

Numerical strategy for model correction using physical constraints



Yanyan He, Dongbin Xiu*

ARTICLE INFO

Article history:

Received 24 September 2015

Received in revised form 4 February 2016

Accepted 22 February 2016

Available online xxxx

Keywords:

Uncertainty quantification

Model-form uncertainty

Model correction

ABSTRACT

In this paper we present a strategy for correcting model deficiency using observational data. We first present the model correction in a general form, involving both external correction and internal correction. The model correction problem is then parameterized and casted into an optimization problem, from which the parameters are determined. More importantly, we discuss the incorporation of physical constraints from the underlying physical problem. Several representative examples are presented, where the physical constraints take very different forms. Numerical tests demonstrate that the physics constrained model correction is an effective way to address model-form uncertainty.

© 2016 Elsevier Inc. All rights reserved.

1. Introduction

The field of uncertainty quantification (UQ) has undergone rapid developments in the last decade. While tremendous progresses have been made in the quantification of parametric uncertainty, much less has been accomplished in the front of model-form uncertainty. One of the primary goals of model-form uncertainty analysis is to quantify the deficiency of the simulation models, as it is widely accepted that all models are approximations to the physical truth. To this end, the approach developed by Kennedy and O'Hagan [8] has found its use in a variety of applications. See, for example, [1,4,5,11,13], and the references therein. In this approach, the simulation model is corrected by adding a discrepancy term. Then, both the simulation model and the discrepancy term are parameterized by Gaussian processes. The hyper-parameters in the Gaussian processes are then estimated using Bayesian inference based on observational data. While effective in many cases, the explicit introduction of the additive discrepancy term often destroys certain important physical properties of the underlying problem, which are often built into the simulation model via tremendous effort. This undesirable side effect has been recognized in the literature, and efforts have been made to mitigate it, for example, via constrained prior construction in the Bayesian inference [2]. Another approach to preserve the physical properties of the model is to directly embed the model discrepancy in the model, as opposed to treating it as an external additive term. This was proposed in [12], where random parameters are introduced inside the model to allow one to tune the model using data. The random variables are then parameterized by polynomial chaos expansion [3,14] and Bayesian inference is then employed to estimate the expansion coefficients.

In this paper, we first frame the model correction problem in a general manner, using model discrepancy terms both external and internal to the model. More importantly, we discuss a means of explicitly incorporating the physical constraints that the underlying problem should satisfy. This is vital to the corrected model, as it is intended to be an improved model to the underlying physical phenomenon. Upon parameterizing the model correction terms, our model correction problem

* Corresponding author at: Department of Mathematics and Scientific Computing and Imaging Institute, University of Utah, Salt Lake City, UT 84112, USA.
E-mail address: dongbin.xiu@utah.edu (D. Xiu).

is formulated as an optimization problem, where the distance between the corrected model predictions and observation data is minimized. The physical constraints are also parameterized and then incorporated in the optimization problem as constraints. This results in the core contribution of this paper – the method of physics constrained model correction (PCMC). A general discussion about the physical constraints is difficult, as they can take vastly different forms depending on the underlying physical problems. Instead, we then proceed to use several representative examples to illustrate the effects of the physical constraints. The examples include a diffusion problem with imperfect modeling of the diffusivity and boundary conditions, a mechanical system with conservation of energy, a differential equation model with missing boundary layer transition, and a two-dimensional Navier–Stokes equations model with incompressibility constraint. Using extensive numerical tests, we demonstrate that the PCMC is an effective method to produce improved model predictions and performs better than the unconstrained model correction (UMC) which does not enforce the important physical constraints. We remark that by “better” we merely refer to the fact that PCMC is able to directly incorporate the physical constraints, if desired, into the prediction. Ultimately, the quality of a model should be judged by its predictability and how it compares against the ground truth. Just because one model incorporates certain physical constraints does not necessarily imply it has better predictability. How to judge the predictability of a model is, however, not the scope of this paper. Here we focus exclusively on the numerical implementations of the incorporation of the physical constraints, if one deems their incorporation necessary. Whether and how the incorporation of the physical constraints improves the model prediction is a problem dependent issue and should be decided prior to the model correction procedure.

This paper is organized as follows. In Section 2 we discuss the general forms of model correction. These include external model correction, internal model correction, and mixed model correction using both external and internal corrections. The physics constrained model correction (PCMC) is then presented in Section 3. Several representative examples of the physical constraints are discussed in Section 4. Then, in Section 5, we present extensive numerical tests for these representative systems and demonstrate the effectiveness of the PCMC framework.

2. Model correction

Throughout this paper, we use $M(x; p(x))$ to denote the simulation model. Here $x \in D$ is the coordinate in a physical domain, and $p(x)$ represents an important process embedded in the model. For example, it can be the diffusivity field of a diffusion model. For notational convenience, we do not include the time variable. This is done without loss of generality, as the incorporation of time is straightforward by introducing $(x, t) \in D \times [0, T]$ for some $T > 0$. Our numerical examples shall include a time dependent problem to demonstrate the applicability to time domain. We also restrict ourselves to the discussion of deterministic models.

We denote y^t the “true” (and unknown) output of the underlying physical system, and let

$$d = y^t + e, \quad (2.1)$$

be the observation data with noise e . Let $y = M(x; p)$ be the model output, we seek to construct a *corrected model* \hat{M} such that it is a better approximation of the truth y^t . While there may exist a variety of ways to introduce model corrections, here we focus on the approaches that can be broadly classified as *external correction* and *internal correction*.

2.1. External model correction

In external model correction, we seek to introduce the correction terms outside the simulation model M . The most obvious approach is of additive type, that is,

$$\hat{M} = M(x; p) + \delta(x), \quad (2.2)$$

where $\delta(x)$ is the correction term, and \hat{M} stands for the corrected model. This is the most widely used model correction form. For example, the well known Kennedy–O’Hagan method [8] takes this form.

One may also consider a multiplicative correction and seek

$$\hat{M} = \delta(x)M(x; p). \quad (2.3)$$

Obviously, the two can be combined into a more general form of external correction.

$$\hat{M} = \delta_m(x)M(x; p) + \delta_a(x), \quad (2.4)$$

where δ_m stands for the multiplicative correction factor and δ_a for the additive correction term.

2.2. Internal model correction

In internal model correction, we seek to “correct” the important internal modeling process $p(x)$, in order to improve the model prediction. The similar correction form used in the external correction, (2.4), can now be employed to the internal

process $p(x)$:

$$\hat{p}(x) = \delta_m(x)p(x) + \delta_a(x), \quad \hat{M} = M(x; \hat{p}(x)), \tag{2.5}$$

where, again, δ_m and δ_a stand for the multiplicative factor and the additive correction term, respectively, for the internal process p . An additive internal model correction can be obtained by setting $\delta_m(x) \equiv 1$; whereas a multiplicative internal model correction can be obtained by setting $\delta_a(x) \equiv 0$.

2.3. Mixed model correction

Obviously, one can combine both the internal and external correction and construct a more general mixed type model correction. It shall take the following form,

$$\begin{aligned} \hat{p}(x) &= \delta_m^{(i)}(x)p(x) + \delta_a^{(i)}(x), \\ \hat{M} &= \delta_m^{(e)}(x)M(x; \hat{p}(x)) + \delta_a^{(e)}(x), \end{aligned} \tag{2.6}$$

where the superscripts (i) and (e) stand for the internal and the external corrections, respectively, and the subscripts a and m stand for the additive term and the multiplicative factor, respectively. The simplifications to the special cases of the external correction form (2.4) and the internal correction form (2.5) are obvious. Throughout this paper, we will often write (2.6) in a more compact form,

$$\hat{y} = \hat{M}(x; \delta(x)), \quad \delta(x) = (\delta_m^{(i)}, \delta_a^{(i)}, \delta_m^{(e)}, \delta_a^{(e)}), \tag{2.7}$$

where $\delta(x)$ includes all the correction terms in the external and/or the internal form.

We remark that even though this form is general, in practice one shall prefer “simpler” models, in order to reduce the complexity of both the modeling efforts and the simulation burden. To this end, intuition and experience of the underlying physical problems and the properties of the existing model M are critical in the construction of the model corrections.

3. Model correction with physical constraint

In this section we present the strategies for implementing the model correction methods (2.6). In particular, we introduce physics constrained model correction (PCMC) and discuss its numerical implementations.

3.1. Physical constraints

For many practical problems, the underlying physical problem possesses certain properties that need to be satisfied. For example, conservation of certain quantities, positivity of some variables, etc. Let us denote the constraints as a set of (nonlinear) functions

$$C_i(y^t) = 0, \quad i = 1, \dots, n_c, \tag{3.1}$$

where n_c is the number of equality constraints the true solution y^t satisfies, and a set of inequality constraints

$$F_i(y^t) \leq 0, \quad i = 1, \dots, n_f, \tag{3.2}$$

where n_f is the number of the inequality constraints.

