Chapter 5

Numerical experiments

In this chapter, we present a number of numerical experiments to examine the practical performance of the previously developed algorithms.

5.1 Experimental setup

We first describe the experimental setup.

5.1.1 Hyperparameter values

The algorithms used in the main theorems (see Tables 4.2 and 4.3) are designed to ensure the desired error bounds. In our numerical experiments, we deviate from these values in a number of minor ways. However, our hyperparameter choices are still closely based on theory. We now discuss the precise hyperparameter choices used in the experiments. These choices are also summarized in Table 5.1.

First, we take the parameter λ to be $\lambda = (\sqrt{25m})^{-1}$. This differs somewhat from the value $\lambda = (4\sqrt{m/L})^{-1}$ used in the theoretical algorithms. The rationale behind doing this is that L is, in practice, a polylogarithmic factor that arises from the compressed sensing theory. It is well known that logarithmic factors appearing in compressed sensing theory are generally quite pessimistic [8, 13, 61]. Therefore, we avoid using L. The choice $\lambda = (5\sqrt{m})^{-1}$ was obtained in [8, Appendix A] after manual tuning.

As shown later, the primal-dual iteration converges subject to the condition $\|\boldsymbol{A}\|_2^2 \leq (\tau\sigma)^{-1}$. See Lemma 9.2. Since the error bound (9.2) scales linearly in τ^{-1} and σ^{-1}, a standard choice for these parameters is

$$\tau = \sigma = 1/\|\boldsymbol{A}\|_2. \tag{5.1}$$

In Tables 4.2 and 4.3 we choose

$$\tau = \sigma = (\Theta(n,d))^{-\alpha},$$

since the latter is an upper bound for $\|\boldsymbol{A}\|_2$, i.e., $\|\boldsymbol{A}\|_2 \leq (\Theta(n,d))^{\alpha}$. See (10.9). This bound is arguably quite crude. The reason for using it in our main theorems is to avoid having to compute $\|\boldsymbol{A}\|_2$, since this generally cannot be done in finitely many arithmetic operations. However, in our numerical experiments we simply use (5.1) instead, as it is simpler and $\|\boldsymbol{A}\|_2$ can be efficiently approximated in practice.

Parameter	Value	Notes
λ	$(\sqrt{25m})^{-1}$	Based on [8, Appendix A]
σ	$\\|\boldsymbol{A}\\|_2^{-1}$	Based on Lemma 9.2
τ	$\\|\boldsymbol{A}\\|_2^{-1}$	Based on Lemma 9.2
r	e^{-1}	Based on Theorem 9.4
T	$\left\lceil \frac{2\\|\boldsymbol{A}\\|_2}{r} \right\rceil$	Based on Theorem 9.4, assuming $C = 1$
s	$\frac{T}{2\\|\boldsymbol{A}\\|_2}$	Based on Theorem 9.4

Table 5.1. Hyperparameter values used in the numerical experiments. The first three parameters are used in both the unrestarted and restarted primal-dual iterations. The final three parameters are used in the restarted scheme only.

For the restarting scheme, we also have the scale parameter $0 < r < 1$, the constant $s > 0$ and the number of inner iterations T. These parameters are inferred from Theorem 9.4. This result shows that the error in the restarted primal dual iteration after l restarts is bounded by

$$r^l \|\boldsymbol{b}\|_{2;v} + \frac{r}{1-r}\zeta', \tag{5.2}$$

provided

$$T = \left\lceil \frac{2C}{r\sqrt{\sigma\tau}} \right\rceil, \quad a_l = \frac{1}{2}\sigma\varepsilon_{l+1}T,\ l = 0, 2, \ldots.$$

Here, as discussed in Theorem 9.4, $C > 0$ is a numerical constant that arises from the compressed sensing theory. This and the choice (5.1) leads immediately to the following value for s:

$$s = \frac{T}{2\|\boldsymbol{A}\|_2}.$$

Unfortunately, the constant C is difficult to determine exactly (it is closely related to the constant $c^\star$ discussed in Remark 4.5). In our experiments, we simply pick the value $C = 1$. This immediately yields

$$T = \left\lceil \frac{2\|\boldsymbol{A}\|_2}{r} \right\rceil.$$

Finally, to determine a value of r we consider the error bound (5.2). This argument is based on [48]. After l restarts, the total number of iterations $t = Tl$. Substituting the value of T, we see that

$$r^l = \exp(\log(r)t/T) = \exp\left(\log(r)\left\lceil \frac{2\|\boldsymbol{A}\|_2}{r} \right\rceil^{-1} t\right). \tag{5.3}$$

Ignoring the ceiling function, it therefore makes sense to choose $0 < r < 1$ to minimize the function $r \mapsto r\log(r)$. This attains its minimum value of $-\mathrm{e}^{-1}$ at $r = \mathrm{e}^{-1}$, which is the value we now use.

