AN EMBEDDED METHOD-OF-LINES APPROACH TO SOLVING PARTIAL DIFFERENTIAL EQUATIONS ON SURFACES

INGRID VON GLEHN, THOMAS MÄRZ, AND COLIN B. MACDONALD *

Abstract. We introduce a method-of-lines formulation of the closest point method, a numerical technique for solving partial differential equations (PDEs) defined on surfaces. This is an embedding method, which uses an implicit representation of the surface in a band containing the surface. We define a modified equation in the band, obtained in a straightforward way from the original evolution PDE, and show that the solutions of this equation are consistent with those of the surface equation. The resulting system can then be solved with standard implicit or explicit time-stepping schemes, and the solutions in the band can be restricted to the surface. Our derivation generalizes existing formulations of the closest point method and is amenable to standard convergence analysis.

Key words. Closest Point Method, implicit surfaces, partial differential equations, method of lines

AMS subject classifications. 65M06, 58J35, 65M20

1. Introduction. Partial differential equations (PDEs) defined on curved surfaces appear in a variety of physical and biological systems and applications. Examples include fluid flow on surfaces [22], the diffusion of chemicals on cell membranes [21], and texture mapping in computer graphics [30].

The numerical solution of these equations and treatment of surface differential operators is an area of active research. Some methods work directly on the surface, using either a parameterization (for a survey, see [10]), or a triangulation of the surface [8]. Embedding methods form an alternative approach, in which the surface is embedded into a larger space, and a related equation is solved in this surrounding space. Finally, a restriction is used to obtain the solution on the surface. The closest point method [27, 17] is an example of such a technique. Other embedding techniques using an implicit representation of the surface include the level set approach of [2, 13] for variational problems, and finite element methods on implicit surfaces [4, 7]. Recently, methods using radial basis functions [11, 24] have been introduced.

This paper is based on the closest point method, which is applicable to a wide variety of surface geometries, and is simple to implement using standard well-studied numerical techniques on Cartesian grids [27]. It has been applied to a variety of problems, including eigenvalue problems [15], image segmentation [28], image denoising [3], and fluid effects on surfaces [1].

We derive a modified formulation of the closest point embedding equation for evolution PDEs, and show that there is a one-to-one correspondence between solutions of the embedding and surface equations. This formulation is simple to discretize and solve numerically using a standard method of lines approach. This generalizes a stabilized implicit method of [17], and has the advantage that it can be adapted to a very general class of problems. An appropriate explicit or implicit time-stepping scheme can be used, depending on the particular problem considered. The new method retains the advantages of the original closest point method.

1.1. Outline. We begin Section 2 with a review of the closest point method followed by an overview of the new formulation in Section 2.1 and an example in

*Mathematical Institute, University of Oxford, OX1 3LB, UK (vonglehn,maerz,macdonald)@maths.ox.ac.uk). This publication is based on work supported by Award No. KUK-C1-013-04, made by King Abdullah University of Science and Technology (KAUST)
Section 2.2. Section 3 then defines a system of embedding equations, and shows that this is consistent with the original surface PDE. This system is reduced to a single equation in Section 4. The numerical solution of this equation is studied, and a discretization to obtain a system of ODEs is presented in Section 5. Numerical studies of an introduced parameter, and convergence studies and examples are given in Sections 6 and 7. Finally we present some conclusions and a discussion of future work.

2. The Closest Point Method. Suppose we want to solve a evolutionary PDE defined on a curved surface. A simple embedding technique known as the closest point method was introduced in [27]. This method uses the fact that the surface is embedded in $\mathbb{R}^n$, and represents the surface by a retraction based on Euclidean distance. For every point $x$ in this surrounding space $\mathbb{R}^n$ the retraction returns a surface point which is closest to $x$. We call this retraction a closest point function denoted by $cp$. If a function is defined on the surface, the data can be extended off the surface into the surrounding space by assigning to each point $x$ the value of the surface function at $cp(x)$. The key observation is that this function is now constant in the direction normal to the surface. Surface gradients and surface divergences of the original function will agree with the standard Cartesian operators of the extended function at the surface [18]. We call these ideas the “gradient principle” and “divergence principle” [27].

These principles are used to derive a simpler analogous PDE problem in the embedding space (for example, replacing surface intrinsic diffusion with the bulk or Cartesian diffusion). However, the closest point principles hold only on the surface. If this analogous PDE is evolved throughout the embedding space, the restriction to the surface may no longer be a solution of the original equation. The approach of the explicit closest point formulation of [27] is to advance the embedding PDE only by a single time step, before a re-extension of the data is performed. At the start of the next time step, the surface PDE will again agree on the surface with the analogous bulk PDE. The resulting scheme can then be expressed as a two-step explicit method in the form of [27, 16], which alternates between time steps of the embedding space PDE, followed by a re-extension of the surface data. We note this approach is not a method of lines.

An implicit version of the closest point method was introduced in [17], allowing application to stiff problems, such as those involving biharmonic or higher-order operators. To ensure stability, this formulation includes a stabilizing term, which can be related to the approach of the current paper. More general forms of the closest point function $cp$ were introduced in [18], using notions other than Euclidean distance to determine the mapping between the surrounding space and the surface.

2.1. A new approach to the embedding equation. Rather than alternating between time-steps and re-extensions as in the original closest point method [27], we investigate an alternative approach, in which a single equation can be evolved throughout the entire embedding space, for all time, without separate extension steps. This is achieved by creating a modified embedding equation with a constraint.

The solution of a given surface evolution PDE is a function $u$, which is defined only for points on the surface. We consider instead the function $v = u \circ cp$, which is defined for all points in a band surrounding the surface. Based on the gradient and divergence principles [27, 18], we formulate a new equation for $v$. The constraint or side condition that $v$ is a closest point extension is enforced by adding a penalty term to the equation. We show that evolving this new equation throughout the space to a
given time $t$, and then restricting to the surface, results in a solution of the original surface PDE at time $t$.

If the resulting Cartesian differential operators and extension operators are discretized in space as in [17], we obtain an ordinary differential equation in the computational band. Thus we have a new method-of-lines formulation of the closest point method, which can be implemented using either explicit or implicit time-stepping.