3.2. Physics constrained model correction (PCMC)

Let (x_j, d_j) , $j = 1, \dots, n_d$, be a set of observational data (2.1). By following the traditional approach, we assume the observation error e has zero mean. We now seek to determine the correction term δ in (2.7) using the observational data.

Let V_δ be a linear space from which the correction δ is constructed. Let $\dim(V_\delta) = n_\delta$ and $(b_1, \dots, b_{n_\delta})$ be a set of basis for V_δ . We can then express $\delta(x)$ as

$$\delta(x) = \sum_{i=1}^{n_\delta} \alpha_i b_i(x), \tag{3.3}$$

and further write the corrected model (2.7) as

$$\hat{y} = \hat{M}(x; \alpha), \quad \alpha = (\alpha_1, \dots, \alpha_{n_\delta})^T, \tag{3.4}$$

where the dependence of the corrected model \hat{M} on the coefficients α of the correction function δ is explicitly shown. The corrected model \hat{M} is now parameterized via the introduction of the basis in V_δ . We remark that it is reasonable to use

different linear spaces for the internal correction term and the external correction term in (2.6). The space V_δ is then the superposition of the different spaces.

Let $\hat{y}_j(\alpha) = \hat{M}(x_j; \alpha)$ be the corrected simulation data at x_j , $j = 1, \dots, n_d$, where the observations are made. Let $d = (d_1, \dots, d_{n_d})$ and $\hat{y} = (\hat{y}_1, \dots, \hat{y}_{n_d})$ be the observational data vector and the simulation data vector, respectively, and $C = (C_1, \dots, C_{n_c})$ be the set of the equality constraint functions in (3.1). We then seek to solve the following optimization problem

$$\begin{aligned} \alpha^\dagger &= \underset{\alpha}{\operatorname{argmin}} \|d - \hat{y}(\alpha)\|, \quad \text{subject to} \\ C_i(\hat{y}(\alpha)) &= 0, \quad i = 1, \dots, n_c, \\ F_j(\hat{y}(\alpha)) &\leq 0, \quad j = 1, \dots, n_f. \end{aligned} \tag{3.5}$$

Alternatively, one can absorb the equality constraints into the objective function and rewrite (3.5) as

$$\begin{aligned} \alpha^\dagger &= \underset{\alpha}{\operatorname{argmin}} \|d - \hat{y}(\alpha)\| + \lambda \|C(\hat{y}(\alpha))\|_w, \quad \text{subject to} \\ F_j(\hat{y}(\alpha)) &\leq 0, \quad j = 1, \dots, n_f, \end{aligned} \tag{3.6}$$

where, for a set of non-negative weights $w = (w_1, \dots, w_{n_c})$ satisfying $w_1 + \dots + w_{n_c} = 1$,

$$\|C\|_w = \sum_{i=1}^{n_c} w_i \|C_i\| \tag{3.7}$$

is a weighted norm for the constraint functions. Both (3.5) and (3.6) are viable choices mathematically. In practice, one may wish to choose one of them depending on the problem at hand.

Note that in (3.5) and (3.6) the norms are left unspecified. In practice, they can be chosen based on the specific problem at hand. For the distance between d and \hat{y} , one can use a standard vector norm, e.g., the 2-norm. The norms in the weighted norm of C in (3.7) are more subtle. Since each constraint C_i may be vastly different, the norms must be chosen properly based on the specific forms of each C_i .

Upon solving the optimization problem (3.6) or (3.5), we can construct the physics constrained model correction (PCMC) term as

$$\delta^\dagger(x) = \sum_{i=1}^{n_\delta} \alpha_i^\dagger b_i(x), \tag{3.8}$$

and consequently, the new physics constrained model

$$\hat{M} = \hat{M}(x; \delta^\dagger(x)) \tag{3.9}$$

in the form of (2.7).

3.3. Unconstrained model correction (UMC)

One obvious simplification of PCMC is the removal of the physical constraints. Or, in certain cases, the physical constraints do exist, but they do not impose any effective constraints on the model corrections. The PCMC is then reduced to unconstrained model correction (UMC)

$$\alpha^\dagger = \underset{\alpha}{\operatorname{argmin}} \|d - \hat{y}(\alpha)\|. \tag{3.10}$$

Most of the existing model correction methods do not explicitly incorporate physical constraints and can be loosely classified as UMC. In this approach, (3.10) achieves the purpose of fitting the observational data. It is obvious that the result of UMC shall not preserve any physical properties in general.

3.4. Effects of physical constraints

The physical constraints have a direct impact on the solution of the PCMC optimization problem (3.5) or (3.6). In general, the constraints may take vastly different forms for different problems. They may also exert highly complex and nonlinear effects on the variable α in the PCMC optimization (3.5) or (3.6). Consequently, the effective number of constraints on the coefficients α may be different. This is because the potentially highly nonlinear constraint functions may possess certain complicated dependence structure, in term of α .

Note that in some cases one may wish to employ only the internal model correction (2.5), as in the case of [12]. If some physical constraints are already enforced by the simulation model M , then the corrected model $\hat{M} = M(x; \hat{p}(x))$ in (2.5) will automatically satisfy these constraints. Consequently, these constraints become redundant in the PCMC formulation (3.5)

or (3.6). The overall effective physical constraints will be less than $n_c + n_f$, which is the total number constraints. Let n_c^{eff} be the number of the effective physical constraints on α . Obviously, $n_c^{eff} \leq n_c + n_f$. We then distinguish the optimization problem (3.6) into two scenarios.

3.4.1. Over-determined case

When $n_d + n_c^{eff} \geq n_\delta$, the overall number of constraints, imposed by both the data and the physical constraints, is larger than the number of unknown coefficients. The PCMC optimization problem (3.6) or (3.5) essentially becomes a regression problem. In fact, when the vector 2-norm is employed, it can be considered a (nonlinear) constrained least squares problem.

3.4.2. Under-determined case

When $n_d + n_c^{eff} < n_\delta$, the PCMC optimization problem (3.6) or (3.5) becomes under-determined, as the number of unknowns (n_δ) is larger than the number of overall constraints. In this case, to ensure the solvability of the optimization problem, we introduce a regularization term and seek to solve, instead of (3.5),

$$\begin{aligned} \alpha^\dagger &= \operatorname{argmin}_\alpha \|d - \hat{y}(\alpha)\| + \beta \|\alpha\|, \quad \text{subject to} \\ C_i(\hat{y}(\alpha)) &= 0, \quad i = 1, \dots, n_c, \\ F_j(\hat{y}(\alpha)) &\leq 0, \quad j = 1, \dots, n_f, \end{aligned} \tag{3.11}$$

where $\beta > 0$. Or, instead of (3.6), we solve

$$\begin{aligned} \alpha^\dagger &= \operatorname{argmin}_\alpha \|d - \hat{y}(\alpha)\| + \lambda \|C(\hat{y}(\alpha))\|_w + \beta \|\alpha\|, \\ \text{subject to} \quad F_j(\hat{y}(\alpha)) &\leq 0, \quad i = 1, \dots, n_f. \end{aligned} \tag{3.12}$$

We leave the norm in the regularization term $\|\alpha\|$ unspecified. In practice, one may choose the vector 1-norm (to enhance sparsity), 2-norm, or else. The under-determined case is more difficult numerically than the over-determined case. Care must be taken to ensure its solution behaves properly. In the rest of the paper we will focus exclusively on the over-determined case and leave the under-determined case to be reported in a separate work.

4. Representative examples of physical constraints

The general discussion of the physics constrained model correction (PCMC) method (3.6) is difficult, without specifying the underlying physical problem. Here we list a few representative examples of constraints. In the next section we will use these examples to conduct extensive numerical investigation. Note that in these examples we use fairly general-purposed physical constraints. In practical problems, however, one may encounter very specialized constraints. The PCMC procedure shall apply equally well.

4.1. Diffusion problem

Our first example is the well studied elliptic problem. Without loss of generality, we consider Dirichlet boundary conditions,

$$\begin{cases} -\nabla \cdot (a(x)\nabla u(x)) = f(x), & x \in D, \\ u(x) = g(x), & x \in \partial D. \end{cases} \tag{4.1}$$

The physical constraints considered in this paper are the conservation of fluxes. More specifically, let $V^{(i)}$, $i = 1, \dots, n_v$, be a number of control volumes, we then impose

$$-\oint_{\partial V^{(i)}} a(x) \frac{\partial u}{\partial n} dx = \int_{V^{(i)}} f(x) dx, \tag{4.2}$$

where n is the outward normal vector along the boundary $\partial V^{(i)}$.

Note that if we define the control volumes to be small elements that discretize the entire computational domain, then the constraints (4.2) effectively recover the governing equation (without the boundary condition). This can be a useful property for benchmarking purpose. Also, in certain situations, the constraint become redundant, and the PCMC and UMC become equivalent. This shall be made clear in the different examples we present in the next section. In practice, one can allocate the control volumes only to a number of selective and “important” locations within the computational domain.

