Universität Stuttgart

SimTech
Cluster of Excellence

D. Wirtz and B. Haasdonk

A-posteriori error estimation for parameterized kernel-based systems

Stuttgart, June 2012

Institute of Applied Analysis and Numerical Simulation,
University of Stuttgart,
Pfaffenwaldring 57
D-70569 Stuttgart, Germany
{daniel.wirtz,bernard.haasdonk}@mathematik.uni-stuttgart.de
www.agh.ians.uni-stuttgart.de

Abstract This work is concerned with derivation of fully offline/online decomposable efficient a-posteriori error estimators for reduced parameterized nonlinear kernel-based systems. The dynamical systems under consideration consist of a nonlinear, time- and parameter-dependent kernel expansion representing the system's inner dynamics as well as time- and parameter-affine inputs, initial conditions and outputs. The estimators are established for a reduction technique originally proposed in [7] and are an extension of the estimators derived in [11] to the fully time-dependent, parameterized setting. Key features for the efficient error estimation are to use local Lipschitz constants provided by a certain class of kernels and an iterative scheme to balance computation cost against estimation sharpness. Together with the affinely time/parameter-dependent system components a full offline/online decomposition for both the reduction process and the error estimators is possible. Some experimental results for synthetic systems illustrate the efficient evaluation of the derived error estimators for different parameters.

Keywords nonlinear parameterized dynamical systems, model reduction, kernel methods, a-posteriori error estimates, offline/online decomposition, subspace projection

Preprint Series
Stuttgart Research Centre for Simulation Technology (SRC SimTech)

SimTech – Cluster of Excellence
Pfaffenwaldring 7a
70569 Stuttgart

publications@simtech.uni-stuttgart.de
www.simtech.uni-stuttgart.de

1 Introduction

Many real world processes naturally lead to models formulated as dynamical systems with inputs and outputs. More often than not, those models come with a variety of different parameters comprising known configuration ranges as well as (optimal) values yet to be determined. In many of those settings the models are embedded into a many-query or real-time context, e.g. optimization or control, which in turn requires fast and repeated evaluation of the model for different parameter configurations. Even though computational power has significantly increased over the last years, high-resolution models still need reduction techniques to make such higher-level applications feasible. Within all those settings fast and rigorous error estimation procedures are important in order to quantify the model errors introduced by the reduction process.

In this work we will introduce the considered class of dynamical systems, outline the reduction technique, derive the error estimators and their offline/online decomposition and present some experimental results.

2 Model reduction technique

The basics of the reduction technique that we consider have been proposed in [7] for non-parameterized nonlinear systems, which we extend to the case of parameterized time-dependent nonlinear systems. The class of dynamical systems we consider is given by

$$x'(t) = f(x(t), t, \mu) + B(t, \mu)u(t), \tag{1}$$

$$x(0) = x_0(\mu), \tag{2}$$

$$y(t) = C(t, \mu)x(t), \tag{3}$$

where $x(t) \in \mathbb{R}^d$ denotes the system's state and $y(t) \in \mathbb{R}^k$ the output for $t \in [0, T]$. In extension to [11] we have a parameter domain $\mathcal{P} \subseteq \mathbb{R}^p$ and parameters $\mu \in \mathcal{P}$. The input/control function $u : [0, T] \to \mathbb{R}^m$ is mapped into the state space by an time- and parameter-affine matrix

$$B(t, \mu) = \sum_{i=1}^{Q_B} \theta_i^B(t, \mu) B_i, \tag{4}$$

with $Q_B \in \mathbb{N}$ small, constant matrices $B_i \in \mathbb{R}^{d\times m}$ and low-complexity coefficient functions $\theta_i^B : [0, T] \times \mathcal{P} \to \mathbb{R}$. Further we have initial states $x_0(\mu) \in \mathbb{R}^d$ and an output mapping matrix $C(t, \mu) \in \mathbb{R}^{k\times d}$ for which the affine dependencies hold similarly. This affine time/parameter dependence of B, C and x_0 is crucial for the computational efficiency, as then the reduction process can be decomposed into an offline/online phase, see [4]; we refer to [2] for methods on how to obtain approximations in such form for more general components.

One key feature of the reduction method is to find a *kernel expansion* representation/approximation of the system's nonlinearity f. As the approximation process and methods are out of the scope of our article, we assume that the system's nonlinearity is in fact already represented by a kernel expansion of the form

$$f(x, t, \mu) = \sum_{i=1}^{N} c_i \Phi_s(x, x_i) \Phi_t(t, t_i) \Phi_{\mathcal{P}}(\mu, \mu_i), \tag{5}$$

with scalar state, time and parameter *kernels* $\Phi_s, \Phi_t, \Phi_{\mathcal{P}}$, expansion centers $x_i \in \mathbb{R}^d,\ t_i \in [0, T],\ \mu_i \in \mathcal{P}$, and coefficient vectors $c_i \in \mathbb{R}^d, i = 1 \dots N$. In our situation the expansion center triples (x_i, t_i, μ_i) are usually chosen to be samples of some full system trajectory $x(t)$ for a $t \in [0, T]$ and $\mu \in \mathcal{P}$. Note here that we will neglect the dependency of x on t when it is not of importance in the given context. A *kernel* is basically a symmetric function $\Phi : \mathbb{R}^d \times \mathbb{R}^d \longrightarrow \mathbb{R}$, and under some additional assumptions $\{\Phi(x, \cdot)\}_{x\in X\subseteq\mathbb{R}^d}$ spans a unique Reproducing Kernel Hilbert Space (RKHS). As simple algebraic combinations of kernels again result in kernels, our base assumption could also be seen as requiring our original system nonlinearity to be a member of the RKHS induced by $\Phi_s\Phi_t\Phi_{\mathcal{P}}$. For more details on kernels and approximation techniques see [8–10], for example.

