Computational and Applied Mathematics (2021) 40:146
https://doi.org/10.1007/s40314-021-01531-5

Optimization algorithms for parameter identification in parabolic partial differential equations

Karl Glasner[1]

Received: 25 February 2021 / Revised: 6 May 2021 / Accepted: 8 May 2021 /
Published online: 20 May 2021

Abstract
The problem of estimating model parameters from data representing near-equilibrium patterns in PDEs is considered. This problem is formulated as an optimization problem by determining the nearest state on a manifold of equilibria. Algorithms to solve this optimization problem are proposed, by first regularizing the problem and using explicit search directions on the tangent space of the equilibrium manifold. Some rigorous results on local converge are obtained. Several examples of pattern forming systems are used to test the proposed methodology. Comparisons to synthetic data are made showing the ability of obtaining excellent estimates even when significant noise is present.

Keywords Parameter estimation · PDE-constrained optimization · Pattern formation

Mathematics Subject Classification 65K10 · 35Q93

1 Introduction

Physical models are often equipped with parameters and other constitutive details which cannot be easily related to first principles. With the abundance of experimental data and recent advances in machine learning, it is natural to seek methods which incorporate real world information into physically inspired models by fitting or estimating system parameters.

In dynamic problems, most approaches to parameter estimation utilize data in the form of a time series. In the context of parabolic partial differential equations, however, only a single data point in time may be available, representing a spatially inhomogeneous signature of the system's behavior. Here it is imagined that this takes the form of a noisy perturbation of either a true steady state or a long-lived transient. This is a particularly common situation in physical models of pattern formation Cross and Hohenberg (1993) and materials science Chen (2002), where observational timescales may be much slower than the dynamics that

Communicated by Gabriel Haeser.

✉ Karl Glasner
kglasner@math.arizona.edu

[1] Department of Mathematics, University of Arizona, Tucson, AZ 85721, USA

forms spatial inhomogeneity. This work focusses on this case, where the input used for parameter estimation takes the form of a single time-independent state.

Early approaches to parameter identification in partial differential equations considered well-posed inverse problems from which analytically exact solutions can be found (e.g. Friedman and Reitich (1992); Knowles (2001)). More typically, the problem of parameter identification is regularized by formulating it as an optimization problem. The most common approach is to determine parameter values which minimize the difference between the model output and the provided data Fullana et al. (1997); Ackleh et al. (1998); Ashyraliyev et al. (2008); Garvie et al. (2010); Jin and Maass (2012); Croft et al. (2015); Sgura et al. (2019); Zhao et al. (2020). This is implemented using techniques of PDE-constrained optimization Hinze et al. (2008), sometimes in conjunction with other qualitative assessments Sgura et al. (2019).

Sparse regression methods comprise a second class of optimization methods Rudy et al. (2017); Schaeffer (2017); Rudy et al. (2019); Wang et al. (2019); Maddu et al. (2019), which involve minimizing the residual error (the discrepancy in the equation itself evaluated using the data). These approaches include a sparsity-inducing penalty term which allows for identification of minimal (parsimonious) models drawn from a large library of possibilities. This methodology is described in the context of data which has both time and space dependence, but can be adapted to the type of input data considered here. A direct comparison to residual minimization parameter estimation is made here, albeit without the sparsity inducing penalization.

Numerous other approaches to data-informed inverse problems in PDEs have been explored. Bayesian inference has been employed in the PDE setting to estimate parameters Dewar et al. (2010). Alternatively, Gaussian process regression has been proposed Raissi et al. (2017); Maziar and George (2018) as a way to naturally accommodate noisy data. Lastly, machine-learning strategies using neural networks are also being studied for identification problems in PDEs Long et al. (2019); Raissi et al. (2017a, b).

This paper is organized as follows. Section 2 reviews and compares two optimization formulations for parameter identification in the context of stationary input data. In Sect. 2.1, a compromise between these two formulations is proposed. Iterative methods for solving the resulting constrained optimization problems are proposed in Sects. 2.2 and 2.3. Local convergence of these methods is demonstrated in Sect. 3. Examples of pattern-forming partial differential equations are discussed in Sects. 4–6. In these applications, synthetic data constructed from numerical simulations with added noise is used to test the performance of the proposed methods.

2 Formulations of parameter estimation

The underlying model which is considered here is a nonlinear evolution equation on some vector space U of the form

$$u_t = N(u; \mu), \tag{1}$$

with $u(t) : \mathbb{R} \to U$. The model parameters constitute $\mu \in P \subset \mathbb{R}^p$, with P being a prescribed set of allowed values. Although the focus of this work is when (1) represents a partial differential equation, in practice, spatial discretization renders U finite dimensional. Throughout this work, the nonlinear operator $N(\cdot)$ is assumed to be sufficiently differentiable with respect to its arguments.

The dynamics in (1) is technically unnecessary here, as we are really interested in the equilibrium set $\mathcal{M}_0$, defined as

$$\mathcal{M}_0 = \{u | N(u; \mu) = 0,\ \mu \in P\}. \tag{2}$$

Typically, $\mathcal{M}_0$ is a collection of manifolds whose dimensions are $\geq p$. In some models, it is certainly possible that multiple parameter values produce the same equilibrium; for example in $u_t = \mu f(u)$ clearly μ cannot be inferred from equilibria at all. To make sense of the parameter identification problem in our context, it will be assumed that for any $u \in \mathcal{M}_0$, there is a unique parameter $\mu^*(u)$ with $N(u, \mu^*(u)) = 0$.

Dynamics only enter the algorithm described below in the sense of a prescribed map $\mathcal{E}^0(u; \mu) : U \to \mathcal{M}_0$, for which the preimage of $u^* = \mathcal{E}^0(u; \mu)$ is regarded as a "basin of attraction" for u^*. For the dynamics (1), this arises naturally by letting $u(0) = u_0$ and defining

$$\mathcal{E}^0(u_0; \mu) = \lim_{t \to \infty} u(t). \tag{3}$$

This only makes sense if the dynamics are "relaxational" in nature, rather than oscillatory of chaotic. Nevertheless, (3) is well defined in a great number of applications, in particular gradient flows and more generally, systems possessing a Lyapunov functional. The method described below does not compute E_0 directly, but rather uses a regularized version (see 8).

The inverse problem for parameter identification arises by supplying "target" data $u_T \in U$, which is assumed to be at least somewhat representative of a member of the set $\mathcal{M}_0$. If it happens that $u_T \in \mathcal{M}_0$, it would be natural to identify the parameters as those associated with a true equilibrium, i.e. $\mu = \mu^*(u_T)$. In general, we expect the supplied target data to represent either a steady state or a a long-lived transient, possibly contaminated with noise from either measurement error or effects not captured by the model. In these cases, the most to hope for is to choose parameters to be optimal in some sense.

Optimization problems for parameter identification have historically been formulated in two distinct ways. One methodology looks to minimize distance between the model output (solution) and supplied data Ackleh et al. (1998); Ashyraliyev et al. (2008); Garvie et al. (2010); Croft et al. (2015); Sgura et al. (2019). In the present context, this amounts to finding $u \in \mathcal{M}_0$ which is as close as possible to the target data, so that the parameter estimate is prescribed by

$$\mu = \mu^*(u^*), \quad u^* = \arg\min_{u \in \mathcal{M}_0} ||u - u_T||, \quad \text{(Minimum distance formulation)}, \tag{4}$$

where the norm $||\cdot||$ is prescribed. The discussion here only supposes that $||\cdot||$ is induced by an inner product $\langle \cdot, \cdot \rangle$, although the later examples specialize to the L^2 norm.

A second approach is to minimize the norm of the residual, that is,

$$\mu = \arg\min_{\mu \in P} ||N(u_T; \mu)||, \quad \text{(Minimum residual formulation)}, \tag{5}$$

where again the norm is prescribed. In the case where N has terms linear in parameter components μ_i,

$$N(u; \mu) = N_0(u) + \sum_{i=1}^{p} \mu_i N_i(u), \tag{6}$$

then (5) is a standard least-squares problem. By including in (5) a term which penalizes the number of non-zero parameter components, the resulting problem constitutes the time-independent version of the sparse regression method Rudy et al. (2017); Schaeffer (2017).

While the examples below happen to be of the form (6), the algorithm we describe does not depend on this, and allows the parameter dependence to be completely general.

Note that both formulations (4) and (5) may produce the same result if it happens that u_T is an element of $\mathcal{M}_0$. For target data near the equilibrium set $\mathcal{M}_0$, both formulations might also produce comparable results, since $||N(u_T;\mu)||$ is generally smallest when $\mu \approx \mu^*(u^*)$, where u^* is defined in (4).

There are advantages and drawbacks to each formulation. The optimization problem (4) is highly nonlinear, and often possesses a vast number of local minimizers which makes finding a global minimizer difficult. Prior studies (e.g. Croft et al. (2015)) utilize the Levenberg–Marquardt algorithm to accomplish the minimization, but it is not clear that this would be efficient in all cases, or always converge to the correct minimizer. On the other hand, our results indicate that adding significant amounts of noise to an otherwise exact steady state does not significantly alter the minimizer in (4).

