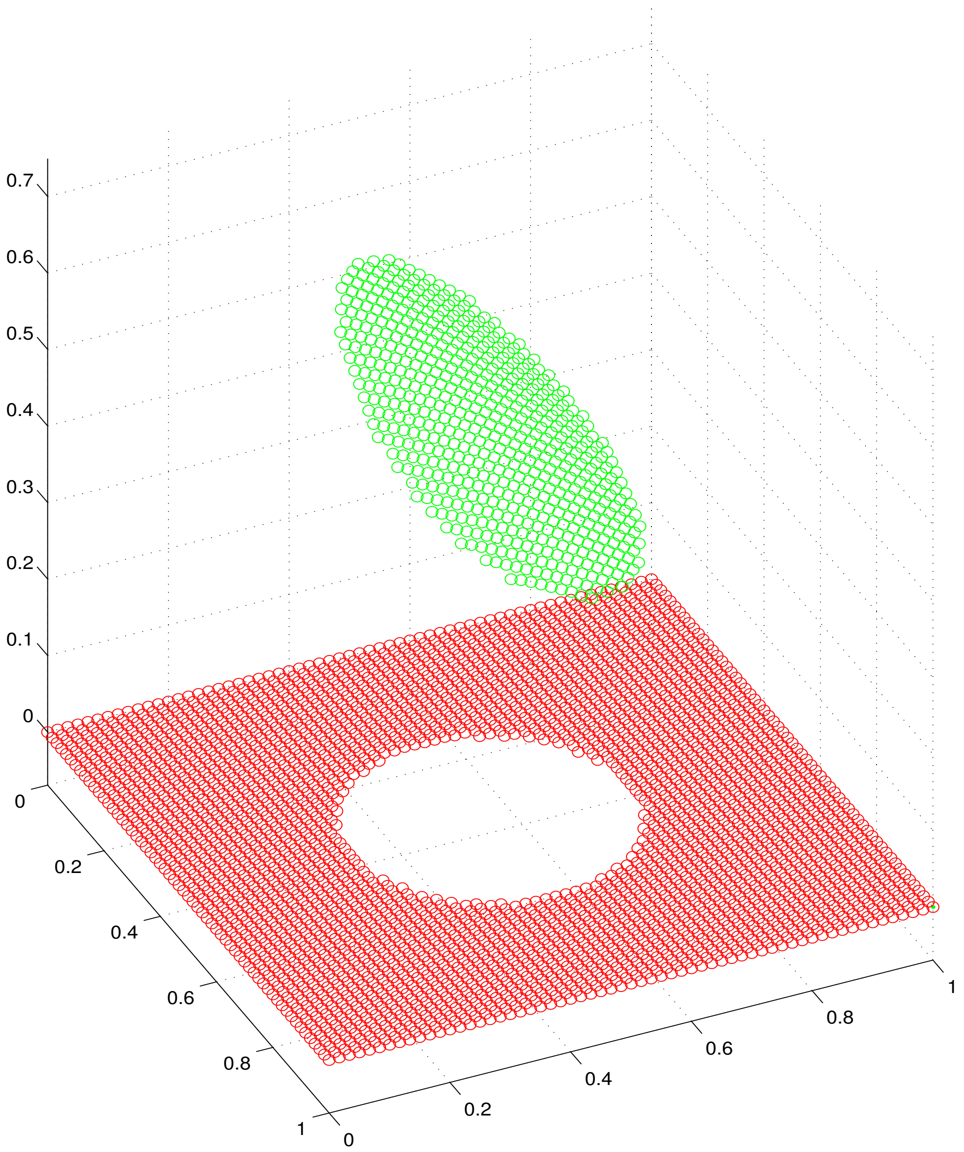

Figure 5: Three Spatial Dimensions (Cross-Section), $\nabla \cdot (\beta \nabla u) = f(x, y, z)$, $[u] \neq 0$, $[\beta u_n] \neq 0$