2.2. Example - diffusion equation. We first illustrate the method with an example, before giving a more detailed derivation. Let $S$ be a smooth closed surface embedded in $\mathbb{R}^n$, and $u$ a scalar function on $S$. Consider the surface diffusion equation

$$u_t = \Delta_S u,$$

subject to an initial condition $u_0$.

If $B(S)$ is a tubular neighbourhood of the surface in $\mathbb{R}^n$ (referred to as the band), then we can define a closest point function $\text{cp} : B(S) \rightarrow S$, as in [18], which maps points in the band to points on the surface. Typically, this will be the point closest in Euclidean distance on the surface, but this may be made more general in certain cases [18].

The extension operator $E$ is then defined as $Eu(x) = u \circ \text{cp}(x)$.

The surface differential operator (Laplace–Beltrami operator) may be replaced by a standard Laplacian using the principles in [27, 18]

$$u_t = \Delta [Eu] \text{ on } S.$$

This equation is valid only for points $x \in S$, since the left hand side $u_t$ is defined only on the surface. In order to obtain an equation defined throughout the entire band $B(S)$, we perform an extension on both sides of the equation

$$Eu_t = E\Delta [Eu] \text{ on } B(S).$$

We now define a function $v = Eu$ in the embedding space. Since the operator $E$ is independent of $t$, the previous equation can be rewritten as

$$v_t = E\Delta v \text{ on } B(S),$$

subject to the condition $v = Eu$. But if $v$ is the extension of $u$ then $v$ must also be its own extension (see also Lemma 3.5) and we obtain a system of two equations in $v$:

$$v_t = E\Delta v,$$  \hspace{1cm} (2.1a)

$$v = Ev.$$  \hspace{1cm} (2.1b)

We will show that the solutions $v$ of this system, when restricted to the surface $S$, agree with the solutions $u$ of the original equation; i.e. $u = v|_S$.

This system could then be approximated using the two-step method of [27], where a single time step of the first equation is carried out, and then the side condition imposed by extending the data $v(\cdot, t_k)$ off the surface.

We propose an alternative method, in which a single equation is solved. The side condition is added to the equation, with a constant multiplication factor $\gamma$,

$$v_t = E\Delta v - \gamma(v - Ev).$$  \hspace{1cm} (2.2)

This equation forms the basis for the method of lines. We show that the solutions of this single equation agree with the solutions of the system (2.1), for any non-zero choice of the parameter $\gamma$. However, in practice the choice of $\gamma$ affects the resulting numerical methods, as investigated in Section 6.
3. Defining an embedding equation. We will construct an equation defined in the embedding band, and show that there is a one-to-one correspondence between solutions of this and the original equation on the surface. In contrast to previous formulations, the embedding equation is satisfied for all time throughout the computational band, not only on the surface.

We first define an extension operator, then review the definition of the closest point gradient, divergence and Laplacian principles. The embedding equation is then defined through the application of these principles to the surface differential operators.

3.1. Closest Point Principles. In the following, let $S$ be a smooth surface of dimension $k$, embedded in $\mathbb{R}^n, n \geq k$, which possesses a tubular neighbourhood or embedding band $B(S) \subset \mathbb{R}^n$ surrounding the surface $S$. A general class of closest point functions mapping points in the neighbourhood to the surface was introduced in [18]. The closest point function based on Euclidean distance is a special case. Given one of these closest point functions, we define the extension operator $E$ which acts on surface functions, and returns a function on the embedding band $B(S)$.

**Definition 3.1 (Closest Point Extension Operator).** If $u : S \times \mathbb{R} \to \mathbb{R}$ is a scalar-valued function on the surface, then the closest point extension $v = Eu$ is a function $v : B(S) \times \mathbb{R} \to \mathbb{R}$ defined as

$$v(x,t) = Eu(x,t) := u(cp(x),t), \quad x \in B(S).$$

This definition can then be generalized to act on functions defined on all of $B(S)$ by operating on the restriction of the function to the surface $S$

$$Ev := E(v|_S) = v(cp(x),t), \quad x \in B(S).$$

Operation on a vector-valued function is defined componentwise.

The closest point principles of [27, 18] can then be formulated using this extension operator.

**Principle 3.2 (Gradient Principle).** If $E$ is a closest point extension operator according to Definition 3.1, then

$$\nabla[Eu](y) = \nabla_S u(y), \quad y \in S,$$

holds for the surface gradient $\nabla_S u$ of a smooth scalar surface function $u : S \to \mathbb{R}$.

**Principle 3.3 (Divergence Principle).** If $E$ is a closest point extension operator according to Definition 3.1, then

$$\text{div}[Eg](y) = \text{div}_S g(y), \quad y \in S,$$

holds for the surface divergence $\text{div}_S g$ of a smooth surface vector-field $g : S \to \mathbb{R}^n$.

**Principle 3.4 (Laplacian Principle).** In the case that $E$ is the particular closest point extension operator corresponding to Euclidean distance to the surface $[18]$, then

$$\Delta[Eu](y) = \Delta_S u(y), \quad y \in S,$$

holds for the surface Laplacian $\Delta_S u$ of $u$.

The gradient and divergence principles above can be combined to apply to a wider class of functions [27, 18]. In general, we assume $A_S$ is any surface-spatial differential operator such that the above principles can be applied to give an operator $A$ with

$$A(t,x,Eu) |_{x=y} = A_S (t,y,u), \quad y \in S. \quad (3.1)$$

That is, the operator $A$ (acting on functions on $B(S)$) is an analog of the operator $A_S$ (acting on functions on $S$) where $A$ has a standard differential operator wherever $A_S$ has a surface differential operator.
3.2. Equivalence of surface and embedding equations. We first give a simple lemma, which will be used frequently in the proofs below.

**Lemma 3.5.** The extension operator is idempotent.

**Proof.** Let \( v \) be a closest point extension of some function \( w : B(S) \times \mathbb{R} \to \mathbb{R} \), so that \( v = Ew \). Since the closest point operator \( cp \) is a retraction,
\[
v(y, t) = w(cp(y), t) = w(y, t), \quad y \in S,
\]
so \( v \) and \( w \) agree on the surface. Then
\[
E^2w = Ev = E(v |_S) = E(w |_S) = Ew. \quad \square
\]

From this, it follows that if a function \( v \) can be written as the extension of another function \( (v = Ew) \) then \( v \) must be its own extension \( (v = Ev) \).

