Journal of Mathematical Biology
https://doi.org/10.1007/s00285-020-01494-x

Mathematical Biology

Parameter estimation for a point-source diffusion-decay morphogen model

Mark B. Flegg[1] · Mario A. Muñoz[2] · Kate Smith-Miles[2] · Wai Shan Yuen[3] · Jennifer A. Flegg[2] · John G. Carroll[3]

Received: 23 August 2019 / Revised: 8 February 2020

Abstract
In this paper we present a novel method for finding unknown parameters for an unknown morphogen. We postulate the existence of an unknown morphogen in a given three-dimensional domain due to the spontaneous arrangement of a downstream species on the domain boundary for which data is known. Assuming a modified Helmholtz model for the morphogen and that it is produced from a single source in the domain, our method accurately estimates the source location and other model parameters. Notably, our method does not require the forward solution of the model to be computed which can often be a challenge for three-dimensional PDE model parameter fitting. Instead, an extension is made from the problem domain to an infinite domain and the analytic nature of the fundamental solution is exploited. We explore in this manuscript strategies for best conditioning the problem and rigorously explore the accuracy of the method on two test problems. Our tests focus on the effect of source location on accuracy but also the robustness of the algorithm to experimental noise.

Keywords Morphogen · Parameter fitting · Inverse problem · Modified Helmholtz equation

Mathematics Subject Classification 92C15 · 35K57 · 35R30

1 Introduction

The distribution of morphogens in space and time are the key determining factor for a wide array of morphological patterning and developmental programs in biology

✉ Mark B. Flegg
mark.flegg@monash.edu

[1] School of Mathematical Sciences, Monash University, Clayton, Australia

[2] School of Mathematics and Statistics, University of Melbourne, Parkville, Australia

[3] Monash Biomedicine Discovery Institute, Monash University, Clayton, Australia

Published online: 25 April 2020

(Tabata and Takei 2004). The ubiquity with which morphogen distributions are applied in biological processes cannot be overstated. Morphogen distributions are often associated with cellular responses to extracellular chemical gradients (Gurdon and Bourillot 2001). However, in a broad sense, morphogens are commonly also applied to models with a variety of different scales and contexts—from intracellular chemical systems (Howard 2012) to resources on ecological scales (Tilman 1984). The majority of studies focus on the effects of the morphogen rather than the morphogen itself. It is assumed in most models that the properties of the morphogen are known so that predictions can be made about downstream consequences. In cases where it is only the consequences that are known, determining the properties of the unknown morphogen can be challenging. It is this latter challenge that we address in this manuscript. That is, we assume a morphogen exists but its parameters are unknown. We wish to determine—or at least estimate—these parameters based on the assumption of a simple morphogen model and using only observed downstream data.

We focus our attention on one of the most common mechanisms for morphogen gradient formation in biology—linear diffusion and decay (Wartlick et al. 2009). We will make a pseudo-steady state assumption and thus are interested in determining the parameters and source of a modified Helmholtz model using downstream data. Specifically, we consider a source of morphogen inside some domain Ω. Nothing about this morphogen is known; its source location, production rate, degradation rate or diffusion constant. We assume that the morphogen is absorbed by the boundary $\partial\Omega$ and promotes the accumulation of a downstream variable A on the boundary that is measureable by experiment. Examples of systems in biology where this would directly apply include parameterisation of recruiting or substrate proteins at cell membranes (Remorino 2017) or long range signalling molecules which are absorbed/detected at the boundary between tissues or at cell membranes (Camley 2018). In the former case, A would be the recruited molecular species and in the latter case A would be a measure of cellular responses to the spatially dependent signal such as proliferation, migration, chemical secretion, morphological changes etc. Here, our motivation is the determination of properties of an upstream promoter of actin formation which forms the actin cap within the cortex of oocytes. These actin caps (A) drive polar body formation during meiosis events but it is unknown what initially causes the caps to form (Dehapiot et al. 2013). This work constitutes an initial step forward in addressing this major biological problem. In biological applications, the data for the measurable A is often limited by the technology required to observe it. In three-dimensional problems, it is common in biomedical experiments for data to be presented in slices in the form of stacked 2D images. This constraint poses a challenge for accurate quadrature of the data where it is required.

There is a large body of work dedicated to solving inverse problems. In the case of inverse problems involving linear PDE models, the forward problem is usually a uniquely solvable and well-posed problem of finding u when $\mathcal{L}_{\boldsymbol{\theta}} u = f$ where $\mathcal{L}_{\boldsymbol{\theta}}$ is some linear partial differential operator that may or may not depend on unknown parameters $\boldsymbol{\theta}$. The model is closed with some boundary conditions which may also depend on unknown parameters and forcing functions. The task of an inverse problem is to instead find any unknown parameters or forcing functions of the model (e.g. $\boldsymbol{\theta}$ or f) when only data $\bar{A}$ determined by the forward solution $u = u_{\boldsymbol{\theta},f}$ is known, for example with some known relationship $\bar{A} = A(u)$. Of these types of inverse problems, the majority deals with finding forcing functions such as f assuming that the parameters which define $\mathcal{L}_{\boldsymbol{\theta}}$ are known (for example—source identification problems (Yan et al. 2008), boundary parameter identification problems (Fasino and Inglese 1999), or inverse geometric problems where the unknown is the boundary manifold itself Marin and Karageorghis 2009). The strategy that is most commonly employed to solve these problems is to find the unknowns which minimize a measure of the error $\epsilon_{\boldsymbol{\theta},f} = |\bar{A} - A(u_{\boldsymbol{\theta},f})|$ (Stuart 2010). This is often a least squares optimisation (minimisation of $\sum \epsilon^2_{\boldsymbol{\theta},f}$, where the sum is taken over the observed data). This solution methodology (direct fitting) has two major drawbacks. The first is that for each iteration of the optimisation step (which evaluates the error $\epsilon_{\boldsymbol{\theta},f}$ of a trial set of unknowns), the forward solution of the model with the trial parameters $u = u_{\boldsymbol{\theta},f}$ is required. When numerical methods are also required to solve the forward PDE model, this can be hugely computationally intensive (especially in 3D and/or irregular domains). The second major drawback is that inverse problems like this are often ill-posed according to the criteria of Hadamard (1923). Research into direct fitting methods such as these focus on addressing these two issues either by improving the optimisation step or introducing regularisation techniques (Engl et al. 1996).

Some inverse problem methods, such as the method of fundamental solutions (MFS), employ known fundamental solutions to avoid the problem of having to iterate the forward solution (Karageorghis et al. 2010). The MFS introduces sources outside of the domain in order that boundary conditions are satisfied. For example, the method of images for the heat equation is an example of an MFS. Since the fundamental solutions are known, the PDE does not have to be solved. Instead, the sources that lie outside of the domain introduce new unknowns.

In this manuscript, we explore a novel method for inverting the modified Helmholtz model with unknown parameters. The model is, in part, loosely inspired by MFS. We use fundamental solutions so that the forward solution to the model is not required in the optimisation step. Whilst we do not introduce new unknown parameters into the optimisation step, our method comes at the cost of having to integrate data numerically over the domain boundary. This can introduce complications which are thoroughly explored in this manuscript. We present strategies for alleviating these complications. It is not within the scope of this paper to discuss or compare methodologies for quadrature or optimisation as these are, in themselves, the topic of whole fields of study. We state the novel method as a general strategy and in examples use simple subalgorithms for surface quadrature and optimisation suitable to the context. We use the method to solve for diffusion, decay and production parameters of a morphogen u as well as its source location based on data A (the flux of morphogen) on the boundary of the domain. Our focus is on identifying the parameters of the morphogen itself and we assume for simplicity that the source of morphogen is concentrated to a single—albeit unknown—location in space so that the problem reduces to finding three coordinates in space as well as parameters such as production rate and decay rate. To help motivate our methodology, after presenting the problem statement and introducing the test problems, we demonstrate the infeasibility of using direct fitting methods using a black box optimisation solver to solve this problem. The strength of our method lies in the fact that the forward PDE model never has to be solved. We are therefore not concerned with computational efficiency but rather we rigorously test the accuracy of our method against the test problems. Since the posedness of this inverse problem is solution-dependent, we investigate the effect of ‘true’ solutions on accuracy of the method. We also investigate for the more challenging test problem the effect of noise in the data (due to, for example, experimental, environmental or instrumentation error) on the estimated parameters.

2 Mathematical foundation

2.1 Problem statement

Consider a morphogen (concentration $\check{U}$) being produced at an unknown location $\mathbf{r}_0$ inside some compact domain with piecewise smooth boundary Ω at an unknown rate $\check{\lambda}_0 > 0$. We shall focus in three dimensional space but outline the theory as a general strategy. This morphogen decays at an unknown rate $\check{\kappa}_0 > 0$ and diffuses with an unknown diffusion constant $\check{D}_0$ until it interacts with the boundary of the domain $\partial\Omega$ at which point it is absorbed and turned into a new chemical (concentration $\check{A}$) which is bound to the boundary of the domain. Assuming a balance between production and linear decay of $\check{A}$ on the boundary, in steady state, $\check{A} = -\nu\nabla\check{U}\cdot\hat{\mathbf{n}}$ for some proportionality constant ν. Here, $\hat{\mathbf{n}}$ is the outward-facing normal on $\partial\check{\Omega}$ and ∇ is the spatial gradient.

We assume that this morphogen reaches steady state quickly and that a static distribution is established. The steady state equation for $\check{U}$ is the modified Helmholtz equation

$$-\check{D}_0 \triangle\check{U}(\mathbf{r}) + \check{\kappa}_0^2\check{U}(\mathbf{r}) = \check{\lambda}_0\delta^3(\mathbf{r}-\mathbf{r}_0), \quad \mathbf{r}\in\Omega, \tag{1}$$

$$\check{U}(\mathbf{r}) = 0, \quad \mathbf{r}\in\partial\Omega, \tag{2}$$

where δ^3 is the three-dimensional Dirac delta distribution. The number of unknown parameters is reduced by non-dimensionalising. We define $\kappa_0^2 = \check{\kappa}_0^2L^2/\check{D}$, $\lambda_0 = \nu\check{\lambda}_0L^2/\check{D}$, $U = \check{U}L^3$ and $A = \check{A}L^4/\nu$, where L is the characteristic spatial scale of Ω and is used to nondimensionalise space. The dimensionless model is

$$-\triangle U(\mathbf{r}) + \kappa_0^2U(\mathbf{r}) = \lambda_0\delta^3(\mathbf{r}-\mathbf{r}_0), \quad \mathbf{r}\in\Omega, \tag{3}$$

$$U(\mathbf{r}) = 0, \quad \mathbf{r}\in\partial\Omega. \tag{4}$$

The boundary-bound species is given in terms of U,

$$A(\mathbf{r}) = -\nabla U(\mathbf{r})\cdot\hat{\mathbf{n}}, \quad \mathbf{r}\in\partial\Omega. \tag{5}$$

In (3)–(5), $\check{\Omega}$ has been normalised into Ω to have order 1 size. The number of unknown parameters in the model (3)–(5) is 5; κ_0, λ_0 and the three coordinates associated with $\mathbf{r}_0$. Whilst it is possible to scale out the parameter λ_0 by dividing through U and A, we have not done this explicitly here in the problem statement because λ_0 is an unknown constant. On the other hand, we have scaled $\check{D}_0$ explicitly out of the problem to avoid the fact that any solution to the inverse problem of finding all of the unknown parameters offers an infinite number of solutions simply by multiplication of $\check{D}_0$, $\check{\kappa}_0$ and $\check{\lambda}_0$ by an arbitrary constant. This is due to the arbitrariness of the constant ν. We therefore attempt to find a unique solution (where a unique solution exists) which is normalised with respect to $\check{D}_0$.