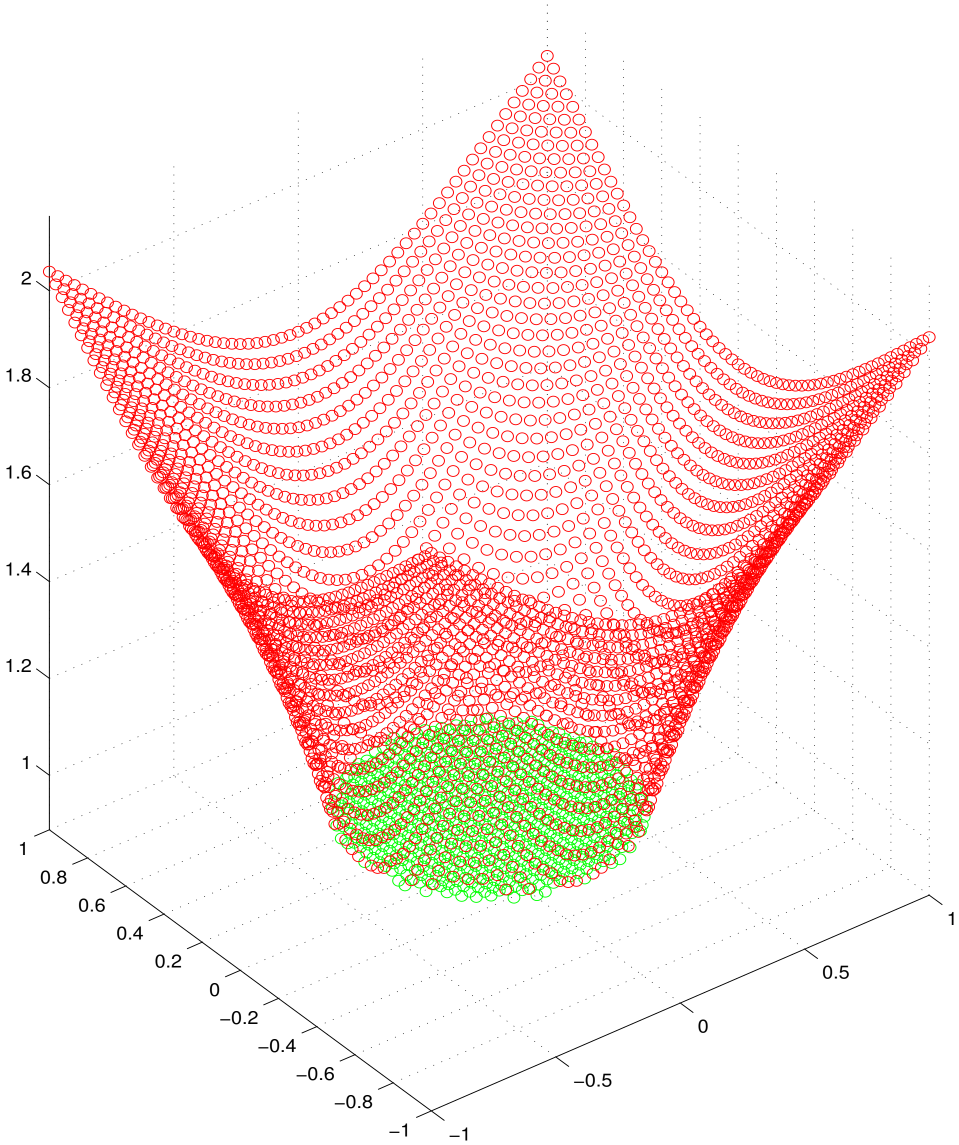

Figure 6: Two Spatial Dimensions, $\triangle u = 0$, $[u] = 0$, $[u_n] \neq 0$