We now show that two problems, one defined only on the surface \( S \), and one defined in the band \( B(S) \), have the same solutions when restricted to the surface.

**Problem 3.6 (Surface Evolution PDE).** Given a smooth closed surface \( S \) in \( \mathbb{R}^n \), let \( u : S \times [0, T) \to \mathbb{R} \), be a smooth solution of the PDE
\[
\begin{align*}
  u_t &= A_S(t, y, u) \quad u(y, 0) = u_0(y), \quad y \in S, \ t \in [0, T)
\end{align*}
\]
where \( A_S(t, y, u) \) is a linear or nonlinear surface differential operator of the class above.

**Remark.** Note that no additional boundary conditions have been specified for Problem 3.7. By (3.2b), the solution everywhere off the surface (including at the boundary of \( B(S) \)) is determined by values on the surface. Extra boundary condition are not necessary (although the extension (3.2b) is consistent in some cases with a Neumann-type boundary condition) and imposing artificial boundary conditions can make this problem ill-posed.

**Theorem 3.8.** Suppose \( S \) is a smooth surface embedded in \( \mathbb{R}^n \) and \( B(S) \subset \mathbb{R}^n \) is a neighbourhood of the surface. Then, for each smooth solution \( u : S \times [0, T) \to \mathbb{R} \) of the surface PDE (Problem 3.6), there exists a unique corresponding solution \( v : B(S) \times [0, T) \to \mathbb{R} \) of the embedding equation (Problem 3.7), which agrees with \( u \) when restricted to the surface \( S \). Conversely, for every solution \( v \) of Problem 3.7, the restriction of \( v \) to \( S \) is a solution of Problem 3.6.

**Proof.** To show existence, let \( v(x, t) = Eu(x, t) \), where \( u \) satisfies Problem 3.6. Then
\[
v_t = \partial_t(Eu) = E(u_t) = E(A_S(t, y, u)),
\]
from the surface PDE, and the fact that \( E \) is time-independent. Now, using (3.1) and the definition of \( E \), we have that
\[
E(A_S(t, y, u)) = E(A(t, x, Eu) |_S) = EA(t, x, Eu) = EA(t, x, v), \quad x \in B(S),
\]
so the first equation (3.2a) is satisfied. The second (3.2b) follows from
\[ Ev = E^2u = Eu = v. \]

Uniqueness follows from the fact that \( v \) agrees with \( u \) on \( S \), and that the off-surface values are uniquely defined by \( v(x, t) = Ev(x, t) \). The smoothness of \( v \) is determined by the smoothness of the surface and smoothness of \( u \) [18].

For the converse, suppose that \( v \) is a solution of Problem 3.7, and let \( u = v|_{S} \). Then
\[ \partial_t u = \partial_t v|_{S} = EA(t, x, v)|_{S}. \]

Since the extension operator leaves values on the surface unchanged, and using the closest point principles on \( A \),
\[ \partial_t u = EA(t, x, v)|_{S} = A(t, x, v)|_{S} = A_S(t, y, v|_{S}) = A_S(t, y, u), \]
so \( u \) is a solution of Problem 3.6 as required.

4. From constrained embedding problem to a single equation. We will show that the system of embedding equations (3.2) defined in the band \( B(S) \) has the same set of solutions as a single equation on \( B(S) \).

**Problem 4.1.** Given \( \gamma \in \mathbb{R} \), let \( v : B(S) \times [0, T) \rightarrow \mathbb{R} \) satisfy
\[ v_t = EA(t, x, v) - \gamma (v - Ev), \quad x \in B(S), \ t \in (0, T) \quad (4.1) \]
with initial condition \( v(x, 0) = v_0(x) \).

**Theorem 4.2.** Suppose that \( v : B(S) \times [0, T) \rightarrow \mathbb{R} \) is a solution of the system of equations (Problem 3.7). Then \( v \) also satisfies the single equation (Problem 4.1) for all \( \gamma \in \mathbb{R} \). Conversely, if \( v \) is a solution of Problem 4.1 with initial condition \( v_0 = Ev_0 \), then \( v \) will satisfy the system of embedding equations (Problem 3.7).

**Proof.** The first part follows directly, since \( v = Ev \) (3.2b) implies that the extra term multiplied by \( \gamma \) is zero, and the single equation (Problem 4.1) becomes equivalent to (3.2a).

For the converse, we operate on both sides of (4.1) with an extension operator \( E \), and use the fact that this operator is idempotent:
\[ Ev_t = EA(t, x, v) - \gamma (Ev - Ev) = EA(t, x, v). \]

Subtracting this from (4.1) gives
\[ (v - Ev)_t = -\gamma (v - Ev). \]

We now define a function \( z = v - Ev \), to obtain an ODE for \( z \):
\[ z_t = -\gamma z, \]
with initial condition \( z_0 = v_0 - Ev_0 = 0. \) This has unique solution \( z \equiv 0 \).

It follows that \( v = Ev \), and so the second equation of the system (3.2b) holds. Again, the extra term in (4.1) is zero, and so the single equation is equivalent to the system of equations.

**4.1. Remark on boundary conditions.** As above for the system of equations, the single equation (4.1) does not require any additional boundary conditions at the boundaries of \( B(S) \). The imposition of other boundary conditions could cause this problem to be ill-posed, for example, if they are contradictory to \( v = Ev \).
4.2. The Poisson Problem. A similar approach can be used for time-independent problems. Here, we show the Poisson equation as an example, but this can be generalized to equations of the form \( A_S(x,t,u) = f \) for the same class of operators as above.

**Theorem 4.3.** Consider the system of embedding equations obtained as above from the Poisson equation \( \Delta_S u = f \) on a surface \( S \),

\[
E \Delta v = Ef \quad \text{(4.2a)} \\
v = Ev, \quad x \in B(S). \quad \text{(4.2b)}
\]

The solutions of this system are the same as the solutions of the single equation

\[
E \Delta v - \gamma (v - Ev) = Ef, \quad x \in B(S), \quad \text{(4.3)}
\]

for any \( \gamma \in \mathbb{R} \setminus \{0\} \).