5.1.2 Test functions

We consider four test functions. The first two are scalar-valued functions, given by

$$f_1(\boldsymbol{y}) = \exp\left(-\frac{1}{2d}\sum_{k=1}^{d} y_k\right), \quad \forall \boldsymbol{y} \in \mathcal{U}, \quad \text{with } d = 2, \tag{5.4}$$

and

$$\begin{gathered} f_2(\boldsymbol{y}) = \exp\left(-\frac{2}{d}\sum_{k=1}^{d}(y_k - w_k)^2\right), \quad \forall \boldsymbol{y} \in \mathcal{U}, \\ \text{with} \quad w_k = \frac{(-1)^k}{k+1}, \quad \forall k \in [d] \text{ and } d = 16. \end{gathered} \tag{5.5}$$

These are standard test functions (see, e.g., [8, Appendix A.1]). The first function varies very little with respect to $\boldsymbol{y}$. Hence, it is expected to be very well approximated by a sparse polynomial approximation. The second has more variation in $\boldsymbol{y}$, therefore we expect a larger approximation error.

We also consider two Hilbert-valued functions. These both arise as solutions of the parametric elliptic diffusion equation

$$\begin{aligned} -\nabla \cdot (a(\boldsymbol{x},\boldsymbol{y})\nabla u(\boldsymbol{x},\boldsymbol{y})) &= g(\boldsymbol{x}), \quad \forall \boldsymbol{x} \in D,\ \boldsymbol{y} \in \mathcal{U}, \\ u(\boldsymbol{x},\boldsymbol{y}) &= 0, \quad \forall \boldsymbol{x} \in \partial D,\ \boldsymbol{y} \in \mathcal{U}, \end{aligned}$$

which is a standard problem in the parametric PDE literature. We take the physical domain D as

$$D = (0,1)^2.$$

For simplicity, we also choose

$$g(\boldsymbol{x}) = 10$$

to be constant. In this case, the solution map

$$\mathcal{U} \to \mathcal{V}, \quad \boldsymbol{y} \mapsto u(\cdot,\boldsymbol{y}), \quad \mathcal{V} = H_0^1(D),$$

is a Hilbert-valued function with codomain being the Sobolev space $H_0^1(D)$. We consider two different setups, leading to smooth and less smooth Hilbert-valued functions, which we denote as f_3 and f_4, respectively. The first is a simple two-dimensional problem with lognormal diffusion coefficient:

$$f_3: \quad d = 2,\ a(\boldsymbol{x},\boldsymbol{y}) = 5 + \exp(x_1 y_1 + x_2 y_2). \tag{5.6}$$

For the second, we consider the diffusion coefficient from [5, equation (24)], modified from an earlier example from [110, equation (5.2)], with 30-dimensional parametric dependence and one-dimensional (layered) spatial dependence given by

$$
\begin{aligned}
&f_4: \; d=30, \; a(\boldsymbol{x},\boldsymbol{y}) = \exp\left(1 + y_1 \left(\frac{\sqrt{\pi}\beta}{2}\right)^{1/2} + \sum_{i=2}^{d} \zeta_i \, \vartheta_i(\boldsymbol{x}) \, y_i\right), \\
&\qquad \zeta_i := (\sqrt{\pi}\beta)^{1/2} \exp\left(\frac{-(\lfloor \frac{i}{2} \rfloor \pi\beta)^2}{8}\right), \quad \vartheta_i(\boldsymbol{x}) := \begin{cases} \sin\left(\lfloor \frac{i}{2} \rfloor \pi x_1/\beta_p\right) & i \text{ even}, \\ \cos\left(\lfloor \frac{i}{2} \rfloor \pi x_1/\beta_p\right) & i \text{ odd}, \end{cases} \\
&\qquad \beta_c = 1/8, \quad \beta_p = \max\{1, 2\beta_c\}, \quad \beta = \beta_c/\beta_p. \qquad (5.7)
\end{aligned}
$$