An obvious inequality constraint for this problem is

$$a(x) \geq 0, \tag{4.3}$$

which may become active if one uses internal model correction to correct the diffusivity field.

4.2. Mechanical dynamics

We now consider the Newton's law,

$$m\ddot{x}(t) = -\nabla V(x), \quad x(0) = x_0, \quad \dot{x}(0) = v_0, \quad (4.4)$$

where m is the mass and V the potential energy. We are interested in the solution in a time domain $t = [0, T]$ for some $T > 0$.

The physical constraint is the conservation of energy, that is,

$$E(t) = \frac{1}{2}m\dot{x}^2 + V(x) = E(0), \quad (4.5)$$

where the constant $E(0)$ is determined by the initial condition.

Note that this is a time dependent problem. The framework discussed in the previous section can be readily applied by replacing the x variable by the temporal variable t and consider the time domain $[0, T]$. For long time integration with $T \gg 1$, a global representation/parameterization of the correction term $\delta(t)$ may not be desirable. In this case, one can adopt a piecewise representation of $\delta(t)$ in the time domain.

4.3. Differential equation with boundary layer

Let us now consider the following ordinary differential equation

$$\epsilon y_{xx} + (1 + \epsilon)y_x + y = 0, \quad y(0) = 0, \quad y(1) = 1, \quad (4.6)$$

where $\epsilon > 0$ is a small parameter. When $\epsilon \ll 1$, the solution exhibits a sharp (and smooth) transition near the left boundary $x = 0$ – the so-called boundary layer. This can be considered as a simplified model for the well known boundary layer problem in PDE (e.g., Navier–Stokes equations).

For this well studied problem, a straightforward asymptotic analysis shows that the solution can be decomposed into two parts (cf. [9]).

- Outer solution for $x \sim O(1)$. Upon dropping the small ϵ terms, one obtains $y_x + y = 0$ and the corresponding outer solution $y^0 = e^{1-x}$.
- Inner solution for $x \sim O(\epsilon)$. Utilizing a change of variable $\tau = x/\epsilon$, one obtains a transformed equation with the leading terms $y_{\tau\tau} + y_\tau = 0$. Upon solving it and matching it with the outer solution, one obtains the inner solution $y^1 = e - e^{1-x/\epsilon}$.

The global solution is then a composition of the two solution, $y = e(e^{-x} - e^{-x/\epsilon})$. It is a first order approximation of order ϵ .

In this paper we will use a numerical model $M(x)$ that does not capture the boundary layer. We then require our physical constraint to re-introduce the “missing” physics. Realizing that inside the boundary layer the derivative terms with the ϵ factor in (4.6) reach a balance, we then require that

$$\epsilon y_{xx} + y_x \approx 0, \quad x \sim O(\epsilon). \quad (4.7)$$

Upon integration once, this gives

$$\epsilon y_x + y \approx \text{const} = e, \quad x \sim O(\epsilon), \quad (4.8)$$

where the constant is determined by the matching condition with the outer solution.

4.4. Incompressible flow

We finally consider the incompressible fluid flows, governed by the incompressible Navier–Stokes equations,

$$\begin{aligned} \nabla \cdot v &= 0, \\ \frac{\partial v}{\partial t} + v \cdot \nabla v &= -\nabla p + \nu \nabla^2 v, \end{aligned} \quad (4.9)$$

where p is the pressure and ν the viscosity.

In this paper, without considering any special cases, we will use the incompressibility as the physical constraint, i.e.,

$$\nabla \cdot v = 0. \quad (4.10)$$

5. Numerical examples of PCMC

In this section, we present various numerical examples to demonstrate the properties of PCMC. We will use the four example problems presented in the previous section. In all of the numerical tests, we use the following setup.

- The true solution y^t : This will be the high resolution numerical solution of the governing equation. That is, our “true” solution is the numerically exact solution of the governing equation.
- The data d : We use synthesized data that are generated at selected locations by adding random noise to the true solution y^t . The noise is i.i.d. Gaussian with zero mean value. The variance of the noise is relative to y^t :

$$d = y^t(1 + \sigma e), \tag{5.1}$$

where $e \sim \mathcal{N}(0, 1)$ is i.i.d. Gaussian with zero mean and unit variance, $\sigma \geq 0$ is the relative standard deviation. In all examples, we will use the phrase 100 $\sigma\%$ noise to refer to the value of σ . Multiple simulation runs are conducted to avoid coincidental matching.

- The model M : The models we use here are imperfect modeling of the governing equations (the truth). The imperfection takes the following forms: crude numerical approximations, incorrect modeling of the inputs (e.g., boundary conditions), or, simplification of the governing equations.

Note that the availability of the true solution y^t here only serves the purpose of testing the numerical results. In all examples, the PCMC algorithms use only the synthetic data d and the model output M . The corrected model output \hat{M} is then compared to the true solution y^t to gauge the effectiveness of the PCMC algorithms. In all the numerical examples, we keep the number of data points (n_d) larger than the number of expansion coefficients (n_δ) in the correction terms. This ensures all of the tests are over-determined and avoids the numerical issues with respect to undersampling, which is a separate topic. We employ the vector 2-norm in the PCMC formulation (3.6). This essentially makes the PCMC a constrained (nonlinear) least squares problem, and the UMC an unconstrained least squares problem. Obviously there exist a large variety of numerical algorithms to solve the (nonlinear) least squares problem. Here we employ one of the most standard algorithms, the quasi-Newton method, to solve it. Which algorithm is best suited for this type of optimization problem is not the focus of this paper.

We conduct extensive tests for the diffusion problem example, where the external, internal, and mixed model corrections are considered. For the rest of the examples, we employ only the additive external model correction. In one example of the diffusion problem, we activate the inequality constraint (4.10). In all the other examples, we only consider equality constraints and use (3.6) to solve the PCMC optimization.

In all examples, we employ Legendre orthogonal polynomials to parameterize the correction terms. This results in better numerical stability, compared to monomials, at higher degrees. Obviously one is free to choose other parameterizations, e.g., radial basis, Gaussian process, etc. The degree of the Legendre polynomials is gradually increased till a reasonable resolution independent solution is reached.

5.1. Diffusion problem

For the ease of benchmarking, we consider a one-dimensional diffusion problem with Dirichlet boundary conditions

$$\begin{cases} \frac{d}{dx} \left(a(x) \frac{du}{dx} \right) = 0, & x \in (-1, 1), \\ u(-1) = u_\ell, & u(1) = u_r, \end{cases} \tag{5.2}$$

where the diffusivity $a(x)$ shall take different forms in our examples. We employ a second-order finite volume method with very dense uniform grids to fully resolve this simple equation. This gives us the true solution u^t .

5.1.1. External model correction

We first define our true solution u^t as the solution of (5.2) with

$$a(x) = a^t(x) = 1 + \frac{1}{2} \sin(2\pi x), \quad u_\ell^t = -0.1, \quad u_r^t = 1.2. \tag{5.3}$$

Our model $M(x)$ is the solution of the same equation (5.2) but with

$$a(x) \equiv 1, \quad u_\ell = 0, \quad u_r = 1. \tag{5.4}$$

This immediately gives us a simple analytical solution of the model

$$M(x) = \frac{x+1}{2}. \tag{5.5}$$

Obviously, the deficiency of the model M stems from the incorrect modeling of the diffusivity a and the boundary conditions.

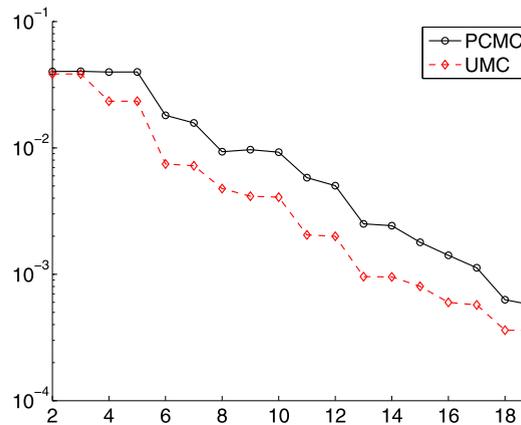


Fig. 5.1. External model correction for the diffusion problem without noise. The L2 errors in the corrected model \hat{M} with respect to the order of the polynomial expansion.

We then seek to construct an improved model by the additive external model correction (2.2) and use a global Legendre polynomial expansion to parameterize the correction term $\delta(x)$. That is, we seek

$$\hat{M}(x) = M(x) + \delta(x) = M(x) + \sum_{i=0}^n \alpha_i L_i(x), \quad (5.6)$$

where n is the order of the polynomial expansion. This implies that the linear space from which δ is parameterized is $V_\delta = \mathbb{P}_n$, the space of polynomials of degree up to n and $n_\delta = \dim(V_\delta) = n + 1$.