Now, for given biorthogonal matrices $V, W \in \mathbb{R}^{d\times r}, V^tW = I_r$ (r-dimensional identity matrix), the second part of the reduction process is to apply a Galerkin projection to the full system (1) in order to obtain the reduced model with $r \ll d$, whose state we shall denote by $z(t) \in \mathbb{R}^r$:

$$\begin{aligned} z'(t) &= \sum_{i=1}^{N} \tilde{c}_i \Phi_s(Vz(t), x_i)\Phi_t(t, t_i)\Phi_{\mathcal{P}}(\mu, \mu_i) && (6)\\ &\quad + \sum_{i=1}^{Q_B} \theta_i^B(t,\mu)\tilde{B}_i u(t) && (7)\\ z(0) &= \sum_{i=1}^{Q_{x_0}} \theta_i^{x_0}(\mu)\tilde{x}_i^0 =: z_0(\mu), && (8)\\ y^r(t) &= \sum_{i=1}^{Q_C} \theta_i^C(t,\mu)\tilde{C}_i z(t), && (9) \end{aligned}$$

with reduced quantities $\tilde{c}_i := W^t c_i \in \mathbb{R}^r, \tilde{B}_i := W^t B_i \in \mathbb{R}^{r\times m}, \tilde{C}_i := C_i V \in \mathbb{R}^{k\times r}, \tilde{x}_i^0 := W^t x_i^0 \in \mathbb{R}^r$. As the choice of V, W is independent from the reduction process, their computation method will be treated as a black-box. However, proper orthogonal decomposition (POD)-related methods like the POD-Greedy are the most widely applied, see e.g. [1, 3, 6] and references therein.

Before we discuss the motivation why to use such kernel representations, we introduce a more general scalar product and norm for the state space as in some applications they are different from the standard Euclidean case. Thus, let $G \in \mathbb{R}^{d\times d}$ be a symmetric positive definite matrix. Then G defines a scalar product $\langle x, y\rangle_G := x^t G y$ on $\mathbb{R}^d$ with induced norm $||x||_G := \sqrt{\langle x, x\rangle_G}$. Choosing $G := I_d$ will yield the standard Euclidean inner product $\langle\cdot,\cdot\rangle$ and 2-norm $||\cdot||$.

Now, the key advantage of having kernel expansion representations (with a certain class of kernels) is that the projection of (5) yields an expansion whose evaluation has a complexity independent of d. For systems without this structure the argument $Vz(t)$ and output of f is still high-dimensional and thus no effective reduction will be achieved. First, the transformation $\tilde{c}_i := W^t c_i \in \mathbb{R}^r$ as shown above is straightforward and yields an expansion of output dimension r. Second, the evaluation of $\Phi_s(Vz(t), x_i)$ can also be performed *loss-less* with a complexity of $\mathcal{O}(r)$ for certain classes of kernels. In extension to the scalar product kernels suggested in [7] we proposed in [11] to use *translation- and rotation invariant* kernels which are also known as radial basis functions. Those kernels have the structure

$$\Phi(x, y) := \phi(||x - y||_G)$$

for some scalar function $\phi : \mathbb{R}_0^+ \to \mathbb{R}$, of which the Gaussian kernel

$$\Phi(x, y) = e^{-\frac{||x-y||^2}{\gamma^2}}$$

for $\gamma > 0$ is probably the most popular example. Let $U := \langle v_1, \ldots, v_r\rangle \subset \mathbb{R}^d$ be the space spanned by the columns $v_i \in \mathbb{R}^d$ of V. With the additional requirement

$$x_i \in U, \text{ i.e. } x_i = Vz_i, z_i \in \mathbb{R}^r,\ i = 1 \ldots N. \tag{10}$$

we obtain

$$\begin{aligned} \Phi_s(Vz, x_i) &= \phi(||Vz - Vz_i||_G) && (11)\\ &= \phi(||z - z_i||_{V^tGV}) =: \Phi_s^r(z, z_i) && (12) \end{aligned}$$

with V^tGV being a small $\mathbb{R}^{r\times r}$ matrix inducing a new norm on $\mathbb{R}^r$. Now with this type of kernel the reduced system (6)-(9) can be solved with a complexity completely *independent* of d. Note that the assumption $x_i \in U$ is only of a technical nature. We either extend U by the span of the x_i, or, if the kernel expansion is created with knowledge of U, one can choose $x_i \in U$ in the first place.