5.1.3 Error metrics and finite element discretization

In our experiments, we consider the relative $L^2_{\varrho}(\mathcal{U})$-norm error

$$
\frac{\|f - \hat{f}\|_{L^2_{\varrho}(\mathcal{U})}}{\|f\|_{L^2_{\varrho}(\mathcal{U})}}, \qquad (5.8)
$$

for the scalar-valued functions f_1 and f_2 and the relative $L^2_{\varrho}(\mathcal{U}; H^1_0(D))$-norm error

$$
\frac{\|f - \hat{f}\|_{L^2_{\varrho}(\mathcal{U};H^1_0(D))}}{\|f\|_{L^2_{\varrho}(\mathcal{U};H^1_0(D))}}, \qquad (5.9)
$$

for the Hilbert-valued functions f_3 and f_4. To (approximately) compute this error we use a high-order isotropic Smolyak sparse grid quadrature rule based on Clenshaw–Curtis points. This rule is generated using the TASMANIAN software package [129]. We set the level of the quadrature rule in each experiment as large as possible within the constraints of computational time and memory.

We now describe the discretization $\mathcal{V}_h$ for the Hilbert-valued functions f_3 and f_4. This is obtained via the finite element method as implemented by Dolfin [95], and accessed through the python FEniCS project [18]. We generate a regular triangulation $\mathcal{T}_h$ of $\overline{D}$ composed of triangles T of equal diameter $h_T = h$. We consider a conforming discretization, which results in a finite-dimensional subspace $\mathcal{V}_h \subset \mathcal{V} = H^1_0(D)$, where $\mathcal{V}_h$ is the space spanned by the usual Lagrange finite elements $\{\varphi_i\}_{i=1}^{K}$ of order $k = 1$. We rely on the Dolfin `UnitSquareMesh` method to generate a mesh with 33 nodes per side, corresponding to a finite element triangulation with $K = 1089$ nodes, 2048 elements and meshsize $h = \sqrt{2}/32$. See [5, 52] for further implementation details.

Explicit forms of the Hilbert-valued functions f_3 and f_4 are not available. Therefore, computing the relative error requires first computing a reference solution. This is

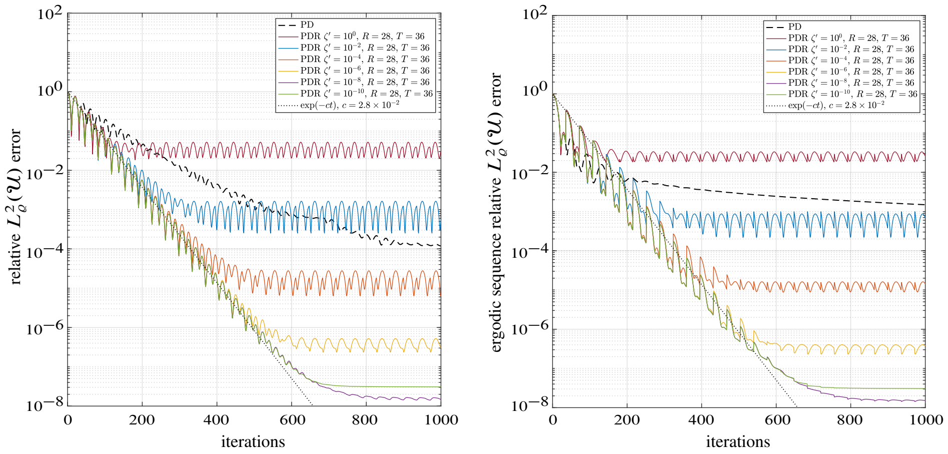

Figure 5.1. Approximation error versus iteration number for the function f_1 from (5.4). This figure shows the relative L^2 errors of the polynomial approximations obtained from (*left*) the iterates $\boldsymbol{c}^{(n)}$ and (*right*) the ergodic sequence $\bar{\boldsymbol{c}}^{(n)}$. These approximations are constructed using the Legendre polynomial basis and $m = 250$ sample points drawn randomly and independently from the uniform measure. The index set $\Lambda = \Lambda^{\mathsf{HC}}_{n,d}$, where $d = 2$ and $n = 184$, which gives a basis of cardinality $N = |\Lambda| = 997$. We compare the primal dual iteration "PD" and the restarted primal dual iteration "PDR" for various values of the tolerance ζ'. We also plot the theoretical error curve (5.10), where t is the iteration number. The quadrature rule used to compute the relative error is a sparse grid rule of level 11 consisting of $M = 7169$ points.

usually done by using a finite element discretization with meshsize an order of magnitude smaller than that used to compute the various approximations. However, our main focus in these experiments is on the polynomial approximation and algorithmic errors E_{app} and E_{alg}. Since our theoretical results assert that the approximations are robust to physical discretization error, we do not perform this additional (and costly) computational step. Instead, we compute reference solutions using the same finite element discretization as that used to construct the various approximations. In other words, there is no physical discretization error present in these experiments.