The physical constraints are the conservation of the fluxes (4.2). In this one-dimensional example and with the absence of the source term f , it reduces to the continuity of the fluxes across the interfaces of the representative control volumes. For any control volume $V^{(j)}$ centered at x_j , $j = 1, \dots, n_v$, we enforce

$$a^t(x) \frac{d\hat{M}}{dx} \Big|_{x=x_{j-\frac{1}{2}}} = a^t(x) \frac{d\hat{M}}{dx} \Big|_{x=x_{j+\frac{1}{2}}}, \quad j = 1, \dots, n_v. \quad (5.7)$$

The constrained PCMC optimization is then conducted to obtain the expansion coefficients α .

The synthetic data are generated at a set of uniformly distributed points in the domain. We first consider the noise-free case, where the data are the true solutions without the Gaussian errors. The convergence of the corrected model output \hat{M} to the true solution u^t is shown in Fig. 5.1, with respect to the order of the Legendre expansions. Here we keep the number of data to be three times the number of unknown coefficients α , i.e., $n_d = 3n_\delta$, to ensure the underlying least squares problem is well behaved. We can clearly observe the exponential convergence of the solution errors, for both the PCMC and UMC. Note that since the data contain no noise, the model correction (5.6), with or without the constraints (PCMC or UMC), essentially becomes a Legendre polynomial approximation to a smooth function (the true solution). The fast convergence of both PCMC and UMC is not unexpected.

We then consider noisy data, by adding 10% Gaussian noise to the true solution u^t . The corrected model outputs \hat{M} are plotted in Fig. 5.2, for both PCMC and UMC at expansion order $n = 9$. We observe that both the PCMC and UMC are able to fit the observation data well. The PCMC, however, also fits the underlying physics because of the incorporation of the constraints. Consequently, the PCMC solution reproduces the true solution much better and overlaps with the true solution in most part of the domain. On the other hand, the UMC solution is a mere least squares fitting of the data and does not show such a strong correlation with the true solution.

We also remark that although the PCMC works well in this example, the implementation of the flux constraints (5.7) requires the knowledge of the true diffusivity field $a^t(x)$. Information like this can be difficult to acquire for complex problems. In practice, constraints without requiring too much information shall be preferred. In the next few examples for the same diffusion problem, the knowledge of a^t is no longer needed.

5.1.2. Internal model correction

We now define the true solution u^t to be the solution of (5.2) with

$$a(x) = a^t(x) = 1 + \frac{1}{2} \sin(2\pi x), \quad u_\ell^t = 0, \quad u_r^t = 1. \quad (5.8)$$

Our model M is again the solution with $a \equiv 1$, $u_\ell = 0$, and $u_r = 1$ and satisfies $M(x) = (x + 1)/2$.

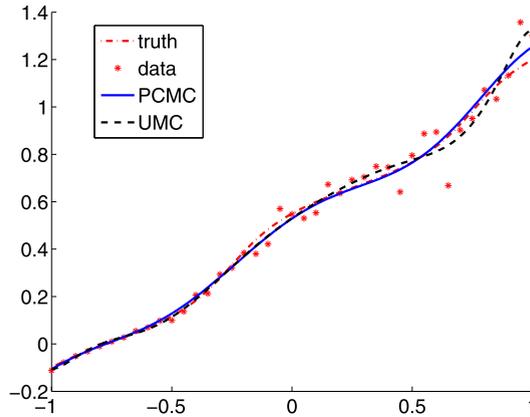


Fig. 5.2. External model correction for the diffusion problem with 10% noise. The true solution u^t and the corrected models \hat{M} by both PCMC and UMC, along with the observational data.

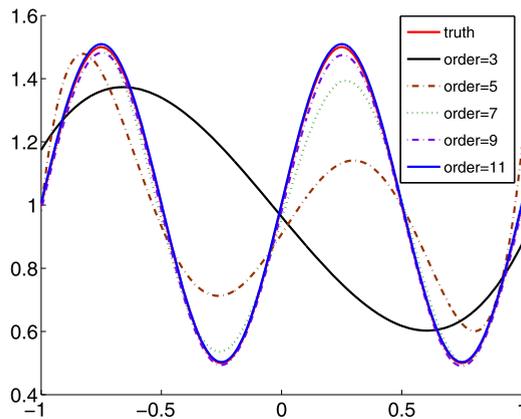


Fig. 5.3. Internal model correction for the diffusion problem without noise. The corrected diffusivity $\hat{a}(x)$ with respect to the order of Legendre expansion.

We now construct an internal additive model correction using Legendre expansion

$$\hat{a}(x) = 1 + \delta(x) = 1 + \sum_{i=0}^n \alpha_i L_i(x), \tag{5.9}$$

and the corrected model $\hat{M}(x; \hat{a})$ to be the solution of (5.2) with

$$a(x) = \hat{a}(x), \quad u_\ell = 0, \quad u_r = 1. \tag{5.10}$$

That is, in this example, the deficiency of the model M lies solely in the “incorrect” modeling of the diffusivity $a \equiv 1$. We then seek to correct the model output by using an internal correction on the incorrect diffusivity. Note that since the modified solution \hat{M} satisfies the governing equation, it automatically satisfies the constraints. The PCMC and UMC become equivalent.

Again, we first test the noise-free case, where clear numerical convergence can be observed. We use $n_d = 41$ equally distributed data, which are the true solution u^t . In Fig. 5.3, the corrected diffusivity field $\hat{a}(x)$ (5.9) is plotted at different order of the Legendre polynomial expansions, along with the true diffusivity field $a^t(x)$. It can be seen that the corrected diffusivity \hat{a} converges to the true diffusivity a^t – the two become visually indistinguishable at order $n = 11$. The errors in the corrected diffusivity \hat{a} and the corrected model solution \hat{M} are tabulated in Table 5.1 and then plotted in Fig. 5.4. We clearly see the fast convergence.

We then consider the noisy data case. The corrected diffusivity \hat{a} and corrected model \hat{M} are plotted in Fig. 5.5. Here the noise level is at 5% and we used $n = 5$ order Legendre expansion for the internal correction term. The results agree with the truth well. Numerical results at higher-order polynomials do not show much improvement, as the presence of the noise prevents the clear convergence seen in the noiseless case.

5.1.3. Mixed model correction

We now present a more comprehensive example requiring the mixed form of model correction (2.6).

Table 5.1

Internal model correction for the diffusion problem without noise. Errors in the modified diffusivity \hat{a} and the modified model solution $\hat{M}(x)$.

Expansion order	3	5	7	9	11
Errors in \hat{a}	4.07×10^{-1}	1.70×10^{-1}	4.67×10^{-2}	1.59×10^{-2}	6.68×10^{-3}
Errors in \hat{M}	6.41×10^{-2}	1.46×10^{-2}	2.85×10^{-3}	1.81×10^{-4}	1.12×10^{-5}

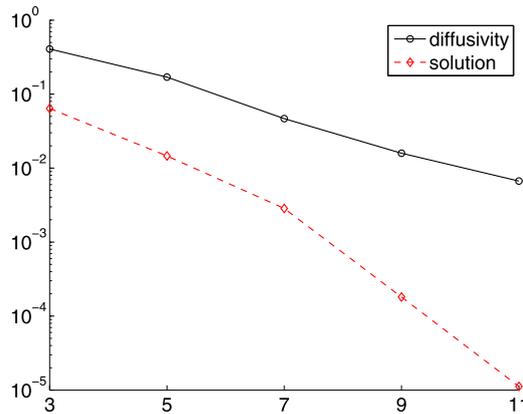


Fig. 5.4. Internal model correction for the diffusion problem without noise. Errors in the modified diffusivity $\hat{a}(x)$ and the modified model solution $\hat{M}(x)$.

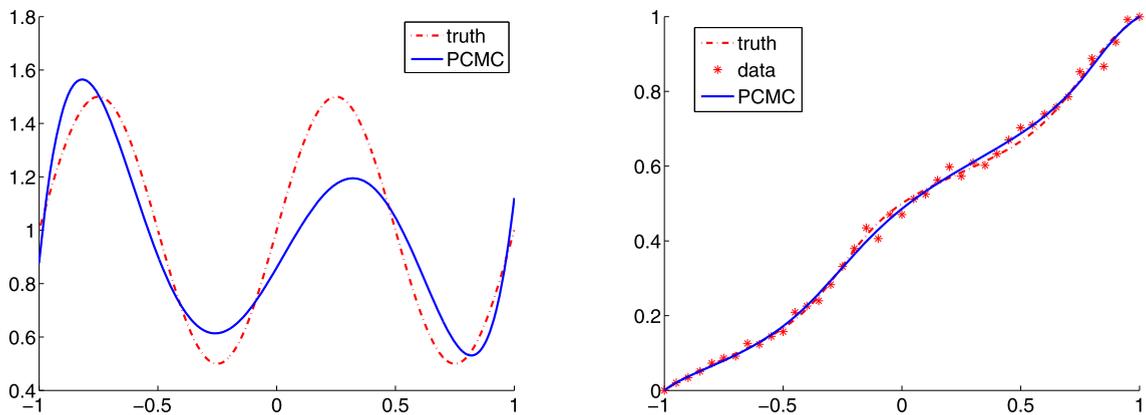


Fig. 5.5. Internal model correction example with 5% noise at polynomial order $n = 5$. Left: the corrected diffusivity $\hat{a}(x)$; Right: the corrected model output \hat{M} .