On the downside, however, we must pay for the increased flexibility of having the nonlinearity structure (5) and fast reduced models by an increased number of choices that have to be made. The kernels and kernel hyperparameters like the Gaussian γ must be chosen as well as the center triples/sample points (x_i, t_i, μ_i), of which we would like as few as possible as the cost of the algorithm is directly related to the number of centers N.

3 A-posteriori error estimation

When dealing with reduced dynamical systems it is essential to have means to compute or at least estimate the error that is made during simulations. Extending the results from our previous work [11], we derive rigorous and efficient error estimators for the time- and parameter-dependent reduced system (6)-(9).

For the following we assume the full and reduced system's solutions to have been obtained using a $\mu \in \mathcal{P}$. For simplicity of notation we suppress the μ-dependencies where clear from context and otherwise indicate it by an extra semicolon-separated μ argument. Let $x^r(t) := Vz(t),\ t \in [0,T]$ be the reconstructed state space variable. Then the state space error $e(t;\mu) := x(t) - x^r(t)$ is given by the solution of the error system

$$\begin{aligned} e'(t) &= f(x(t),t,\mu) - VW^t f(x^r(t),t,\mu) \\ &\quad + \left(I_d - VW^t\right) B(t,\mu)u(t), && (13) \\ e(0) &= x_0(\mu) - VW^t x_0(\mu). && (14) \end{aligned}$$

Let $E_{x_0}(\mu) := ||(I_d - VW^t)x_0(\mu)||_G$ be the initial error for the remainder of this work.

3.1 Global Lipschitz constant error estimations

Estimating the norm of the derivative and application of the comparison lemma [5, p.32] yields a first a-posteriori error estimator when assuming Lipschitz continuity of the state space kernel.

Theorem 1 (Global Lipschitz Estimator) *Let f be a kernel expansion as in (5). Further, let the state space kernel Φ_s be L-Lipschitz continuous with respect to the first variable. Then $\forall\ \mu \in \mathcal{P}$ the state space error is bounded via*

$$||e(t;\mu)||_G \leq \Delta_{GLE}(t,\mu) \quad \forall\ t \in [0,T],$$

with

$$\Delta_{GLE}(t,\mu) := \int\limits_0^t \alpha(s,\mu) \exp\left(\int\limits_s^t \beta(r,\mu)dr\right) ds + \exp\left(\int\limits_0^t \beta(r,\mu)dr\right) E_{x_0}(\mu)$$

and

$$\begin{aligned} \alpha(t,\mu) &:= \left|\left|\left(I_d - VW^t\right)\left(f(x^r(t),t,\mu) + B(t,\mu)u(t)\right)\right|\right|_G, \\ \beta(t,\mu) &:= L \sum_{i=1}^N ||c_i||_G \, |\Phi_t(t,t_i)\Phi_{\mathcal{P}}(\mu,\mu_i)|. \end{aligned}$$

Proof (Proof) With (5) we directly obtain

$$\begin{aligned} &||f(x(t)) - f(x^r(t))||_G \\ &= \left|\left|\sum_{i=1}^N c_i\big(\Phi(x(t),x_i) - \Phi(x^r(t),x_i)\big)\Phi_t(t,t_i)\Phi_{\mathcal{P}}(\mu,\mu_i)\right|\right|_G \\ &\leq \sum_{i=1}^N ||c_i||_G \left|\Phi(x(t),x_i) - \Phi(x^r(t),x_i)\right| |\Phi_t(t,t_i)\Phi_{\mathcal{P}}(\mu,\mu_i)| \\ &\leq L \sum_{i=1}^N ||c_i||_G \, |\Phi_t(t,t_i)\Phi_{\mathcal{P}}(\mu,\mu_i)| \, ||e(t)||_G \\ &= \beta(t,\mu) \, ||e(t;\mu)||_G \,. \end{aligned}$$

By inserting $-f(x^r(t)) + f(x^r(t))$ we get from (13):

$$\begin{aligned}||e'(t;\mu)||_G &= \Big|\Big|f(x(t)) - f(x^r(t)) + \big(I_d - VW^t\big)f(x^r(t) \\ &\quad + \big(I_d - VW^t\big)B(t,\mu)u(t)\Big|\Big|_G \\ &\leq \beta(t,\mu)\,||e(t;\mu)||_G + \alpha(t,\mu).\end{aligned}$$

Finally, application of the comparison lemma for the ODE $v'(t) = \beta(t,\mu)v(t) + \alpha(t,\mu), v(0) = E_{x_0}(\mu)$ yields the error estimator as analytical solution. □

Note that the error estimator $\varDelta_{GLE}(t,\mu)$ is rigorous, continuous and monotonically increasing. Once a bound for the state error is available the output error $e_y(t;\mu) := y(t) - y^r(t)$ can be bounded by

$$||e_y(t;\mu)|| \leq C_o(\mu)\,||e(t;\mu)||_G \leq C_o(\mu)\varDelta_{GLE}(t,\mu)$$

where

$$C_o(\mu) := \sup_{t\in[0,T]} ||C(t,\mu)||_G\,. \tag{15}$$

with the G-induced matrix norm

$$||C(t,\mu)||_G = \sup_{x\in\mathbb{R}^d} \frac{||C(t,\mu)x||_G}{||x||_G}.$$

3.2 Local Lipschitz constants

Closer investigation of the error estimator from Theorem 1 shows that if the kernel Lipschitz constant L is too large, the estimator will grow fast and the estimation will be overly conservative.