**Proof.** If \( v \) is a solution of (4.2a) and (4.2b), then the additional term in (4.3) is zero, so the single equation is satisfied. Conversely, if \( v \) satisfies (4.3), then we may extend the equation to obtain

\[
E^2 \Delta v - \gamma (Ev - E^2 v) = E^2 f,
\]

and using the idempotence of \( E \), this is

\[
E \Delta v - \gamma (Ev - Ev) = Ef.
\]

It follows that

\[
E \Delta v = Ef,
\]

and substituting back in (4.3), for any non-zero \( \gamma \), we have that \( v = Ev \). \( \square \)

This approach could easily be implemented for a more general differential operator as defined in (3.1). Poisson problems are investigated in another work [5].

5. A method-of-lines discretization. The conversion of the system of two equations — the extended PDE (3.2a) and the constraint (3.2b) — into a single equation (4.1) can now be used to define a method-of-lines discretization. The band \( B(S) \) is discretized using a standard uniform Cartesian grid in \( \mathbb{R}^n \), with \( N \) points in the embedding band. The vector \( v \in \mathbb{R}^N \) is defined as the set of values of the function \( v \) at these points. Following [17], we discretize the spatial differentiation operators on this grid using standard finite difference schemes to obtain matrices. For example, in 2D the Cartesian Laplacian is discretized by a matrix \( L \), the standard 5-point discrete Laplacian.

Multiplication by the matrix \( E \) [17] implements the discrete extension of a surface function, approximating the extension \( E \) using interpolation on the grid points surrounding the closest point. We note that this matrix operator is no longer idempotent, which complicates the theory in the semi-discrete case, and indeed in this work our analysis is mostly applied to the continuous operator.

The necessary size of the band \( B(S) \) to contain the differentiation and interpolation stencils is discussed in [27, 17] and is a small multiple of the mesh parameter \( \Delta x \).

With the discrete operators inserted into the equation, we obtain a system of ordinary differential equations for the vector \( v \)

\[
\partial_t v = ELv - \gamma (I - E)v, \quad \text{(5.1a)}
\]
or more generally

\[ \partial_t v = E A v - \gamma (I - E) v, \tag{5.1b} \]

where \( A \) is the matrix discretization, in the linear case, of the operator \( A \) in (3.1).

This system of ODEs can then be solved using either implicit or explicit time-stepping (or a combination). The consistency, convergence and stability of the method will depend on the interpolation, spatial discretization and time-stepping schemes. We discuss some of these issues, in particular how these relate to the choice of the parameter \( \gamma \) in Section 6.

5.1. Comparison to other semi-discrete formulations. In the formulation of [27], time steps of the discretized PDE are alternated with an extension step:

1. complete one time step of \( \partial_t v = L v \);
2. perform a re-extension \( v = E v \).

This approach is not a method of lines; it forces the solution to be constant in the direction normal to the surface after each time step. In our method-of-lines approach, this requirement is imposed with the penalty term in the PDE itself, so that no explicit re-extension step is required.

In [17] an initial suggestion for a method-of-lines approach in the particular case of the diffusion equation was the equation \( \partial_t v = L E v \). However, as this was seen to be unstable, a stabilized version was proposed, solving \( \partial_t v = M v \), where the matrix \( M \) was given by

\[ M = L E - \frac{2d}{(\Delta x)^2} (I - E). \]

At least for the diffusion equation, this is very similar to our (5.1) which also solves \( \partial_t v = M v \) but with

\[ M = E L - \gamma (I - E). \]

Note that the order of the matrices \( E \) and \( L \) is reversed, and the factor \( \frac{2d}{(\Delta x)^2} \) is generalized with the introduction of a new parameter \( \gamma \) (although in practice we recommend this same value for the Laplace–Beltrami operator). In [17], the discrete operator \( M \) was defined based on a special treatment of the diagonal of the discretized operator. The new formulation (5.1) is based on a different concept: we penalize the equation, not the operator and this makes the approach more general.

5.2. Nonlinear and higher-order operators. Previous formulations of the stabilized operator, such as those in [17] and [15], were stated for the Laplace–Beltrami operator, and did not include a general methodology for variable coefficient or nonlinear equations. The new method can easily be formulated to include such operators. In Section 7, we show numerical results on nonlinear curvature-dependent diffusion and reaction-diffusion equations. As an example of higher-order operators (which require further extensions \( E \)), we consider here the biharmonic operator \( \Delta_2 \).

The surface biharmonic equation \( u_t = -\Delta^2_2 u \) can be converted using the closest point principles to the form \( u_t = -\Delta E \Delta E u \) on the surface. Now operating with an extension on both sides of the equation, and substituting \( v = E u \), we have

\[ v_t = -E \Delta E \Delta v, \quad \text{subject to } v = Ev. \]
Forming a single equation and discretizing as before in (5.1) gives the semi-discrete form

\[ v_t = -\text{ELEL}v - \gamma(I - E)v. \]

We see that in general, the penalty term \( \gamma(v - Ev) \) remains, and an additional extension operator \( E \) is included. Note that this differs from the procedure in [17], which uses the squared matrix operator \( MM \) in the biharmonic case. Both approaches seem to work in practice. The advantages of each (or perhaps even of combinations) remains to be studied. Fully non-linear problems also warrant further study.

5.3. Summary of the method-of-lines approach. The resulting algorithm can be summarized as:

1. Extend the surface equation into the band by applying the extension operator \( E \), and then use Principles 3.2 and 3.3 to replace surface differential operators with their Cartesian analogs.
2. Add the penalty term \(-\gamma(v - Ev)\) to the PDE.
3. Use standard discretizations in space for the differential and extension operators, and an appropriate time-stepping scheme to solve the resulting system.

6. Effect of the parameter \( \gamma \). It should be emphasized that the parameter \( \gamma \) is not a Lagrange multiplier; it is not necessary to solve for a value of \( \gamma \) as part of the solution procedure. Rather, \( \gamma \) is a numerical parameter that controls how strongly the constraint is imposed. The parameter \( \gamma \) may affect the consistency and stability of the method. Numerical tests below suggest that a wide range of values result in convergent schemes.