The true solution u^t is the solution of (5.2) subject to

$$a(x) = a^t(x) = 1 + \frac{1}{2} \sin(10\pi x), \quad u_\ell^t = 0, \quad u_r^t = 1. \tag{5.11}$$

Note that this results in a rather oscillatory true solution. The true diffusivity $a^t(x)$ and the true solution $u^t(x)$ are shown in Fig. 5.6.

Our model focuses on a region of interest – a small interval $I = [x_\ell, x_r] = [-0.125, 0.075]$, which is marked by the rectangle area in Fig. 5.6. This is to mimic the situation where the global solution has fine structure and is expensive to fully resolve. In such a case, one often resorts to study the local solution behavior in certain sub-domain. (Obviously in this simple example the global solution is easy to obtain for benchmarking.) The location of the particular interval I is arbitrarily chosen in this example.

The numerical model $M(x)$ is the diffusion equation (5.2) restricted to $x \in I = [x_\ell, x_r] = [-0.125, 0.075]$ and subject to

$$a \equiv 1, \quad u|_{x_\ell} = \frac{1}{2}(x_\ell + 1), \quad u|_{x_r} = \frac{1}{2}(x_r + 1). \tag{5.12}$$

That is, the numerical model M is the restriction on the interval I of the trivial solution (5.5) under the assumption (5.4).

Since both the boundary conditions and the diffusivity in the model M are incorrect, we construct the corrected model \hat{M} as a mixed type (2.6) with only additive terms. That is, we let

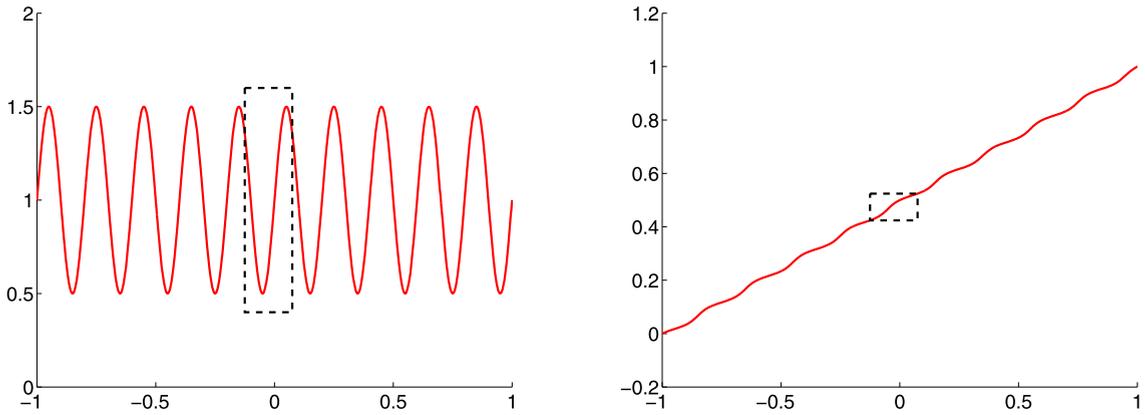


Fig. 5.6. The true solution for the mixed model correction example. Left: the diffusivity field; Right: the solution. The black rectangle box indicates the region of interest, where the numerical model is constructed.

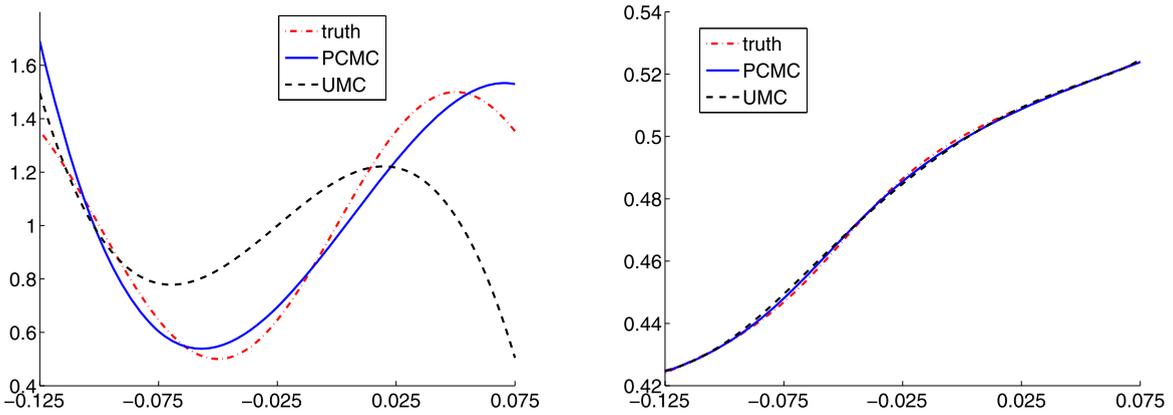


Fig. 5.7. Mixed model correction example without noise at $n_a = n_u = 3$. Left: the corrected diffusivity $\hat{a}(x)$; Right: the corrected model output \hat{M} .

$$\hat{a}(x) = 1 + \delta^{(i)}(x) = 1 + \sum_{k=0}^{n_a} \alpha_k^{(i)} L_k(x), \quad x \in I = [x_\ell, x_r], \tag{5.13}$$

and the corrected model

$$\hat{M}(x; \hat{a}) = M(x; \hat{a}) + \delta^{(e)}(x), \tag{5.14}$$

where $M(x; \hat{a})$ is the solution of (5.2) restricted to $x \in I$ with

$$a(x) = \hat{a}(x), \quad u|_{x_\ell} = \frac{1}{2}(x_\ell + 1), \quad u|_{x_r} = \frac{1}{2}(x_r + 1). \tag{5.15}$$

The external correction term $\delta^{(e)}$ is also expressed as a Legendre polynomial series

$$\delta^{(e)}(x) = \sum_{k=0}^{n_u} \alpha_k^{(e)} L_k(x). \tag{5.16}$$

The continuity of the flux constraints (5.7) are then applied to \hat{M} in a set of small control volumes in I in the PCMC scheme. In this example, we use $n_v = 40$ equally-sized control volumes to enforce the constraints.

We first study the model corrections (both PCMC and UMC) with noise-free data. In this case, the data are the true solution at $n_d = 51$ equally spaced points in I . The polynomial expansion orders for both \hat{a} and \hat{M} are kept the same, i.e., $n_a = n_u = n$. This implies that in the PCMC optimization (3.6), there are $n_\alpha = 2n + 2$ number of variables to optimize. The solutions obtained at $n = 3$ are shown in Fig. 5.7 and the solutions at $n = 7$ are shown in Fig. 5.8. While the corrected model outputs by PCMC and UMC are both close to the true solution, their behaviors in the corrected diffusivity \hat{a} are noticeably different. First, we notice that in PCMC, the corrected diffusivity \hat{a} matches the true a well, and the result at $n = 7$ is much better than that at $n = 3$. This is expected. For UMC, however, the diffusivity field completely misses the true solution at the

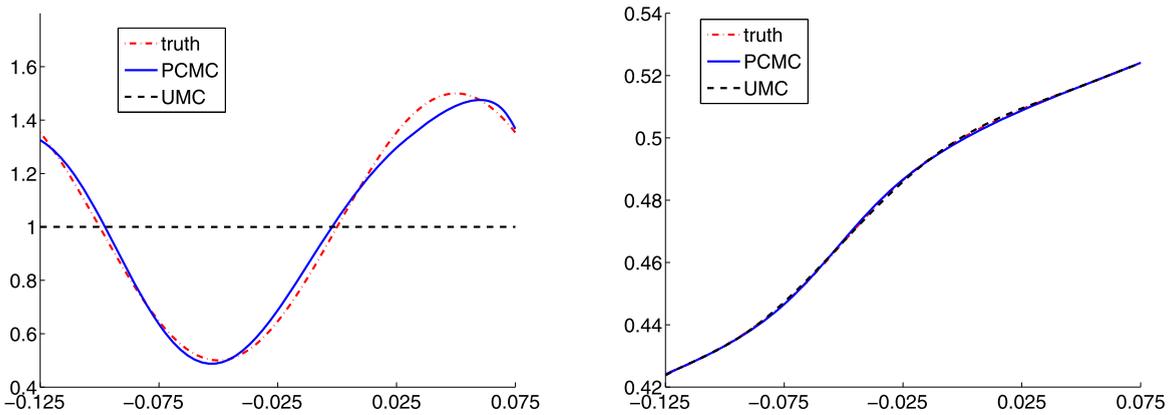


Fig. 5.8. Mixed model correction example without noise at $n_a = n_u = 7$. Left: the corrected diffusivity $\hat{a}(x)$; Right: the corrected model output \hat{M} .

