# Appendix: Gaussian Processes for Efficient Numerical Simulations in Physics

## A Neural Network from Section 2.3.1

For comparison with the GP model, a 5-layer simple neural network was used. This neural network had 100 hidden units per layer, and was fully connected (100 activations between layers).

## B Details of Time Extrapolation Model

In this section we describe the time extrapolation model for the Kortweg-de Vries system and for the Kuramoto-Sivashinsky system in further detail. Recall the nonlinear operators for the two systems, given by equations REF and REF in the main text:

$$\text{Kortweg-de Vries: } \frac{\partial u}{\partial t} = -\lambda_1 u \frac{\partial u}{\partial x} - \lambda_2 \frac{\partial^3 u}{\partial x^3},$$

$$\text{Kuramoto-Sivashinksy: } \frac{\partial u}{\partial t} = -\lambda_1 u \frac{\partial u}{\partial x} - \lambda_2 \frac{\partial^2 u}{\partial x^2} - \lambda_3 \frac{\partial^4 u}{\partial x^4}$$

We begin with the Kortweg-de Vries (KdV) equation. The nonlinear operator (where we use subscripts to denote derivatives)

$$\mathcal{N}_x u = -\lambda_1 u u_x - \lambda_2 u_{xx} - \lambda_3 u_{xxx}$$

must be linear when applied to $u_n$. So, we must approximate the coefficient $u$ in the first term by something which does not depend on $u_n$. Since we assume that the time between measurements $\Delta t$ is small, we approximate $u_n \approx u_{n-1}$. That is, we approximate $\mathcal{N}_x$ by the linear operator

$$\mathcal{L}_x u = -\lambda_1 \bar{u} u_x - \lambda_2 u_{xx} - \lambda_3 u_{xxx}$$

where $\bar{u}(x,t) = u(x, t - \Delta t)$. Then, after placing the joint GP prior specified in section 3.3 of the main text, we interpret the joint kernel as

$$\begin{bmatrix} u_n \\ u_{n-1} \end{bmatrix} = \mathcal{GP}\left(0, \begin{bmatrix} k_{1,1} & k_{1,0} \\ k_{0,1} & k_{0,0} \end{bmatrix}\right)$$

where we have defined, with $\mathcal{L}_x$ referring to derivatives with respect to $x$, and $\mathcal{L}_{x'}$ referring to derivatives with respect to $x'$, the first and second arguments of $k$ respectively,

$$\begin{aligned} k_{1,0}(x,x') &= (I - \Delta t \mathcal{L}_x)k(x,x') \\ k_{0,1}(x,x') &= (I - \Delta t \mathcal{L}_{x'})k(x,x') \\ k_{0,0}(x,x') &= (I - \Delta t \mathcal{L}_{x'})(I - \Delta t \mathcal{L}_x)k(x,x') \end{aligned}$$

Here we note that since $k$ is assumed to be a stationary kernel, we have $k(x,x') = k(x - x') = k(\tau)$, and so

$$\frac{\partial^n}{\partial x^n} k(x,x') = (-1)^n \frac{\partial^n}{\partial x'^n} k(x,x') = \frac{\partial^n}{\partial \tau^n} k(\tau)$$

Notating $\frac{\partial^n}{\partial \tau^n} k(\tau)$ by $k^{(n)}$, we then can compute, for the KdV model,

$$k_{1,1} = k$$

$$\begin{aligned} k_{1,2} &= (I - \Delta t \mathcal{L}_{x'})k \\ &= k - \Delta t \lambda_1 \bar{u}' k^{(1)} + \Delta t \lambda_2 k^{(3)} \end{aligned}$$

$$\begin{aligned} k_{2,2} &= (I - \Delta t \mathcal{L}_x)(I - \Delta t \mathcal{L}_{x'})k \\ &= k + \Delta t \lambda_1 (\bar{u} - \bar{u}')k^{(1)} - \Delta t^2 \lambda_1^2 \bar{u}' \bar{u} k^{(2)} - \Delta t^2 \lambda_1 \lambda_2 (\bar{u} + \bar{u}')k^{(4)} - \Delta t^2 \lambda_2^2 k^{(6)} \end{aligned}$$