The problem that we address in this manuscript is an inverse problem but also a parameter fitting problem. *How should the unknown parameters of* (3)–(5), κ_0, λ_0 *and* $\mathbf{r}_0$, *be determined given data in the form of* $A(\mathbf{r})$ *as defined by* (5) *distributed over* $\partial\Omega$*?* Whilst we will discuss this problem with sufficient generality that it may be extended to general data formats, we will focus most of our attention on solving this problem in cases where data for A is known in the form of regular or sliced 3D data and we also acknowledge that this strategy requires careful consideration depending on the geometry of Ω and availability of data A. Breaking 3D volumes up into 2D regular slices is a common format for biomedical 3D image data (MRI, histology, etc). We define sliced data as data which is known at high resolution 'pixels' on parallel planes separated by some finite (and usually significant) distance h. In general, we denote that $A(\mathbf{r})$ is known only at discrete points/pixels $\mathbf{r} = \bar{\mathbf{r}}_p$, $p = 1, \ldots, N_p$, on the surface $\partial\Omega$. We will use the notation $\bar{A}_p = A(\bar{\mathbf{r}}_p)$ and $\bar{A} = \left\{\bar{A}_p\right\}_{p=1}^{N_p}$. For the remainder of this manuscript, the overbar notation indicates the association with either (1) the discrete numerical data $\bar{A}$ (rather than the underlying continuous data A) or (2) a discrete numerical/computational analogue of the indicated continuous function/operation acting upon this data. It is important to remember that operations with an overbar are numerical approximate operations and are subject to numerical error.

We will demonstrate our numerical algorithm on two main test problems which are designed so that the data may be generated exactly by solving analytically the forward problem (3)–(5) with known parameters. These test problems are outlined in the following two subsections.

2.1.1 Test problem 1: the unit cube

To demonstrate the accuracy of our proposed algorithm on the simplest domain, Test problem 1 will make use of the special case $\Omega = (-1/2, 1/2)^3 = C$, a unit cube centred at the origin.

For this proof-of-concept test problem, we will use a simple regular lattice on which raw data $\bar{A}$ is known. That is, for each of the six faces of the cube, the square surface is subdivided into $N_s \times N_s$ regular squares and $\bar{A}_p = A(\bar{\mathbf{r}}_p)$ is evaluated at $\bar{\mathbf{r}}_p$ for $p = 1, \ldots, 6N_s^2$; the centres of each of these small squares. For Test problem 1, we naturally use Cartesian coordinates $\mathbf{r} = (x, y, z)$ (with all relevant indices and markings associated with their corresponding position vector).

Test problem 1 also has an analytically-tractable forward solution which we use to generate test data. The solution to (3)–(4) can be found using the method of images:

$$U(\mathbf{r}) = \lambda_0 \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} \frac{s_i s_j s_k \exp\left(-\kappa_0 \left|\delta\mathbf{r}_{i,j,k}\right|\right)}{4\pi \left|\delta\mathbf{r}_{i,j,k}\right|}, \tag{6}$$

where $\delta \mathbf{r}_{i,j,k} = \mathbf{r} - \mathbf{r}_{i,j,k}$, s_m is -1 if m is odd and 1 if m is even and, in Cartesian coordinates, $\mathbf{r}_{i,j,k} = (i, j, k) + (s_i x_0, s_j y_0, s_k z_0)$. The summation in (6) is taken over all combinations of integer indices i, j, and k from $-\infty$ to ∞, however, for practical computation and noting that the summand decays rapidly when $\mathbf{r} \in C$ as $|i|$, $|j|$, and $|k|$ become large, this summation is truncated (we found truncating at $i, j, k = \pm 20$ to be ample for very high accuracy—even when $\kappa_0 = 0$). The flux A can be found directly using (5) and (6)

$$A(\mathbf{r}) = \lambda_0 \sum_{(i,j,k)=-\infty}^{\infty} s_i s_j s_k \frac{(\kappa_0 |\delta \mathbf{r}_{i,j,k}| + 1)(\delta \mathbf{r}_{i,j,k} \cdot \hat{\mathbf{n}})}{|\delta \mathbf{r}_{i,j,k}|^3} \exp(-\kappa_0 |\delta \mathbf{r}_{i,j,k}|), \quad (7)$$

where $\hat{\mathbf{n}}$ is the outward-facing normal to C and depends on the face of ∂C on which $\mathbf{r}$ lies.

2.1.2 Test problem 2: the unit ball

Test problem 2 is the special case where $\Omega = B$, a unit ball centred at the origin. As a sphere presents a significantly more complex computational challenge, we shall dedicate the bulk of the analysis of our proposed algorithm on this problem, a computational analysis which thoroughly tests for accuracy and robustness. There are three reasons for choosing this test problem: (1) the forward problem of calculating A given a set of parameters $\{\kappa_0, \lambda_0, \mathbf{r}_0\}$ has an analytic solution so that our results may be compared with known solutions, (2) our biological motivation for the methodology presented in this manuscript is as a pilot study for a more ambitious task of finding multiple point sources of morphogen within (roughly) spherical biological cells using experimental 2D slice images and (3) more generally, spherical/rounded domains are common in biological applications involving morphogens (tumours, cells, developing tissues, etc.).

For Test problem 2, we will naturally use spherical coordinates. All position vectors $\mathbf{r}$ are represented with coordinates (r, ϕ, θ) where $r = |\mathbf{r}|$ and ϕ and θ are the polar and azimuthal angles of $\mathbf{r}$, respectively (coordinates for position vectors with indices and other markings also have these associated markings). We consider experimental data $\bar{A}$ to be provided on the boundary of a series of N_z 2D circular images of 'slices' through the volume Ω. That is, we consider that known flux data is $\bar{A}_p = A(\bar{\mathbf{r}}_p)$ for $p = 1, \ldots, N_z N_\theta$ on ∂B where $N_\theta \gg N_z$ is the nominally large number of data points around the circular circumference of each image. We shall also use the notation $\bar{\mathbf{r}}_p = \bar{\mathbf{r}}_{i,j}$ for all $i = 1, \ldots, N_z$ and all $j = 1, \ldots, N_\theta$ corresponding to the ith slice and jth azimuthal data point. Slices are evenly spaced from pole to pole of the ball (the poles are defined by the orientation of the slices) whilst the N_θ data points in the azimuthal direction are evenly spaced (albeit at a much higher resolution) around the 2D disk made by each slice. That is, $\bar{z}_i = \cos(\bar{\phi}_i) = -1, -1 + 2/(N_z - 1), -1 + 4/(N_z - 1), \ldots, 1$ respectively and $\bar{\theta}_j = 0, 2\pi/N_\theta, 4\pi/N_\theta, \ldots, 2\pi(N_\theta - 1)/N_\theta$ respectively. Presenting the data $\bar{A}$ in this (sliced) way on $\partial\Omega$ significantly complicates defining quadrature rules which accurately approximate integrals over the surface, however this

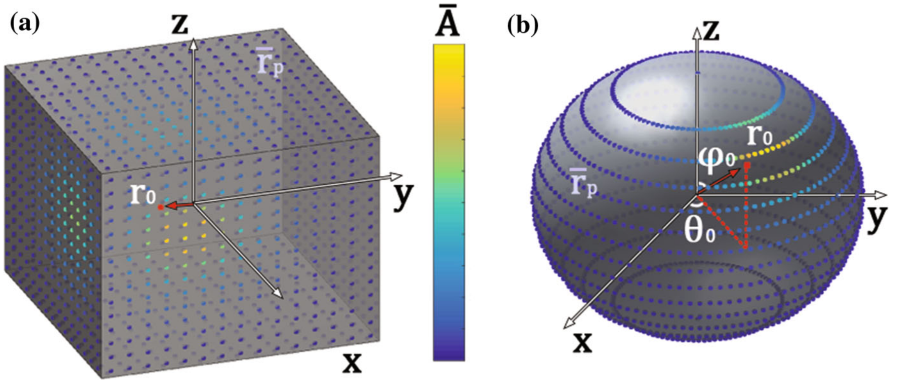

Fig. 1 Diagrams of test problem geometries Ω for **a** Test problem 1 and **b** test problem 2. Indicated on each diagram is the position of a known source $\mathbf{r}_0$ and distribution of data on the surface $\partial\Omega$. At each point $\bar{\mathbf{r}}_p$ the data $\bar{A}_p = A(\mathbf{r}_p)$ is simulated by Eqs. (7) and (9) respectively. The problem addressed in this manuscript is how can one find $\mathbf{r}_0$ and the unknown parameters κ_0 and λ_0 given only $\bar{A}$ at the points $\bar{\mathbf{r}}_p$

is a challenge that is common when biomedical imaging data is provided for modelling purposes.

It is possible to derive the solution to (3)–(4) by separation of variables in spherical coordinates, for Test problem 2:

$$U(\mathbf{r}) = \frac{\kappa_0\lambda_0}{2\pi^2}\sum_{l=0}^{\infty} (2l+1)\left(i_l(\kappa_0 r_<)k_l(\kappa_0 r_>) - \frac{k_l(\kappa_0)}{i_l(\kappa_0)}i_l(\kappa_0 r)i_l(\kappa_0 r_0)\right)P_l\left(\cos\alpha_0\right), \quad (8)$$

where $r_< = \min(r, r_0)$, $r_> = \max(r, r_0)$ and $\cos\alpha_0$ is the cosine of the angle between $\mathbf{r}$ and $\mathbf{r}_0$. The functions P_l, i_l and k_l are the lth Legendre polynomial and lth order modified spherical Bessel functions of the first and second kind, respectively. The flux A can be computed directly from (5) and (8).

$$A(\mathbf{r}) = \frac{\lambda_0}{4\pi}\sum_{l=0}^{\infty}(2l+1)\frac{i_l(\kappa_0 r_0)}{i_l(\kappa_0)}P_l\left(\cos\alpha_0\right). \quad (9)$$

We generate test data $\bar{A}_p = A(\bar{r}_p)$ for which we know the true parameters $\{\kappa_0, \lambda_0, \mathbf{r}_0\}$ using (9). Figure 1 is a diagram of the domains for both Test problem 1 and 2 together with their notation.

2.1.3 Black box optimisation method

We used a variant of the Covariance Matrix Adaptation Evolutionary Strategy (CMA-ES) algorithm to find a numerical solution to the inverse problem. CMA-ES is a stochastic, population-based search algorithm for $\mathbb{R}^n$ spaces. At each iteration step, new solutions are generated by sampling a multivariate normal distribution $\mathcal{N}(\boldsymbol{\mu}, \mathbf{C})$, where $\boldsymbol{\mu}$ is the current solution and $\mathbf{C}$ is a $n \times n$ covariance matrix. The best solutions are used to update the distribution for the next iteration. In particular, $\mathbf{C}$ is adapted in such a way that its eigenvectors align with the direction of the gradient. As such, CMA-ES can be considered a stochastic equivalent to a quasi-Newton method, because $\mathbf{C}$ fills the same role of the inverse Hessian of retaining information about the curvature of the space (Hansen and Ostermeier 2001).

The variant employed in this paper, known as BIPOP-CMA-ES, uses randomized restarts whenever the method converges to a local optima (Hansen 2009). At each restart, the algorithm uses one of two regimes to modify the size of the population of solutions, aiming to improve its global or local search capabilities. This approach has been shown to be outperform several state-of-the-art algorithms on poorly conditioned non-separable functions (Hansen et al. 2010). As such, this algorithm is a suitable candidate for tackling our inverse fitting problem.

2.2 Computational intractability of common parameter fitting techniques

At first glance, the closed-form solutions to the forward problem (6) and (8) respectively suggests an obvious method to determine the parameters $\mathbf{r}_0$, λ_0 and κ_0 given $\bar{A}_p = A(\bar{r}_p)$.

For example, a brute force least-squares optimisation could take the form of

$$\{\kappa_0, \lambda_0, \mathbf{r}_0\} = \underset{\{\kappa,\lambda,\boldsymbol{\rho}\}}{\operatorname{argmin}} \sum_{p=1}^{N_p} \left(A(\bar{\mathbf{r}}_p; \kappa, \lambda, \boldsymbol{\rho}) - \bar{A}_p\right)^2 ,$$

where $A(\bar{\mathbf{r}}_p; \kappa, \lambda, \boldsymbol{\rho})$ are the actin values predicted by the model (5) at the pixel points $\bar{\mathbf{r}}_p$. Alternatively, Bayesian methods have been used to parameterise dynamical systems models in biology, including Approximate Bayesian Computation (Toni et al. 2009) however these need to be used with care for deterministic models (Alahmadi et al. 2020). The common bottleneck for all of these methods is that they require the repeated computation of $A(\bar{\mathbf{r}}_p; \kappa, \lambda, \boldsymbol{\rho})$, data predicted by the model given the 5 parameters (κ, λ and 3 coordinates of $\boldsymbol{\rho}$). This is not practical for the class of problems considered in this paper.