In our preceding work we proposed a *local Lipschitz constant* computation that utilizes the current position $x^r(t)$ of the reduced state variable and used an a-priori error bound $||e(t)||_G \leq \varTheta(t)$ to improve the estimation procedure. The benefit of knowing $x^r(t)$ is that one argument of the estimation $||f(x(t)) - f(x^r(t))||_G \leq \beta(t,\mu)\,||x(t) - x^r(t)||_G$ can be seen as fixed, which can be used to compute a local Lipschitz constant using secant gradients. Also, the a-priori bound $\varTheta(t)$ enables efficient computation of those secant gradients utilizing the kernel expansion center positions.

As the derivation of those estimations is independent from the parameter and time, we will not state the whole derivation here and refer the reader to our previous work. Instead, we define the necessary ingredients and state our main estimation result for the parameter- and time-dependent case.

The key to our local Lipschitz constant computations is to use radial basis functions $\varPhi(x,y) := \phi(||x-y||),\ x,y\in\mathbb{R}^d$ which are induced by a certain class of scalar functions ϕ:

Definition 1 (Bell functions) A function $\phi : \mathbb{R}_0^+ \to \mathbb{R}_0^+$ is called a *bell function*, if

$$\begin{aligned} i)&\quad \phi \in C^2(\mathbb{R}_0^+), \\ ii)&\quad ||\phi||_{L^\infty} \leq B,\ B > 0, &(16)\\ iii)&\quad \phi'(0) \leq 0, &(17)\\ iv)&\quad \exists\ r_0 > 0 :\ \phi''(r)(r-r_0) > 0\ \forall\ r \neq r_0. &(18)\end{aligned}$$

The Gaussian is induced by a bell function $\phi(r) = e^{-r^2/\gamma^2}$, for example. Additionally, we assume to have an a-priori coarse error bound

$$||e(t;\mu)||_G \leq \varTheta(t,\mu)\ \forall\ t\in[0,T], \tag{19}$$

which can be $\varTheta(t,\mu) = \infty$ if no further knowledge is available. We introduce the notation

$$d_i(t) := ||x^r(t) - x_i||\,, \quad i = 1\ldots N, \tag{20}$$

for the distance of the reduced state $x^r(t)$ to the i-th expansion center x_i during the reduced simulation. With those prerequisites local secant gradients are computable efficiently, yielding the following a-posteriori error estimator:

Theorem 2 (Local Secant Lipschitz Error estimator) *Let f be a kernel expansion as in* (5) *with $\Phi_s(x, y) = \phi(||x - y||)$ for a bell function ϕ. Further let $||e(t;\mu)|| \leq \Theta(t, \mu)\ \forall\ t, \mu$. Then $\forall\ \mu \in \mathcal{P}$ the state space error is bounded via*

$$||e(t;\mu)||_G \leq \Delta_{LSLE}(t, \mu) \quad \forall\ t \in [0, T],$$

with

$$\Delta_{LSLE}(t, \mu) := \int\limits_0^t \alpha(s, \mu) \exp\left(\int\limits_s^t \beta(r, \mu) dr \right) ds + \exp\left(\int\limits_0^t \beta(r, \mu) dr \right) E_{x_0}(\mu)$$

and

$$\begin{aligned} \alpha(t, \mu) &:= \left|\left| \left(I_d - VW^t\right)\left(f(x^r(t), t, \mu) + B(t, \mu)u(t)\right) \right|\right|_G, \\ \beta(t, \mu) &:= \sum_{i=1}^N L_\Theta(d_i(t))\, ||c_i||_G\, |\Phi_t(t, t_i)\Phi_{\mathcal{P}}(\mu, \mu_i)|, \end{aligned} \tag{21}$$

$$\begin{aligned} L_\Theta(d_i(t)) &:= \left| \frac{\phi(d_i(t)) - \phi(r_{\Theta,i})}{d_i(t) - r_{\Theta,i}} \right|, \\ r_{\Theta,i} &:= \begin{cases} d_i(t) + \operatorname{sign}(r_i - d_i(t))\, \Theta(t, \mu) & r_i \notin \Omega(t, \mu) \\ r_i & r_i \in \Omega(t, \mu), \end{cases} \\ r_i &:= \arg \min_{r \in \mathbb{R}_0^+} \frac{\phi(d_i(t)) - \phi(r)}{d_i(t) - r}, \end{aligned} \tag{22}$$

with $\Omega(t, \mu) := B_{\Theta(t,\mu)}(d_i(t)) \cap \mathbb{R}_0^+$.