6.1. Penalty term and zero-stability. The term \(-\gamma(v - Ev)\) imposes the side condition \( v = Ev \) to the PDE and can be viewed as a penalty term in the equation. If \( v \) is not constant in the direction normal to the surface, then the term \( v - Ev \) can be large, and dominate the term containing the differential operator. We analyze this term by considering the trivial time-dependent PDE \( u_t = 0 \) which leads to the equation

\[ vt = -\gamma(v - Ev). \quad (6.1) \]

Any deviation in the normal direction will be penalized by a large right hand side. We will show that positive values of \( \gamma \) will return the system to the stable equilibrium, while negative values of \( \gamma \) may lead to instability.

Operating on both sides of this equation with an extension \( E \), and using the idempotence of the operator gives \( Ev_t = 0 \). As in the proof of Theorem 4.2, we subtract this from (6.1), and define the function \( z = v - Ev \). We can then study this as an ODE system

\[ z_t = -\gamma z, \quad \text{with} \quad z_0 = v_0 - Ev_0, \quad (6.2) \]

which has unique solution \( z(x, t) = z_0(x)e^{-\gamma t} \). If the initial condition is perturbed slightly, so that it is no longer zero (i.e., \( v_0 \) is not exactly a closest point extension), then the function \( z \) will still decay to zero, provided that \( \gamma \) is positive.

As (6.2) is essentially the Dahlquist test equation [14], the region of absolute stability is determined by the time-stepping method used. For example, for the forward Euler method, the region of absolute stability is \(|1 - \gamma \Delta t| \leq 1\). With positive \( \gamma \), this
I. VON GLEHN, T. MÄRZ, AND C. B. MACDONALD

implies a time step restriction $\Delta t \leq \frac{2}{\gamma}$. Likewise, for the explicit four-stage Runge–Kutta scheme we have a stability restriction of $\Delta t \leq \frac{2\gamma}{\gamma_3}$. For A-stable methods such as the implicit Euler method, all positive values of $\gamma$ give stable solutions for the trivial PDE.

Although this analysis was based on the continuous operator $E$ rather than the discrete $E$, our computations below suggest good agreement. Thus returning to the semi-discrete problem (5.1), we expect a stability restriction (when using an explicit scheme) based on each of the two terms in the equation and we will take $\Delta t$ based on the minimum of the two restrictions. One reasonable strategy in choosing $\gamma$ is to avoid increasing the stiffness of the system (compared to that of the Cartesian discretization of the equivalent non-surface PDE problem).

Example: surface diffusion equation. If we discretize (5.1a) using forward Euler in time and the standard second-order scheme for the Laplacian, we might expect a time step restriction of

$$\Delta t \leq \min \left(\frac{\Delta x^2}{2d}, \frac{2\gamma}{\gamma_3}\right).$$

(6.3)

Thus, at least for the surface diffusion equation, we can recommend a value of $\gamma$ of

$$\gamma = \frac{2d}{\Delta x^2},$$

(6.4)

and with this choice we can expect the usual choice $\Delta t \leq \frac{\Delta x^2}{2d}$ to result in a stable scheme (with a factor of two to spare).

Figure 6.1 illustrates that the above theory correctly predicts the practical stability properties. On the unit circle, we consider the equation $u_t = \Delta_S u - u$ with semi-discrete form $v_t = \mathbf{E}v - v - \gamma(I - \mathbf{E})v$. We discretize with forward Euler and estimate the largest possible stable time-step; the results are very close to (6.3). Using our suggested value of $\gamma$ from (6.4) allows the time-step predicted by the standard non-surface Cartesian finite difference scheme. In practice if a larger value of $\gamma$ is desirable, then the time-step $\Delta t$ could simply be reduced for stability.
6.2. Consistency. Requiring the method to be consistent also places certain restrictions on the penalty parameter $\gamma$, as well as the interpolation order $p$ of the extension operator. Again we consider the case of the Laplace–Beltrami operator as an example. With a second-order spatial discretization and first-order explicit time-stepping, the scheme can be written

$$\frac{v^{n+1} - v^n}{\Delta t} = E_pL v^n - \gamma (v^n - E_p v^n). \quad (6.5)$$

As before, $L$ and $E_p$ are discretizations of the Laplacian and extension operators, with polynomial interpolation of order $p$ in the extension [17]. The truncation error will include standard terms of order $O(\Delta t) + O(\Delta x^2)$ from the discretization, as well as a term proportional to $\Delta x^{p+1}$ from the extension operator. If $\gamma$ is chosen to scale with $\Delta x^{-\alpha}$, then the final term in the truncation error is a contribution of $O(\Delta x^{p+1-\alpha})$ from the penalty term (note that the exact solution is an extension so the truncation error in the penalty term is simply that of the discrete operator $E_p$). Combining these results gives an overall order of accuracy of the method of

$$O(\Delta t) + O(\Delta x^2) + O(\Delta x^{p+1}) + O(\Delta x^{p+1-\alpha}).$$

For first-order consistency, it is necessary that $p \geq \alpha$, so if $\gamma = O(\Delta x^{-2})$ (which may be required for stability), then at least degree 2 interpolation in $p$ is needed. To maintain second order convergence in $\Delta x$, at least $p = 3$ is required.

Note that if in some situation, the dependence of $\gamma$ on $\Delta x$ could be freely chosen, then setting $\gamma$ to be a constant and using $p = 1$ should also give second order convergence. This would be computationally more efficient, since then only bilinear/trilinear interpolation matrices could be used. Further details of the consistency of the closest point method are given in [19].

6.3. Stability. The stability of the system will also depend on the choice of $\gamma$, for the equation and discretization considered. In the case that $\gamma$ is zero, the side condition is not enforced at each time step. In practice, in this case small errors in the normal direction tend to grow over time, eventually leading to instability. Figure 6.2 shows how the maximum error in the solution depends on $\gamma$ for the particular case of the heat equation on the unit circle at time $t = 0.5$, using forward and backward Euler time-stepping with $\Delta t = \frac{1}{4} \Delta x^2$ and $\Delta t = \frac{1}{4} \Delta x$ respectively. The vertical line in the first figure is at $\gamma \Delta x^2 = 8$, where the solution becomes unstable at large $\gamma$ due to the loss of zero-stability described in Section 6.1. For implicit time-stepping, the large $\gamma$ instability does not occur. As $\gamma$ becomes too small ($\gamma \Delta x^2 \approx 0.1$, the solution may also become unstable. Intuitively, this is because the penalty is not strong enough to impose the constraint. For the heat equation, a suggested value is $\gamma \approx \frac{1}{4 \Delta x^2}$ (and this is the same value chosen in [17]). In [5], a case is considered where the two extension operators in the scheme (6.5) have different degrees $p$ of interpolation. For the Poisson equation on closed curves in $\mathbb{R}^2$, the scheme with two extension operators with polynomial interpolations of order 1 and 3, and $\gamma \approx \frac{1}{4 \Delta x^2}$ is shown to be second-order and stable.