The obvious method to finding $A(\bar{\mathbf{r}}_p; \kappa, \lambda, \boldsymbol{\rho})$ is to solve the PDE (3) directly. This is very computationally inefficient, especially in 3D. Alternatively, one can simulate trajectories of the SDE which has a Fokker–Planck equation given by the modified Helmholtz model to estimate $A(\bar{\mathbf{r}}_p; \kappa, \lambda, \boldsymbol{\rho})$. That is, one starts with many initial random walkers starting at $\boldsymbol{\rho}$. Each walker moves a distance $\sqrt{6\delta}\xi$ (where $\xi \sim \mathcal{N}(0, 1)$ is a unit variance normally distributed random number with mean zero) in a random direction (in 3D) and has a probability $\kappa\delta$ of decay for each walking step. The parameter δ should be chosen to be small for improved accuracy. When the walkers are incident on $\partial\Omega$ they are removed and the probability density of walkers on $\partial\Omega$ is an approximation for $A(\bar{\mathbf{r}}_p; \kappa, \lambda, \boldsymbol{\rho})/\lambda$. In either case, evaluating $A(\bar{\mathbf{r}}_p; \kappa, \lambda, \boldsymbol{\rho})$ is very costly in most situations. In the test examples in this manuscript, we use special cases of Ω where

$A(\bar{\mathbf{r}}_p; \kappa, \lambda, \boldsymbol{\rho})$ has analytic forms and even in these cases, repeated evaluation is costly when using direct fitting methods.

As a demonstration of the computational cost of solving the problem when $A(\bar{\mathbf{r}}_p; \kappa, \lambda, \boldsymbol{\rho})$ is repeatedly required, we shall present the least-squares fitting of data to the model $A(\bar{\mathbf{r}}_p; \kappa, \lambda, \boldsymbol{\rho})$ using a state-of-the-art black-box optimisation technique described in Sect. 2.1.3. This black-box optimisation algorithm is also as part of our method which is the focus of the manuscript but in that case is used on a different optimisation problem.

Since the least-squares fitting method will converge depending on the initial guess, we ran 30 instances of the optimisation process, from 30 different random initialisations (selected within a reasonable truncated subset of the constraints of the parameter space). We ran this optimisation for demonstrative purposes only on Test problem 2 ($\Omega = B$), using data $\bar{A}$ generated by setting $N_z = 15$, $N_\theta = 150$, $r_0 = 0.8$, $\phi_0 = \pi/6$, $\theta_0 = \pi/3$, $\kappa_0 = \pi/2$ and $\lambda_0 = 1$. As the optimisation process works by iteratively sampling the model outputs using updated parameter estimates which eventually converge on close approximations of the true parameter values, a large number model output evaluations may be needed for a good approximations. We evaluate the errors associated with a given set of parameters estimates by taking the absolute value (Euclidean distance in the case of $\mathbf{r}_0$) of the difference between the estimate and the true parameter. We then find the relative error by normalising the absolute error with respect to the initial error averaged over the 30 iterations of the optimisation process so that the mean of the initial relative errors for each parameter is 1.

In Fig. 2, we plot the mean (and interquartile range) over the 30 reinitialisations of the relative error in each of the three parameters κ, λ and $\boldsymbol{\rho}$ as a function of the cumulative number of times the complete model data $A(\bar{\mathbf{r}}_p; \kappa, \lambda, \boldsymbol{\rho})$ was computed (that is, the number of times a solution to (3)–(5) needs to be computed). As can be seen clearly from Fig. 2, there is a substantial period of exploration of the parameter space. Between approximately 1000 and 20,000 solutions to (3)–(5) are required (depending on the initial guess) for the optimisation process to explore the solution space before converging towards the true value. Whilst the accuracy is eventually assured using this algorithm the real cost of its use for this problem is the number of times the model needs to be computed. For the test cases in this manuscript, this computation amounts to evaluating large sums which have analytic expressions but these are special cases. In the vast majority of real-world applications, each new model computation involves the numerical solution of the full 3D model (3)–(5). In most cases, 20,000 model solutions would be prohibitively costly from a computational perspective.

It should be noted that this is purely for motivational purposes only and whilst other methods might improve on computational costs, these types of direct methods are limited when calculation or estimation of $A(\bar{\mathbf{r}}_p; \kappa, \lambda, \boldsymbol{\rho})$ is difficult. Furthermore, due to the ill-conditioned nature of this inverse problem, if A cannot be estimated with extreme accuracy, large inaccuracies in the fitted parameters will result.

Our proposed method does not require the calculation of the forward problem at all and therefore is not affected by the limitations of the direct methods. Instead, our method assumes that the data A can be numerically integrated easily. Because the limitations of our method and traditional direct methods are so different, they cannot be easily compared and their suitability depends on the context of the problem.

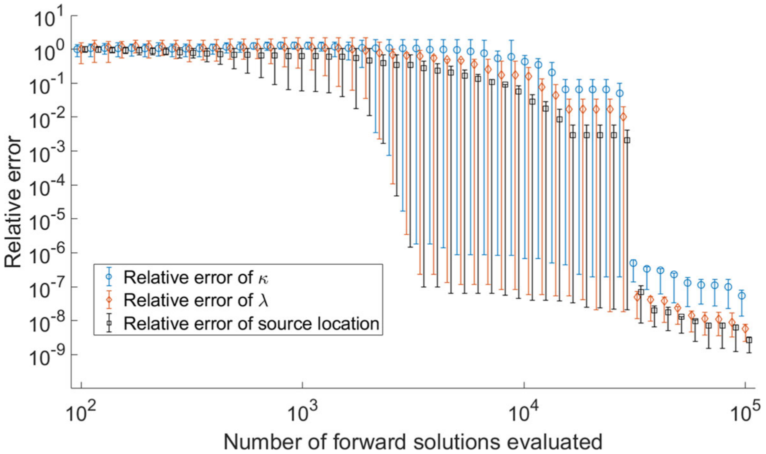

Fig. 2 The error associated with an estimated solution, normalised to the mean error of the initial guess, of a standard least squares fit of model $A(\mathbf{r}_p; \kappa, \lambda, \boldsymbol{\rho})$ to data $\bar{A}_p$ as a function of the number of times the forward model solutions $A(\mathbf{r}_p; \kappa, \lambda, \boldsymbol{\rho})$ are evaluated. For each point, the interquartile range distributed over 30 reinitialisations of the optimisation algorithm is presented. The normalised errors are plotted for the parameters κ, λ and the source location. The first two of these parameters uses the absolute difference to measure error where as the latter uses the Euclidean norm

Furthermore, even if a closed-form solution such as (9) is available, it may be difficult to compute model outputs, even when the parameters are known (or guessed). For example in (9), since $0 < r_0 < 1$, it is possible to show that the series in (9) converges but it may not be obvious where this series should be truncated and furthermore, as more terms are taken, the quotient $i_l(\kappa_0 r_0)/i_l(\kappa_0)$ may rapidly be subject to substantial floating point errors and various asymptotic approximations need to be used. In our case, in order to generate data, we use the asymptotic approximation $i_l(\kappa_0 r_0)/i_l(\kappa_0) \sim r_0^l$ for large l to avoid floating point errors as l increases and truncate the sum when the summand becomes insignificantly small (Paris 1984). We found that $l = 20$ is sufficiently high to truncate the sum except in cases where $\kappa_0 \gg 1$ or $0 < (1 - r_0) \ll 1$. Since both of these cases result in highly localised δ function-like distributions of A, we truncate our solution space to exclude such extremes. The posedness of the problem depends on both the domain $\varOmega$ and the true values $\{\kappa_0, \lambda_0, \mathbf{r}_0\}$ from which $\bar{A}$ is generated. For example, whilst the example used to generate Fig. 2 has a clearly unique optimal solution (towards which all 30 initialisations converge) this is only because κ_0 is not too large or small, ϕ_0 is not near the extremes, etc. In other situations, the problem may be more ill-posed which will effect convergence and accuracy. In fact, there are cases in which A is not uniquely defined by model parameters. For example, for Test problem 2, if $\mathbf{r}_0$ is at the origin, Eq. (9) gives $A = \lambda_0/(4\pi i_0(\kappa_0))$, a constant at all points on ∂B and thus there are an infinite number of solutions for the parameter pair $\{\kappa_0, \lambda_0\}$ (despite that in each solution $\mathbf{r}_0$ is at the origin). As we shall see later and as it was alluded to in Fig. 2, the position vector $\mathbf{r}_0$ can be consistently determined with more accuracy than either λ_0 or κ_0.

In this manuscript, we will propose a method for solving this inverse fitting problem without needing to solve the forward modified Helmholtz model (3), irrespective of the geometry of Ω. In the next section we outline some of the mathematical background required for this algorithm.

2.3 Theory in the limit of continuous data

Consider the closed modified Helmholtz model defined by (3)–(5) with a compact domain $\bar{\Omega}$ embedded in the Euclidean three-dimensional space $\mathbb{R}^3$ ($\bar{\Omega} \subset \mathbb{R}^3$) with piecewise smooth boundary $\partial\Omega$ and unknown parameters $\kappa_0 \in [0, \infty)$, $\lambda_0 \in (0, \infty)$ and $\mathbf{r}_0 \in \Omega$, where $\bar{\Omega}$ is the closure of the set Ω.

Consider an extension of the domain from Ω to $\mathbb{R}^3$. At the same time, to simplify working, we define the normalised solution $u = U/\lambda_0$ and the normalised flux $a = A/\lambda_0$. We extend the domain such that the solution u remains continuous everywhere and also satisfies the PDE (3) outside of $\bar{\Omega}$.

$$-\triangle u(\mathbf{r}) + \kappa_0^2 u(\mathbf{r}) = \delta^3(\mathbf{r} - \mathbf{r}_0), \quad \mathbf{r} \in \left(\mathbb{R}^3 \backslash \partial\Omega\right), \tag{10}$$

$$u(\mathbf{r}) = 0, \quad \mathbf{r} \in \partial\Omega. \tag{11}$$

Note that this extension to $\mathbb{R}^3$ seems uninteresting at this point since $u = 0$ for $\mathbf{r} \notin \Omega$ due to the fact the only source is inside the Dirichlet boundary condition $\partial\Omega$. Next, we define the (free) Green's function for the system in $\mathbb{R}^3$ using an unknown parameter $\kappa \in [0, \infty)$ (not necessarily equal to κ_0) as the solution to

$$-\triangle G_\kappa(\mathbf{r}; \mathbf{r}') + \kappa^2 G_\kappa(\mathbf{r}; \mathbf{r}') = \delta^3(\mathbf{r} - \mathbf{r}'), \quad \mathbf{r} \in \mathbb{R}^3, \quad \mathbf{r}' \in \mathbb{R}^3, \tag{12}$$

which can be computed analytically and is well-known as the fundamental solution of the modified Helmholtz equation:

$$G_\kappa(\mathbf{r}; \mathbf{r}') = \frac{\exp\left(-\kappa \left|\mathbf{r} - \mathbf{r}'\right|\right)}{4\pi \left|\mathbf{r} - \mathbf{r}'\right|}. \tag{13}$$