The minimum residual problem (5), especially if it is of linear least squares form, is easily computed, and appears to provide reliable parameter estimates for low-noise target data. Computing spatial derivatives in $N(u_T,\mu)$ presents a challenge, however, and readily amplifies any noise in the the target data. We find that the estimate is easily corrupted even in cases where noise is very moderate. It has been suggested Schaeffer (2017) that filtering or smoothing the data to begin with may improve this situation; in the example of Sect. 4, this idea is tested and compared to the present formulation.

2.1 A regularized approach

To mollify the disadvantages inherent in (4) and (5), consider a modified of (1),

$$u_t = N(u;\mu) + C(u_T - u), \tag{7}$$

which includes a "fidelity" term $C(u_T - u)$, where C will be called the interpolation parameter, and satisfies $0 \le C < \infty$. Analogous to (2), the equilibrium set may be defined as

$$\mathcal{M}_C = \{u | N(u;\mu) + C(u_T - u) = 0,\ \mu \in P\}. \tag{8}$$

Associated with this set is a mapping $\mathcal{E}^C(u_0;\mu) : U \to \mathcal{M}_C$, defined analogous to (3) using the flow (7) instead. This is computed in practice by evolving (7) up to a large prescribed time, using u_0 as the initial data. Of course, this is only practical if good numerical methods are available; we describe in detail below semi-implicit, time-adaptive spectral methods which generally require very few timesteps.

Letting $\mu^*(u)$ define the mapping from elements in $\mathcal{M}_C$ to corresponding parameter values, a constrained optimization problem analogous to (9) can be formulated:

$$\mu^C = \mu^*(u^C), \quad u^C = \arg\min_{u\in\mathcal{M}_C} ||u - u_T||. \tag{9}$$

Note that if $u_T \in \mathcal{M}_0$, then u_T is also in every set $\mathcal{M}_C$, and moreover $\mu^C = \mu^0$, so solving (9) is the same as solving (4). More generally, the optimization problem (9) can be viewed as an interpolation between the minimum distance and minimum residual formulations. Indeed, as $C \to 0$, we will show that under a broad set of circumstances, minimizers to (9) converge to minimizers of (4). Conversely for large C, the set $\mathcal{M}_C$ is asymptotically described by the solutions to

$$N(u_T;\mu) + C(u_T - u) = 0. \tag{10}$$

In this case, minimizing $||u - u_T||$ over $\mathcal{M}_C$ is the same as minimizing $||N(u_T;\mu)||$ over μ.

We find there is a trade-off in the choice of interpolation parameter C, reflective of the limiting cases $C = 0$ and $C = \infty$. If C is chosen to be very small, then the difficulties of finding a true global minimum in (4) are still present, whereas as C is increased the fidelity term has a convexifying effect, and fewer local minima might be expected (see Fig. 2). On the other hand, if C is too large then the quality of the parameter estimate is often poor, as noise in the target data is not smoothed enough by the flow (9).

The optimum choice of C is entirely problem dependent. For the examples below, the initial choice was partly made by trial and error: if C was too large, the parameter estimate was sometimes well outside the allowed parameter set P, whereas if C was initially too small, the minimization algorithm proposed in the next section struggles to find a reasonable local minimum. The former issue is dealt with in Sect. 2.3 by systematically decreasing C to improve the estimate.

2.2 The projection-relaxation algorithm

A variety of options exist for solving the constrained minimization problem (9), including constrained gradient descent, Levenberg–Marquardt methods Ashyraliyev et al. (2008) and Newton-type methods Goodman (1985); Bonnans et al. (2006). Here we describe a version of the latter, specialized to our situation where the objective function is quadratic. The constraint is satisfied exactly at each step, and the optimization is done on the tangent space of $\mathcal{M}_C$. Since this is low dimensional, the associated Hessian is easy to invert.

The iterative algorithm requires an initial state $u^{(0)} \in \mathcal{M}_C$ and parameter values $\mu^{(0)}$ to be given. The former is computed by letting $u^{(0)} = \mathcal{E}^C(u_{guess}; \mu^{(0)})$, where $u_{guess} \in U$ is provided; the choice $u_{guess} = u_T$ appears to be reliable. Further iterates $u^{(j)}$, $\mu^{(0)}$ will be constructed so as to guarantee that $||u^{(j)} - u_T||$ will be a strictly decreasing sequence in j.

If $\mathcal{M}_C$ is roughly flat near some iterate $u^{(j)}$, then the closest point on $\mathcal{M}_C$ to u_T is well-approximated by the projection of $u_T - u^{(j)}$ onto the tangent space at $u^{(j)}$. Provided the parameter set near $\mu^{(j)}$ provides a local coordinate system for $\mathcal{M}_C$, then the tangent space may be identified by the basis elements

$$v_i = \frac{\partial u^*}{\partial \mu_i}(\mu^{(j)}), \quad i = 1, \ldots, p. \tag{11}$$

Here $u(\mu)$ is the local $1-1$ mapping from parameters to elements of $\mathcal{M}_C$. This can be computed from the defining equation $N(u(\mu); \mu) + C(u_T - u(\mu)) = 0$ by differentiation, giving the linear systems

$$\left(\nabla_u N(u^{(j)}, \mu^{(j)}) - C\right) v_i = -\frac{\partial N}{\partial \mu_i}(u^{(j)}, \mu^{(j)}), \quad i = 1, \ldots, p. \tag{12}$$

For parabolic PDEs, this represents a set of linear elliptic equations. This could problematic numerically if the system is ill-conditioned, for example. Empirically, however, this does not appear to be the case provided C is not too small. Methods for efficient inversion of this system are described in the examples.

Projection of u_T onto the tangent space can now be accomplished by finding

$$\min_{\tilde{\mu}} \left\| u^{(j)} + \sum_{i=1}^{p} \tilde{\mu}_i v_i - u_T \right\|, \tag{13}$$

where $\tilde{\mu}$ is the parameter increment. This least squares problem is equivalent to the linear problem $A\tilde{\mu} = b$ where

$$A_{ik} = \langle v_i, v_k \rangle, \quad b_k = \langle u_T - u^{(j)}, v_k \rangle. \tag{14}$$

After projection, the new parameter estimate and state are

$$\mu^{(j+1)} = \mu^{(j)} + \tilde{\mu}, \quad u_p = u^{(j)} + \sum_{i=1}^{p} \tilde{\mu}_i v_i; \tag{15}$$

however, u_p is generally not in $\mathcal{M}_C$. This can be accomplished by a relaxation step, where the new iterate is determined by $u^{(j+1)} = \mathcal{E}^C(u_p, \mu^{(j+1)})$ (see Fig. 1).

If the manifold $\mathcal{M}_C$ is highly curved, there is no guarantee that $u^{(j+1)}$ is closer to u_T than the previous state $u^{(j)}$, or that the new parameters $\mu^{(j+1)}$ are in the allowed set P. Moreover, the iterative algorithm may have stability problems when using the full size parameter increment $\tilde{\mu}$ in (15). To avoid these issues, step-limiting criteria are employed which reduce the magnitude of $\tilde{\mu}$. The first is to check that both the objective function $||u^{(j+1)} - u_T||$ is smaller than $||u^{(j)} - u_T||$, and $\mu^{(j+1)} \in P$. If not, the parameter increment is reduced (by a factor of two, for example), and the the relaxation step is repeated. Stability of the iteration can also depend on $\tilde{\mu}$ not being too large; to this end (15) is replaced with

$$\mu_p = \mu^{(j)} + \alpha\tilde{\mu}, \quad u_p = u^{(j)} + \alpha \sum_{i=1}^{p} \tilde{\mu}_i v_i. \tag{16}$$

where

$$\alpha = \min(1, C_\alpha ||u_T - u^{(j)}||). \tag{17}$$

The parameter $C_\alpha > 0$ is system dependent and can be determined by an explicit stability calculation (see Sect. 3.1). In the examples below, no stability problems where encountered by simply setting α to unity. The complete method is summarized in algorithm 1.