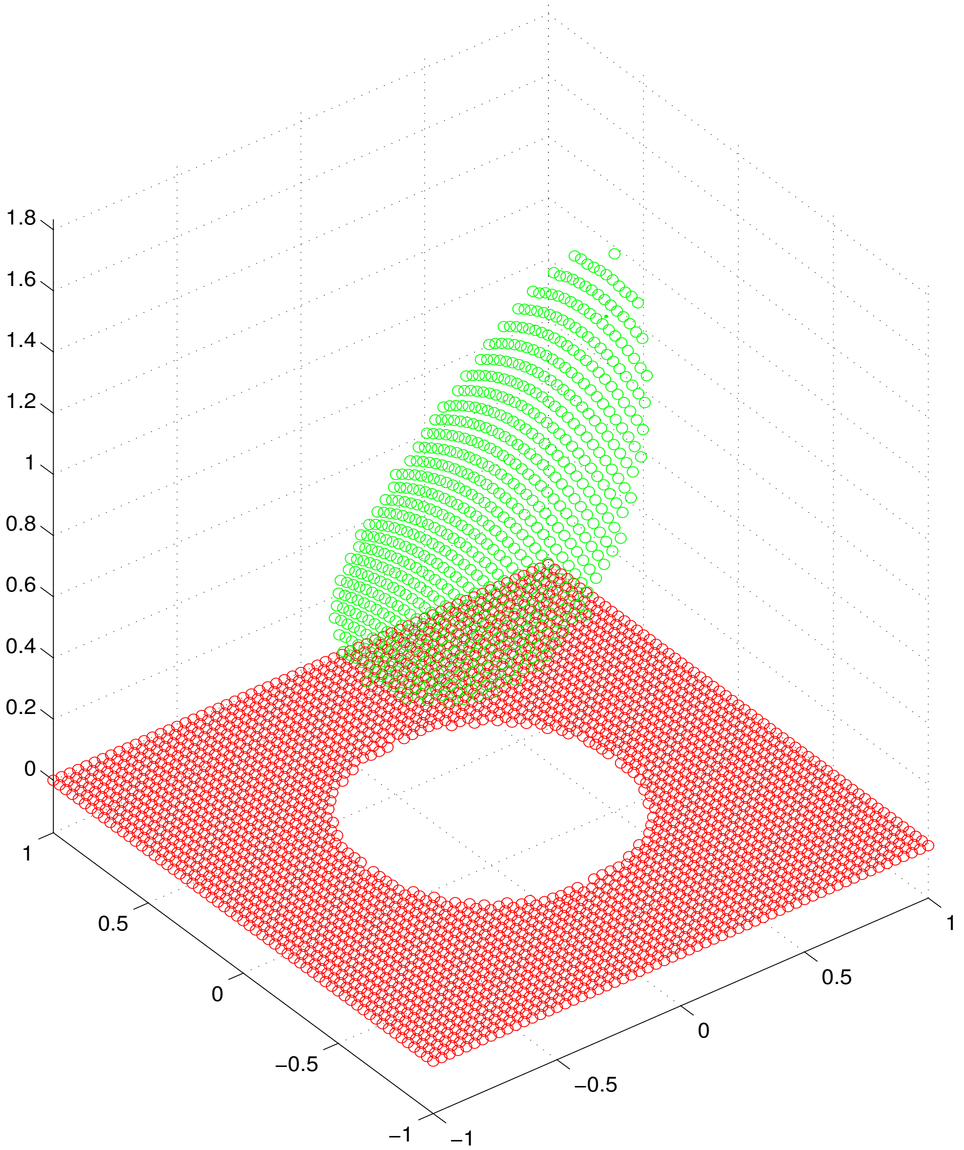

Figure 7: Two Spatial Dimensions, $\triangle u = 0$, $[u] \neq 0$, $[u_n] \neq 0$