We define the integral transform $\mathcal{T}_\kappa$ which takes $a(\mathbf{r})$ defined on $\mathbf{r} \in \partial\Omega$ and generates a new function $\mathcal{T}_\kappa(\mathbf{r}; a)$ defined on $\mathbf{r} \in \mathbb{R}^3$

$$\mathcal{T}_\kappa(\mathbf{r}; a) = \oint\!\!\!\oint_{\partial\Omega} G_\kappa(\mathbf{r}; \mathbf{r}') a(\mathbf{r}')\, \mathrm{d}S', \quad \mathbf{r} \in \mathbb{R}^3, \tag{14}$$

where the integral is taken over the surface $\partial\Omega$ with respect to the coordinates of $\mathbf{r}'$. Using (5) and the divergence theorem,

$$\mathcal{T}_\kappa(\mathbf{r}; a) = -\iiint_{\Omega} \nabla' \cdot \left(G_\kappa(\mathbf{r}; \mathbf{r}') \nabla' u(\mathbf{r}')\right)\, \mathrm{d}V', \tag{15}$$

where ∇' is the gradient taken with respect to coordinates of $\mathbf{r}'$ and the integral is taken over $\mathbf{r}' \in \Omega$. The integrand in (15) can be rewritten:

$$\mathcal{T}_\kappa(\mathbf{r}; a) = \iiint_\Omega -\nabla' \cdot \left(u(\mathbf{r}')\nabla' G_\kappa(\mathbf{r}; \mathbf{r}')\right) + \tag{16}$$

$$- \; G_\kappa(\mathbf{r}; \mathbf{r}')\, \triangle' u(\mathbf{r}') + u(\mathbf{r}')\, \triangle G_\kappa(\mathbf{r}; \mathbf{r}')\, \mathrm{d}V', \tag{17}$$

where $\triangle'$ is the Laplacian with respect to coordinates of $\mathbf{r}'$, and we note that $\triangle G_\kappa(\mathbf{r}; \mathbf{r}') = \triangle' G_\kappa(\mathbf{r}; \mathbf{r}')$ due to labelling symmetry in (13). Of the three terms in the integrand, the first integrates to zero after application of the divergence theorem and noting (11). For the second and third terms, we can substitute (10) and (12) respectively

$$\begin{aligned} \mathcal{T}_\kappa(\mathbf{r}; a) &= \iiint_\Omega G_\kappa(\mathbf{r}; \mathbf{r}') \left(\delta^3(\mathbf{r}' - \mathbf{r}_0) - \kappa_0^2 u(\mathbf{r})\right) \\ &\quad + u(\mathbf{r}') \left(\kappa^2 G_\kappa(\mathbf{r}; \mathbf{r}') - \delta^3(\mathbf{r} - \mathbf{r}')\right) \mathrm{d}V' \qquad (18) \\ &= G_\kappa(\mathbf{r}; \mathbf{r}_0) - u(\mathbf{r}) + \left(\kappa^2 - \kappa_0^2\right) \tilde{u}(\mathbf{r}; \kappa), \qquad (19) \end{aligned}$$

where

$$\tilde{u}(\mathbf{r}; \kappa) = \iiint_\Omega G_\kappa(\mathbf{r}; \mathbf{r}') u(\mathbf{r}')\, \mathrm{d}V'. \tag{20}$$

Since, by construction, $u(\mathbf{r}) = 0$ for all $\mathbf{r} \in \mathbb{R}^3 \backslash \Omega$, if $\mathbf{r} \in \mathbb{R}^3 \backslash \Omega$ and $\kappa = \kappa_0$, (19) reduces to $\mathcal{T}_\kappa(\mathbf{r}; a) = G_\kappa(\mathbf{r}; \mathbf{r}_0)$. This is important because G_κ has a succinct analytical form (13) which is independent of the domain Ω and we expect this analytic form for all points outside the domain Ω. Using (13) and (19), for $\mathbf{r} \in \mathbb{R}^3 \backslash \Omega$, we define the function f;

$$\begin{aligned} f(\mathbf{r}; a, \kappa, \boldsymbol{\rho}) &= \log\left(4\pi \mathcal{T}_\kappa(\mathbf{r}; a)\, |\mathbf{r} - \boldsymbol{\rho}|\right) + \kappa\, |\mathbf{r} - \boldsymbol{\rho}| \qquad (21) \\ &= \kappa\left(|\mathbf{r} - \boldsymbol{\rho}| - |\mathbf{r} - \mathbf{r}_0|\right) + \log\left(\frac{|\mathbf{r} - \boldsymbol{\rho}|}{|\mathbf{r} - \mathbf{r}_0|}\right) \\ &\quad + \log\left(1 + \frac{\left(\kappa^2 - \kappa_0^2\right) \tilde{u}(\mathbf{r}; \kappa)}{G_\kappa(\mathbf{r}; \mathbf{r}_0)}\right), \qquad (22) \end{aligned}$$

The function f is computed from a and parameter estimates κ and $\boldsymbol{\rho}$ (21). What we have shown here is that if the parameters match that of the model which generated the data a ($\kappa = \kappa_0$ and $\boldsymbol{\rho} = \mathbf{r}_0$), $f(\mathbf{r}; a, \kappa, \boldsymbol{\rho}) = 0$ for all $\mathbf{r} \notin \bar{\Omega}$. The function f has no real biological significance. Roughly speaking it measures the difference between the parameters κ and $\boldsymbol{\rho}$ and their true values by its variation between different points $\mathbf{r}$.

Importantly, a is unknown since the value of λ_0, which is used to normalise the known flux A, is unknown. Note however that since $a = A/\lambda_0$, by the definitions (14) and (21)

$$f(\mathbf{r}; A, \kappa, \boldsymbol{\rho}) = f(\mathbf{r}; a, \kappa, \boldsymbol{\rho}) + \log(\lambda_0). \tag{23}$$

That is, evaluating f using an unnormalised flux A and correct guesses $\kappa = \kappa_0$ and $\boldsymbol{\rho} = \mathbf{r}_0$ returns $\log(\lambda_0)$, a constant, for all $\mathbf{r} \notin \bar{\Omega}$ irrespective of the morphology of the domain Ω. Thus, instead of fitting the data $\bar{A}$ to the model A which requires the solution of a PDE model in 3D, we simply fit $f(\mathbf{r}; \bar{A}, \kappa, \boldsymbol{\rho}) - \log(\lambda) = 0$ using (21) and (23). The cost of this simplification, however, is that it requires the numerical computation of $\mathcal{T}_\kappa(\mathbf{r}; A)$ for some point(s) $\mathbf{r} \notin \bar{\Omega}$ which is not as trivial as it may seem and relies on the data $\bar{A}$ being collected in a manner which allows for accurate quadrature. As we shall also see, accurate quadrature may be assisted by strategic choices for $\mathbf{r}$.

3 Computational method

3.1 Foundations of the inversion algorithm

Inspired by the result (22) and (23), the main strategy of our algorithm is to perform the following constrained optimisation

$$\{\kappa_0, \lambda_0, \mathbf{r}_0\} = \underset{\{\kappa,\lambda,\boldsymbol{\rho}\}}{\operatorname{argmin}} \sum_{i=1}^{N} w_i' \left(f(\mathbf{r}_i; A, \kappa, \boldsymbol{\rho}) - \log(\lambda)\right)^2, \tag{24}$$

for weights w_i' and some chosen set of points $\mathbf{r}_i \notin \bar{\Omega}$, $i = 1, \ldots, N$, constrained by $\kappa \in [0, \infty)$, $\lambda \in (0, \infty)$ and $\boldsymbol{\rho} \in \Omega$. Dealing with discrete data, in practice we are only capable of computing

$$\{\kappa_0, \lambda_0, \mathbf{r}_0\} \approx \underset{\{\kappa,\lambda,\boldsymbol{\rho}\}}{\operatorname{argmin}} \sum_{i=1}^{N} w_i' \left(\bar{f}(\mathbf{r}_i; \bar{A}, \kappa, \boldsymbol{\rho}) - \log(\lambda)\right)^2. \tag{25}$$

In (25), we use the bar notation $\bar{f}$ to represent a numerical approximation to f utilising numerical integration schemes in the transformation $\bar{\mathcal{T}}_\kappa(\mathbf{r}; \bar{A})$ (in turn, a numerical approximation of $\mathcal{T}_\kappa(\mathbf{r}; A)$ as defined by (14)—see Sect. 3.2);

$$\bar{f}(\mathbf{r}_i; \bar{A}, \kappa, \boldsymbol{\rho}) = \log\left(4\pi \bar{\mathcal{T}}_\kappa(\mathbf{r}_i; \bar{A}) \left|\mathbf{r}_i - \boldsymbol{\rho}\right|\right) + \kappa \left|\mathbf{r}_i - \boldsymbol{\rho}\right|. \tag{26}$$

The choice of $\mathbf{r}_i$ is, at least in principle, arbitrary but the best choice, in practice, is not obvious. Since the domain is not one-dimensional, choosing $\mathbf{r}_i$ in a regular deterministic way with constant weights w_i' leads to biases in the evaluation of the objective function depending on the location of $\boldsymbol{\rho}$. That is, as $\boldsymbol{\rho}$ changes, so too does the density of points $\mathbf{r}_i$ as a function of distance $|\mathbf{r} - \boldsymbol{\rho}|$ which appears in (26)—note that $\bar{\mathcal{T}}_\kappa(\mathbf{r}_i; \bar{A})$ is dependent only on $|\mathbf{r}_i - \boldsymbol{\rho}|$ near the optimum set of parameters [see (19)]. We will therefore choose $\mathbf{r}_i$ in a random but constrained way (outlined in Sect. 3.3). We ensure that the weights w_i' are dependent on $\mathbf{r}_i$ according to their distance from $\boldsymbol{\rho}$

such that the weighting is independent of the density of points as a function of distance from $\boldsymbol{\rho}$. We first label each $\mathbf{r}_i$ in order of closest to furthest from $\boldsymbol{\rho}$ and choose

$$w_i' = \tau_i w_i = \frac{w_i}{2(d_N - d_1)} \begin{cases} d_2 - d_1, & i = 1, \\ d_N - d_{N-1}, & i = N, \\ d_{i+1} - d_{i-1}, & \text{otherwise} \end{cases}, \tag{27}$$

where $d_i = |\mathbf{r}_i - \boldsymbol{\rho}|$. Here, w_i does not explicitly depend on $\mathbf{r}_i$. By using (27) we are using the trapezoidal method to find the average value of the weighted summand as a function of distance from $\boldsymbol{\rho}$, thereby making the scheme as unbiased as possible due to clustering of points $\mathbf{r}_i$ a distance from an unknown point $\boldsymbol{\rho}$. The weights w_i will be used to better condition the optimisation (25) against inaccurate objective function minimums caused by numerical error in the calculation of $\bar{f}$.

The optimisation (25) itself can, in principle, be performed using any number of techniques but we use the BIPOP-CMS-ES algorithm described earlier in Sect. 2.2. In the proceeding subsections we discuss two remaining outstanding issues that need to be addressed in order to perform the optimisation (25) appropriately. These issues are; how best to apply quadrature to the data $\bar{A}$ [see (14)] to calculate $\bar{\mathcal{T}}_\kappa(\mathbf{r}; \bar{A})$ (Sect. 3.2), as well as the choice of the weights w_i, and the choice of $\mathbf{r}_i \notin \bar{\Omega}$ to obtain the most reliable parameter estimates (Sect. 3.3).

The summary of the inversion/fitting algorithm is as follows. Start with some compact volume Ω and data $\bar{A}_p$ which is assumed by the model to be the solution A to (3)–(5) evaluated at data points $\mathbf{r}_p \in \partial\Omega$ for some unknown parameters κ_0, λ_0 and $\mathbf{r}_0$. We can obtain a good approximation for the unknown parameters by taking the following steps

1. Determine quadrature rules for accurate numerical integration over $\partial\Omega$ using only the points $\mathbf{r}_p$. For the two test problems in this manuscript, we discuss this in Sect. 3.2.
2. Generate N points $\mathbf{r}_i$ which lie outside of the volume Ω. Care should be taken to generate suitable points and a discussion of this can be found in Sect. 3.3. We find that $N = 30$ is sufficiently large in most cases.
3. For a given guess of the parameters κ_0, λ_0 and $\mathbf{r}_0$ (labelled κ, λ and $\boldsymbol{\rho}$) evaluate the cost/objective function [see (25)]. This objective function is constructed as follows
   (a) At each point $\mathbf{r}_i$ use the quadrature rule from Step 1 to calculate $\bar{\mathcal{T}}_\kappa(\mathbf{r}_i; \bar{A})$, a numerical approximation to the transform described in (14).
   (b) Calculate at each point $\mathbf{r}_i$ the value of $w_i \left(\bar{f}(\mathbf{r}_i; \bar{A}, \kappa, \boldsymbol{\rho}) - \log(\lambda)\right)^2$ where $\bar{f}$ is defined in (26) and w_i is discussed in Sect. 3.3.
   (c) Arrange the points $\mathbf{r}_i$ in order of smallest $|\mathbf{r}_i - \boldsymbol{\rho}|$ to largest and either use (27) to change w_i to w_i' and sum from $i = 1$ to $i = N$ to obtain the objective function found in (25) or simply find the average value of $w_i \left(\bar{f}(\mathbf{r}_i; \bar{A}, \kappa, \boldsymbol{\rho}) - \log(\lambda)\right)^2$ as a function of $|\mathbf{r}_i - \boldsymbol{\rho}|$ using a reliable integration technique [we use the trapezoidal method to generate (27)].
4. Optimise the objective/cost function [see (25)] to find an approximation of the parameters κ_0, λ_0 and $\mathbf{r}_0$. The landscape of the objective function in the parameter

space may not be ‘nice’ and a sophisticated optimisation algorithm may be necessary. The BIPOP-CMA-ES we use in this manuscript is commonly used when no analytical expression is available for the objective function. This is the case when considering a general domain morphology Ω and is the main reason we use it here.