Proof (Proof) The argumentation is identical to the case of Theorem 1, the only difference is the now implicitly time-dependent local Lipschitz constant. For the derivation of $L_\Theta(d_i(t))$ and the well-definedness of (22) we refer the reader to [11]. □

To illustrate the principle of the local Lipschitz constant estimations of Theorem 2, Figure 1 shows an example for minimum secant gradients for some r_i, using the Gaussian inducing bell function $\phi(r) = \exp(-r^2/\gamma^2)$ with $\gamma = 2$. The a-priori bound $\Theta(t, \mu)$ can be obtained via an iterative scheme

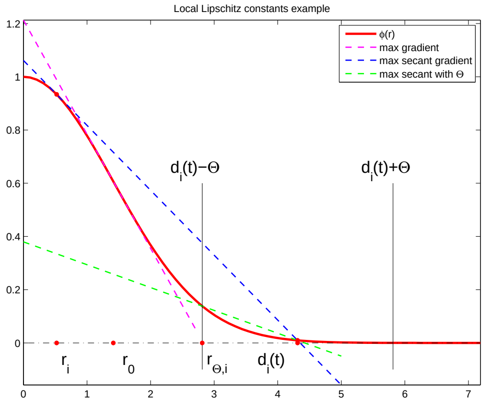

Fig. 1 Local Lipschitz constant comparison

starting with $\Theta^0 \equiv \infty$, and recycling the estimation result within another iteration of the error bound computation. As shown in previous work, those iterations converge uniformly to a limit estimate Δ_{LSLE}^∞, which holds also for the parameterized case.

3.3 Computation and offline/online decomposition

From the proof of Theorem 1 we see that the actual computation of the error estimators only involves solving a small auxiliary ODE

$$v'(t) = \beta(t,\mu)v(t) + \alpha(t,\mu), \qquad v(0) = E_{x_0}(\mu),$$

which can in fact be computed “on the fly” with the reduced simulation by adding an extra dimension to the reduced system (6)-(9). Now, in order to allow an efficient computation of the term $\alpha(t,\mu)$ we decompose it in an offline/online fashion similar to [4]. In this setting, the offline phase comprises expensive computations with complexity depending on the high dimension d and has to be performed once. The online phase is performed for different parameters $\mu \in \mathcal{P}$ and inputs $u(t)$ and its complexity is dependent only on the reduced dimension r.

Proposition 1 (Offline/online decomposition)
Let the conditions from either Theorem 1 or 2 hold. Define

$$\begin{aligned} M &:= (c_1 \dots c_N) \in \mathbb{R}^{d\times N},\\ \varphi(t) &:= \big(\Phi_s^r(z(t), z_i)\Phi_t(t,t_i)\Phi_{\mathcal{P}}(\mu,\mu_i)\big)_{i=1}^N \in \mathbb{R}^N, \end{aligned}$$

and recall the norm inducing matrix G from Section 2. In addition to (4) *assume that $x_0(\mu)$ is given as*

$$x_0(\mu) = \sum_{i=1}^{Q_0} \theta_i^0(\mu)x_i^0. \tag{23}$$

Then the offline computations are $||c_i||_G$, $i = 1\dots N$ for the $\beta(t,\mu)$ term and

$$\begin{aligned} \tilde{x}_i^0 &:= \big(I_d - VW^t\big)x_i^0,\ i = 1\dots Q_0,\\ \tilde{x}_{ij}^0 &:= (\tilde{x}_i^0)^t G\tilde{x}_j^0,\ i,j = 1\dots Q_0,\\ \tilde{M} &:= \big(I_d - VW^t\big)M,\ M_1 := \tilde{M}^t G\tilde{M} \in \mathbb{R}^{N\times N},\\ \tilde{B}_i &:= \big(I_d - VW^t\big)B_i,\ i = 1\dots Q_B,\\ M_{2,i} &:= \tilde{M}^t G\tilde{B}_i \in \mathbb{R}^{N\times m},\ i = 1\dots Q_B,\\ M_{3,ij} &:= \tilde{B}_i^t G\tilde{B}_j \in \mathbb{R}^{m\times m},\ i,j = 1\dots Q_B \end{aligned}$$

The online part consists of computing the ODE

$$\begin{aligned} v(0) &= \sqrt{\sum_{i,j}^{Q_0} \theta_i^0(\mu)\theta_j^0(\mu)\tilde{x}_{ij}^0}\\ v'(t) &= \beta(t,\mu)v(t) + \alpha(t,\mu)\\ \alpha(t,\mu) &= \Big(\varphi(t)^t M_1\varphi(t) + 2\varphi(t)^t M_2(t,\mu)u(t)\\ &\quad + u(t)^t M_3(t,\mu)u(t)\Big)^{\frac{1}{2}},\\ M_2(t,\mu) &:= \sum_{i=1}^{Q_B} \theta_i^B(t,\mu)M_{2,i}\\ M_3(t,\mu) &:= \sum_{i,j=1}^{Q_B} \theta_i^B(t,\mu)\theta_j^B(t,\mu)M_{3,ij}. \end{aligned}$$

The proof follows directly from the definitions. Notice that if there are no inputs $u(t)$ for a given system only the constant matrix M_1 has to be computed as M_2 and M_3 do not appear. However, all the offline matrices M_1, M_2, M_3 are small matrices only depending on Q_B, N and m. For an efficient scheme on how to compute the local Lipschitz constant estimations we refer to [11].