Algorithm 1: Projection-relaxation (PR) optimization

Given: u_{guess}, $\mu^{(0)}$, u_T
$u^{(0)} = \mathcal{E}^C(u_{guess}; \mu^{(0)})$
repeat
  Compute tangent space basis elements by solving (12) for $i = 1, \ldots, p$
  Construct matrix $A_{ik} = \langle v_i, v_i \rangle$ and vector $b_k = \langle u_T - u^{(j)}, v_k \rangle$
  Solve $A\tilde{\mu} = b$
  repeat
    $\mu^{(j+1)} := \mu^{(j)} + \alpha\tilde{\mu}$
    $u_p := u^{(j)} + \alpha \sum_{i=1}^{p} \tilde{\mu}_i v_i$
    $u^{(j+1)} := \mathcal{E}^C(u_p; \mu^{(j+1)})$
    $\tilde{\mu} := \tilde{\mu}/2$
  until $||u^{(j+1)} - u_T|| < ||u^{(j)} - u_T||$ *and* $\mu^{(j+1)} \in P$
until $|\tilde{\mu}| < \tilde{\mu}_{min}$

2.3 The regularized minimum distance (RMD) algorithm

In some problems, it might be desirable to locate a solution to the minimum distance problem (4), rather then the regularized version where $C > 0$. Rather than attempting to locate the

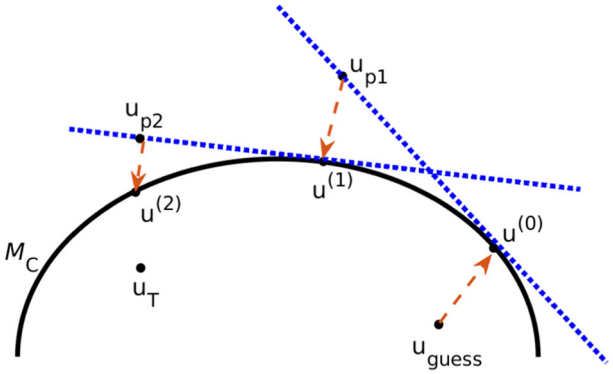

Fig. 1 The first two steps of the projection-relaxation method. The provided initial state u_{guess} (typically just u_T) is mapped via $\mathcal{E}^C()$ (shown as red/dashed arrows) to $u^{(0)} \in \mathcal{M}_C$, and then the nearest point to the target data on the tangent space (blue/dotted) is found (u_{p1}). The new iterate $u^{(1)} \in \mathcal{M}_C$ is determined again by the mapping $\mathcal{E}^C()$, and then the process is repeated

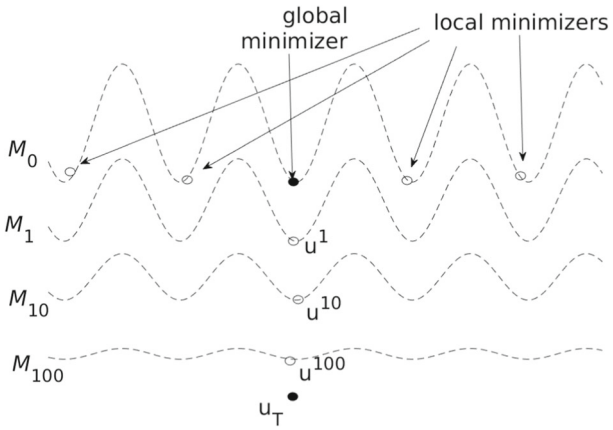

Fig. 2 A schematic view of the RMD algorithm. If the true equilibrium set $\mathcal{M}_0$ is highly convoluted, the optimization problem possesses numerous local minima. The algorithm provides a sequence of minimizers to the regularized problem (here values of C were arbitrarily chosen to be 100, 10, 1) which ideally converge to the global minimizer

global minimum in (4) by setting $C = 0$ immediately, we can use C as a continuation parameter, and progressively solve (9) for smaller and smaller values of C. This is done by specifying an initial parameter estimate, and initial state $u_{guess} = u_T$. The projection-relaxation algorithm is used to solve (9) for each value of C, producing both a new estimate for the parameter μ^C and state u^C. These are then used as inputs for projection-relaxation after C is reduced by some prescribed factor $C_f > 1$. This process is repeated until C is as small as desired, producing a sequence of parameter estimates.

A sketch of the geometric interpretation of the algorithm is given in Fig. 2. Because the true equilibrium set $\mathcal{M}_0$ may be extremely convoluted, the numerous local minima for (4) may exist. The effect of the interpolation parameter is to smooth out the equilibrium manifold $\mathcal{M}_0$, improving the chances that a global minimum of (9) can be found. A progressive

reduction of C essentially performs continuation on this problem, leading gradually toward the unregularized optimization problem with $C = 0$.

The initial choice of interpolation parameter C somewhat depends on the quality of the target data. If C is chosen to be too small initially, then fidelity to the input data is lost, and the algorithm may converge to a parameter estimate which encapsulates little information about the data. Conversely, if C is chosen to be too large initially, noise from the target data is not sufficiently damped, which can lead to poor performance in the projection-relaxation step. Note that the PR algorithm will always terminate in a finite number of steps, since the halting criteria $|\tilde{\mu}| < \tilde{\mu}_{min}$ will be reached even if the minimum is not obtained. In applications, this appears to be a sign that C is simply too large, and reducing C systematically will correct this issue. A summary of the method described above is provided in algorithm 2.

Algorithm 2: Regularized distance minimization (RMD)

Given: μ_{guess}, u_T, C
Initialize $u^C = u_T$ and $\mu^C = \mu_{guess}$
repeat
  Set (u, μ) to the output of the projection-relaxation algorithm with inputs (u^C, μ^C)
  Let $C := C/C_f$ and $(u^C, \mu^C) = (u, \mu)$
until $C < C_{min}$

3 Local convergence

Local convergence of the foregoing algorithms is now studied. Global behavior is much more difficult to assess and will be reserved for future study, since it depends both on the quality of the initial guesses and the structure of the underlying model. The two specific questions to be addressed are

(1) Does the projection-relaxation iterates converge to a minimum (global or otherwise) of problem (9)?
(2) Does a sequence of minimizers u^{C_n} for $C_n \to 0$ converge to the minimizer in (4)?

Central to both questions is the structure and dimensionality of the manifolds $\mathcal{M}_0$ and $\mathcal{M}_C$ near minimizers. In the simplest case, $\nabla_u N(u_0; \mu_0)$ is non-singular, and an implicit function argument shows that these manifolds are locally parameterizable by μ. Moreover, such an argument shows that $\mathcal{M}_C$ is close to $\mathcal{M}_0$ in the sense

$$\text{dist}(\mathcal{M}_C, u_0) = o(1) \text{as} C \to 0. \tag{18}$$

In theorem 2, it is shown that property (18) is sufficient to guarantee that solutions to the regularized ($C > 0$) optimization problem converge to the original one.

The situation of *degenerate equilibria*, where the dimensionality of $\mathcal{M}_0$ is greater than p, is more complicated. This arises quite naturally, for example, in cases where the evolution equation (1) has conserved quantities or symmetries, which function as "hidden" parameters. One possible remedy is to extend the parameter set to account for these; this is discussed in Sect. 3.4. For gradient systems, on the other hand, it is shown that the closeness property (18) holds without any modification.

3.1 Stability of the PR algorithm

The behavior of the projection-relaxation algorithm in a neighborhood B of the minimizer u^C is now investigated. The manifold $\mathcal{M}_C$ is assumed to be locally p-dimensional within B; this is verified directly for gradient systems in Sect. 3.3. Specifically, it is assumed that $\mathcal{M}_C \cap B$ is the range of a smooth map $u^*(\mu) : P' \to U$ where $P' \subset \mathbb{R}^p$ is a neighborhood of μ^C, and

$$v_i^0 \equiv \frac{\partial u^*}{\partial \mu_i}(\mu^C) \tag{19}$$

are linearly independent. It is also supposed that within the neighborhood B, the map $\mathcal{E}^C(u;\mu)$ is independent of u, so that the relaxation step will select a unique state $u^*(\mu)$.

Provided B is sufficiently small,

$$u^*(\mu) = u^C + \sum_{i=1}^{p} v_i^0(\mu_i - \mu_i^C) + \frac{1}{2}\sum_{i=1}^{p}\sum_{j=1}^{p} Q_{ij}^0(\mu_i - \mu_i^C)(\mu_j - \mu_j^C)$$
$$+O(||\mu - \mu^C||^3), \quad Q_{ij}^0 \equiv \frac{\partial^2 u^*}{\partial \mu_i \partial \mu_j}(\mu^C). \tag{20}$$

Notice that μ^C is obtained by minimizing $||u_T - u^*(\mu)||$, which means that

$$\langle u_T - u, v_i^0 \rangle = 0. \tag{21}$$

The minimum is also assumed to be locally quadratic; in other words, the corresponding Hessian satisfies

$$\text{The matrix} A^0 - Q \text{is positive definite, where} Q_{ij} = \langle u_T - u, Q_{ij}^0 \rangle \text{and} A_{ij}^0 = \langle v_i^0, v_j^0 \rangle. \tag{22}$$

The tangent space basis elements v_i can be estimated

$$v_i = v_i^0 + \sum_{j=1}^{p} Q_{ij}^0(\mu_j - \mu_j^C) + O(||\mu - \mu^C||^2). \tag{23}$$