3.2 Calculation of $\bar{\mathcal{T}}_\kappa\left\{\bar{A}\right\}$

Performing the numerical transformation $\bar{\mathcal{T}}_\kappa$ based on (14) numerically and accurately for use in $\bar{f}$ in (25) depends heavily on the format of the data.

In Test problem 1 (see Sect. 2.1.1), a regular square grid is used for the raw data on $\partial\Omega$ and numerical integration can be achieved simply using a Riemann sum. Specifically,

$$\bar{\mathcal{T}}_\kappa(\mathbf{r};\,\bar{A}) = \frac{1}{N_s^2}\sum_{p=1}^{N_p} G_\kappa(\mathbf{r};\,\bar{\mathbf{r}}'_p)\bar{A}_p, \quad \mathbf{r}\in\mathbb{R}^3, \tag{28}$$

In Test problem 2 (see Sect. 2.1.2), the data is arranged in slices. In this case, it makes sense to split the integration into its azimuthal and polar parts.

$$\begin{aligned}
\bar{\mathcal{T}}_\kappa(\mathbf{r};\,\bar{A}) &= \overline{\int_0^{\pi}} \bar{\gamma}_\kappa(\mathbf{r},\bar{\phi}';\,\bar{A})\sin\bar{\phi}'\,\mathrm{d}\bar{\phi}', && (29)\\
&= \overline{\int_{-1}^{1}} \bar{\gamma}_\kappa(\mathbf{r},\bar{\phi}';\,\bar{A})\,\mathrm{d}(\cos\bar{\phi}'), && (30)\\
&= \sum_{i=1}^{N_z} W_i\bar{\gamma}_\kappa(\mathbf{r},\bar{\phi}'_i;\,\bar{A}), \quad \mathbf{r}\in\mathbb{R}^3,\ \text{where} && (31)\\
\bar{\gamma}_\kappa(\mathbf{r},\bar{\phi}';\,\bar{A}) &= \overline{\int_0^{2\pi}} G_\kappa(\mathbf{r};\,\bar{\mathbf{r}}'(\bar{\theta}',\bar{\phi}'))\bar{A}(\bar{\theta}',\bar{\phi}')\,\mathrm{d}\bar{\theta}', && (32)\\
&= \frac{2\pi}{N_\theta}\sum_{j=1}^{N_\theta} G_\kappa(\mathbf{r};\,\bar{\mathbf{r}}'(\bar{\theta}'_j,\bar{\phi}'))\bar{A}(\bar{\theta}'_j,\bar{\phi}'), && (33)
\end{aligned}$$

where W_i are weights associated with a chosen quadrature. Importantly, we emphasize that quadrature on surfaces are a large field of study. Whilst general methods exist (Reeger et al. 2016), we recommend that care should be taken in choosing a method that is best for data provided. As always, an overbar indicates a numerical approximation has been used. The numerical integral (32) can be done with a low order numerical integration scheme since N_θ is large. We found the Riemann sum (33) over each raw data point on each slice sufficient for (32) however, in principle, this can be done in other ways. The summation in (33) is taken over all data points in the slice with polar angle $\bar{\phi}'$. The numerical integration (30) requires a higher order scheme since N_z may be small. Using a trapezoidal rule for this integration ($W_i = W_i^{\mathrm{tr}} = 1/N_z$ if $i = 1$ or $i = N_z$ otherwise $W_i = W_i^{\mathrm{tr}} = 2/N_z$) gives accurate results when $\bar{\gamma}_\kappa(\mathbf{r},\bar{\phi};\,\bar{A})$ is small near $\bar{\phi} = 0$ and $\bar{\phi} = \pi$ (when the unknown source is not nearly aligned with the z-axis). To improve robustness of our algorithm in these extreme cases, we instead apply Gregory corrections to these weights $W_i = W_i^{\mathrm{tr}} + W_i^{\mathrm{G}}$. The Gregory corrections W_i^{G} are calculated by performing forward and backward finite difference approximation to the error terms of the Euler–Maclaurin formula (Bocher et al. 1994). Specifically, we use weights W_i corresponding to the *order 6 Gregory formula* as defined by Javed and Trefethen (2015) and find that this gives accurate results in most (non-extreme) circumstances. Such a rule, in principle, integrates the smooth function $\bar{\gamma}_\kappa(\mathbf{r}, \bar{\phi}; \bar{A})$ with an accuracy of $\mathcal{O}((2/N_z)^7)$ (Brass and Petras 2011; Hildebrand 1987). Note that choosing a really large order for the Gregory corrections can lead to significant numerical instability.

We have outlined here our quadrature rules for evaluating $\bar{\mathcal{T}}_\kappa$ [an approximation of (14)] for each of our test problems, however for other geometries, suitable quadrature rules should be carefully considered Ω.

3.3 Choice of weights w_i and sample points $\mathbf{r}_i \notin \Omega$

We define the error ϵ_κ associated with calculation of $\bar{\mathcal{T}}_\kappa$;

$$\epsilon_\kappa(\mathbf{r}; \bar{A}) = \bar{\mathcal{T}}_\kappa(\mathbf{r}; \bar{A}) - \mathcal{T}_\kappa(\mathbf{r}; A). \tag{34}$$

Substitution of $\bar{\mathcal{T}}_\kappa$ into the RHS objective function of (25) and using (26) gives

$$\begin{aligned}&\sum_{i=1}^{N} \tau_i w_i \left(\bar{f}(\mathbf{r}_i; \bar{A}, \kappa, \boldsymbol{\rho}) - \log(\lambda)\right)^2 \\ &\quad = \sum_{i=1}^{N} \tau_i w_i \left(f(\mathbf{r}_i; A, \kappa, \boldsymbol{\rho}) - \log(\lambda) + \frac{\epsilon_\kappa(\mathbf{r}_i; \bar{A})}{\bar{\mathcal{T}}_\kappa(\mathbf{r}_i; \bar{A})} + \mathcal{O}\left(\left(\frac{\epsilon_\kappa}{\bar{\mathcal{T}}_\kappa}\right)^2\right)\right)^2.\end{aligned} \tag{35}$$

We require therefore that $|\epsilon_\kappa/\bar{\mathcal{T}}_\kappa| \ll 1$ for good approximation. The accuracy of the optimisation is achieved with the use of a number of heuristically defined strategies which ignore higher order terms in (35) and are outlined in the following subsections.

3.3.1 Constraining κ from above and choosing $\mathbf{r}_i$ sufficiently close to $\partial\Omega$

Given the integrand in (14) and noting (13), if $\mathbf{r}_i$ is either chosen too far from $\partial\Omega$ or κ is large, $\bar{\mathcal{T}}_\kappa(\mathbf{r}_i; \bar{A})$ becomes vanishingly small and is subject to wild floating point errors. In this situation the size of the term $|\epsilon_\kappa/\bar{\mathcal{T}}_\kappa|$ cannot be guaranteed. We therefore constrain further our optimisation (25) to 'reasonable' κ values. To get a handle on 'reasonable' κ values, we note that the diffusion and decay of the morphogen from a source point inside Ω reduces the concentration (in the absence of boundary interference) according to the distribution (13) and is therefore dependent most strongly (exponentially) on the quantity κd where d is the distance from the source. Since we non-dimensionalise Ω so that distances d are $\mathcal{O}(1)$, values of κ any larger than 5 result in highly localised distributions of A on $\partial\Omega$. It is not only unlikely that $\bar{A}$ will have the resolution to resolve these distributions, but also highly localised A results in a very ill-posed parameter fit. We therefore restrict our search of κ to values less than $\kappa_{\max} = 5$. Furthermore, distances from $\partial\Omega$ to $\mathbf{r}_i$ should also necessarily decay $\bar{\mathcal{T}}_\kappa$ in the same way and thus we restrict $\mathbf{r}_i$ to being no further away from $\partial\Omega$ than 1 at its nearest point.

3.3.2 Choosing $\mathbf{r}_i$ sufficiently far from $\partial\Omega$ and away from regions of sparse data points $\bar{\mathbf{r}}_p$

The integrand of $\bar{\mathcal{T}}_\kappa(\mathbf{r}_i; \bar{A})$, see (14), becomes singular if $\mathbf{r}_i$ approaches $\partial\Omega$. Regardless of the accuracy of numerical integration, we therefore expect $|\epsilon_\kappa(\mathbf{r}; \bar{A})|$ to be large compared to $\bar{\mathcal{T}}_\kappa(\mathbf{r}_i; \bar{A})$ if the distance between $\mathbf{r}_i$ and $\partial\Omega$ is on the order of the spacing between data points $\mathbf{r}_p$ (for example, slice separation in Test problem 2). In our numerical examples, we allow the minimum distance between $\mathbf{r}_i$ and $\partial\Omega$ to be no less than 0.3. Together with the constraint raised in Sect. 3.3.1, all of the points $\mathbf{r}_i$ are chosen within the interval of distances 0.3 to 1 of the closest point on $\partial\Omega$. For Test problem 1 this means $\mathbf{r}_i$ lie inside a box of thickness 0.7 surrounding Ω, whilst for Test problem 2 $\mathbf{r}_i$ lie inside a spherical shell of thickness 0.7 surrounding Ω. To further improve results, it is important that $\mathbf{r}_i$ be chosen close to higher densities of data points $\bar{\mathbf{r}}_p$. In the case of the cube (Test problem 1), we restrict further the choice of $\mathbf{r}_i$ to only those points which have at least one Cartesian coordinate in the range of coordinates of Ω—between $-1/2$ and $1/2$ (no points in the corners of the currently permitted region). In the case of the sphere (Test problem 2), we restrict points $\mathbf{r}_i$ to lie within an interval of z values—between $z = \pm 0.5$—so that they are not near poles where data is more sparse.

The choice of N can increase the optimisation computational time if it is too large but needs to be large enough that a reasonable distribution of points is sampled. We found $N = 30$ to be enough for sufficiently smooth objective functions limiting numerical fluctuations in the solution due to the stochastic choice of $\mathbf{r}_i$ in relation to the data points $\bar{\mathbf{r}}_p$. Within their constrained regions, all of the $\mathbf{r}_i$ points are chosen with density proportional to the square of the distance from the centre of mass of the domain Ω. This was done so that, in 3D, points that are further away from a source (uniformly distributed) inside the domain Ω are not biased—or at least bias is minimised.

3.3.3 Choosing weights w_i

The problem posed in this manuscript is ill-posed. The practical reason is that for a given distribution $A(\mathbf{r})$ on $\partial\Omega$ generated by particular $\mathbf{r}_0$, κ_0 and λ_0, large deviations in $\mathbf{r}_0$ coupled with complementary changes in κ_0 and λ_0 can generate only small deviations in the observed $A(\mathbf{r})$. For example a deviation of $\mathbf{r}_0$ away from the surface $\partial\Omega$ may be, at least approximately, compensated for by a decrease in κ_0, an increase in λ_0, or combination thereof. There exists, therefore, a shallow valley in the parameter space for the objective function in (24). Without appropriate weights w_i, 'good'—in terms of fitness—but inaccurate solutions to (24) are likely to be found as the solution to (25). These incorrect solutions arise because of the error term ϵ_κ in (35). Our argument for choosing appropriate weights w_i is heuristic as the rigorous error analysis of this problem is prohibitively challenging. In Fig. 3, we present contour plots of the cost function landscapes. These landscapes are for Test problem 2 and assume the correct values for ϕ_0 and θ_0. The value of λ_0 is chosen for each pair of r_0 and κ_0 so that the cost function is minimised. For each of the figures $r_0 = 0.7$, $\theta_0 = \pi/4$, $\lambda_0 = 2$, $\kappa_0 = 1$, $N_z = 15$, $N_\theta = 100$, $N = 200$ and, for demonstrative purposes, we choose $\phi_0 = \pi/8$ which leads to problematic ill-posedness in the problem. Figure 3a presents the contours of the cost function where no weights are used ($w_i = 1$). The landscape is presented dependent on $\rho = |\boldsymbol{\rho}|$ and κ and represent the value of the objective function when the other parameters are optimised. It is important to note that whilst $\epsilon_\kappa/\bar{\mathcal{T}}_\kappa$ in (35) may or may not be small which allows for neglecting of high order terms, the perturbation to the objective function can lead to large errors in the optimal parameters. This is especially the case because $\epsilon_\kappa/\bar{\mathcal{T}}_\kappa$ depends on κ. It is important to note that as $\kappa \to \infty$, $\bar{\mathcal{T}}_\kappa \to 0$, and thus the optimisation (25) biases small values of κ along the valley in the cost function landscape. This can be clearly seen in Fig. 3a where it is clear that the optimal parameter set involves a value of r_0 which is too high and a value of $\kappa = 0$ (instead of 1). Assuming that second order error terms in (35) are much smaller than the first order terms, we have

$$\sum_{i=1}^{N} \tau_i w_i \left(\bar{f}(\mathbf{r}_i; \bar{A}, \kappa, \boldsymbol{\rho}) - \log(\lambda)\right)^2$$

$$= \sum_{i=1}^{N} \tau_i \left(\sqrt{w_i} \left(f(\mathbf{r}_i; A, \kappa, \boldsymbol{\rho}) - \log(\lambda)\right) + \frac{\sqrt{w_i}\epsilon_\kappa(\mathbf{r}_i; \bar{A})}{\bar{\mathcal{T}}_\kappa(\mathbf{r}_i; \bar{A})} \right)^2. \qquad (36)$$