6.4. Relationship to the explicit method of Ruuth & Merriman. Note that if we choose $\gamma = \frac{1}{\Delta t}$, the resulting iteration for the surface heat equation will be the same as the two-step method of [27]. The scheme (6.5) becomes

$$v^{n+1} = E_p(\Delta t Lv^n + v^n),$$
which corresponds to applying one step of first-order explicit time-stepping, followed by performing an extension.

7. Numerical examples. We demonstrate the effectiveness of the new method with various examples in 2D and 3D.

7.1. Diffusion equation on the unit circle and unit sphere. The diffusion equation example of Section 2.2 is studied on the unit circle embedded in 2D, and the unit sphere embedded in 3D. Starting from the surface equation \( u_t = \Delta_S u \), the resulting embedding equation with the penalty term is

\[ v_t = E\Delta v - \gamma(v - Ev). \]

We take the standard parameterization \( \sigma : [0, 2\pi) \to \mathcal{S}, \sigma(\theta) = (\cos(\theta), \sin(\theta))^T \), for the unit circle, and write \( \bar{u}(t, \theta) = u(t, \sigma(\theta)). \) The initial condition on the circle is taken to be \( \bar{u}(0, \theta) = \cos \theta + \cos 3\theta \), giving exact solution \( \bar{u}(t, \theta) = e^{-t}\cos \theta + e^{-9t}\cos 3\theta \). Similarly, the parameterization of the sphere is given by \( \sigma : (-\pi, \pi] \times [-\pi/2, \pi/2] \to \mathcal{S}, \sigma(\theta, \phi) = (\cos \theta \cos \phi, \sin \theta \cos \phi, \sin \phi) \). The initial condition is \( \bar{u}(0, \theta, \phi) = \cos(\phi + 1/2) \), so that \( \bar{u}(t, \theta, \phi) = e^{-2t}\cos(\phi + 1/2) \).

Standard second-order central differences are used to discretize the Laplacian, and the order \( p \) of the polynomial interpolation is varied. Figures 7.1a and 7.1b show convergence studies with explicit and implicit time-stepping, using a forward Euler and BDF2 scheme respectively. The parameter \( \gamma \) is fixed to be \( \frac{2d}{\Delta x^2} \), where \( d \) is the dimension of the embedding space. The solution is run in time until \( t = 0.5 \), using \( \Delta t = \frac{1}{4}\Delta x^2 \) in the explicit case, or \( \Delta t = \frac{1}{4}\Delta x \) for the implicit BDF2 scheme. The figures demonstrate the expected second-order convergence for \( p \geq 3 \). We compute the error by restricting the solution of the embedding equation to the surface, and computing the max-norm error over the discrete approximation to \( \mathcal{S} \).

7.2. Biharmonic equation. As an example of a higher-order operator requiring more extensions, consider the biharmonic equation \( u_t = -\Delta^2_S u \), again on the unit circle in 2D. As in Section 5.2, the resulting embedding PDE is

\[ v_t = -E\Delta E\Delta v - \gamma(v - Ev) \]

(7.1)
AN EMBEDDED METHOD-OF-LINES APPROACH TO SURFACE PDES

Fig. 7.1: Numerical convergence studies for the diffusion equation on the unit circle (a) and unit sphere (b), using forward Euler and BDF2 time-stepping respectively, with $\gamma = \frac{4}{\Delta x^2}$.

Fig. 7.2: Convergence study for the biharmonic equation on the unit circle, using BDF2 implicit time stepping, with $\gamma = \frac{4}{\Delta x^2}$.

The initial condition $u(\theta, 0) = \cos \theta + \cos 3\theta$ results in the exact solution $u(\theta, t) = e^{-t} \cos \theta + e^{-81t} \cos 3\theta$ at time $t$.

Explicit time step restrictions become prohibitive for the higher-order operators, so only implicit schemes are considered. We do not yet know how to choose $\gamma$ in this biharmonic case; further work is required. However, with $\gamma = \frac{4}{\Delta x^2}$ and $p \geq 4$, we do observe second order convergence in Figure 7.2, which shows the error in a BDF2 implicit time-stepping scheme, with $\Delta t = \frac{1}{4} \Delta x$.

7.3. Reaction-diffusion equations on a triangulated surface. The Gray–Scott reaction-diffusion equations are used as a model of pattern formation [12, 23]. Formulated on a surface, the equations are given by

\begin{align}
   u_t &= \nu_u \Delta_S u - u v^2 + F(1 - u) \tag{7.2a} \\
   v_t &= \nu_v \Delta_S v + u v^2 - (F + k)v \tag{7.2b}
\end{align}
This example involves nonlinear terms in $u$ and $v$, which are simple to treat with the method of lines. After extension and discretization in space, the system of equations in the computational band becomes

$$\begin{align*}
    u_t &= \nu_u \text{EL}u - uv^2 + F(1 - u) - \gamma(u - Eu) \\
    v_t &= \nu_v \text{EL}v + uv^2 - (F + k)v - \gamma(v - Ev)
\end{align*}$$

The surface is a triangulated genus 3 shape [25], from which a closest point function is calculated [16]. This system of equations is solved with an implicit-explicit IMEX scheme treating the Laplace–Beltrami operators implicitly, and the nonlinear terms explicitly [26]. Diffusion constants used are $\nu_u = (\Delta x)^2/9$, $\nu_v = \nu_u/2$, with parameters $k = 0.063$, $F = 0.054$ [20]. The results for $u$ at steady state are shown in Figure 7.3.