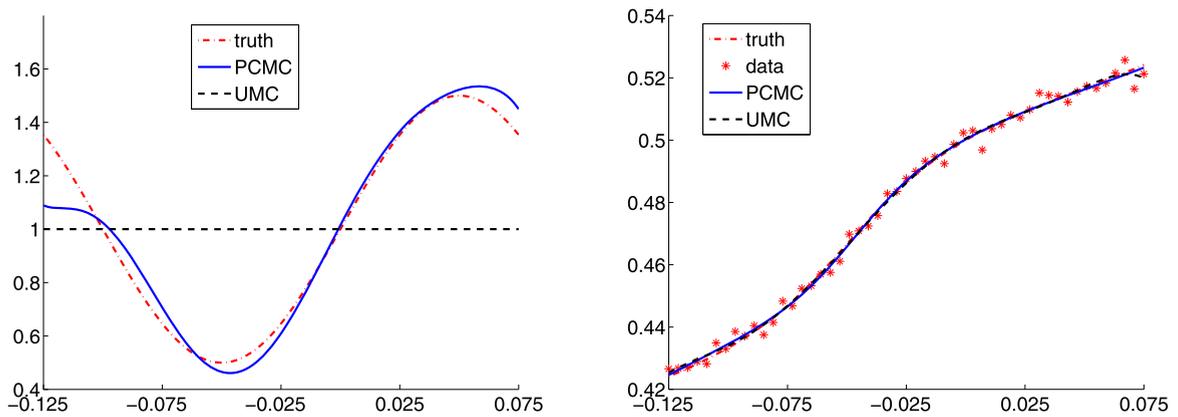


Fig. 5.9. Mixed model corrections with 0.5% noise at polynomial order $n = 7$. Left: the corrected diffusivity \hat{a} ; Right: the corrected model \hat{M} .

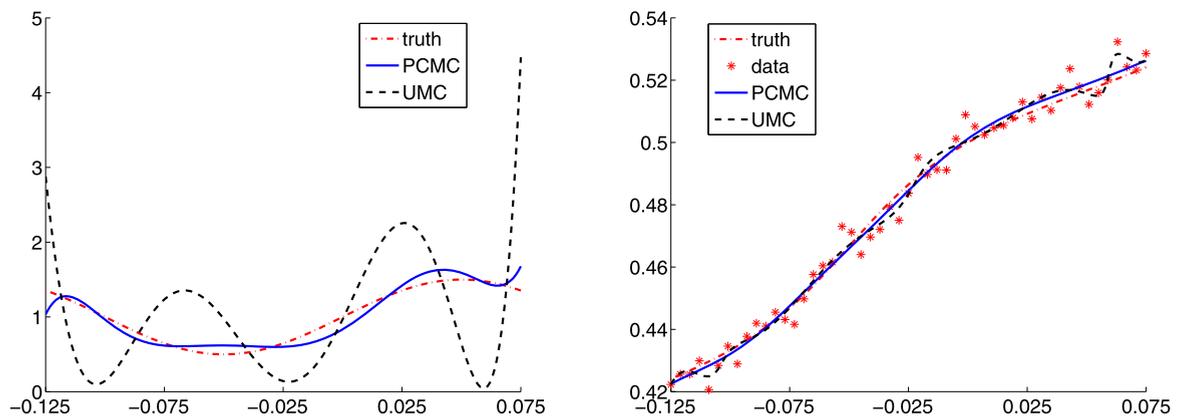


Fig. 5.10. Mixed model corrections with 1% noise at polynomial order $n = 7$. Left: the corrected diffusivity \hat{a} ; Right: the corrected model \hat{M} .

higher order $n = 7$. This is due to the fact that the external correction term $\delta^{(e)}$ is highly effective in correcting the solution discrepancy. At $n = 7$, the presence of $\delta^{(e)}$ alone is able to reduce the residue of the UMC optimization (3.10) to the error tolerance. Consequently, the internal correction $\delta^{(i)}$ becomes inactive. It is only at the lower order $n = 3$, when the external correction $\delta^{(e)}$ alone is not able to fully correct the solution, that the internal correction $\delta^{(i)}$ becomes active in the UMC minimization (3.10).

We then consider the problem with noisy data. More specifically, we present the cases of 0.5% and 1% noises. The results are shown in Fig. 5.9 and Fig. 5.10, respectively. While both PCMC and UMC are able to fit the solution data well, the fits to the diffusivity field are quite different. The PCMC can reproduce the diffusivity field very well, and the UMC certainly

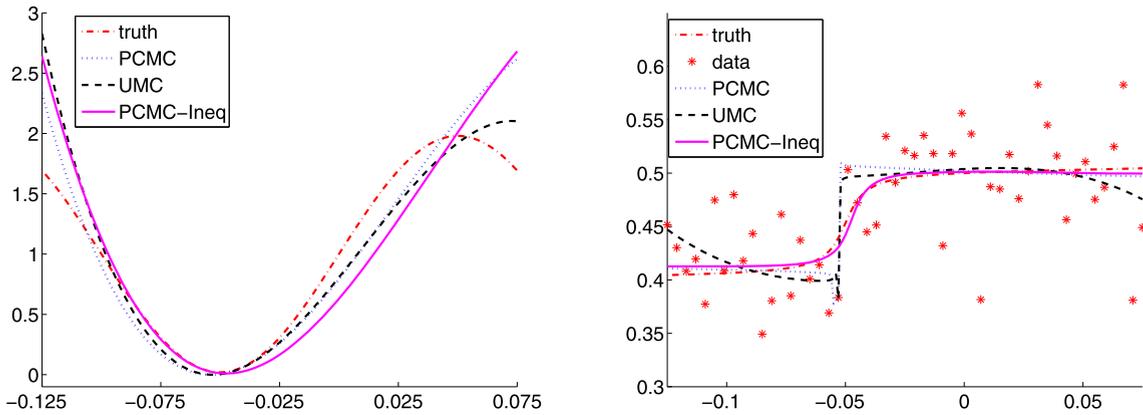


Fig. 5.11. Mixed model corrections with 10% noise and near zero crossing diffusivity field at low polynomial order $n = 3$. Left: the corrected diffusivity \hat{a} ; Right: the corrected model \hat{M} . Here PCMC stands for the PCMC method without inequality constraint, and PCMC-ineq stands for PCMC with the inequality constraint.

cannot. This clearly indicates the need for the physical constraints in PCMC. Although the UMC can indeed match the data in a reasonable way, its corrected model solution can be considered “unphysical”, for it does not inherit the internal physical properties.

5.1.4. Mixed model correction with inequality constraint

We now repeat the same example in the previous section. The only difference is in the true diffusivity. Instead of (5.11), we now set

$$a^t(x) = 1 + 0.98 \sin(10\pi x). \tag{5.17}$$

We then add 10% relative noise to the true solution u^t to generate the synthetic observational data. The same mixed model correction (5.13) to (5.16) is then applied, where we use a Legendre polynomial expansion at a low order $n = 3$ for the correction terms. Since the minimum of the true diffusivity field is dangerously close to zero, the noise level in the data is high (at 10%), and the polynomial fit is of low order ($n = 3$), we activate the inequality constraint (4.3) to avoid the zero crossing of the corrected diffusivity field \hat{a} . The PCMC problem (3.6) with the inequality constraint is then solved. The results are shown in Fig. 5.11. While not visually obvious, the corrected diffusivity $\hat{a}(x)$ by UMC and PCMC without the inequality constraint both contain negative values, whereas the PCMC with the inequality constraint stays strictly positive. The zero crossing by UMC and PCMC without the inequality constraint has a more obvious impact on the corrected solutions – sharp oscillations are visible at the location of zero crossing. On the other hand, the PCMC with the inequality constraint is able to produce smooth and much better corrected solution.

5.2. Mechanical dynamics

We now turn to the mechanical dynamics example in Section 4.2. The true solution x^t is the high resolution solution of (4.4) subject to

$$V(x) = -\frac{1}{2}x^2 + \frac{1}{4}x^4, \quad x_0 = 2, \quad v_0 = 1. \tag{5.18}$$

By fixing the mass at $m = 1$, we readily obtain the energy constant to be conserved at $E(0) = 2.5$.

Our model $M(t)$ is a crude numerical discretization to the governing equation (4.4). In particular, we employ the first order Euler forward method with a time step $\Delta t = 0.1$. This numerical model obviously contains non-trivial numerical errors and does not conserve the energy constant.

We only consider the additive type external model correction (2.2), as there is no internal process p in this problem. We also employ a piecewise Legendre polynomial representation of the correction term. For $k = 1, \dots$,

$$\hat{M}(t) = M(t) + \delta^{(k)}(t) = M(t) + \sum_{i=0}^n \alpha_i^{(k)} L_i(t), \quad t \in [T_{k-1}, T_k], \tag{5.19}$$

where $T_0 = 0$ and $T_k > T_{k-1}$. Without loss of generality, we employ uniform pieces of length 1 with $T_k = k$, and fix the polynomial order at $n = 6$ in each piece.