4 Experimental results

In this part we present numerical experiments for synthetic parameterized dynamical systems using kernel expansions as nonlinearity. As in our previous work, we additionally consider two more error estimator modifications. The convergence of the iterative scheme to a limit estimate Δ^{∞}_{LSLE} motivates a heuristic variant of our local estimators, where we use the estimation from the previous time-step as a-priori error bound for the next one. This originates from the fact that the limit function reproduces itself when used as a-priori bound within the iterations. We will refer to this time-discrete method by "LSLE TD". Second, the choice

$$\beta(t) := \frac{||f(x(t),t,\mu) - f(x^r(t),t,\mu)||_G}{||x(t;\mu) - x^r(t;\mu)||_G}$$

yields the smallest possible estimation for this estimator structure and is included as expensive "Lower Bound" for comparison reasons.

Our test setting is the following: We set $d = 240000$ to represent a large-scale system, $T = 20$, $G = I_d$ and let $\mathbf{1} := (1\dots 1)^t \in \mathbb{R}^d$. We choose $\mathcal{P} = [0,1] \times [0,10] \times [-1,1]$ as our parameter domain. The kernel expansion (5) uses $N = 20$, $\Phi_t \equiv 1$ and centers $x_i := \frac{50(i-1)}{N-1}\mathbf{1},\ \mu_i := \frac{10(i-1)}{N-1}(0,1,0)^t,\ i = 1\dots N$. Further, Φ_s and $\Phi_{\mathcal{P}}$ are Gaussians with $\gamma_s = 224, \gamma_{\mathcal{P}} = 5.3733$, where γ_s is chosen so that $\Phi_s(x_i,x_j) < 10^{-5}\ \forall\ |i-j| \geq 2$, i.e. a certain locality of the kernel expansion is ensured. Note that $\Phi_{\mathcal{P}}$ only uses the second entry of μ, $\mu_{\{1,3\}}$ are ignored. Finally, the expansion coefficient vectors are given via $c_i = \exp(-x_i/15) \in \mathbb{R}^d,\ i = 1\dots N$ and we define $x_0(\mu) = \mu_3\mathbf{1}$ as initial value. So parameter μ_2 is an *"expansion parameter"* influencing the system's inner dynamics and μ_3 sets the *"initial value"*; μ_1 will be discussed later. As in our referenced work, we use $\theta = 0.05$ to control V and hence the subspace quality, and we average the output using $C(t,\mu) = \sqrt{d}\mathbf{1}$. We use an explicit Euler scheme with time-step $\Delta t = 0.05$ as solver.

The next three figures compare the different estimated output errors $||e_y(t)||$ using (15) and $\mu_2 = 5, \mu_3 = -0.2$. Figure 2 shows the absolute errors over time. The improvement from the GLE over the LSLE variant is due to the local Lipschitz constant estimations and the LSLE iterations further improve the estimation by several orders of magnitude. Figure 3 shows the relative output errors for the same setting. The fact that the LSLE is indistinguishable from the discrete LSLE TD variant after five iterations shows a fast convergence of the iteration scheme and strongly supports the applicability of the heuristic LSLE TD variant as substitute for $\Delta^{\Theta_\infty}_{LSLE}(t,\mu)$. Figure 4 plots the computation times against the estimated output errors at $T = 20$. All estimators are $10-20$ times faster than computing the full error. The GLE is the cheapest but coarsest estimator, and the LSLE TD is slightly slower but yields the best estimation results.

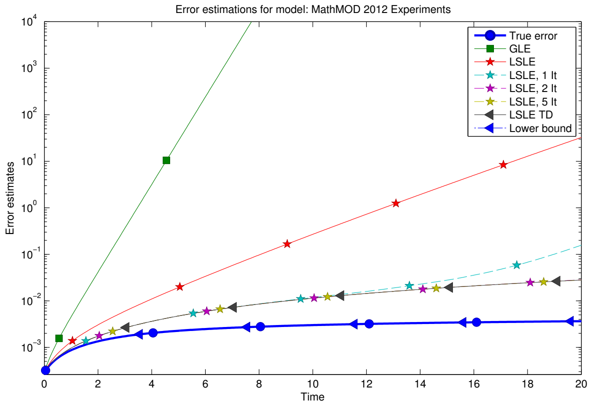

Fig. 2 Absolute L^2 state space errors of estimators

Table 1 shows the output errors at $T = 20$ along with the computation times and the overestimation factors. We also observe that too many iterations of the LSLE estimation do not necessarily yield a relevant improvement. The LSLE TD estimator only overestimates by a factor of 7.6.