The linear problem for the parameter increment is $A\tilde{\mu} = b$ where

$$A_{ij} = A^0 + O(||\mu - \mu^C||), \tag{24}$$

and by (20)

$$b_i = \langle u_T - u^C, v_i^0 \rangle - \left\langle \sum_{j=1}^{p} v_j^0(\mu_j - \mu_j^C), v_i^0 \right\rangle + \left\langle u_T - u^C, \sum_{j=1}^{p} Q_{ij}^0(\mu_j - \mu_j^C) \right\rangle + O(||\mu - \mu^C||^2)$$
$$= \sum_{j=1}^{p} [A^0 - Q]_{ij}(u_j^C - \mu_j) + O(||\mu - \mu^C||^2), \tag{25}$$

where (21) was used. Also note that the symmetric matrix $A^0 \equiv \langle v_i^0, v_j^0 \rangle$ is positive definite since the set $\{v_i^0\}$ is linearly independent, and for any nonzero $x \in \mathbb{R}^p$

$$A^0 x \cdot x = \left(\sum_{i=1}^{p} x_i v_i \right)^2 > 0. \tag{26}$$

Provided $||\mu - \mu^C||$ is small enough, a standard perturbation calculation leads to

$$\tilde{\mu} = \mu^C - \mu + (A^0)^{-1} Q(\mu - \mu^C) + O(||\mu - \mu^C||^2). \tag{27}$$

If $\mu^{(j)}$ are parameter value iterates produced by the algorithm, then they obey a mapping of the form

$$\mu^{(j+1)} = \alpha\mu^C + (1-\alpha)\mu^{(j)} + \alpha(A^0)^{-1} Q(\mu^{(j)} - \mu^C) + O(||\mu^{(j)} - \mu^C||^2). \tag{28}$$

Linear stability of the fixed point μ^C is, therefore, satisfied if

$$\left|\sigma\left((1-\alpha)I + \alpha(A^0)^{-1} Q\right)\right| < 1, \tag{29}$$

where $\sigma()$ represents the spectrum of the matrix.

Lemma 1 *The matrix $(A^0)^{-1} Q$ has real eigenvalues smaller than one.*

Proof The product of positive definite and symmetric matrices has real eigenvalues, and (22) means that for any $x \in \mathbb{R}^p$, $A^0 x \cdot x \geq Qx \cdot x$. The eigenvalues solve $Qx = \lambda A^0 x$, so that the maximum eigenvalue is equal to

$$\max_{x \neq 0} \frac{Qx \cdot x}{A^0 x \cdot x} < 1. \tag{30}$$

It follows from the lemma that $\sigma\left((1-\alpha)I + \alpha(A^0)^{-1} Q\right) < 1$. Then to satisfy (29), one needs

$$\alpha < \frac{1}{1 - \min \sigma\left((A^0)^{-1} Q\right)}. \tag{31}$$

The prescription (17) satisfies (31) provided

$$C_\alpha = ||(A^0)^{-1} Q^0||^{-1}, \tag{32}$$

where $||\cdot||$ denotes the operator norm. We have proved

Theorem 1 *If $\mu^{(0)}$ is sufficiently close to μ^C, then the projection-relaxation algorithm gives $\mu^{(j)} \to \mu^C$ and $u^{(j)} \to u^C$ as $j \to \infty$.*

3.2 Convergence of the RMD algorithm

The question of whether minimizers to the regularized problem converge to the minimizer in (4) is now addressed.

Theorem 2 *Suppose that (18) holds. If u^{C_n}, $C_n \to 0$, is a sequence of global minimizers of the problem (9) converging to u_0, then u_0 is the unique global minimizer in (4).*

Proof Suppose that instead, there is a global minimizer in (4) $\tilde{u} \neq u_0$, and therefore

$$||u_0 - u_T|| - ||\tilde{u} - u_T|| \equiv d > 0. \tag{33}$$

By (18), there are $\tilde{u}^C \in \mathcal{M}_C$ with $||\tilde{u}^C - \tilde{u}|| = o(1)$ as $C \to 0$. It follows

$$||\tilde{u}^C - u_T|| \leq ||u_T - \tilde{u}|| + ||\tilde{u}^C - \tilde{u}|| \tag{34}$$

$$= ||u_T - u_0|| + ||\tilde{u}^C - \tilde{u}|| - d \tag{35}$$

$$\leq ||u_T - u^C|| + ||u_0 - u^C|| + ||\tilde{u}^C - \tilde{u}|| - d. \tag{36}$$

C can be chosen small enough so that $||\tilde{u}^C - \tilde{u}|| + ||u_0 - u^C|| < d$, which means that

$$||\tilde{u}^C - u_T|| < ||u_T - u^C||, \tag{37}$$

which contradicts the assumption that u^C was a global minimizer to (9).

3.3 Gradient systems

In the case, where there is an associated potential,

$$N(u;\mu) = \nabla_u F(u;\mu), \quad \text{for some} F : U \times \mathbb{R}^p \to \mathbb{R} \text{with} F \in C^3, \tag{38}$$

the role of C as a bifurcation parameter is much easier to assess, and it will be shown that minimization over degenerate equilibria is automatically accounted for.

It is assumed that the objective function $J = ||u - u_T||$ admits a structurally stable unique global minimum $u_0 \in \mathcal{M}_0$ with corresponding parameter $\mu_0 = \mu^*(u_0)$, and $\mathcal{M}_0$ is a smooth manifold in the neighborhood of that point. Structural stability means that the corresponding Hessian $\nabla^2 J(u_0)$, viewed as a bilinear form on the tangent space of $\mathcal{M}_0$ at u_0, must be positive definite.

To study the structure of $\mathcal{M}_0$ and $\mathcal{M}_C$ near the global minimizer in (4), a Lyapunov–Schmidt decomposition is used. Let Y be the nullspace of $\nabla_u N(u_0, \mu_0)$ and denote by X its orthogonal complement. Define coordinates x and y as the projection of $u - u_0$ onto these subspaces, so that there is an orthogonal matrix Λ, where $(x, y) = \Lambda(u_0 - u_T)$. Similarly let $f(x, y, \mu)$ and $g(x, y, \mu)$ be the projection of $N(u;\mu)$ onto X and Y, respectively. Observe that the system, written in new coordinates, retains the variational structure since

$$\begin{pmatrix} f \\ g \end{pmatrix} = \nabla_{x,y} F'(x, y) \equiv \nabla_{x,y} F\Big(\Lambda^{-1}(x, y)^T + u_0\Big). \tag{39}$$

The system of interest (7) can be written

$$f(x, y, \mu) + C(x_T - x) = 0, \tag{40}$$

$$g(x, y, \mu) + C(y_T - y) = 0. \tag{41}$$

Since $\nabla_u N(u_0, \mu_0)v = 0$ for any $v \in Y$, $f_y(0, 0, \mu_0) = 0 = g_y(0, 0, \mu_0)$, and (39) means that $g_x = F'_{xy} = f_y$ so that $g_x(0, 0, \mu_0) = 0$ also. (To simplify notation, derivatives are to be interpreted tensorially, i.e. $f_{xy} = [\partial^2 f/\partial x_i \partial y_j]$. Contraction over an index, where unambiguous, will be indicated with a dot product). Since u_0 is the minimizer of $||u_T - u||$ subject to (40-41), then $(u_T - u_0) \cdot v = 0$ for any $v \in Y$. In new coordinates, this means that $y_T = 0$.

In (40), $f_x(0, 0, \mu_0)$ is nonsingular and therefore for $C = 0$, the implicit function theorem provides a smooth map $x^*(y, \mu)$ for which $f(x^*(y, \mu), y, \mu) = 0$ in a neighborhood of $(x, y, \mu) = (0, 0, \mu_0)$. Furthermore, it is supposed that the graph of x^* coincides with $\mathcal{M}_0$ in this neighborhood; in other words, the coordinate y accounts for the hidden parameters. This mapping has the following properties.

Lemma 2 *The following hold:*

(a) $x^*_y(0, \mu_0) = 0$,
(b) $f_{yy}(0, 0, \mu_0) = -f_x(0, 00, \mu_0) \cdot x^*_{yy}(0)$.

Proof Differentiation of $f(x^*(y,\mu),y,\mu_0)=0$ leads to

$$f_x\cdot x^*_y+f_y=0,\quad f_x\cdot x^*_{yy}+f_{xx}\cdot x^*_y\cdot x^*_y+f_{yy}=0.$$

Setting $x=y=0$ in each expression gives (a) and (b).

To ensure that $\mathcal{M}_C$ can be constructed locally to u_0, the following property is required:

Lemma 3 *For gradient systems obeying (38), the matrix*

$$I+g_{xy}(0,0,\mu)\cdot f_x^{-1}(0,0,\mu_0)\cdot x_T \tag{42}$$

(where I is the identity) is positive definite.