7.4. Curvature-dependent diffusion on surfaces. The geometry of the surface itself can be included in the PDE. This is demonstrated with an example of a diffusion equation, where the diffusivity depends on the curvature of the surface. Consider the equation

$$u_t(y) = \text{div}_S(a(y)\nabla_S u(y)), \quad (7.4)$$

where we choose the inhomogeneous diffusivity $a(y)$ related to the mean curvature $\kappa(y)$ of the surface by

$$a(y) = \frac{1}{1 + |\kappa(y)|}.$$ 

We can use the closest point function representing the surface to calculate the curvature directly. The mean curvature $\kappa$ on the surface is given by

$$\kappa(y) = ||\Delta \text{cp}(y)||_2. \quad (7.5)$$

This follows from the fact that mean curvature vector $HN$ can be written as the Laplace–Beltrami operator of the identity function on the surface [6]

$$H(y)N(y) = -\Delta_S Id_S(y).$$
Here, \( N(y) \) is the normal to the surface at the point \( y \). Applying the closest point principles, we have

\[
H(y)N(y) = -\Delta(EI ds)(y) = -[\Delta cp](y),
\]

since the closest point function is the extension of the identity on the surface. The magnitude \( \kappa \) of the mean curvature is found by taking the two-norm of this expression. Following our discretization of Section 5, we compute the mean curvature based on (7.5) on the grid of the embedding space \( B(S) \) by

\[
\kappa = E \sqrt{(L cp_1)^2 + (L cp_2)^2 + (L cp_3)^2},
\]

where \( cp_1, cp_2, \) and \( cp_3 \) are vectors of the components of the closest point associated with each grid point. From this we compute the vector \( a \) consisting of the values of \( a \) at each grid point.

The PDE (7.4) is simple to solve numerically using the method of lines approach detailed in Sections 4 and 5. The embedded surface with penalty term is

\[
v_t = E \div (a \nabla v) - \gamma (v - Ev).
\]

Now a standard scheme is used to discretize the variable coefficient diffusion term which yields the semi-discrete form

\[
v_t = E \left[ D_b^x (A_x f^a D_x^y v) + D_b^y (A_y f^a D_y^x v) + D_b^z (A_z f^a D_z^x v) \right] - \gamma (I - E) v,
\]

where \( D_b \) and \( D_f \) are the backward and forward finite difference matrices in the direction indicated by the superscripts. Similarly, the \( A_f \) matrices refer to forward two-point averages of the point-wise diffusivity values. That is, the half-point diffusivities are approximated by the averages:

\[
a_{i+1/2,j,k} \approx \frac{a_{i+1,j,k} + a_{i,j,k}}{2}, \quad a_{i,j+1/2,k} \approx \frac{a_{i,j+1,k} + a_{i,j,k}}{2}, \quad a_{i,j,k+1/2} \approx \frac{a_{i,j,k+1} + a_{i,j,k}}{2},
\]

This scheme can then be evolved with explicit Euler time-stepping.

Figures 7.4a and 7.4c show the curves used to demonstrate this approach: an ellipse and the curve parameterized by \( x = (1 + \frac{1}{3} \cos(6s)) \cos s, y = (1 + \frac{1}{3} \cos(6s)) \sin s \). Initial conditions are \( u(s,0) = \cos(3s) \). The resulting solutions at time \( t = 0.5 \) are shown in Figures 7.4b and 7.4d. Solutions are also computed using Chebfun [29] (based on the parameterization) and plotted for comparison.

7.5. Reaction-diffusion with curvature-dependent parameters. Through dependence on curvature, the geometry of the surface could influence systems such as reaction-diffusion equations. In diffusion-driven instability, the difference in diffusion coefficients of two chemical species drives an instability leading to pattern formation [23]. If the diffusivities vary across the surface, patterns may form only in certain areas.

In the Gray–Scott model above, the ratio of diffusion coefficients \( \nu_v = \frac{\nu}{2} \) is used to form a patterned steady state. With equal coefficient values, no patterns are formed. We now consider a case where \( \nu_v \) varies with curvature of the surface.

The approaches of the two previous numerical examples are combined. The Gray–Scott scheme (7.3) is solved on a surface of non-constant curvature, with \( \nu_v \) related to \( \nu_u \) by

\[
\nu_v = \nu_u / \left( 3 - \frac{2}{c_1 - c_2}(\kappa - c_2) \right),
\]
where $c_1$ and $c_2$ are the maximum and minimum curvatures of the surface. At areas of low curvature, the ratio will be close to 3, while areas of high curvature will have equal coefficients.

We expect patterns to form preferentially in low-curvature areas, as demonstrated in Figure 7.5. Initial conditions are taken to be the steady state $(u_0, v_0) = (1, 0)$, with random Gaussian noise added. Figure 7.5a shows the ratio of the diffusivities calculated from the mean curvature of an ellipsoid. Steady states for $u$ demonstrating spot and stripe formation on this surface are shown in Figures 7.5b and 7.5c. Parameters used are $F = 0.026$, $k = 0.061$ for spots and $F = 0.054$, $k = 0.063$ for stripes, as expected on flat domains [20]. A similar system is solved on a parameterized red blood cell shape (derived in [9] and used with reaction-diffusion models in [11]), this time with nonequal coefficients at areas of high curvature. Figure 7.6 shows spot and stripe formation on the high-curvature regions of the surface.

8. Conclusions. We have introduced a new formulation of an embedding method for solving partial differential equations (PDEs) on surfaces, based on the closest point representation. Our formulation results from the addition of a penalty term to the surface PDE, which helps ensure that the solution in the embedded space stays constant in the normal direction. Like the original closest point method of Ruuth and
Fig. 7.5: Curvature-dependent reaction-diffusion on an ellipsoid. Ratio of diffusion coefficients (a) — inversely proportional to surface curvature, stripe (b) and spot (c) formation in regions of low curvature using the Gray–Scott model (7.3).

Fig. 7.6: Curvature-dependent reaction-diffusion on a red blood cell shape. Ratio of diffusion coefficients (a) — proportional to surface curvature, stripe (b) and spot (c) formation in regions of high curvature using the Gray–Scott model (7.3).

Merriman, the method is simple and very general with respect to surface geometry, dimension and co-dimension.