5.2 Numerical results 1: The optimization error

Our first experiments, Figures 5.1–5.4, compare the behaviour of the unrestarted primal-dual iteration to the restarted primal-dual iteration with several different values of the tolerance parameter ζ'. In all cases, we observe a consistent improvement from the restarted scheme. This is particularly noticeable for the functions f_1 and f_3, since the underlying approximation error ζ is smaller in these cases. Recall that these functions are well approximated by polynomials. As predicted by our theoretical results,

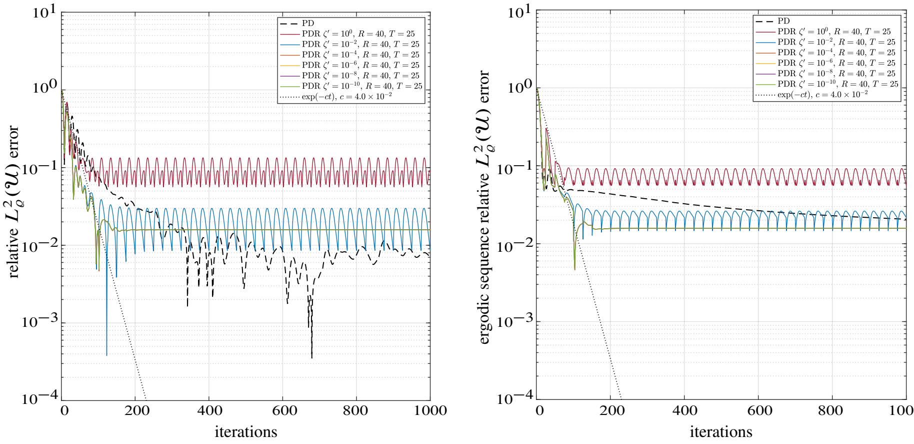

Figure 5.2. Approximation error versus iteration number for the function f_2 from (5.5). This figure shows the relative L^2 errors of the polynomial approximations obtained from (*left*) the iterates $\boldsymbol{c}^{(n)}$ and (*right*) the ergodic sequence $\bar{\boldsymbol{c}}^{(n)}$. These approximations are constructed using the Legendre polynomial basis and $m = 2000$ sample points drawn randomly and independently from the uniform measure. The index set $\Lambda = \Lambda^{\mathsf{HC}}_{n,d}$, where $d = 16$ and $n = 16$, which gives a basis of cardinality $N = |\Lambda| = 8277$. We compare the primal dual iteration "PD" and the restarted primal dual iteration "PDR" for various values of the tolerance ζ'. We also plot the theoretical error curve (5.10), where t is the iteration number. The quadrature rule used to compute the relative error is a sparse grid rule of level 5 consisting of $M = 51137$ points.