The observation data are generated at $n_d^{(k)} = 11$ equal distance points inside each sub-interval $[T_{k-1}, T_k]$, by adding 10% relative error of i.i.d. Gaussian to the true solution x^t .

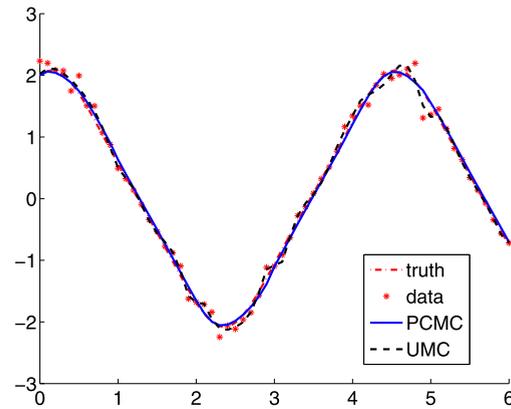


Fig. 5.12. Mechanical dynamics problem for time $t \in [0, 6]$.

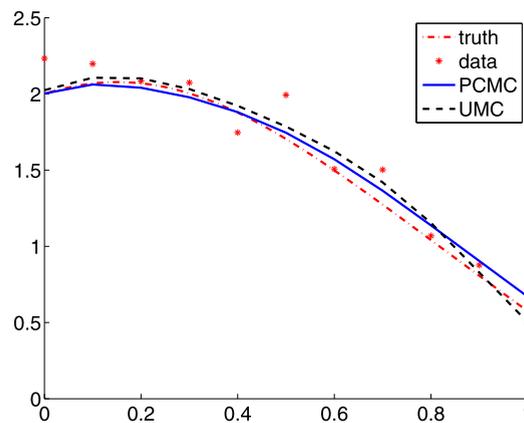


Fig. 5.13. Zoomed in view of the solutions of the mechanical dynamics problem for time $t \in [0, 1]$.

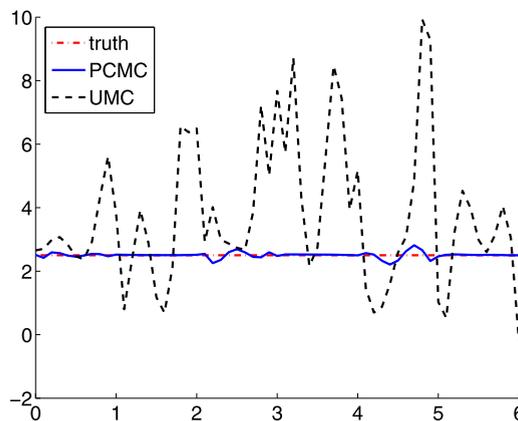


Fig. 5.14. Mechanical dynamics problem. The evolution over time of the energy constants by PCMC and UMC.

The numerical results are shown in Fig. 5.12, where we observe a better approximation to the true solution by the PCMC than the UMC. The zoomed-in view of the solutions, along with the noisy data, are shown in Fig. 5.13, for the first time interval $t \in [0, 1]$. The time history of the energy constants $E(t)$, obtained by PCMC and UMC, are shown in Fig. 5.14. We observe that the PCMC results conserve the energy constant much better. It is now clear that the use of the physical constraint in PCMC enforces the corrected model \hat{x} to not only fit the noisy observational data but also the underlying true physical model much better. The UMC, on the other hand, merely fits the noisy data and does not preserve the important physical property. We also remark that the constraint in this case, the conservation of the energy constant, is not directly embedded in the model M . Consequently, it is not clear how to preserve the property via internal correction.

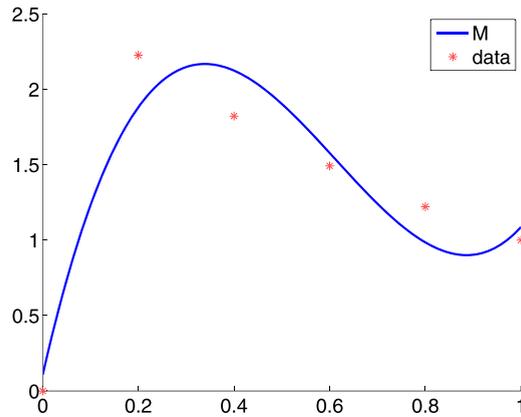


Fig. 5.15. Illustration of the crude numerical model M for the boundary layer example. It is a third order polynomial fit of 6 true solution data.

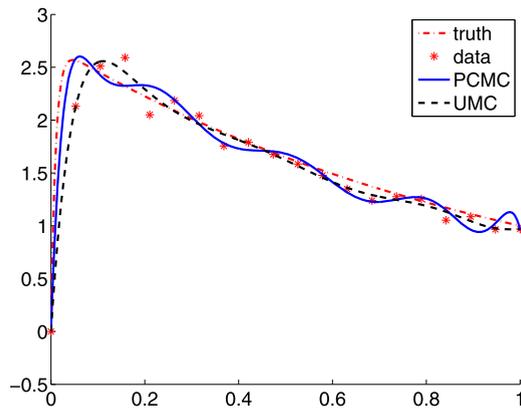


Fig. 5.16. Boundary layer problem. Solutions by PCMC and UMC.

5.3. Differential equation with boundary layer

We now turn to the boundary layer problem in Section 4.3. The true solution y^t is the high resolution numerical solution to the governing equation (4.6).

We construct our model M as a very crude approximation to the true solution. More specifically, the model M is a very low third order polynomial fit of the true solution using 6 data points. This is shown in Fig. 5.15. It is obvious that this crude model completely misses the boundary layer.

Our corrected model is the additive external correction (2.2), where we use Legendre polynomial expansion of order n to parameterize the correction term

$$\hat{y} = \hat{M}(x) = M(x) + \delta(x) = M(x) + \sum_{i=0}^n \alpha_i L_i(x). \tag{5.20}$$

The physical constraint is (4.8). It is enforced via an integral form, i.e.,

$$C(\hat{y}) = \left(\int_0^\epsilon (\epsilon \hat{y}_x + \hat{y} - e)^2 dx \right)^{1/2}. \tag{5.21}$$

We choose $n_d = 20$ equidistant data points, which are generated by adding 10% Gaussian errors to the true solution. The parameter ϵ is set to be $\epsilon = 0.01$. This implies that all the data point lie outside the boundary layer. Therefore, in this example the data points do not provide much information about the missing physics. We do, however, enforce the corrected boundary condition at $x = 0$ in the model corrections, i.e., $\hat{y}(0) = 0$.

Fig. 5.16 shows the computational results, for polynomial order $n = 10$. We clearly observe that the PCMC is able to capture the correct boundary layer transition very well. On the other hand, the UMC does not quite capture the boundary layer. It does produce a transition region. However, this transition is a numerical artifact, and it is the result of the numerical fitting of the left boundary condition $\hat{y}(0) = 0$ and the data points. The quality of this numerical layer depends solely on

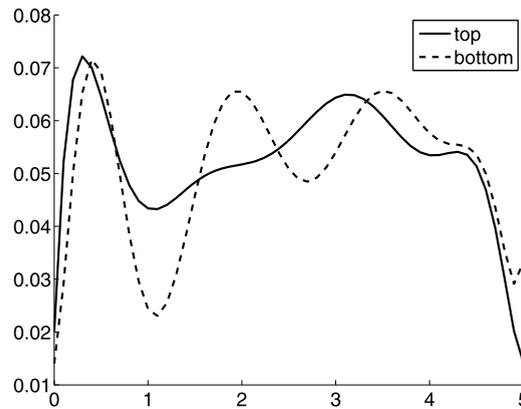


Fig. 5.17. The incompressible channel flow: the true boundary conditions for v_1 at the top and bottom walls.

the quality and locations of first few data points. In PCMC, the boundary layer is captured by enforcing the correct physics. The mild oscillations in the results are numerical artifacts of using a high order polynomial to fit a function with sharp gradients.

5.4. Incompressible flow

Finally, we consider the incompressible flow in Section 4.4. Here, we consider a two-dimensional channel flow in a domain $x = (x_1, x_2) \in [0, 5] \times [0, 1]$.

The true solution $v^t = (v_1^t, v_2^t)$ is the solution of the Navier–Stokes equations subject to the following boundary conditions:

$$\begin{aligned} \text{Inlet } (x_1 = 0) : \quad & v_1^t = x_2(1 - x_2), \quad v_2^t = 0, \\ \text{Outlet } (x_1 = 5) : \quad & \frac{\partial v_1^t}{\partial x_1} = 0, \quad \frac{\partial v_2^t}{\partial x_1} = 0. \end{aligned} \quad (5.22)$$

At the top and bottom boundaries, we let $v_2^t = 0$ and then v_1^t be two independent realizations of a Gaussian process with zero mean and a very short correlation length. The profiles of v_1^t at these two boundaries are shown in Fig. 5.17. This setup can be considered as a model for micro-flow or nano-flow in the channel. As widely recognized, fluid flows at such small scales allow slippery boundary conditions (cf. [7]), as opposed to the non-slip condition at macroscopic scales. The use of the two particular realizations of the Gaussian process is rather arbitrary. They only serve the purpose of numerical testing. The true solution v^t are then the steady state solution obtained by a high-order *hp*-spectral element method [6], via the software package Nektar++ [10]. The simulation domain is decomposed into five equally-sized element, each of which is of size 1×1 . Inside each element, we use order 8 spectral expansion to obtain the numerically exact solution. The viscosity is set at $\nu = 0.01$.