One might propose that choosing $w_i = \bar{\mathcal{T}}_\kappa(\mathbf{r}_i)^2$ will help rectify this problem. Whilst it is clear that this may help regularise the problem in the neighbourhood of the true value, using this weight (see Fig. 3b) we note that a global minimum is achieved if $\kappa \to \infty$. This is because as κ gets large ϵ_κ vanishes and so does $\bar{\mathcal{T}}_\kappa$. That said, a local minimum can be seen near the true parameter values in Fig. 3b. We noticed numerically for Test problem 2 that the rate at which ϵ_κ vanishes as κ gets asymptotically large is at an exponential rate of $d = (1 - \rho)$. By multiplying the weights by an exponential term which grows at a rate d with respect to κ we can further improve the conditioning of this problem. We define $d(\boldsymbol{\rho})$ heuristically as the minimum distance between $\boldsymbol{\rho}$ and $\partial\Omega$ and choose the weights

$$w_i = \exp\left(2\kappa d(\boldsymbol{\rho})\right) \bar{\mathcal{T}}_\kappa(\mathbf{r}_i; \bar{A})^2. \qquad (37)$$

Whilst this choice of weights does not give perfect results and has not been rigorously derived, we observe that they result in global minimums in the truncated parameter space of the cost function which are not biased towards the extremes with respect to κ except in the special case where also $d(\boldsymbol{\rho})$ is small (see Fig. 3c). We shall investigate the accuracy of using (37) in (25) as a way of approximating the true solution (24) in Sect. 4.

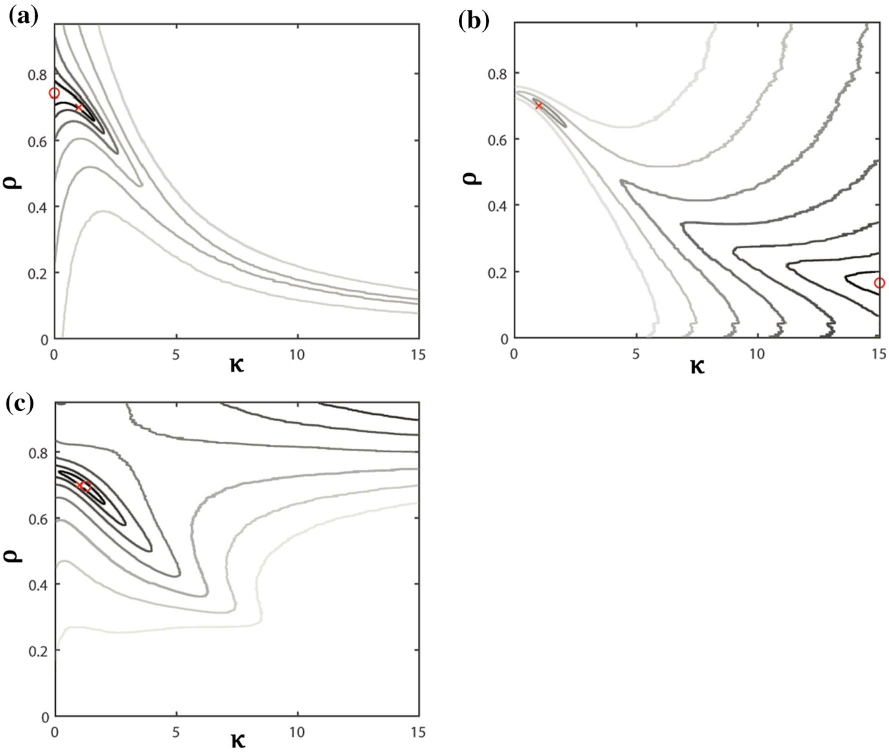

Fig. 3 Contours of the cost function for the optimisation problem (25) against ρ and κ for Test problem 2 with weights **a** $w_i = 1$, **b** $w_i = \bar{\mathcal{T}}_\kappa(\mathbf{r}_i)^2$, and **c** $w_i = \exp(2\kappa d(\boldsymbol{\rho}))\,\bar{\mathcal{T}}_\kappa(\mathbf{r}_i)^2$. In each subfigure, contours represent level curves of the cost function whose values are a geometric progression with ratio of 4. The brightness of each line indicates the relative magnitude of the level curve; small values of the cost function are indicated by darker lines. In each case presented in this figure, the true values of r_0 and κ_0 are indicated with a red 'X'. For comparison, the minimum of the cost function over ρ and κ in the truncated parameter space displayed in the figure is indicated with a red 'O'. In **a**, **b** we find errors in the computation of the objective function force the minimal computed value to have small and large κ values respectively, whilst in **c** the problem is regularised and the optimal parameters coincide with the true ones (color figure online)

4 Experimental results

Here we present experimental results which test the accuracy and robustness of the algorithm proposed in Sect. 3. To do this we require a measure of error. This is done for each of the parameters κ_0, λ_0 and $\mathbf{r}_0$. Using the notation $\{\bar{\kappa}_0, \bar{\lambda}_0, \bar{\mathbf{r}}_0\}$ to describe the output of the algorithm—the LHS of (25), we will present all errors as the normalised difference between true and estimated parameters

$$E_\kappa = \left|\frac{\bar{\kappa}_0 - \kappa_0}{\kappa_0}\right|, \tag{38}$$

$$E_\lambda = \left|\frac{\bar{\lambda}_0 - \lambda_0}{\lambda_0}\right|, \tag{39}$$

$$E_{\mathbf{r}} = ||\bar{\mathbf{r}}_0 - \mathbf{r}_0||, \tag{40}$$

Notice that in the case of $\mathbf{r}_0$, the error is not normalised. This is because $\mathbf{r}_0$ should be normalised with respect to its characteristic scale which is the spatial scale of Ω; in our case where the model has been nondimensionalised this is 1.

4.1 Test problem 1: = *C*

4.1.1 r_0-Dependent accuracy in fitted parameters

We focus our attention on the importance of $\mathbf{r}_0$ in relation to $\partial\Omega$ and distribution of $\mathbf{r}_p$ on the accuracy of the proposed fitting algorithm from Sect. 3. The parameters λ_0 and κ_0 are not very interesting in this regard. The parameter λ_0 is just a scaling constant whilst κ_0 dictates the decay rate in space and for the most part simply parametrises the effect of changing $\mathbf{r}_0$. For all the test cases in Sect. 4.1 $\kappa_0 = \lambda_0 = 2$ was used. Due to the regularity of the data points $\mathbf{r}_p$ for this test problem, there are a number of symmetries in Test problem 1 that may be exploited. To explore the full domain $\mathbf{r}_0 \in C$ we explore just the points in the subset $C' = \{(x_0, y_0, z_0)|0 < z_0 \leq y_0 \leq x_0 < 1/2\}$. All other points $\mathbf{r}_0 \in C$ give indistinguishable results to those for a particular point in C'. This can be demonstrated simply by relabelling the axes. Initially we present results for a fixed value of $x_0 = 0.35$. On the surface $x_0 = 0.35$, the region of interest forms a triangular region on a cube of length $2x_0$ concentric with the cube C as indicated in Fig. 4a. We choose an array of points $\mathbf{r}_0$ on this triangle (represented with red dots). We use $n_0 = 8$ points in both the y_0 and z_0 direction indicated in Fig. 4. For each point $\mathbf{r}_0$, together with $\kappa_0 = \lambda_0 = 2$, data A_p is generated on the cube surface ∂C using (7) on a regular grid of $N_s = 15$ points in each direction ($h = 1/15$ separation between grid points). The algorithm described in Sect. 3 is applied for each source point $\mathbf{r}_0$ with 10 reinitialisations of the optimisation process. The errors E_κ, E_λ and $E_{\mathbf{r}}$ as described by (38)–(40) are summarised in Fig. 4b–d respectively using a logarithmic (natural, base e) color map. There are three main points to take from these figures. The first is that reconstruction of parameters seems to be less robust closer to $\partial\Omega$ but also far away from $\partial\Omega$ (that is near the centre of the domain). We shall soon provide further validation of this observation, we observe that there is an optimal source position at an intermediate position between the centre and boundary of the domain for which estimating the parameters (especially κ_0 and λ_0) of the model becomes easier. The second observation is that the reconstruction of $\mathbf{r}_0$ is much more robust overall than either κ_0 or λ_0. The reason for this is that increases in both κ_0 and λ_0 can lead to very similar data with only small changes in r_0 and therefore it is difficult to be sure of the values of the (κ_0, λ_0) pair. The last observation is that stochasticity in both the optimisation algorithm and method by which $\mathbf{r}_i$ are chosen can produce noticeable stochasticity in the results, however it is clear that good results are obtained for all experiments (the worst error over the whole experiment was found in the predicted κ_0 of about 10% error).

To more explicitly explore the nature of the reconstructed parameters as a function of distance from ∂C, we define an approximation of the average errors E_κ, E_λ and $E_{\mathbf{r}}$

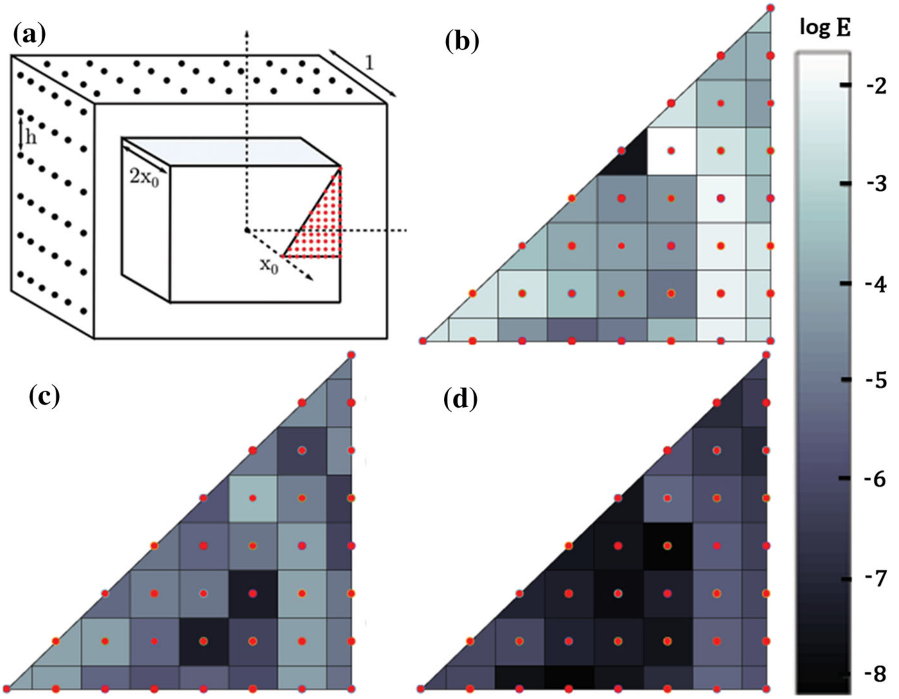

Fig. 4 A plot of errors **b** E_κ, **c** E_λ and **d** $E_\mathbf{r}$ for Test problem 1. The errors are calculated on data with $N_s = 15$, $\kappa_0 = \lambda_0 = 2$ and r_0 at the red point positions which lie on a cube with $x_0 = 0.35$ and $0 < z_0 \leq y_0 \leq x_0$ as indicated by the diagram in (**a**) (color figure online)

over all points r_0 which lie on the cube of length $2x_0$ centred at the origin. We define these average errors as a function of $0 < x_0 < 1/2$

$$\langle E_\theta \rangle (x_0) = \frac{2E_{\theta,r}(x_0) + \sum_v E_{\theta,v}(x_0) + 4\sum_s E_{\theta,s}(x_0) + 8\sum_m E_{\theta,m}(x_0)}{4(n_0-1)^2}, \quad (41)$$

where θ represents either κ, λ or $\mathbf{r}$ and $E_{\theta,i} = E_\theta$ evaluated at various classes of point r_0 in the triangle $0 \leq z_0 \leq y_0 \leq x_0$ that were displayed in Fig. 4a. The i in this context is either r, v, s or m which reference the points at the right angle, the other two vertices, the sides, and the middle/internal points respectively. The sums in (41) are the sums over each of these classifications of points r_0 respectively. The reason for the different coefficients in (41) is due to the multiplicity of these points and their representative domain on the full surface of the cube with side length $2x_0$. It will be important to note that in the presentation of all average results, we take the natural logarithm of the mean error $\langle E_\theta \rangle$ and also plot the interquartile range of the distribution of this error over the triangular region associated with a given x_0. Whilst the interquartile range takes into consideration some stochastic effects and its use usually implies noise, it is rather more accurately representative of the range of variation of the error over the cube surface. Presented in Fig. 5 is the average error (41) at different values of x_0 together with the