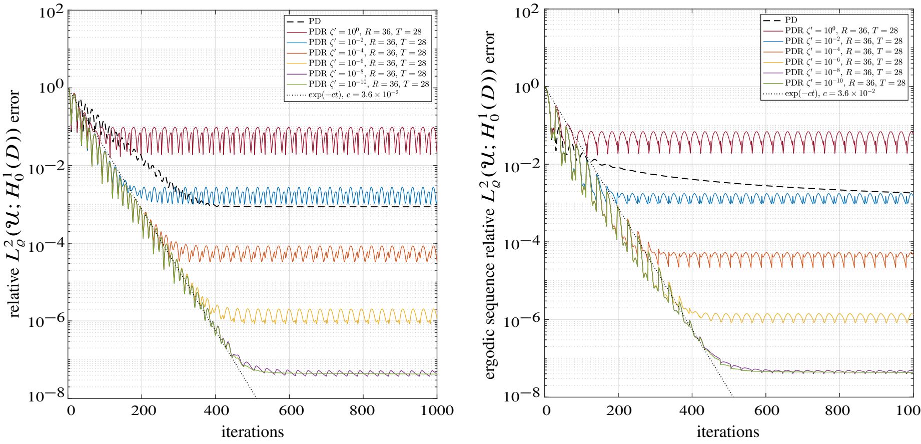

Figure 5.3. Approximation error versus iteration number for the function f_3 from (5.6). This figure shows the relative L^2 errors of the polynomial approximations obtained from (*left*) the iterates $\boldsymbol{c}^{(n)}$ and (*right*) the ergodic sequence $\bar{\boldsymbol{c}}^{(n)}$. These approximations are constructed using the Legendre polynomial basis and $m = 250$ sample points drawn randomly and independently from the uniform measure. The index set $\Lambda = \Lambda^{\mathsf{HC}}_{n,d}$, where $d = 2$ and $n = 184$, which gives a basis of cardinality $N = |\Lambda| = 997$. We compare the primal dual iteration "PD" and the restarted primal dual iteration "PDR" for various values of the tolerance ζ'. We also plot the theoretical error curve (5.10), where t is the iteration number. The quadrature rule used to compute the relative error is a sparse grid rule of level 9 consisting of $M = 1537$ points.

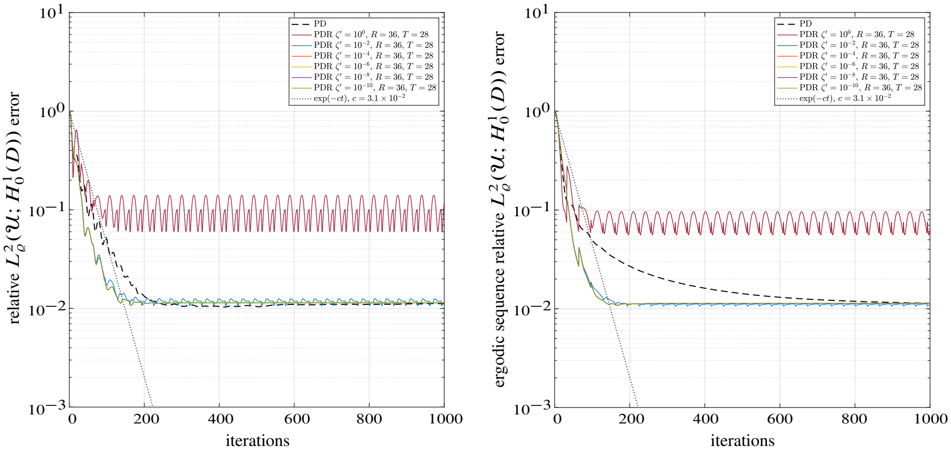

Figure 5.4. Approximation error versus iteration number for the function f_4 from (5.7). This figure shows the relative L^2 errors of the polynomial approximations obtained from (*left*) the iterates $\boldsymbol{c}^{(n)}$ and (*right*) the ergodic sequence $\bar{\boldsymbol{c}}^{(n)}$. These approximations are constructed using the Legendre polynomial basis and $m = 1000$ sample points drawn randomly and independently from the uniform measure. The index set $\Lambda = \Lambda^{\mathsf{HC}}_{n,d}$, where $d = 30$ and $n = 10$, which gives a basis of cardinality $N = |\Lambda| = 7841$. We compare the primal dual iteration "PD" and the restarted primal dual iteration "PDR" for various values of the tolerance ζ'. We also plot the theoretical error curve (5.10), where t is the iteration number. The quadrature rule used to compute the relative error is a sparse grid rule of level 3 consisting of $M = 1861$ points.

the error for the restarted scheme decays exponentially fast with respect to the number of iterations to this limiting accuracy. For example, in the case of f_1 the restarted scheme (with sufficiently small ζ') achieves a relative error of less than 10^{-6} using only 500 iterations. However, the unrestarted scheme only achieves an error of around 10^{-3} after 1000 iterations.

An important takeaway from these experiments is the insensitivity of the algorithm to the parameter ζ'. Our theoretical results only show exponential convergence (with respect to iteration number) when $\zeta' \geq \zeta$, where ζ is a certain upper bound for the error. This appears unnecessary in practice. For instance, in Figures 5.2 and 5.4 we expect the underlying error ζ to be roughly 10^{-2} in magnitude, since this is the limiting error achieved by the unrestarted scheme. Yet setting $\zeta' = 10^{-10}$ has no noticeable effect on the performance of the restarted scheme. Moreover, for $\zeta' \in \{10^{-4}, 10^{-6}, 10^{-8}, 10^{-10}\}$ the results are nearly identical in both Figures 5.2 and 5.4, and hence the plot lines are overlaid in the case of the restarted scheme.