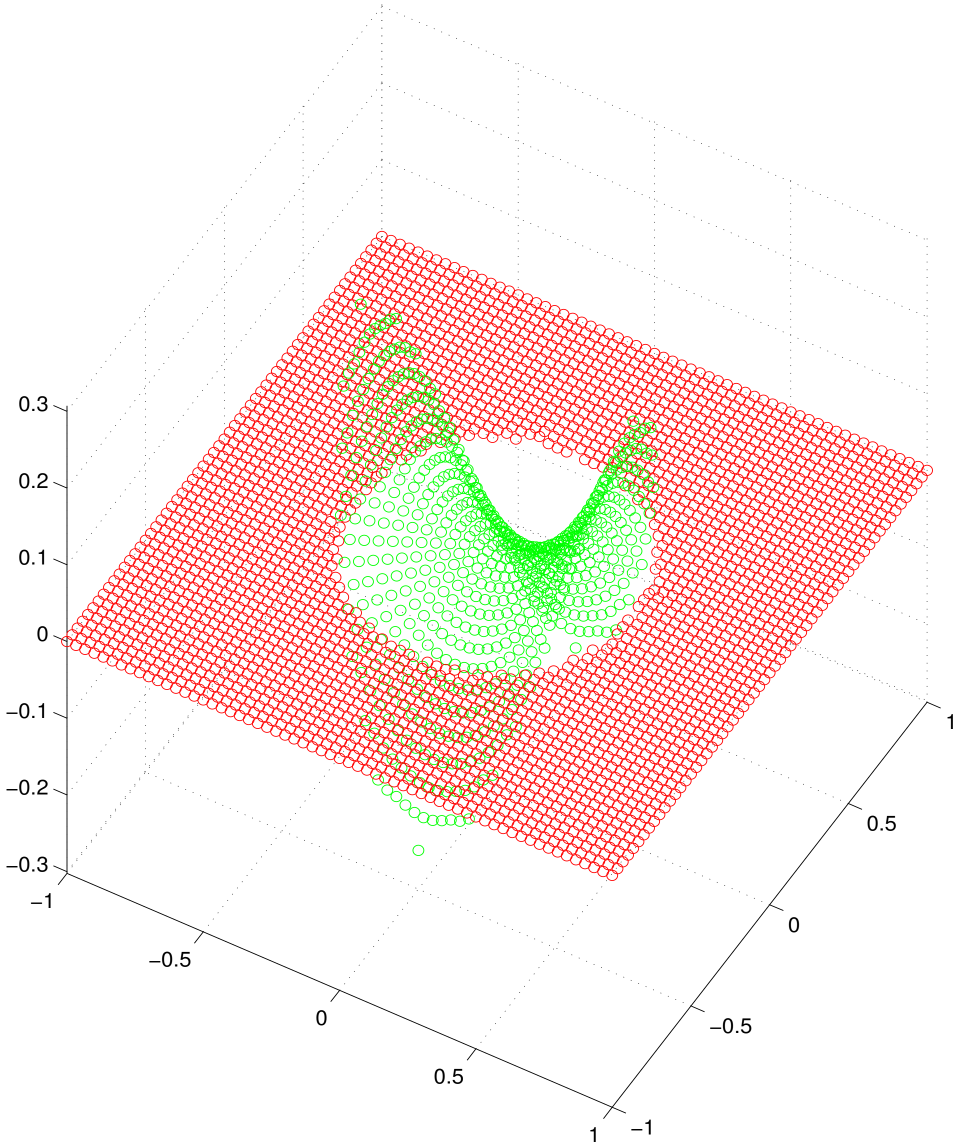

Figure 8: Two Spatial Dimensions, $\triangle u = 0$, $[u] \neq 0$, $[u_n] \neq 0$