Proof By (39) then $g_{xy}=f_{yy}$, so that using lemma 2(b),

$$g_{xy}(0,0,\mu_0)\cdot f_x^{-1}(0,0,\mu_0)=-x^*_{yy}(0,\mu_0), \tag{43}$$

(note the order of multiplication is not important since f_x^{-1} is also self-adjoint). The problem (4) is equivalent to minimizing $G(y,\mu)\equiv||x_T-x^*(y,\mu)||^2+||y||^2$. It follows that $G_{yy}(0,\mu_0)=-x^*_{yy}(0)\cdot x_T+I$ is positive definite, which combined with (43) verifies (42).

Theorem 3 *With the foregoing hypotheses, property (18) holds, and moreover $\mathcal{M}_C$ can be locally parameterized by μ for sufficiently small C.*

Proof The system (40-41) describing $\mathcal{M}_C$ has the form

$$0=f_x(0,0,\mu_0)\cdot x+C(x_T-x)+f_\mu(0,0,\mu_0)\cdot(\mu-\mu_0)+O(||x||^2+||y||^2+||\mu-\mu_0||^2), \tag{44}$$

and

$$0=(g_{xy}(0,0,\mu_0)\cdot x-C)\cdot y+g_\mu(0,0,\mu_0)\cdot(\mu-\mu_0)+O(||x||^2+||y||^2+||\mu-\mu_0||^2). \tag{45}$$

The implicit function theorem guarantees there is a smooth map $x=x(y,\mu;C)$ in a neighborhood of $y=0$, $\mu=\mu_0$ and $C=0$ solving (44) with

$$x=x(y,\mu;C)=-f_x^{-1}(0,0,\mu_0)\Big(f_\mu(0,0,\mu_0)\cdot(\mu-\mu_0)+Cx_T\Big)+O(||y||^2+||\mu-\mu_0||^2+C^2). \tag{46}$$

Inserting into (45) gives

$$\begin{aligned}0=&-C\Big(I+g_{xy}(0,0,\mu)\cdot f_x^{-1}(0,0,\mu_0)\cdot x_T+f_x^{-1}(0,0,\mu_0)f_\mu(0,0,\mu_0)\cdot(\mu-\mu_0)\Big)\cdot y\\&+g_\mu(0,0,\mu_0)\cdot(\mu-\mu_0)+O(||y||^2+||\mu-\mu_0||^2+C^2).\end{aligned} \tag{47}$$

Provided $||\mu-\mu_0||=O(C^2)$ and C is small, lemma 3 ensures the y-coefficient is nonsingular. The implicit function theorem may be used to again to determine a smooth solution $y=y(\mu;C)=O(C)$ in the neighborhood of $C=0$ and $\mu=\mu_0$.

3.4 Non-variational problems

In cases where degenerate equilibria in $\mathcal{M}_0$ arise from a known continuous family of symmetries such as translation, setting $C\neq 0$ may break the symmetry, constraining the solutions of (9) to a lower dimensional (typically p-dimensional) manifold compared to $\mathcal{M}_0$. The symmetry in (9) may be restored by redefining the set $\mathcal{M}_C$ to be

$$\mathcal{M}_C=\{w|N(S_\sigma(w),\mu)+C(u_T-S_\sigma(w))=0,\mu\in P,\sigma\in\Sigma\}, \tag{48}$$

where the symmetry $S_\sigma : U \to U$ is a smooth mapping satisfying $S_0(u) = u$ and $N(S_\sigma(u); \mu) = N(u, \mu)$. Typically, Σ is a subset of $\mathbb{R}^q$ or a compact manifold. For example, for translation symmetry in two dimensions with periodic boundary conditions, Σ is the two-dimensional torus T^2.

The problem in (9) may be reformulated to use the constraint (48) and the augmented parameter set (μ, σ). If the goal is to use the RMD algorithm to solve (4), then it remains to check if property (18) holds with the redefinition (48). This question is highly problem specific, and is reserved for future work. Section 6, however, explores a system without a gradient structure and tests the proposed reformulation.

4 Example: the Swift–Hohenberg equation

As a first test of the method, consider the Swift–Hohenberg equation

$$u_t = -(\Delta + K^2)^2 u + \alpha u + \beta u^2 - \gamma u^3 \equiv N(u; \mu), \tag{49}$$

which arises in many spatial pattern-forming applications Cross and Hohenberg (1993). The parameter set considered here is $\mu = (K, \alpha, \beta, \gamma)$, and the spatial domain is a square $[0, 100]^2$, equipped with periodic boundary conditions.

Practical implementation requires an efficient way of solving (7), specifically to compute the equilibrium map $\mathcal{E}^C()$. The simulations in this paper utilize spectral methods for spatial discretization, and a semi-implicit, adaptive time-stepping procedure. Details of these schemes are provided in the appendix.

In this problem, the linear system (12) in the projection-relaxation step has the form $\mathcal{L}v = w \equiv -dN/d\mu(j)$ with

$$\mathcal{L}v = -(\Delta + K^2)^2 v + (\alpha - C)v + 2\beta uv - 3\gamma u^2 v. \tag{50}$$

In two (or more) space dimensions, direct methods to solve this system become undesirable because the associated matrices have a banded diagonal structure. In the context of a spectral discretization, however, one can take advantage of the rapid inversion of the constant coefficient part of the operator in (50). This suggests an iterative method

$$\mathcal{L}_- v_{n+1} = w - \mathcal{L}_+ v_n, \tag{51}$$

where the operator $\mathcal{L}$ is split

$$\mathcal{L}_- = -(\Delta + K^2)^2 - r, \quad \mathcal{L}_+ = (\alpha - C + r) + 2\beta u - 3\gamma u^2, \tag{52}$$

and

$$r = \sup_x \left(C - \alpha - 2\beta u(x) + 3\gamma u(x)^2 \right). \tag{53}$$

Rapid convergence of this iterative scheme is seen in practice, except in cases where C is particularly small.

4.1 Small noise target data

To test the method, a true (numerical) equilibrium was found using parameters $(K, \alpha, \beta, \gamma) = (1.1, 0.6, 0.3, 1.4) \equiv \mu_{exact}$ by evolving from small random initial conditions. The target pattern was then determined by adding 5% Gaussian noise (see Fig. 3).

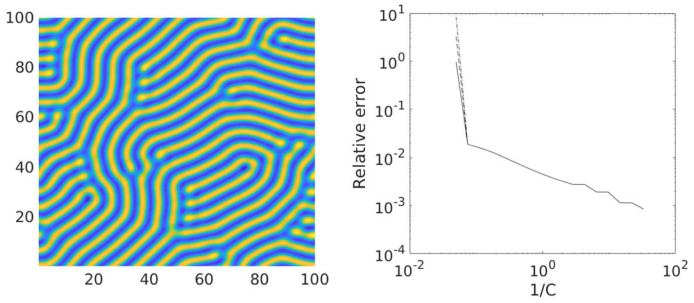

Fig. 3 Left: target data constructed from an equilibrium using random initial and adding 5% Gaussian noise; Right: relative error in the parameter estimate as a function of the inverse interpolation parameter $1/C$. All three initial guesses (given by solid, dashed and dot-dashed curves) yield the same sequence of approximations

Three guesses for parameters were tried: $\mu = (.1, 0, 0, 0), (2, -5, 1, 4,)$, and $(3, 10, 10, 10)$. These were selected somewhat arbitrarily, but within a range of reasonable possibilities, and not too close to the exact parameter set. In this problem, the initial value for C was 20; the other algorithm parameters were $\tilde{\mu}_{min} = 10^{-3}$ and $C_f = 1.3$.

For each value of C, the relative parameter estimate error defined as $||\mu^C - \mu_{exact}||/||\mu_{exact}||$ was calculated. Figure 3 shows that the parameter estimate improves as C is decreased, essentially recovering the exact parameters used to generate the data. There is only a difference in the parameter estimate at the first step, suggesting that the PR algorithm did not encounter multiple local minima. This is not at all surprising, since for large values of C, the optimization problem is an approximation of the least squares regression (5), which necessarily has a unique solution.

Numerous other initial parameter guesses and similar target data were also tried, with nearly identical results. It should be noted that extreme guesses (e.g. $K = 200$) can lead to numerical problems for the solution of (7), either because of timestepping instabilities or lack of spatial resolution, which would be an expected limitation of most model problems. This can potentially be rectified either by an ad-hoc procedure for selection of the initial guess, or developing more robust methods for solution to (7).

4.2 Large noise and a comparison to the minimal residual formulation

A more extreme test was conducted using parameters $\mu_{exact} = (K, \alpha, \beta, \gamma) = (1.1, 0.2, -0.5, 1)$. Again a true equilibrium was found by evolving the equation from random initial data, but in this case a significant amount of noise (50%) was added (Fig. 4).

Three arbitrarily chosen initial guesses for parameters were used, $\mu = (.1, 0, 5, 0)$, $(1, 1, 1, 1)$, and $(.5, -1, 10, 8)$. The algorithm parameters were the same as above. Figure 4 shows how the parameter estimate improves as C is decreased. Notice that the estimate after the first RMD iteration is quite poor, and only improves after a gradual reduction of C. The intuitive reason for this is that noise in the input data becomes less important for smaller values of C since the magnitude of the fidelity term in (10) is reduced.