Another noticeable feature of these experiments is the close agreement between the theorized rate of exponential decay of the restarted scheme, which is given by the right-hand side of (5.3) and what is observed in practice. Since the value $r = \mathrm{e}^{-1}$ is

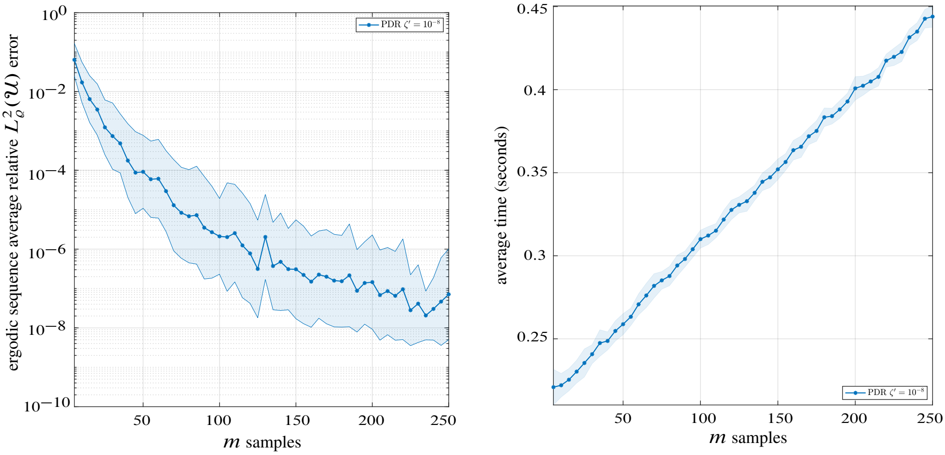

Figure 5.5. (*left*) Approximation error and (*right*) average run time versus number of samples m for the function f_1 from (5.4). This figure shows the relative L^2 errors of the polynomial approximations obtained from the ergodic sequence $\bar{\boldsymbol{c}}^{(n)}$. These approximations are constructed using the Legendre polynomial basis and various sets of m sample points drawn randomly and independently from the uniform measure for each trial. The index set $\Lambda = \Lambda^{\mathsf{HC}}_{n,d}$, where $d = 2$ and $n = 184$, which gives a basis of cardinality $N = |\Lambda| = 997$. We use the restarted primal dual iteration "PDR" with $\zeta' = 10^{-8}$, and display the average error over 50 trials measured in the sample mean in blue and the corrected sample standard deviation after a log transformation in shaded blue, see [8, Appendix A.1.3] for more details. The quadrature rule used to compute the relative error is a sparse grid rule of level 11 consisting of $M = 7169$ points.

used in these experiments, in Figures 5.1–5.4 we also plot the function

$$\exp(-ct), \quad c := \lceil 2\mathrm{e}\|\boldsymbol{A}\|_2 \rceil^{-1} \tag{5.10}$$

versus the iteration number t. This theoretical curve exactly predicts the observed rate of exponential decay of the restarted schemes.

Finally, in all four figures we also show the error of the (restarted) primal-dual iterates, as well as the ergodic sequences. Despite the theoretical results only holding for the latter, we see similar error decay for the iterates. In fact, the iterates give slightly better performance in the case of the unrestarted scheme. As expected, the ergodic sequence reduces the variation in the error for the restarted scheme. Moreover, plotting the ergodic sequence we can see more clearly the benefit of using restarts over not restarting.

5.3 Numerical results 2: Approximation error and run time

In the second set of experiments, our aim is to study the approximation error versus the number of samples m. Having compared different solvers in the previous