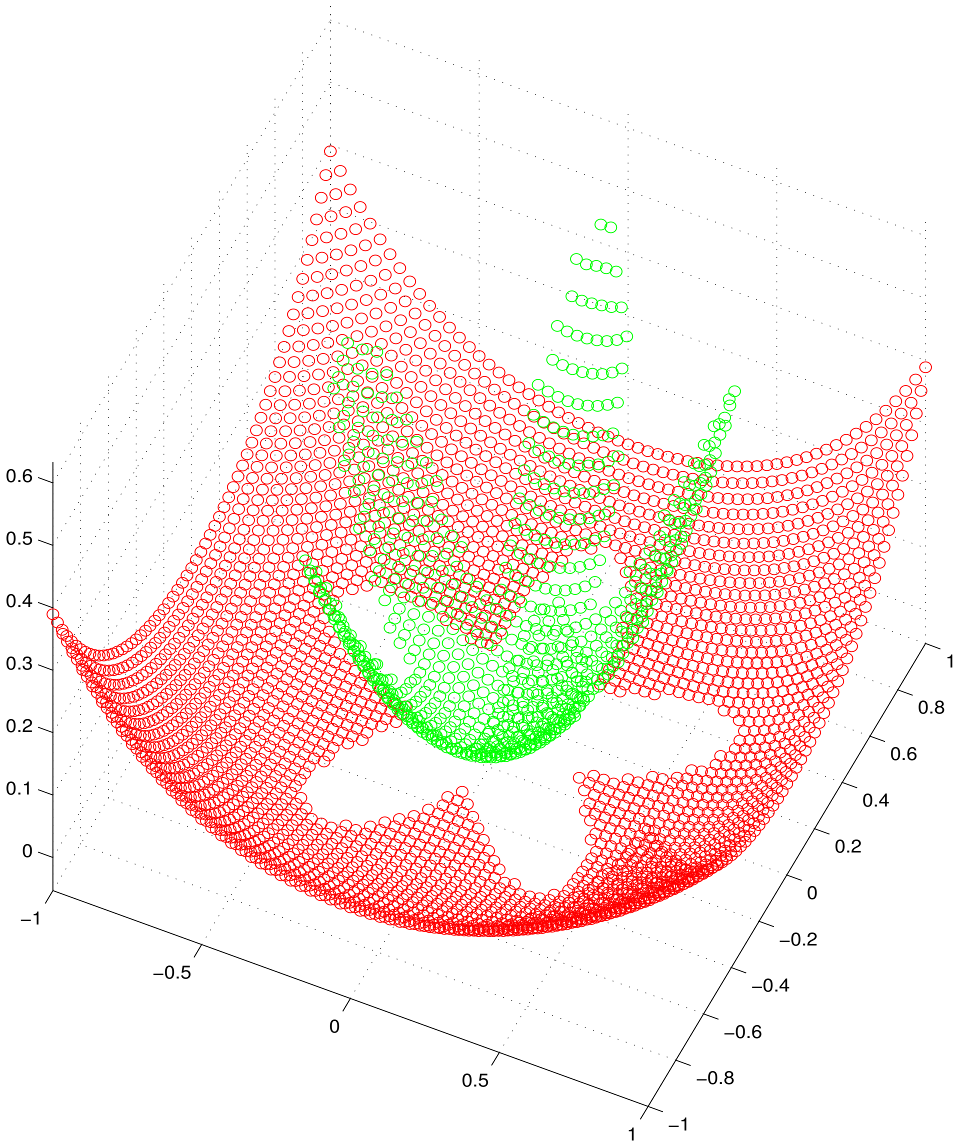

Figure 9: Two Spatial Dimensions, $\nabla \cdot (\beta \nabla u) = f(x, y)$, $[u] \neq 0$, $[\beta u_n] \neq 0$