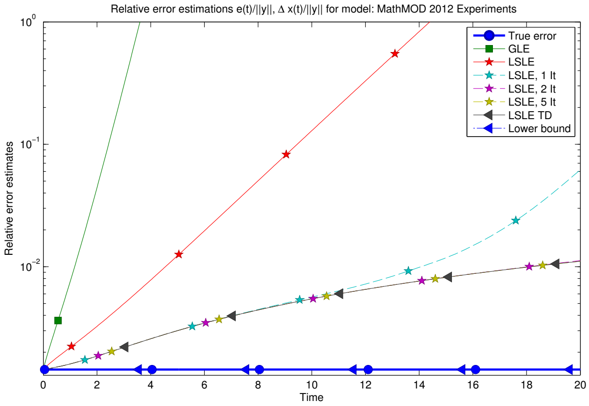

Fig. 3 Relative errors for estimators, using $\mu_2 = 5, \mu_3 = -0.2$

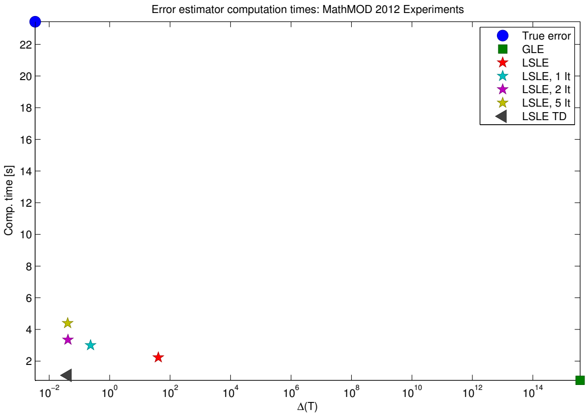

Fig. 4 Computation times for estimator variants

Finally, Figure 5 contains a parameter sweep for μ_2 ranging over $[0, 10]$. Shown are the simulation outputs up to $T = 20$ along with the error bounds of the LSLE TD in transparent light-red. The error bounds stay sharp over the whole parameter range $[0, 10]$, even though the system's dynamics change considerably for different μ_2.

In order to show the influence of external input we choose an affine-parametric input matrix

$$B(t,\mu) = \mu_1 \begin{pmatrix} \mathbf{1} & \mathbf{0} \end{pmatrix} + (1-\mu_1) \begin{pmatrix} \mathbf{0} & \mathbf{1} \end{pmatrix} \in \mathbb{R}^{d \times 2}$$

with inputs

$$u_1(t) = \begin{pmatrix} \frac{2}{5}\sin(\frac{t}{3}) \\ e^{-(12-t)^2} \end{pmatrix}, \quad u_2(t) = \begin{pmatrix} \frac{1}{2}\sin(\frac{t}{2}) \\ 4e^{-7(12-t)^2} - \frac{1}{2}e^{-(5-t)^2} \end{pmatrix}.$$

Both represent each an oscillating and a local stimulation of different kind. For both settings Figure

Name	$\varDelta(20)$	Time	Overestimation
True error	$3.650e-03$	21.43s	$1.000e+00$
GLE	$3.682e+15$	0.62s	$1.009e+18$
LSLE	$3.251e+01$	2.05s	$8.907e+03$
LSLE, 1 It	$1.568e-01$	2.79s	$4.295e+01$
LSLE, 2 It	$2.839e-02$	3.11s	$7.779e+00$
LSLE, 5 It	$2.801e-02$	4.13s	$7.674e+00$
LSLE TD	$2.801e-02$	0.90s	$7.674e+00$
Lower bound	$3.652e-03$	44.02s	$1.001e+00$

Table 1 Estimator statistics at $T = 20$

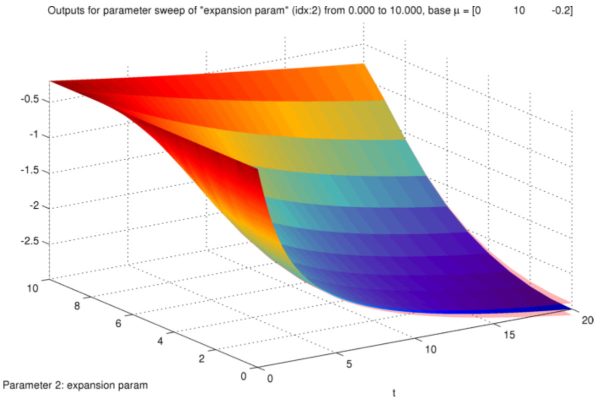

Fig. 5 Parameter sweep for $\mu_2 \in [0, 10], \mu_1 = 0, \mu_3 = -0.2$

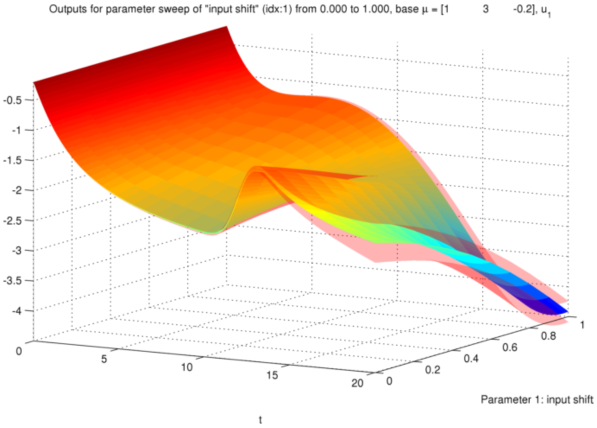

Fig. 6 Parameter sweep for *input shift* $\mu_1 \in [0, 1]$ and u_1

6 and 7 show that the LSLE TD estimator gives good estimations over the parameter range. Finally,