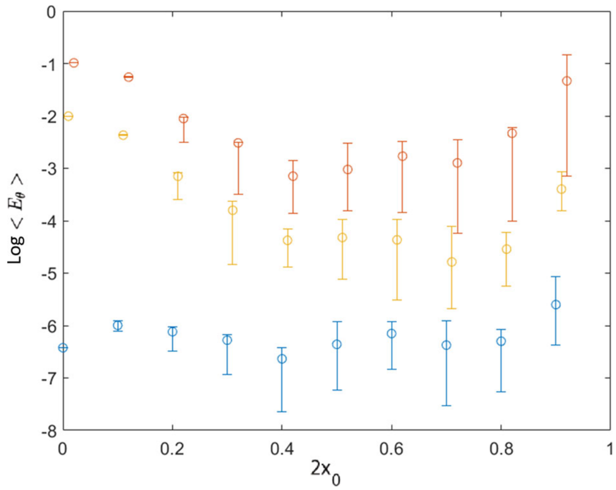

Fig. 5 A plot of the natural log of errors (red) $\langle E_\kappa \rangle$, (orange) $\langle E_\lambda \rangle$ and (blue) $\langle E_{\mathbf{r}} \rangle$ for Test problem 1 averaged over a cube concentric with C and side length $2x_0$ using (41)—see Fig. 4a for reference. For every set of data $N_s = 15$ and $\kappa_0 = \lambda_0 = 2$. The errorbars indicate the interquartile range of error over the cube surface. At each value of x_0 the data points for both $\langle E_\lambda \rangle$ and $\langle E_\kappa \rangle$ have been slightly offset horizontally so that the bars which indicate the interquartile range can be easily distinguished (color figure online)

interquartile range. It is clear that error is higher near the centre of the domain and near the edge of the domain $\partial\Omega$ and more robust in intermediate values of x_0. This is due to the fact that when the source is close to the surface $\partial\Omega$, A is a sharp spike which suffers from high errors during numerical integration whilst sources near the middle of the domain undergo the maximum amount of decay internally and therefore require larger perturbations to change A significantly.

4.1.2 Convergence of fitted parameters in the high resolution limit

To test the accuracy of the method outlined in Sect. 3 as the resolution of the data $\bar{A}$ is improved, we calculate the error (41) holding $2x_0 = 0.7$ and vary the resolution of the points $\mathbf{r}_p$ on ∂C. We use $N_s = \{5, 6, 8, 10, 13, 15, 20, 32\}$ points in each dimension on each surface of the cube to evaluate the input data $\bar{A}$. In Fig. 6, the log of errors associated with each of these resolutions is displayed as a function of $h = 1/N_s$, the spacing between data points. It is clear, given the logarithmic scale of $\langle E_\theta \rangle$ and the relatively linear trend in Fig. 6, that the convergence of the method to giving exact solutions has roughly a power law relationship with h. This is unsurprising as in the limit $h \to 0$, (25) becomes (24) and the results of Sect. 2.3 related to continuous data hold.

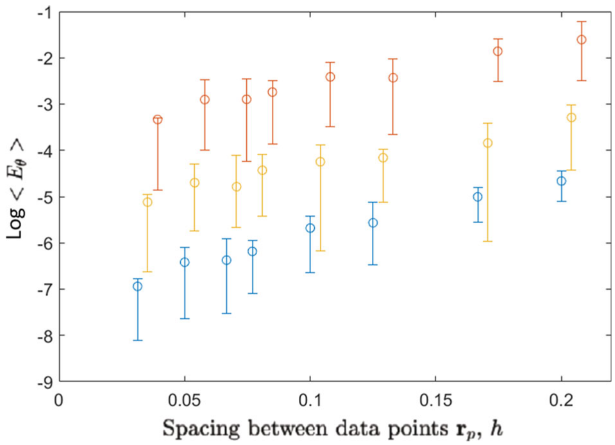

Fig. 6 A plot of the natural log of errors (red) $\langle E_\kappa \rangle$, (orange) $\langle E_\lambda \rangle$ and (blue) $\langle E_\mathbf{r} \rangle$ for Test problem 1 averaged over a cube concentric with C and side length $2x_0 = 0.7$ using (41)—see Fig. 4a for reference. For every set of data $\kappa_0 = \lambda_0 = 2$ and N_s is varied over the values {5, 6, 8, 10, 13, 15, 20, 32}. The errors are plotted against $h = 1/N_s$ so that they show data spacing rather than total amount of data. The errorbars indicate the interquartile range of error over the cube surface. At each value of h the data points for both $\langle E_\lambda \rangle$ and $\langle E_\kappa \rangle$ have been slightly offset horizontally so that the bars which indicate the interquartile range can be easily distinguished (color figure online)

4.2 Test problem 2: $= B$

4.2.1 r_0-Dependent accuracy in fitted parameters

As was done for Test problem 1, we focus first on the dependence of the source location within Ω on the accuracy of the fitting algorithm. We use the following default parameters in our parameter sensitivity analysis; $r_0 = 0.7, \theta_0 = 7\pi/5, \kappa_0 = 1, \lambda_0 = 2$, $N_z = 15$, and $N_\theta = 100$. We do not investigate variation in κ_0 or λ_0 explicitly here. The variation of λ_0 is uninteresting as it's effect is simply to change the scale of the morphogen concentrations U and A whereas varying κ_0 has a phenomenologically similar consequence to changing r_0. The default number of slices $N_z = 15$ was chosen to highlight errors when there is a significant sparsity in the data in the z-direction. Finally, θ_0 was chosen arbitrarily since the problem has rotational symmetry about the z-axis (assuming large N_θ and the accuracy does not depend on the value of θ_0.

In Fig. 7, the errors $E_\mathbf{r}$, E_κ and E_λ are presented as a function of ϕ_0 on the interval $(0, \pi/2)$ whilst holding $r_0 = 0.7$ constant. The error significantly depends on ϕ_0. Small values of ϕ_0—sources pointed towards the poles—result in data A which vary most rapidly near the poles where there is a severe lack of data given equally spaced slices of B. Without data near where A is most dynamic, the quadrature of the domain $\partial\Omega$ is inaccurate. In Fig. 7, we include best fit lines for each of the three errors to show that (visually) each of the errors improve roughly at the same rate, subject also to some stochasticity in the method, as ϕ_0 is changed to align with the x-y plane. This figure also reveals an oscillation of the error as it decreases from $\phi_0 = 0$ to $\phi_0 = \pi/2$. This effect is due to the relative position of ϕ_0 compared with the ϕ coordinates associated

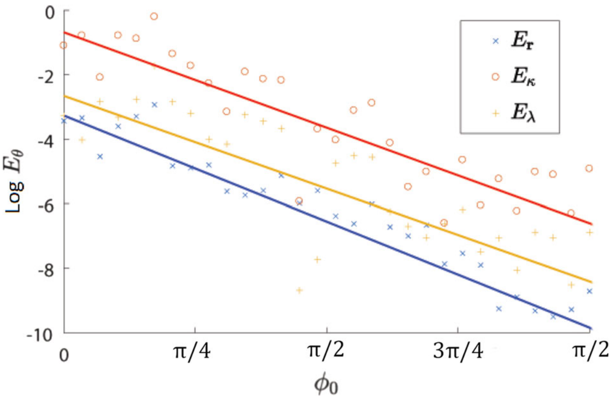

Fig. 7 Errors (red) E_κ, (orange) E_λ and (blue) $E_{\mathbf{r}}$ for Test problem 2 where $r_0 = 0.7$, $\theta_0 = 7\pi/5$, $\kappa_0 = 1$, $\lambda_0 = 2$, $N_z = 15$, and $N_\theta = 100$. Best fit lines are included for visualisation purposes and are colored the same as the algorithm output (color figure online)

with each slice and whether or not the peak of A lies on a slice or between slices. The sampling of A near its peak significantly affects the accuracy of determining $\bar{\mathcal{T}}_\kappa$. The values of ϕ_0 between $\pi/2$ and π have been omitted due to reflection symmetry with those values on the interval 0 to $\pi/2$.

We define the average errors over the concentric inner sphere of radius r_0 analogously to the concentric inner cube in Test problem 1.

$$\langle E_\theta \rangle(r_0) = \overline{\int_0^1 E_\theta(r_0, \phi_0)\, \mathrm{d}(\cos(\phi_0))}, \tag{42}$$

$$= \sum_{i=2}^{N_{\phi_0}} \frac{E_\theta(r_0, \phi_{0,i-1}) + E_\theta(r_0, \phi_{0,i})}{2} \left(\cos(\phi_{0,i-1}) - \cos(\phi_{0,i})\right), \tag{43}$$

where, in this specific context, the subscript θ represents any of the three parameters κ, λ or $\mathbf{r}$ (not to be confused with θ, the coordinate of $\mathbf{r}$). The value N_{ϕ_0} is the number of errors measured as a function of ϕ_0 (for Fig. 7 and all graphs herein $N_{\phi_0} = 30$) and the ith value of ϕ_0 (monotonically increasing from 0 to $\pi/2$) is $\phi_{0,i}$. In (43) we use a trapezoidal method to evaluate the integral (42). The interquartile range which is represented in plots of average error is calculated over the sphere rather than just as a distribution over ϕ to give a true measure of the interquartile range over the domain surface $\partial\Omega$.

In Fig. 8, a plot of the natural log of $\langle E_\theta \rangle$ is displayed against the radius r_0. As with Test problem 1 (Fig. 5), the accuracy of the algorithm seems to be greatest not at the surface $\partial\Omega$ nor at the centre of the domain Ω but in some region in between. Reasons for this behaviour have already been discussed. That being said, accuracy in cases where the parameters r_0 (and by association κ_0) are not extreme tend to give accurate results even when only 15 slices of data A are used.

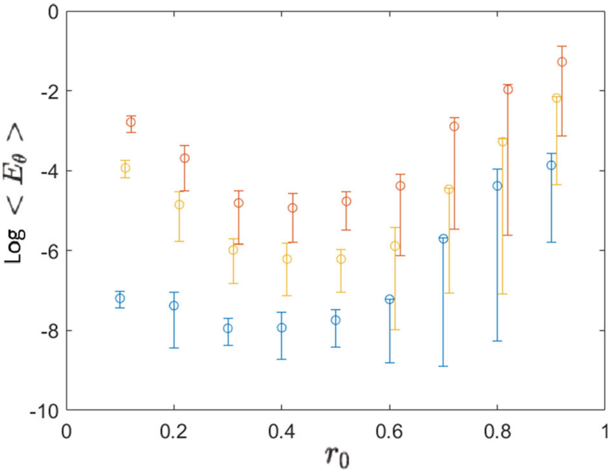

Fig. 8 A plot of the natural log of errors (red) $\langle E_\kappa \rangle$, (orange) $\langle E_\lambda \rangle$ and (blue) $\langle E_{\mathbf{r}} \rangle$ for Test problem 2, averaged over a sphere concentric with B and radius r_0 using (43). For every set of data $\kappa_0 = 1$, $\lambda_0 = 2$ and $N_z = 15$. The errorbars represent the interquartile range of the error over the sphere. At each value of r_0 the data points for both $\langle E_\lambda \rangle$ and $\langle E_\kappa \rangle$ have been slightly offset horizontally so that the bars which indicate interquartile range can be easily distinguished (color figure online)

4.2.2 Convergence of fitted parameters in the high resolution limit

To validate that the algorithm for Test problem 2 results in more accurate results if given more data (slices) we display in Fig. 9 the natural log of the average error as a function of number of slices. Unsurprisingly, it is clear that increasing the number of slices by decreasing the slice thickness gives results which seemingly converge on the true solution.