The observation data are then generated by adding i.i.d. Gaussian noises to the true solution. More specifically, in the interior of each element we distribute uniformly 8×8 points and then add the Gaussian i.i.d. noises to both v_1^t and v_2^t to generate the synthetic data: $d_1 = v_1^t(1 + \sigma e_1)$ and $d_2 = v_2^t(1 + \sigma e_2)$. Here $\sigma \geq 0$ controls the relative size of the error. The total number of data points is $n_d = 8 \times 8 \times 5 = 320$.

Our model solution $M(x) = (M_1, M_2)$ is the steady state solution with the boundary conditions (5.22), along with the non-slip boundary conditions at the top and bottom boundaries. This results in the well known plane Poiseuille flow satisfying

$$M_1 = x_2(1 - x_2), \quad M_2 = 0. \quad (5.23)$$

This implies that the deficiency of the model M stems from the incorrect modeling of the boundary conditions.

We employ the external additive model correction (2.2) to both velocity components, i.e.,

$$\hat{M}_1(x) = M_1(x) + \delta_1(x), \quad \hat{M}_2(x) = M_2(x) + \delta_2(x). \quad (5.24)$$

The correction terms are constructed using two-dimensional Legendre polynomial series of degree of degree n .

In the PCMC optimization (3.6), the data component takes the following form.

$$\|d - \hat{M}(\alpha)\| = \sum_{i=1}^{n_d} \|d_{1,i} - \hat{M}_1(x_i)\|_2 + \tau \|d_{2,i} - \hat{M}_2(x_i)\|_2, \quad (5.25)$$

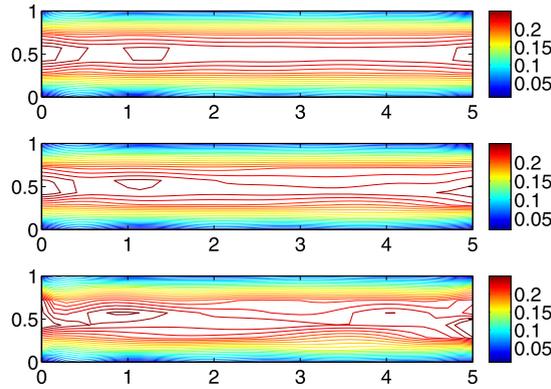


Fig. 5.18. The incompressible channel flow. The contours of the horizontal velocity with 10% relative errors in data. Top: true solution; Middle: PCMC solution; Bottom: UMC solution.

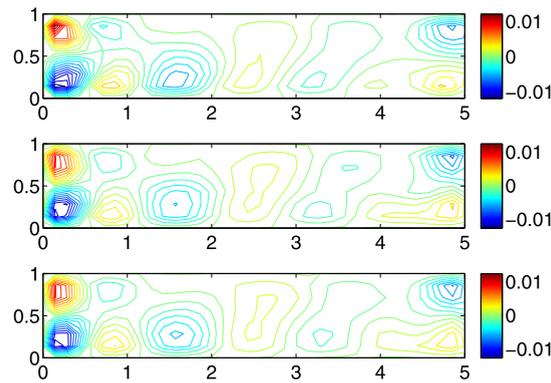


Fig. 5.19. The incompressible channel flow. The contours of the vertical velocity with 10% relative errors in data. Top: true solution; Middle: PCMC solution; Bottom: UMC solution.

where the parameter $\tau > 0$ is introduced to balance the contributions from the two velocity components. This is because in this channel flow the magnitude of the vertical velocity is much smaller than that of the horizontal velocity. We use $\tau = 10$ in all numerical tests. The physical constraint is the incompressibility

$$C(\hat{M}) = \frac{\partial \hat{M}_1}{\partial x_1} + \frac{\partial \hat{M}_2}{\partial x_2} = \frac{\partial \delta_1}{\partial x_1} + \frac{\partial \delta_2}{\partial x_2} = 0. \tag{5.26}$$

Note that since the model M already satisfies the incompressibility, the constraint is effectively on the external correction terms. We enforce this constraint in the PCMC optimization (3.6) at a set of selected points $x_j, j = 1, \dots, n_c$. The points are uniformly distributed in the domain with 40×20 points in each element. This is a dense set of points. They do not cause additional computational burden, as the constraint is evaluated on the correction terms. They can be readily evaluated using the analytical form of the correction terms. The polynomial order of the model corrections is fixed at $n = 12$.

The numerical solutions with 10% relative noise are shown in Figs. 5.18 and 5.19, for the horizontal and vertical velocities, respectively. All contour lines are plotted at the same levels, for the true solution, PCMC solution and UMC solution. We observe that the PCMC produces better results, particularly visible in the horizontal velocity. The results in the vertical velocity are much less clear, primarily because the magnitude of the vertical solution is rather small. This reinforces our early statement that the incorporation of certain physical constraints does not necessarily lead to better predictability. However, the incorporation of the important physical constraints, the incompressibility in this case, is indeed much preferred in practice and leads to more “realistic” result. In this particular case, the incompressibility is measured by the $\|\nabla \cdot u\|_\infty$ norm over a set of dense grids uniformly in the domain. For PCMC, $\|\nabla \cdot u\|_\infty = 1.58 \times 10^{-6}$; while for UMC, $\|\nabla \cdot u\|_\infty = 2.93 \times 10^{-3}$. Results at different noise levels are similar and thus not shown.

6. Summary

In this paper we present a framework of physics constrained model correction (PCMC), for addressing model-form uncertainty. Unlike the existing limited methods for model correction, the PCMC method explicitly incorporates the important physical constraints of the underlying physical problems. A variety of examples are presented in the paper. It is shown

that the PCMC method consistently produces more desirable model outputs, in the sense that its prediction embeds the preferred physical constraints. This suggests that the PCMC can be a viable tool for model-form uncertainty analysis.

Acknowledgement

This work is in part supported by AFOSR (FA95501410022), DARPA (N660011524053), and NSF (1418771).

References

- [1] M.J. Bayarri, J.O. Berger, R. Paulo, J. Sacks, J.A. Cafeo, C.H. Lin, J. Cavendish, J. Tu, A framework for validation of computer models, *Technometrics* 49 (2007) 138–154.
- [2] J. Brynjarsdottir, A. O'Hagan, Learning about physical parameters: the importance of model discrepancy, *Inverse Probl.* 30 (2014) 114007.
- [3] R.G. Ghanem, P. Spanos, *Stochastic Finite Elements: A Spectral Approach*, Springer-Verlag, 1991.
- [4] D.M. Higdon, M.C. Kennedy, J.C. Cavendish, J.A. Cafeo, R.D. Ryne, Combining field data and computer simulations for calibration and prediction, *SIAM J. Sci. Comput.* 26 (2004) 448–466.
- [5] V.R. Joseph, S.N. Melkote, Statistical adjustments to engineering models, *J. Qual. Technol.* 41 (2009) 362–375.
- [6] G.E. Karniadakis, S.J. Sherwin, *Spectral/hp Element Methods for CFD*, Oxford University Press, 1999.
- [7] G.E. Karniadakis, A. Beskok, N. Aluru, *Microflows and Nanoflows: Fundamentals and Simulation*, Springer, 2005.
- [8] M.C. Kennedy, A. O'Hagan, Bayesian calibration of computer models, *J. R. Stat. Soc., Ser. B, Stat. Methodol.* 63 (2001) 425–464.
- [9] A.H. Nayfeh, *Perturbation Methods*, Wiley-Interscience, 2000.
- [10] Nektar++, An efficient h to p finite element framework, <http://www.nektar.info/>.
- [11] Z.G. Qian, C.F. Wu, Bayesian hierarchical modeling for integration low-accuracy and high-accuracy experiments, *Technometrics* 50 (2008) 192–204.
- [12] K. Sargsyan, H.N. Najm, R. Ghanem, On the statistical calibration of physical models, *Int. J. Chem. Kinet.* (2015), <http://dx.doi.org/10.1002/kin.20906>.
- [13] S. Wang, W. Chen, K.L. Tsui, Bayesian validation of computer models, *Technometrics* 51 (2009) 439–451.
- [14] D. Xiu, G.E. Karniadakis, The Wiener–Askey polynomial chaos for stochastic differential equations, *SIAM J. Sci. Comput.* 24 (2) (2002) 619–644.