It is worthwhile to compare the RMD estimates to those produced from the minimal residual formulation (5) alone. Even small amounts of noise will easily corrupt the computation of derivatives in $N(u_T; \mu)$, and smoothing of the raw data u_T is needed to produce reasonable answers. In the context of a spatial spectral discretization, this can be accomplished with a

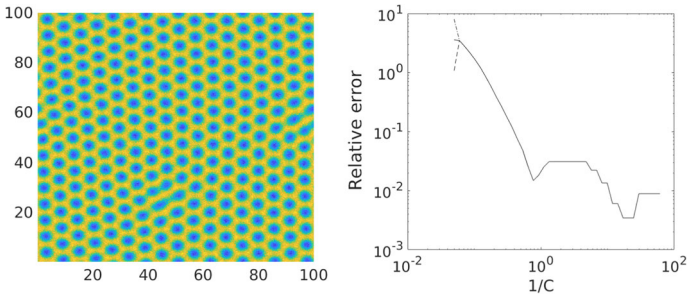

Fig. 4 Left: target data constructed from an equilibrium using random initial and adding 50% noise; Right: relative error in the parameter estimate as a function of $1/C$. All three initial guesses (solid, dashed,dash-dot) yield the same sequence of approximations

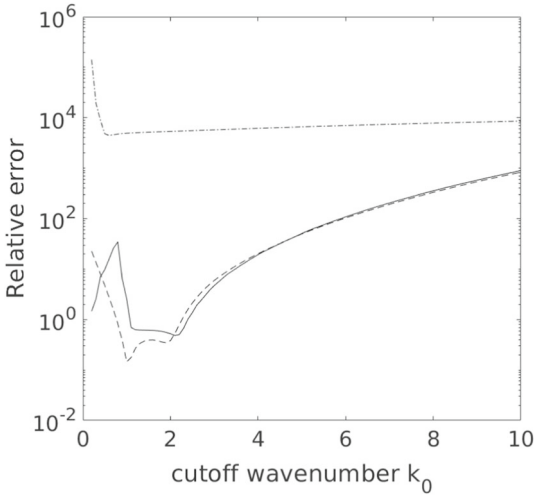

Fig. 5 Relative error of parameter estimate using filtered residual minimization, using the sharp cutoff filter (solid), the exponentially decaying filter (dashed), and the algebraically decaying filter (dash-dot)

low-pass filter defined by $F^{-1}g(k)F$, where F is the (two dimensional) Fourier transform. Three different specifications for the symbol $g(k)$ were tried:

$$g(k) = \begin{cases} \chi(|k| < k_0), & \text{(sharp cutoff)} \\ 1/(1 + \exp(A(|k| - k_0)), & \text{(exponential cutoff)} \\ 1/(1 + |k|^2/k_0^2), & \text{(algebraic cutoff)} \end{cases} \tag{54}$$

To give the filtering procedure the benefit of the doubt, the cutoff wavenumber k_0 was varied. For the exponential cutoff, the parameter A was chosen to obtain the best fit, which turned out to be $A = 4$.

As in the preceding sections, the relative parameter estimate error was computed for each filter type over a range of cutoff values. The results of least squares regression are shown in Fig. 5. In general, parameter estimates were quite poor, regardless of the choice of cutoff wavenumber. The only exception was in a narrow range for the exponentially decaying cutoff, but since k_0 is not given to begin with, it is not clear how this would be determined absent prior knowledge of μ_{exact}. In any case, the quality of estimates using regression alone are not nearly as good as the RMD procedure.

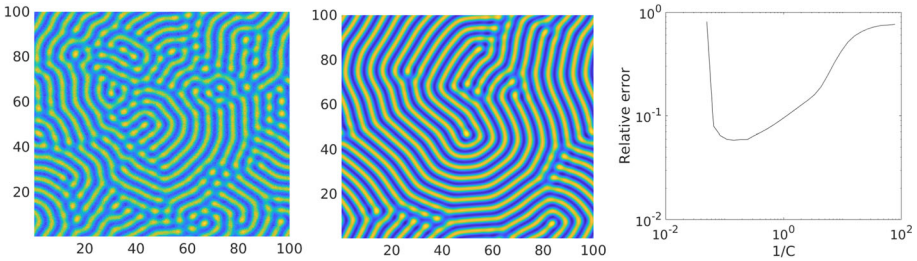

Fig. 6 Left: target data constructed from a transient state at time $t = 30$ with added noise; middle: the equilibrium state, obtained by evolving to $t = 10^8$, which is far from the equilibrium; Right: the relative error of the parameter estimate as the interpolation parameter C is varied

4.3 Transient target data

An even more extreme situation was examined where the data derives from a dynamically transient pattern. This was obtained from evolving the equation from random initial data with parameters $(K, \alpha, \beta, \gamma) = (1.3, .7, .5, 1.5)$ up to $t = 30$, and then adding a significant (50%) amount of noise. The time was chosen so that a recognizable pattern was generated, but yet was very far from the true equilibrium (Fig. 6).

In this test, only the initial guess $\mu = (1, 0, 0, 0)$ was used, although other guesses of the same order of magnitude produced similar results. The other algorithm parameters were the same as the previous examples. The relative error (as defined above) of the parameter estimate was measured as the interpolation parameter C was continually reduced (Fig. 6,right).

For larger values of C, the estimate was poor as is the previous cases. This largely reflects the influence of noise, since the estimate in this regime is similar to the minimum residual estimate. Lowering the value of C effectively filters the input data, and produces better estimates. On the other hand, as C becomes very small, the fidelity to the input data is eroded, and the estimate more reflects the parameters corresponding to a true equilibrium closest to u_T, which is not necessarily correlated with the parameters used to generate the target data to begin with. In this case, the best choice for the interpolation parameter appears to be finite, although presently it is not clear how to choose C in an optimal way.

5 Example: localized patterns in a polymer system

A multiphase model of complex polymers is

$$\begin{pmatrix}\phi_A\\ \phi_B\end{pmatrix}_t = \Delta\left(-G\Delta\begin{pmatrix}\phi_A\\ \phi_B\end{pmatrix} + \nabla W(\phi_A, \phi_B)\right) + \begin{pmatrix}\phi_B - \phi_A\\ \phi_A - \phi_B\end{pmatrix}. \tag{55}$$

The dependent variables ϕ_A, ϕ_B represent densities of monomers in a block copolymer; a detailed discussion of this model can be found in Glasner and Orizaga (2018). Here

$$W = \sigma_{AB}\phi_A^2\phi_B^2 + \sigma_{AS}\phi_A^2(1-\phi_A-\phi_B)^2 + \sigma_{BS}\phi_B^2(1-\phi_A-\phi_B)^2 + \Lambda\phi_A^2\phi_B^2(1-\phi_A-\phi_B)^2, \tag{56}$$

and

$$G = \begin{bmatrix} 2\sigma_{AS} & \sigma_{AS}+\sigma_{BS}-\sigma_{AB} \\ \sigma_{AS}+\sigma_{BS}-\sigma_{AB} & 2\sigma_{BS}\end{bmatrix}. \tag{57}$$

The parameter Λ plays a minor qualitative role in the model, and is simply set to 50 in the following numerical calculations, whereas surface energies $(\sigma_{AB}, \sigma_{AS}, \sigma_{BS})$ are part of the parameter set to be estimated. It should be noted that the evolution is only well posed if G is positive definite; this criteria defines the parameter region over which the optimization problem is considered. The domain and boundary conditions are the same as in Sect. 4.

Written in the form (55), the system is not a gradient, at least with respect to the L^2 metric. The equilibrium problem, on the other hand, can be recast as

$$N(\phi_A, \phi_B; \mu) = -G\Delta \begin{pmatrix} \phi_A \\ \phi_B \end{pmatrix} + \nabla W(\phi_A, \phi_B) + \Delta^{-1} \begin{pmatrix} \phi_B - \phi_A \\ \phi_A - \phi_B \end{pmatrix} + \begin{pmatrix} \lambda_A \\ \lambda_B \end{pmatrix}, \tag{58}$$

where $\lambda_{A,B}$ are undetermined Lagrange multipliers that can be treated as two extra system parameters. The inverse Laplacian Δ^{-1} is defined via the Fourier transform $\hat{u}(k)$ of $u(x)$ as the inverse transform of $\hat{u}(k)/|k|^2$ (setting the $k = 0$ term to zero), and is easily handled in the spectral discretization framework. Written in this form, (58) is a gradient system with respect to the augmented potential

$$F = \int \frac{1}{2} \left| G^{1/2} \nabla \begin{pmatrix} \phi_A \\ \phi_B \end{pmatrix} \right|^2 + W(\phi_A, \phi_B) + \frac{1}{2} \left| \Delta^{-1/2} (\phi_A - \phi_B) \right|^2 dx + \lambda_A \int \phi_A \, dx + \lambda_B \int \phi_B \, dx. \tag{59}$$

The linear systems (12) are solved by the iterative method like (51), where the linear operators are

$$\mathcal{L}_+ = \nabla^2 W - sI, \quad I \equiv \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{60}$$

and

$$\mathcal{L}_- = -G\Delta + (s - C)I - \Delta^{-1} I_2, \quad I_2 \equiv \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}. \tag{61}$$

The choice $s = \max_x \sigma(x)$ where $\sigma(x)$ are principal eigenvalues of $\nabla^2 W(x)$ ensures that L_+ has positive eigenvalues, and guarantees convergence of the iterations. Other aspects of the numerical computation are similar to Sect. 4.