4.2.3 Robustness in the presence of noisy data

Being an ill-posed fitting/inverse problem, it is possible that the algorithm might suffer from large inaccuracies in the presence of noise. Since we are proposing that this method is likely to be useful in biomedical applications, it is important to establish that the results are robust in the presence of data with noise. To simulate noisy input data, instead of input data $\bar{A}_p$ at each point $\mathbf{r}_p$ we replace with a log-normally distributed value $\tilde{A}_p$ with mean $\mu = \bar{A}_p$ and variance $\sigma^2 = (\bar{A}_p \omega)^2$. That is, ω is the noise σ to signal $\bar{A}_p$ ratio at each point $\mathbf{r}_p$. We shall test the loss in accuracy of our algorithm applying it to $\tilde{A}_p$ as though it were $\bar{A}_p$. We use the default parameters used for Fig. 7 and, in Fig. 10, we present the natural log of the average error $\langle E_\theta \rangle$ for noise to signal ratios ω between 0 and 1. The most striking behaviour that can be observed here is that the algorithm suffers in accuracy for the introduction of a small amount of noise but as this noise is increased the marginal loss of accuracy is reduced. This robustness to noisy data stems from the way that the method described in this manuscript integrates over the data. Each node $\mathbf{r}_p$ is integrated first with respect to azimuthal angle θ in the

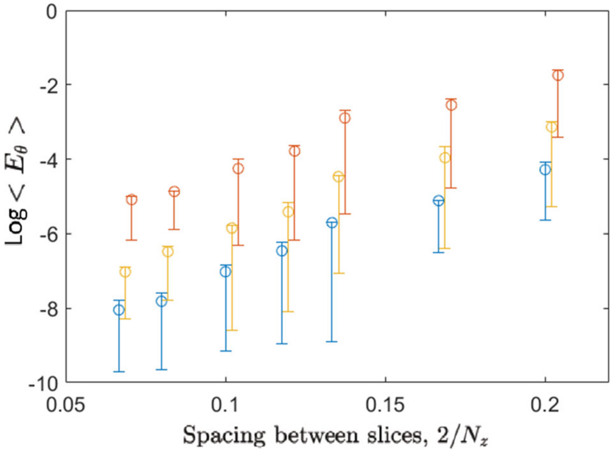

Fig. 9 A plot of the natural log of errors (red) $\langle E_\kappa \rangle$, (orange) $\langle E_\lambda \rangle$ and (blue) $\langle E_{\mathbf{r}} \rangle$ for Test problem 2 averaged over a sphere concentric with B and radius $r_0 = 0.7$ using (43). For every set of data $\kappa_0 = 1$ and $\lambda_0 = 2$. The error is plotted as a function of slice separation/thickness $2/N_z$ for $N_z = \{10, 12, 15, 17, 20, 25, 30\}$. The errorbar width represents the interquartile range of error over the sphere. At each value of r_0 the data points for both $\langle E_\lambda \rangle$ and $\langle E_\kappa \rangle$ have been slightly offset horizontally so that the bars which indicate interquartile range can be easily distinguished (color figure online)

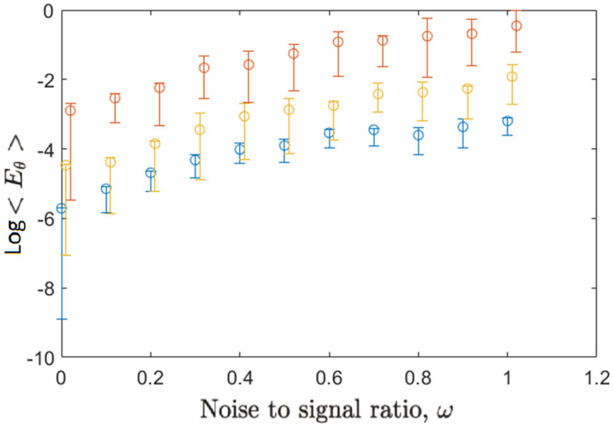

Fig. 10 A plot of natural log of errors (red) $\langle E_\kappa \rangle$, (orange) $\langle E_\lambda \rangle$ and (blue) $\langle E_{\mathbf{r}} \rangle$ for Test problem 2 averaged over a sphere concentric with B and radius $r_0 = 0.7$ using (43). For every set of data $\kappa_0 = 1$, $\lambda_0 = 2$, $N_z = 15$, and $N_\theta = 100$. The error is plotted as a function of noise to signal ratio ω as defined in Sect. 4.2.3. The errorbar width represents the interquartile range of the data over the sphere. At each value of r_0 the data points for both $\langle E_\lambda \rangle$ and $\langle E_\kappa \rangle$ have been slightly offset horizontally so that the bars which interquartile range can be easily distinguished (color figure online)

calculation of $\bar{\mathcal{T}}_\kappa$ (32). Since there are many data points on each slice, integrating over θ smooths a lot of the noise.

5 Conclusions

In this manuscript we present a novel method for finding a morphogen point source in some three-dimensional domain using data describing the flux of the morphogen on the domain boundary. The morphogen is produced, diffuses and decays at unknown rates which are determined, relative to the diffusion constant, by the method. We have focused on two main test problems in which the domain is (1) the unit cube and (2) the unit sphere. These test problems were chosen because they offer exact analytical forward solutions which can be used to generate data with known solutions. The biggest advantage to using this method is that, unlike many inverse problems or fitting techniques, it does not require a single forward solution to be computed. The cost that is introduced for this property is that quadrature of the data on the domain boundary is necessary and should be accurate and this can sometimes be difficult. In Test problem 2 (the unit sphere) we consider the data only on parallel and equidistant slices, inspired by the types of data which are experimentally feasible.

Our method focuses on converting the problem on the bounded domain to that of the infinite domain and exploiting the analytical nature of the fundamental solution. One of the biggest challenges that is presented by this method is that small integration error can lead to large errors in the optimal set of parameters. This is due to the ill-conditioning of the problem. We mitigate against this source of inaccuracy by regularisation which takes a heuristic form; a careful consideration of the error term, especially in the case of large decay rate.

The results are accurate, fast, and robust. Whilst the accuracy of the method depends heavily on the density of data and the position of the source, we found for even conservative conditions that our method gave good results. The recovery of the position of the source was the most accurate, followed by source production rate and the decay rate—the accuracies of which are separated by a number of orders of magnitude. We found that the best accuracy is obtained for dense data, small decay rates and source points which are close to the data on the boundary (for example, not at the poles for Test problem 2) but not too close (somewhere intermediate to the centre of the domain the and the boundary).

This result is a test study for a more general challenge of source reconstruction. In the more general case, the source will not be a single location but a distributed function over the domain. In the case of the linear Helmholtz model it may be possible to extend the method presented here but instead of (19) which describes the relationship between the integrated data $\mathcal{T}_\kappa$ and the fundamental solution G_κ it would be expected that if $\kappa = \kappa_0$ and $\mathbf{r} \notin \Omega$ (19) becomes

$$\mathcal{T}_\kappa(\mathbf{r}; a) = \iiint_\Omega G_\kappa(\mathbf{r}; \mathbf{r}_0)\sigma(\mathbf{r}_0)\,\mathrm{d}V_0, \tag{44}$$

where the integral is taken over coordinates of $\mathbf{r}_0$ and σ is the distribution of the unknown source. We shall leave this for a future work.

References

Alahmadi AA, Flegg JA, Cochrane DG, Drovandi CC, Keith JM (2020) A comparison of approximate versus exact techniques for Bayesian parameter inference in non-linear ODE models. Royal Society Open Science. Accepted 27 Jan 2020

Bocher P, De Meyer H, Vanden Berghe G (1994) On Gregory- and modified Gregory-type corrections to Newton–Cotes quadrature. J Comput Appl Math 50(1–3):145–158. https://doi.org/10.1016/0377-0427(94)90296-8

Brass H, Petras K (2011) Quadrature theory: the theory of numerical integration on a compact interval. American Mathematical Society, Providence

Camley BA (2018) Collective gradient sensing and chemotaxis: modeling and recent developments. J Phys Condens Matter 30(22):223001. https://doi.org/10.1088/1361-648X/aabd9f

Dehapiot B, Carriere V, Carroll J, Halet G (2013) Polarized Cdc42 activation promotes polar body protrusion and asymmeteric division in mouse oocytes. Dev Biol 377(1):202–212. https://doi.org/10.1016/j.ydbio.2013.01.029

Engl HW, Hanke M, Neubauer A (1996) Regularization of inverse problems, mathematics and its applications, vol 375. Kluwer, Dordrecht

Fasino D, Inglese G (1999) An inverse Robin problem for Laplace's equation: theoretical results and numerical methods. Inverse Probl 15:41–48. https://doi.org/10.1088/0266-5611/15/1/008

Gurdon JB, Bourillot P-Y (2001) Morphogen gradient interpretation. Nature 413:797–803. https://doi.org/10.1038/35101500

Hadamard J (1923) Lectures on Cauchy problem in linear partial differential equations. Yale University Press, New Haven

Hansen N (2009) Benchmarking a bi-population CMA-ES on the BBOB-2009 Function Testbed. In: GECCO '09, pp 2389–2396. https://doi.org/10.1145/1570256.1570333

Hansen N, Ostermeier A (2001) Completely derandomized self-adaptation in evolution strategies. Evol Comput 9(2):159–195. https://doi.org/10.1162/106365601750190398

Hansen N, Auger A, Ros R, Finck S, Pošík P (2010) Comparing results of 31 algorithms from the Black-Box optimization benchmarking BBOB-2009. In: GECCO '11, pp 1689–1696. https://doi.org/10.1145/1830761.1830790

Hildebrand FB (1987) Introduction to numerical analysis, 2nd edn. Dover, New York

Howard M (2012) How to build a robust intracellular concentration gradient. Trends Cell Biol 22(6):311–317. https://doi.org/10.1016/j.tcb.2012.03.002

Javed M, Trefethen LN (2015) Euler–Maclaurin and Gregory interpolants. Numer Math 132:201–216. https://doi.org/10.1007/s00211-015-0713-x

Karageorghis A, Lesnic D, Marin L (2010) A survey of applications of the MFS to inverse problems. Inverse Probl Sci Eng 19(3):309–336. https://doi.org/10.1080/17415977.2011.551830

Marin L, Karageorghis A (2009) Regularized MFS-based boundary identification in two-dimensional Helmholtz-type equations. CMC Comput Mater Contin 10:259–293. https://doi.org/10.3970/cmc.2009.010.259

Paris RB (1984) An inequality for the Bessel function $j_\nu(\nu x)$. SIAM J Math Anal 15(1):203–205. https://doi.org/10.1137/0515016?journalCode=sjmaah

Reeger JA, Fornberg B, Watts ML (2016) Numerical quadrature over smooth, closed surfaces. J R Soc A 472:20160401. https://doi.org/10.1098/rspa.2016.0401

Remorino A et al (2017) Gradients of Rac1 nanoclusters support spatial patterns of Rac1 signalling. Cell Rep 21(7):1922–1935. https://doi.org/10.1016/j.celrep.2017.10.069

Stuart AM (2010) Inverse problems: a Bayesian perspective. Acta Numer 19:451–559. https://doi.org/10.1017/S0962492910000061

Tabata T, Takei Y (2004) Morphogens, their identification and regulation. Development 131:703–712. https://doi.org/10.1242/dev.01043

Tilman GD (1984) Plant dominance along an experimental nutrient gradient. Ecology 65(5):1445–1453. https://doi.org/10.2307/1939125

Toni T, Welch D, Strelkowa N, Ipsen A, Stumpf MPH (2009) Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. J R Soc Interface 6(31):187–202

Wartlick O, Kicheva A, Gonzalez-Gaitan M (2009) Morphogen gradient formation. Cold Spring Harb Perspect Biol 1(3):a001255. https://doi.org/10.1101/cshperspect.a001255

Yan L, Fu CL, Yang FL (2008) The method of fundamental solutions for the inverse heat source problem. Eng Anal Bound Elem 32:216–222. https://doi.org/10.1016/j.enganabound.2007.08.002

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.