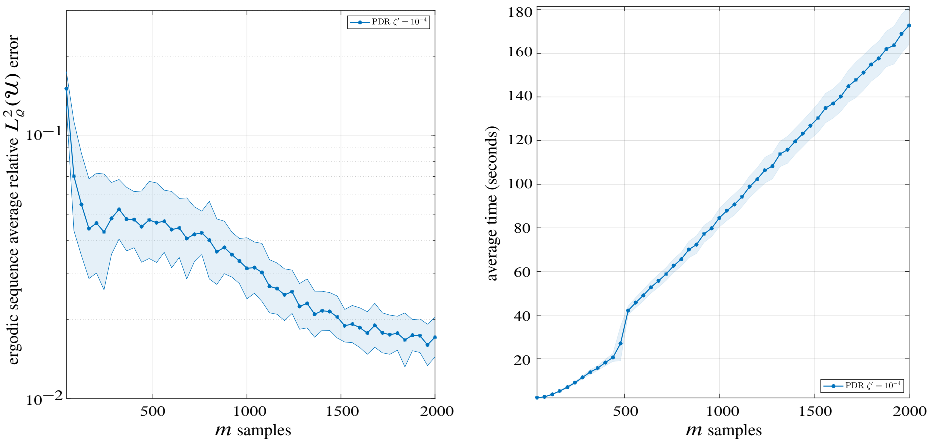

Figure 5.6. (*left*) Approximation error and (*right*) average run time versus number of samples m for the function f_2 from (5.5). This figure shows the relative L^2 errors of the polynomial approximations obtained from the ergodic sequence $\bar{\boldsymbol{c}}^{(n)}$. These approximations are constructed using the Legendre polynomial basis and various sets of m sample points drawn randomly and independently from the uniform measure for each trial. The index set $\Lambda = \Lambda^{\mathsf{HC}}_{n,d}$, where $d = 16$ and $n = 16$, which gives a basis of cardinality $N = |\Lambda| = 8277$. We use the restarted primal dual iteration “PDR” with $\zeta' = 10^{-4}$, and display the average error over 50 trials measured in the sample mean in blue and the corrected sample standard deviation after a log transformation in shaded blue, see [8, Appendix A.1.3] for more details. The quadrature rule used to compute the relative error is a sparse grid rule of level 5 consisting of $M = 51137$ points.

experiments, we now limit our attention to the restarted primal-dual iteration. The only modification we make is to introduce a stopping criterion for the number of restarts. Specifically, given a tolerance ζ', we halt the iteration if the difference between two consecutive iterates is less than $5 \cdot \zeta'$. Specifically, if

$$\|\tilde{\boldsymbol{c}}^{(l)} - \tilde{\boldsymbol{c}}^{(l-1)}\|_2 \leq 5 \cdot \zeta',$$

in the scalar-valued case or

$$\|\tilde{\boldsymbol{c}}^{(l)} - \tilde{\boldsymbol{c}}^{(l-1)}\|_{2;\mathcal{V}} \leq 5 \cdot \zeta',$$

in the Hilbert-valued case, where $\tilde{\boldsymbol{c}}^{(l)}$ is the output of the restarted primal-dual iteration after l restarts, then we halt and take $\tilde{\boldsymbol{c}}^{(l)}$ as the polynomial coefficients of the resulting approximation.

In the following experiments, we perform multiple trials for each value of m. For each trial, we generate a set of sample Monte Carlo points $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m$, then compute the relative error (5.8) or (5.9) of the approximation using a sparse grid quadrature as before. Having done this, we then compute the sample mean and (corrected) sample

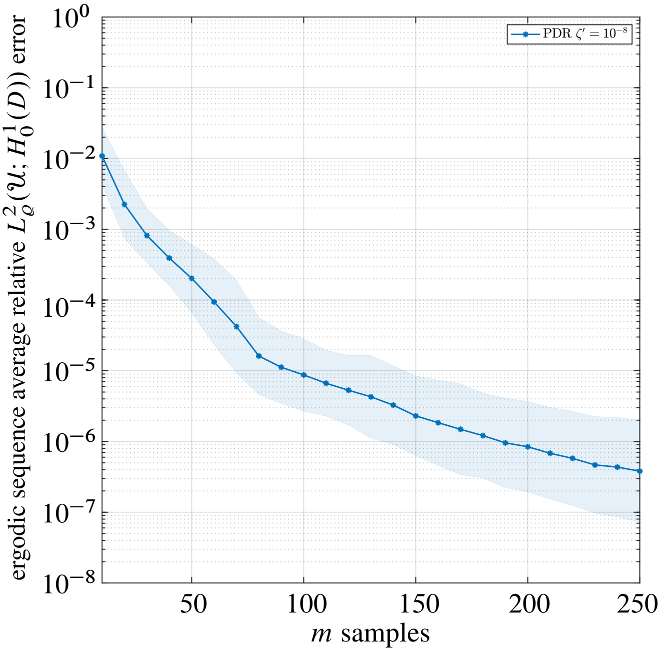

Figure 5.7. Approximation error versus number of samples m for the function f_3 from (5.6). This figure shows the relative L^2 errors of the polynomial approximations obtained from the ergodic sequence $\bar{c}^{(n)}$. These approximations are constructed using the Legendre polynomial basis and various sets of m sample points drawn randomly and independently from the uniform measure for each trial. The index set $\Lambda = \Lambda^{\mathsf{HC}}_{n,d}$, where $d = 2$ and $n = 184$, which gives a basis of cardinality $N = |\Lambda| = 997$. We compare the restarted primal dual iteration "PDR" with $\zeta' = 10^{-8}$ with the average performance over 50 trials measured in the sample mean in blue and the corrected sample standard deviation after a log transformation in shaded blue, see [8, Appendix A.1.3] for more details. The quadrature rule used to compute the relative error is a sparse grid rule of level 11 consisting of $M = 7169$ points.

standard deviation after a log transformation. See [8, Appendix A.1.3] for further discussion and rationale behind this computation.