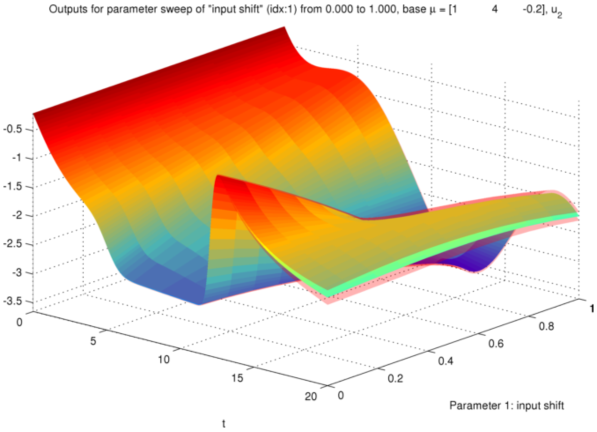

Fig. 7 Parameter sweep for *input shift* $\mu_1 \in [0, 1]$ and u_2

Figure 8 displays a 2D sweep for μ_1, μ_2 and the output and error bounds by LSLE TD at $T = 20$. One can see that the estimation sharpness is more sensitive to changes of the inner dynamics compared to different inputs, which is explainable by the strong influence of μ_2 to the critical (21) term.

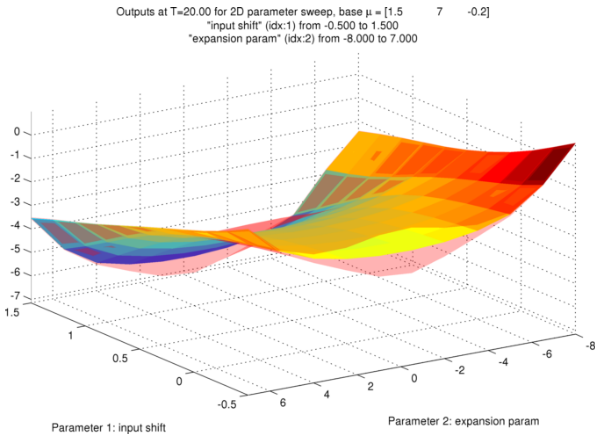

Fig. 8 Parameter sweep for *input shift* and *expansion parameter*, $\mu_1 \in [0,1], \mu_2 \in [-8,7]$ and using u_1

5 Conclusion & Perspectives

In this work we extended the error estimation results in [11] to the parameterized case and showed applicability of the error estimators also for this setting. With the time/parameter affine components B, C and x_0, the parameterized system and the proposed error estimators are completely offline/online decomposable and thus allow for fast evaluation. Our experiments show the applicability of the error estimators for systems where different components depend on parameters. Future work will comprise investigation of techniques to obtain kernel expansion approximations (5) of arbitrary nonlinear dynamical systems, see [8–10]. Finally, incorporation of those extensions into the error estimators will be subject of further research.

6 Acknowledgements

The authors would like to thank the German Research Foundation (DFG) for financial support of the project within the Cluster of Excellence in Simulation Technology (EXC 310/1) at the University of Stuttgart and the Baden-Württemberg Stiftung gGmbH.

References

1. A.C. Antoulas. *Approximation of Large–Scale Dynamical Systems.* SIAM Publications, Philadelphia, PA, 2005.
2. M. Barrault, Y. Maday, N.C. Nguyen, and A.T. Patera. An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations. *C. R. Math. Acad. Sci. Paris Series I*, 339:667–672, 2004.
3. B. Haasdonk. Convergence rates of the pod-greedy method. SimTech Preprint 2011-23, University of Stuttgart. Submitted to M2AN., 2011.
4. B. Haasdonk and M. Ohlberger. Efficient reduced models and a posteriori error estimation for parametrized dynamical systems by offline/online decomposition. *Math. Comp. Model. Dyn.*, 17(2):145 – 161, 2011.
5. J.K. Hale. *Ordinary differential equations*, volume XXI of *Pure and Applied Mathematics.* Wiley-Interscience, 1969.
6. I. T. Jolliffe. *Principal Component Analysis.* Springer-Verlag, 2002.
7. J. Phillips, J. Afonso, A. Oliveira, and L.M. Silveira. Analog macromodeling using kernel methods. In *Proc. of ICCAD 2003*, pages 446–453, November 2003.
8. B. Schölkopf and A. J. Smola. *Learning with Kernels.* Adaptive Computation and Machine Learning. The MIT Press, 2002.

9. I. Steinwart, D. Hush, and C. Scovel. Training SVMs Without Offset. *J. Mach. Learn. Res.*, 12:141–202, February 2011.
10. H. Wendland. *Scattered data approximation*, volume 17 of *Cambridge Monographs on Applied and Computational Mathematics*. Cambridge University Press, 2005.
11. D. Wirtz and B. Haasdonk. Efficient a-posteriori error estimation for nonlinear kernel-based reduced systems. *Systems & Control Letters*, 61(1):203 – 211, 2012.