Compared to previous attempts to construct an implicit closest point method, our method has an advantage in that it works for variable coefficient and nonlinear PDEs. Because the method allows a method-of-lines discretization, it can be used
with either implicit or explicit time-stepping (and, although not our focus here, for elliptic problems). Our approach also seems simpler to analyze.

The solutions of the new embedding equation, when restricted to the surface, are shown to correspond with a one-to-one map to the solutions of the original PDE. The modified equation involves a parameter; we show that, while in the continuous problem any value will work, in numerical discretizations the value is important. In particular, the effect of this penalty parameter on stability is analyzed, and numerical studies of convergence are shown for the Laplace–Beltrami operator and surface biharmonic operators. Examples demonstrate the effectiveness of the method for nonlinear operators on various parameterized and triangulated surfaces, in particular relating to curvature-dependent diffusion.

Future work could investigate fully nonlinear problems and the role of the penalty parameter in higher-order problems, for example, a more thorough treatment of surface biharmonic problems.

Acknowledgements. We thank Dr Anotida Madzvamuse (Sussex) for suggesting the application of curvature-influenced reaction-diffusion.

REFERENCES

[1] S. Auer, C. B. Macdonald, M. Treib, J. Schneider, and R. Westermann, Real-time fluid effects on surfaces using the Closest Point Method, Computer Graphics Forum, 31 (2012), pp. 1909–1923.
[2] M. Bertalmío, L. Cheng, S. Osher, and G. Sapiro, Variational problems and PDEs on implicit surfaces, J. Comput. Phys, 174 (2001), pp. 759–780.
[3] H. Biddle, I. von Glehn, C. B. Macdonald, and T. März, A volume-based method for denoising on curved surfaces, (2013). To appear in Proc. ICIP13, 20th IEEE International Conference on Image Processing.
[4] M. Burger, Finite element approximation of elliptic partial differential equations on implicit surfaces, Computing and Visualization in Science, 12 (2009), pp. 87–100.
[5] Y.-J. Chen and C. B. Macdonald, The Closest Point Method and multigrid solvers for elliptic equations on surfaces. Submitted, 2013.
[6] U. Clarenz, U. Diewald, and M. Rumpf, Processing textured surfaces via anisotropic geometric diffusion, IEEE Transactions on Image Processing, 13 (2004), pp. 248–261.
[7] K. Deckelnick, G. Dziuk, C. M. Elliott, and C. J. Heine, An h-narrow band finite-element method for elliptic equations on implicit surfaces, IMA J. Numer. Anal., 30 (2010), p. 351.
[8] G. Dziuk and C. M. Elliott, Surface finite elements for parabolic equations, Journal of Computational Mathematics - International Edition, 25 (2007), p. 385.
[9] E. Evans and Y. C. Fung, Improved measurements of the erythrocyte geometry, Microvascular research, 4 (1972), pp. 335–347.
[10] M. S. Floater and K. Hormann, Surface parameterization: a tutorial and survey, Advances in Multiresolution for Geometric Modelling, (2005), pp. 157–186.
[11] E. J. Fuselier and G. B. Wright, A high-order kernel method for diffusion and reaction-diffusion equations on surfaces, Journal of Scientific Computing, (2013), pp. 1–31.
[12] P. Gray and S. K. Scott, Autocatalytic reactions in the isothermal, continuous stirred tank reactor: isolas and other forms of multistability, Chem. Eng. Sci., 38 (1983), pp. 29–43.
[13] J. B. Greer, An improvement of a recent Eulerian method for solving PDEs on general geometries, Journal of Scientific Computing, 29 (2006), pp. 321–352.
[14] E. Hairer, S. P. Norsett, and G. Wanner, Solving Ordinary Differential Equations. I: Nonstiff Problems, Springer, 1993.
[15] C. B. Macdonald, J. Brandman, and S. J. Ruuth, Solving eigenvalue problems on curved surfaces using the Closest Point Method, J. Comput. Phys., 230 (2011), pp. 7944–7956.
[16] C. B. Macdonald and S. J. Ruuth, Level set equations on surfaces via the Closest Point Method, J. Sci. Comput., 35 (2008), pp. 219–240.
[17] _____, The implicit Closest Point Method for the numerical solution of partial differential equations on surfaces, SIAM J. Sci. Comput., 31 (2009), pp. 4330–4350.
[18] T. März and C. B. Macdonald, Calculus on surfaces with general closest point functions, SIAM J. Numer. Anal., 50 (2012), pp. 3303–3328.
[19] Consistency and stability of closest point iterations. In preparation, 2013.
[20] R. Munafo, Reaction-Diffusion by the Gray-Scott Model: Pearson’s Parameterization. http://www.mrob.com/pub/comp/xmorphia.
[21] J. D. Murray, Mathematical biology II: spatial models and biomedical applications, Springer, 2003.
[22] T. G. Myers, J. P. F. Charpin, and S. J. Chapman, The flow and solidification of a thin fluid film on an arbitrary three-dimensional surface, Physics of Fluids, 14 (2002), pp. 2788–2803.
[23] J. E. Pearson, Complex patterns in a simple system, Science, 261 (1993), pp. 189–192.
[24] C. Piret, The orthogonal gradients method: A radial basis functions method for solving partial differential equations on arbitrary surfaces, Journal of Computational Physics, (2012).
[25] AIM@SHAPE Shape Repository, Genus3 shape.
[26] S. J. Ruuth, Implicit-explicit methods for reaction-diffusion problems in pattern formation, Journal of Mathematical Biology, 34 (1995), pp. 148–176.
[27] S. J. Ruuth and B. Merriman, A simple embedding method for solving partial differential equations on surfaces, Journal of Computational Physics, 227 (2008), pp. 1943 – 1961.
[28] L. Tian, C. B. Macdonald, and S. J. Ruuth, Segmentation on surfaces with the Closest Point Method, in Proc. ICIP09, 16th IEEE International Conference on Image Processing, 2009, pp. 3009–3012.
[29] L. N. Trefethen et al., Chebfun Version 4.2, The Chebfun Development Team, 2011. http://www.maths.ox.ac.uk/chebfun.
[30] G. Turk, Generating textures on arbitrary surfaces using reaction-diffusion, SIGGRAPH Comput. Graph., 25 (1991), pp. 289–298.