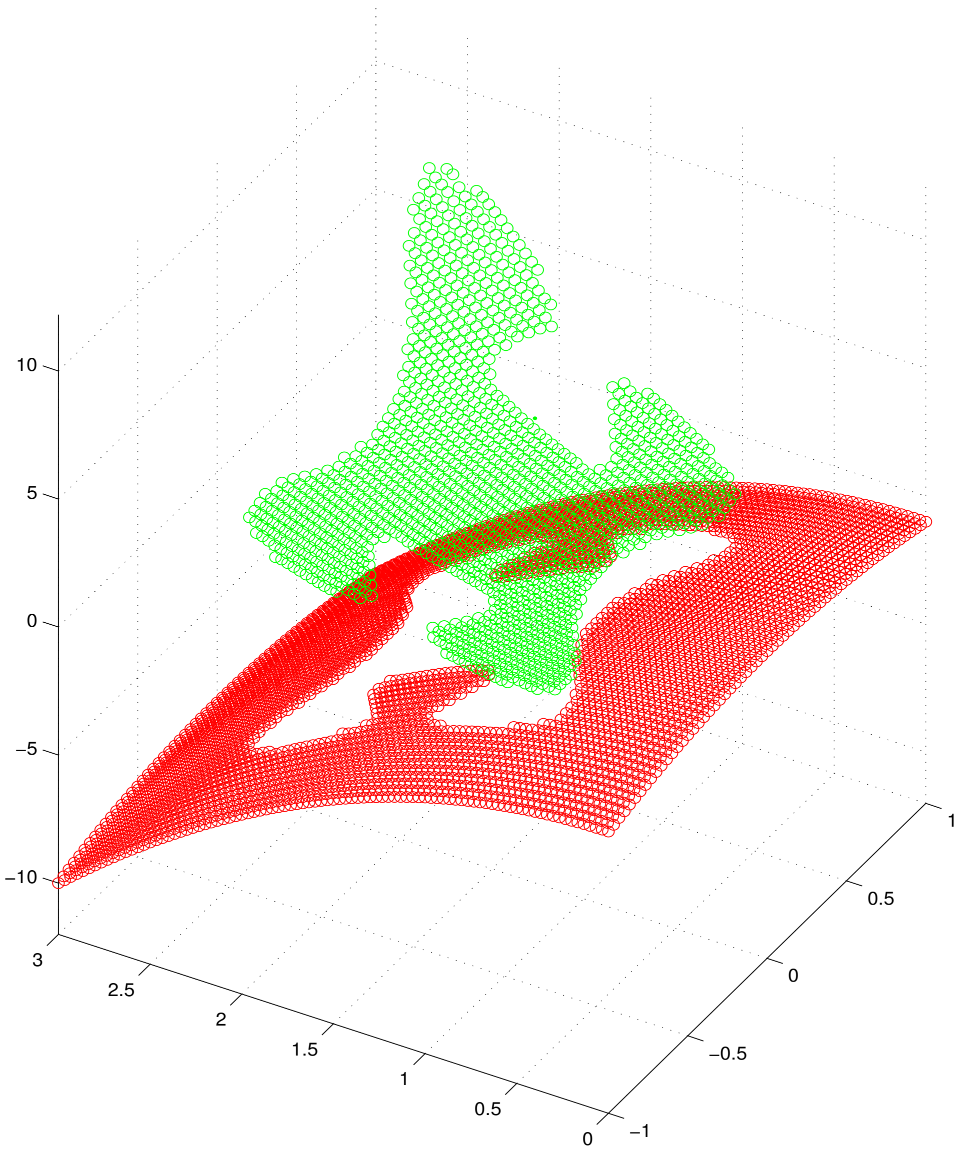

Figure 10: Two Spatial Dimensions, $\nabla \cdot (\beta \nabla u) = f(x, y)$, $[u] \neq 0$, $[\beta u_n] \neq 0$

References

[1] Brackbill, J.U., Kothe, D.B. and Zemach, C., *A Continuum Method for Modeling Surface Tension*, J. Comput. Phys., vol. 100, 335-354 (1992).

[2] Fedkiw, R., Aslam, T., and Xu, S., *The Ghost Fluid Method for deflagration and detonation discontinuities*, J. Computational Physics 154, n. 2, 393-427 (1999).

[3] Fedkiw, R., Aslam, T., Merriman, B., and Osher, S., *A Non-Oscillatory Eulerian Approach to Interfaces in Multimaterial Flows (The Ghost Fluid Method)*, J. Computational Physics, vol. 152, n. 2, 457-492 (1999).

[4] Fedkiw, R., and Liu, X.-D., *The Ghost Fluid Method for Viscous Flows, Progress in Numerical Solutions of Partial Differential Equations*, Arcachon, France, edited by M. Hafez, July 1998.