The results for the four functions f_1, f_2, f_3, f_4 are shown in Figures 5.5–5.8. Figure 5.5 shows the average approximation error and run times for f_1. As discussed, this function is expected to be well approximated by polynomials. In accordance, the error decreases rapidly, achieving roughly 10^{-7} relative L^2 error when $m \approx 200$. This is in broad agreement with the exponential decay rate of the error shown in our main theorems. In Figure 5.6 we consider the more challenging, higher-dimensional function f_2, plotting the average approximation error and run time. Here, as expected, the error decreases significantly more slowly. Both figures exhibit a linear scaling of the run time with the number of samples m. This is consistent with our analysis, since

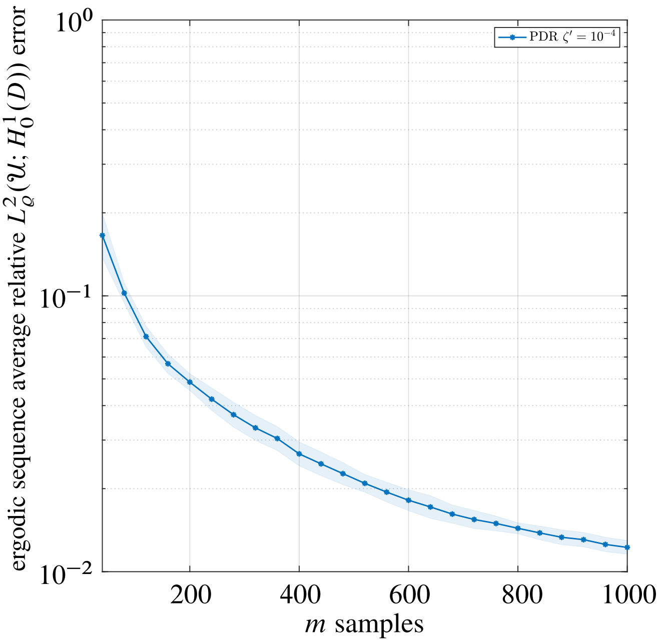

Figure 5.8. Approximation error versus number of samples m for the function f_4 from (5.7). This figure shows the relative L^2 errors of the polynomial approximations obtained from the ergodic sequence $\bar{c}^{(n)}$. These approximations are constructed using the Legendre polynomial basis and various sets of m sample points drawn randomly and independently from the uniform measure for each trial. The index set $\Lambda = \Lambda^{\mathsf{HC}}_{n,d}$, where $d = 30$ and $n = 10$, which gives a basis of cardinality $N = |\Lambda| = 7841$. We compare the restarted primal dual iteration “PDR” with $\zeta' = 10^{-4}$ with the average performance over 50 trials measured in the sample mean in blue and the corrected sample standard deviation after a log transformation in shaded blue, see [8, Appendix A.1.3] for more details. The quadrature rule used to compute the relative error is a sparse grid rule of level 3 consisting of $M = 1861$ points.

each algorithm iteration involves dense matrix-vector multiplications with an $m \times N$ matrix. Also, comparing Figures 5.5 and 5.6 when $m = 250$, we notice the run time is roughly 16 times larger for the latter. This is also in agreement with our analysis. Indeed, $N \approx 1000$ in Figure 5.5 while $N \approx 8000$ in Figure 5.6. However, the number of inner iterations $T = \lceil 2\|\boldsymbol{A}\|_2/r \rceil$ is roughly twice as large in Figure 5.6, where $\|\boldsymbol{A}\|_2 \approx 13$ when $m = 250$, as it is in Figure 5.5, where $\|\boldsymbol{A}\|_2 \approx 7$. The combination of these two factors accounts for the roughly 16-fold increase in run time.

Figure 5.7 displays the performance of the restarted scheme on the Hilbert-valued function f_3. Here we also observe rapid decrease in the error with respect to increasing number of samples m, with relative L^2 error approximately 10^{-6} when $m \approx 200$. Finally, Figure 5.8 shows the results for the less smooth high-dimensional Hilbert-valued function f_4. For this function, we expect slower decrease in the error with respect to m, which is reflected in this set of results. Nonetheless, despite its high dimensionality ($d = 30$) we still achieve two digits of relative accuracy using only $m \approx 1000$ samples.