5.1 Tests of parameter estimation

The method's performance was tested for equilibria generated using the original dynamics (55), and adding noise. In the first test, $(\sigma_{AS}, \sigma_{AB}, \sigma_{BS}) = (4, 3, 4)$ and the initial condition was a circular domain shape, divided into two semicircular subdomains $(\phi_A, \phi_B) = (1, 0)$ on one half and $(0, 1)$ on the other half. Outside of these regions, it was necessary to set $(\phi_A, \phi_B) = (.03, .03)$ so that mass does not entirely flow out of the polymer phase domains. After evolving the system to its final equilibrium configuration, noise was added (Fig. 7). The additional exact parameters $(\lambda_A, \lambda_B) = (.15, .15)$ were then determined via (58).

The results for data which have both small (10%) and large (50%) amounts of noise are reported in Fig. 7. The initial parameter guesses were $(\sigma_{AS}, \sigma_{AB}, \sigma_{BS}, \lambda_A, \lambda_B) = (2, 8, 5, 0, 0)$, although other choices (within reason) gave similar results. The error of the estimates became smaller as C was decreased, nearly converging to the original exact parameters in both test cases.

A test problem with a more elaborate domain configuration was also considered (Fig. 8). The exact surface energy parameters were $(\sigma_{AS}, \sigma_{AB}, \sigma_{BS}) = (2, 2, 2)$, the initial conditions which generate the equilibrium state had random values between 0 and 1 within a circular patch, and a moderate amount of noise (10%) was added. The initial guess here was

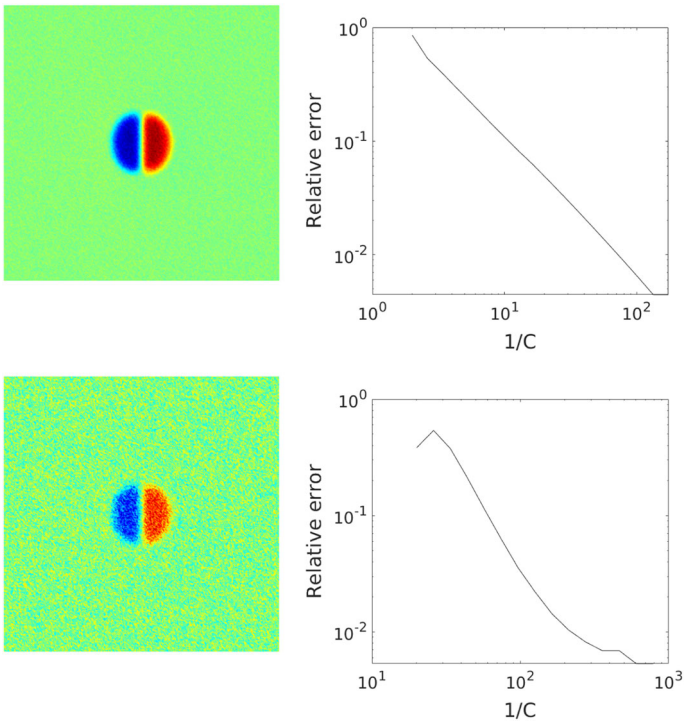

Fig. 7 Left top: target data with 10% added noise; right top: the relative error of the parameter estimate for small noise data. Left bottom: target data constructed with 50% added noise; Right bottom: the relative error for large noise data

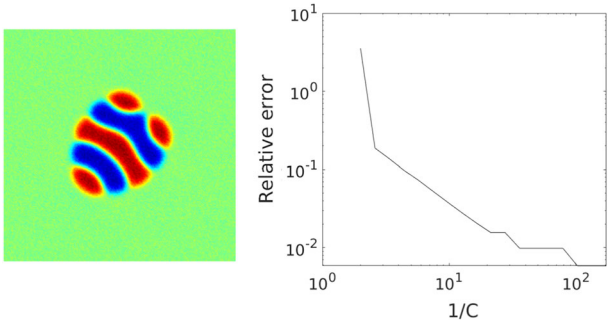

Fig. 8 Left: multi-domain target data; Right: the relative error of the parameter estimate as a function of $1/C$

$(\sigma_{AS}, \sigma_{AB}, \sigma_{BS}, \lambda_A, \lambda_B) = (7, 7, 7, 0, 0)$. The results are very similar to the previous test, where the parameter estimate becomes almost exact for $C \to 0$.

6 Example: A non-variational pattern forming system

As a final illustration of the parameter estimation methodology, we consider the Gray-Scott reaction-diffusion model

$$u_t = D_u \Delta u - uv^2 + f(1-u), \tag{62}$$

$$v_t = D_v \Delta v + uv^2 - (f+k)v. \tag{63}$$

This system has been widely studied because of its diverse pattern forming ability Pearson (1993). The domain and boundary conditions are the same as before.

In addition to the explicit parameter set (D_u, D_v, f, k), there are two hidden parameters corresponding to translation invariance (τ_x, τ_y). These are included in the set of parameters over which optimization is performed. It is convenient to rewrite (48) by changing spatial variables so that the translation symmetry acts only on the target data (u_T, v_T). This results in the minimization of $||(u, v) - (u_T, v_T)||$ subject to the constraint

$$0 = D_u \Delta u - uv^2 + f(1-u) + C[u - u_T(x - \tau_x, y - \tau_y)], \tag{64}$$

$$0 = D_v \Delta v + uv^2 - (f+k)v + C[v - v_T(x - \tau_x, y - \tau_y)], \tag{65}$$

where u, v are evaluated at (x, y).

As in the other test problems, target data was obtained by choosing random initial data for (62–63), evolving this system until some prescribed time, and in this case significant (50%) noise was added. The system produces localized patterns which evolve on a rather slow time scale. As a consequence, the state used for the target pattern was more of a transient than a true equilibrium.

The explicit system parameters were $D_u = 2 \times 10^{-3}$, $D_v = 10^{-3}$, $f = 0.060$, and $k = 0.0609$, and the initial guesses were $D_u = 0.1$, $D_v = 0.1$, $f = 0.1$, $k = 0.1$, and $\tau_x = 0 = \tau_y$. Figure 9 shows the target pattern before and after noise was added. It also shows the relative error of the parameter estimate for noisy data; this quantity only measures the difference of the four explicit system parameters. The algorithm was also used without hidden parameters (Fig. 9, dashed curve), and it produced parameter estimates which were nearly as good. Note that for large values of C, the algorithm failed to improve the estimate, but eventually began to converge toward the true parameters when C became small enough. In contrast to the example in Sect. 4.3, the transient nature of the target configuration did not cause the estimate to eventually worsen as C became very small.

7 Conclusion

This paper introduced a constrained optimization formulation for parameter estimation which combines aspects of previous approaches but avoids some of their difficulties. Efficient algorithms for solving the associated optimization problems were also described. The intractability of finding a global minimizer among numerous local minimizers in (4) appears to be mitigated with this approach. Conversely, the difficulties of residual minimization (5) in handling noisy data can be partially overcome as well.

The projection-relaxation algorithm appears to provide a robust method for locating minimizers to (9). Convergence of this algorithm was noted for a surprisingly wide range of initial parameters, although performance became worse as noise was added. It would be interesting to compare its efficiency and reliability to alternative approaches such as gradient descent-type or Levenberg–Marquardt methods.

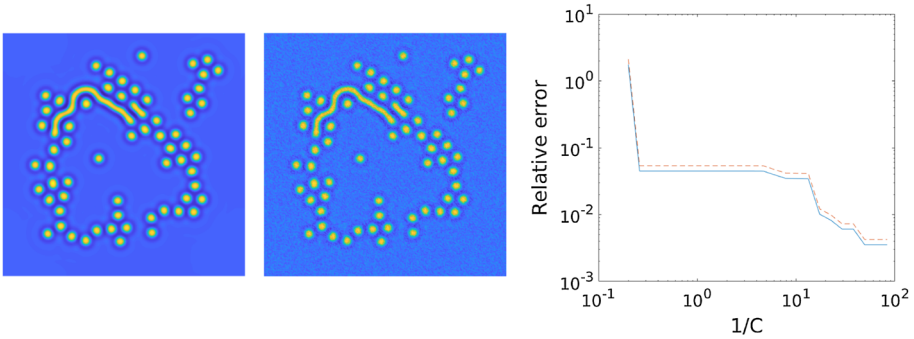

Fig. 9 Left: transient state without noise; Middle: target obtained by adding 50% noise; Right: the relative error of the parameter estimate as the interpolation parameter C is varied, both including hidden parameters (solid) and without (dashed)

Although our method is presented as an attempt to relate the input data to exact steady state solutions, parameter estimates can be quite reasonable even when the target data is a slowly-evolving transient. The reasons for this are not entirely clear at present. For some test problems there is an optimal value for the interpolation parameter C which provides the best parameter estimate. More work should be done to understand how this value should be selected.