[5] Greenbaum, A. and Mayo, A., *Rapid Parallel Evaluation of Integrals in Potential Theory on General Three-Dimensional Regions*, J. Comput. Phys., v. 145, pp. 731-742 (1998).

[6] Hou, T., Li, Z., Osher, S., Zhao, H., *A Hybrid Method for Moving Interface Problems with Application to the Hele-Shaw Flow*, Journal of Comput. Phys., vol. 134, pp. 236-252 (1997).

[7] Johansen, H. and Colella, P., *A Cartesian Grid Embedded Boundary Method for Poisson's Equation on Irregular Domains*, Journal of Comput. Phys., vol. 147, pp. 60-85 (1998).

[8] Kang, M., Fedkiw, R., and Liu. X.-D., *A Boundary Condition Captring Mathod for Multiphase Incompressible Flow*, Journal of Comput. Phys. (submitted).

[9] Karni, S., *Multicomponent Flow Calculations by a Consistent Primitive Algorithm*, Journal of Computational Physics, v. 112, 31-43 (1994).

[10] LeVeque, R.J. and Li, Z., *The Immersed Interface Method for Elliptic Equations with Discontinuous Coefficients and Singular Sources*, SIAM J. Numer. Anal., vol. 31, 1019 (1994).

[11] Li, Z., *A Fast Iterative Algorithm for Elliptic Interface Problems*, SIAM J. Numer. Anal., vol. 35, no. 1, pp. 230-254, (1998).

[12] Li, Z., *A Note on Immersed Interface Method for for Three Dimensional Elliptic Equations*, Computers Math. Appl. 35, 9-17, 1996.

[13] Liu, X.-D. and Sideris, T., (in preparation).

[14] Mayo, A., *Fast High Order Accurate Solutions of Laplace's Equation on Irregular Domains*, SIAM J. Sci. Stat. Comput., vol. 6, no. 1, pp. 144-157 (1985).

[15] Mayo, A., *The Fast Solution of Poisson's and the Biharmonic Equations in Irregular Domains*, SIAM J. Numer. Anal., vol. 21, no. 2, pp. 285-299 (1984).

[16] Mayo, A., *The Rapid Evaluation of Volume Integrals of Potential Theory on General Regions*, J. Comput. Phys., v. 100, pp. 236-245 (1992).

[17] Mayo, A., and Greenbaum, A., *Fast Parallel Iterative Solution of Poisson's and the Biharmonic Equations on Irregular Regions*, SIAM J. Sci. Stat. Comput., vol. 13, no. 1, pp. 101-118 (1992).

[18] McKenney, A., Greengard, L., and Mayo, A., *A Fast Poisson Solver for Complex Geometries*, J. Comput. Phys., v. 118, pp. 348-355 (1995).

[19] Mulder, W., Osher, S., and Sethian, J.A., *Computing Interface Motion in Compressible Gas Dynamics*, J. Comput. Phys., v. 100, 209-228 (1992).

[20] Peskin, C., *Numerical Analysis of Blood Flow in the Heart*, Journal of Comput. Phys., vol. 25, pp. 220-252 (1977).

[21] Peskin, C. and Printz, B., *Improved Volume Conservation in the Computation of Flows with Immersed Elastic Boundaries*, J. Comput. Phys., vol. 105, pp. 33-46 (1993).

[22] Osher, S. and Sethian, J.A., *Fronts Propagating with Curvature Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations*, Journal of Comput. Phys., vol. 79, n. 1, pp. 12-49, (1988).

[23] Sussman, M., Smereka, P. and Osher, S., *A level set approach for computing solutions to incompressible two-phase flow*, J. Comput. Phys., vol. 114, pp. 146-154 (1994).

[24] Unverdi, S.O. and Tryggvason, G., *A Front-Tracking Method for Viscous, Incompressible, Multi-Fluid Flows*, J. Comput. Phys., vol. 100, pp. 25-37 (1992).