The algorithms presented have several associated "meta-parameters" such as $\tilde{\mu}_{min}$ and the initial and final values of C whose choice is problem-specific. It would be advantageous to have a systematic way of determining these, based on criteria for algorithmic efficiency and accuracy of the estimates. In addition, heuristics for providing initial parameter guesses would be helpful.

Perhaps the greatest virtue of the new method is dealing with noisy inputs, sometimes even to the point where the target data pattern is nearly unrecognizable or just a mere caricature of an actual model solution. This seems to be closely tied to the mollifying nature of the underlying evolution equation. In a sense, the equation itself acts as a filter, eliminating fine scale noise in a very natural way. This same phenomenon might be useful when dealing with the residual minimization formulation (5) alone and its sparsity-inducing counterparts.

Acknowledgements The author was supported through NSF awards DMS-1514689 and DMS-1908968.

Appendix

Here we provide details about the numerical methods employed in this paper. In light of periodic boundary conditions, it is natural to employ a standard Fourier pseudo-spectral discretization Trefethen (2000). For square domain $[0, 2\pi]^2$, this approximation takes the form

$$u \approx \sum_{k_x=1}^{N} \sum_{k_y=1}^{N} \hat{u}(k_x, k_y, t) \exp\left[2\pi i \Big(\omega(k_x)x + \omega(k_y)y\Big)\right],$$

where $\omega(k_x), \omega(k_y)$ are the correctly de-aliased wavenumbers. The discrete Fourier transform is used to compute $\hat{u}$, and can be written as a linear map $\hat{u} = \mathcal{F}u$. The Laplacian is therefore

discretized as

$$\Delta u \approx \mathcal{F}^{-1}\Lambda\mathcal{F}u$$

where Λ is just the diagonal multiplication operator $\Lambda\hat{u} = -[(\omega(k_x)^2 + \omega(k_y)^2]\hat{u}$. Note that any rational function of the Laplacian can be similarly computed by spectral mapping, for example,

$$(\Delta - c)^{-1}u = \mathcal{F}^{-1}(\Lambda - cI)^{-1}\mathcal{F}u.$$

We note that the discrete Fourier transform is easily parallelized; for the computations we utilize MATLAB's built-in support for GPU processing.

The discretization in time is based on semi-implicit approaches(e.g. Bertozzi et al. (2011); Song (2016); Glasner and Orizaga (2016)) which only require inversion of constant coefficient operators. A first order method of this form is

$$(u_{i+1} - u_i)/\Delta t = N^+(u_{i+1}) + N^-(u_i^*)$$

where u_i is the ith timestep. Here $N = N^+ + N^-$ is an operator splitting chosen so that N^+ is linear (typically involving the highest order derivatives) and can be rapidly inverted by spectral means. The term u_i^* is a prediction for u_{i+1}, which in the present set of examples is provided by simple linear extrapolation. For gradient flows, this method can be shown to be unconditionally stable when N^+, N^- are negative gradients of the convex and concave parts of the energy Glasner and Orizaga (2016).

To expedite convergence to an equilibrium, the timestep Δt is adapted to match the dynamics at it slows. An easy way to accomplish this is to increase Δt by some factor (usually 1.5) when $||u_{i+1} - u_i||$ is smaller than some tolerance. This is done until a large time T is reached. Since the initial condition u_0 for the operator $E^C(u_0)$ is often already close to an equilibrium, very few (typically $10 - 50$) timesteps are ultimately required.

References

Ackleh AS, Ferdinand RR, Reich S (1998) Numerical studies of parameter estimation techniques for nonlinear evolution equations. Kybernetika 34(6):693–712

Ashyraliyev M, Jaeger J, Blom JG (2008) Parameter estimation and determinability analysis applied to drosophila gap gene circuits. BMC Syst Biol 2(1):83

Bertozzi AL, Ju N, Lu H-W (2011) A biharmonic-modified forward time stepping method for fourth order nonlinear diffusion equations. Discrete Contin Dyn Syst 29(4):1367–1391

Bonnans J-F, Gilbert JC, Lemaréchal C, Sagastizábal CA (2006) Numerical optimization: theoretical and practical aspects. Springer Science & Business Media, New York

Chen L-Q (2002) Phase-field models for microstructure evolution. Ann Rev Mater Res 32(1):113–140

Croft W, Elliott CM, Ladds G, Stinner B, Venkataraman C, Weston C (2015) Parameter identification problems in the modelling of cell motility. J Math Biol 71(2):399–436

Cross MC, Hohenberg PC (1993) Pattern formation outside of equilibrium. Rev Mod Phys 65(3):851

Dewar MA, Kadirkamanathan V, Opper M, Sanguinetti G (2010) Parameter estimation and inference for stochastic reaction-diffusion systems: application to morphogenesis in d. melanogaster. BMC Syst Biol 4(1):21

Friedman A, Reitich F (1992) Parameter identification in reaction-diffusion models. Inverse Probl 8(2):187

Fullana JM, Le Gal P, Rossi M, Zaleski S (1997) Identification of parameters in amplitude equations describing coupled wakes. Phys D Nonlinear Phenomena 102(1–2):37–56

Garvie MR, Maini PK, Trenchea C (2010) An efficient and robust numerical algorithm for estimating parameters in turing systems. J Comput Phys 229(19):7058–7071

Glasner K, Orizaga S (2016) Improving the accuracy of convexity splitting methods for gradient flow equations. J Comput Phys 315:52–64

Glasner K, Orizaga S (2018) Multidimensional equilibria and their stability in copolymer-solvent mixtures. Phys D Nonlinear Phenomena 373:1–12
Goodman J (1985) Newton's method for constrained optimization. Math Program 33(2):162–171
Jin B, Maass P (2012) Sparsity regularization for parameter identification problems. Inverse Probl 28(12):123001
Knowles I (2001) Parameter identification for elliptic problems. J Comput Appl Math 131(1–2):175–194
Long Z, Lu Y, Dong B (2019) Pde-net 2.0: Learning pdes from data with a numeric-symbolic hybrid deep network. J Comput Phys 399:108925
Maddu S, Cheeseman BL, Sbalzarini IF, Müller CL (2019) Stability selection enables robust learning of partial differential equations from limited noisy data. arXiv preprint arXiv:1907.07810
Maziar R, George EK (2018) Hidden physics models: machine learning of nonlinear partial differential equations. J Comput Phys 357:125–141
Michael H, René P, Michael U, Stefan U (2008) Optimization with PDE constraints, vol 23. Springer Science & Business Media, New York
Pearson JE (1993) Complex Patterns in a Simple System. Science 261:189–192
Raissi M, Perdikaris P, Karniadakis GE (2017) Machine learning of linear differential equations using gaussian processes. J Comput Phys 348:683–693
Raissi M, Perdikaris P, Karniadakis GE (2017a) Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations. arXiv preprint arXiv:1711.10561
Raissi M, Perdikaris P, Karniadakis GE (2017b) Physics informed deep learning (part ii): Data-driven discovery of nonlinear partial differential equations. arXiv preprint arXiv:1711.10566
Rudy SH, Brunton SL, Proctor JL, Kutz JN (2017) Data-driven discovery of partial differential equations. Sci Adv 3(4):e1602614
Rudy S, Alla A, Brunton SL, Kutz JN (2019) Data-driven identification of parametric partial differential equations. SIAM J Appl Dyn Syst 18(2):643–660
Schaeffer H (2017) Learning partial differential equations via data discovery and sparse optimization. Proc R Soc A Math Phys Eng Sci 473(2197):20160446
Sgura I, Lawless AS, Bozzini B (2019) Parameter estimation for a morphochemical reaction-diffusion model of electrochemical pattern formation. Inverse Probl Sci Eng 27(5):618–647
Song H (2016) Energy SSP-IMEX Runge-Kutta methods for the Cahn-Hilliard equation. J Comput Appl Math 292:576–590
Trefethen LN (2000) Spectral methods in MATLAB, volume 10. Siam, 2000
Wang Z, Huan X, Garikipati K (2019) Variational system identification of the partial differential equations governing the physics of pattern-formation: Inference under varying fidelity and noise. Comput Methods Appl Mech Eng 356:44–74
Zhao H, Storey BD, Braatz RD, Bazant MZ (2020) Learning the physics of pattern formation from images. Phys Rev Lett 124(6):060201