(recall $\bar{u}(x,t) = u(x, t - \Delta t)$ and $\bar{u}'(x,t) = u(x', t - \Delta t)$). From this we may compute derivatives with respect to parameters $\lambda_i$, which are used for optimization:

$$\begin{aligned} \frac{\partial k_{1,1}}{\partial \lambda_1} &= \frac{\partial k_{1,1}}{\partial \lambda_2} = 0 \\ \frac{\partial k_{1,0}}{\partial \lambda_1} &= -\Delta t \bar{u}' k^{(1)} \\ \frac{\partial k_{1,0}}{\partial \lambda_2} &= \Delta t k^{(3)} \\ \frac{\partial k_{0,0}}{\partial \lambda_1} &= \Delta t (\bar{u} - \bar{u}')k^{(1)} - 2\Delta t^2 \lambda_1 \bar{u}' \bar{u} k^{(2)} - \Delta t^2 \lambda_2 (\bar{u} + \bar{u}')k^{(4)} \\ \frac{\partial k_{0,0}}{\partial \lambda_2} &= -\Delta t^2 \lambda_1 (\bar{u} + \bar{u}')k^{(4)} - 2\Delta t^2 \lambda_2 k^{(6)} \end{aligned}$$

Doing the same for the KS system, which has precisely the same type of nonlinearity, we obtain

$$k_{1,1} = k$$

$$\begin{aligned}
k_{1,0} &= (I - \Delta t \mathcal{L}_{x'})k \\
&= k - \Delta t \lambda_1 \bar{u}' k^{(1)} + \Delta t \lambda_2 k^{(2)} + \Delta t \lambda_3 k^{(4)}
\end{aligned}$$

$$\begin{aligned}
k_{0,0} &= (I - \Delta t \mathcal{L}_x)(I - \Delta t \mathcal{L}_{x'})k \\
&= k + \Delta t \lambda_1 (\bar{u} - \bar{u}')k^{(1)} + (2\Delta t \lambda_2 - (\Delta t \lambda_1)^2 \bar{u}\bar{u}')k^{(2)} + (\Delta t^2 \lambda_1 \lambda_2 (\bar{u} - \bar{u}'))k^{(3)} \\
&\quad + (2\Delta t \lambda_3 + \Delta t^2 \lambda_2^2)k^{(4)} + (\Delta t^2 \lambda_1 \lambda_3 (\bar{u} - \bar{u}'))k^{(5)} + 2\Delta t^2 \lambda_2 \lambda_3 k^{(6)} + \Delta t^2 \lambda_3^2 k^{(8)}
\end{aligned}$$

and for derivatives,

$$\begin{aligned}
\frac{\partial k_{1,1}}{\partial \lambda_1} &= \frac{\partial k_{1,1}}{\partial \lambda_2} = \frac{\partial k_{1,1}}{\partial \lambda_3} = 0 \\
\frac{\partial k_{1,0}}{\partial \lambda_1} &= -\Delta t \bar{u}' k^{(1)} \\
\frac{\partial k_{1,0}}{\partial \lambda_2} &= \Delta t k^{(2)} \\
\frac{\partial k_{1,0}}{\partial \lambda_3} &= \Delta t k^{(4)} \\
\frac{\partial k_{0,0}}{\partial \lambda_1} &= \Delta t (\bar{u} - \bar{u}')k^{(1)} - 2\Delta t^2 \lambda_1 \bar{u}\bar{u}' k^{(2)} + \Delta t^2 \lambda_2 (\bar{u} - \bar{u}')k^{(3)} \\
&\quad + \Delta t^2 \lambda_3 (\bar{u} - \bar{u}')k^{(5)} \\
\frac{\partial k_{0,0}}{\partial \lambda_2} &= 2\Delta t k^{(2)} + \Delta t^2 \lambda_1 (\bar{u} - \bar{u}')k^{(3)} + 2\Delta t^2 \lambda_2 k^{(4)} + 2\Delta t^2 \lambda_3 k^{(6)} \\
\frac{\partial k_{0,0}}{\partial \lambda_3} &= 2\Delta t k^{(4)} + \Delta t^2 \lambda_1 (\bar{u} - \bar{u}')k^{(5)} + 2\Delta t^2 \lambda_2 k^{(6)} + 2\Delta t^2 \lambda_3 k^{(8)}
\end{aligned}$$

# C Spectral Mixture Kernel Derivatives

In one dimension, the spectral mixture (SM) kernel is given by equation 14 in the main text,

$$k(\tau) = \sum_{q=1}^{Q} w_q k_q(\tau) = \sum_{q=1}^{Q} w_q \exp(-2\pi^2 \tau^2 \nu_q) \cos(2\pi \tau \mu_q)$$

where $Q$ is the number of mixture components. In order to compute the covariances in appendix A, we require analytic derivatives of $k(\tau)$. From now on we consider only derivatives of a single mixture component,

$$k_q(\tau) = \exp(-2\pi^2\tau^2\nu_q)\cos(2\pi\tau\mu_q)$$

and note that when computing a full derivative one sums over all such mixture components. Making the substitutions $a = -2\pi^2\nu_q$ and $b = 2\pi\mu_q$, we can easily see that arbitrary derivatives take the form

$$k_q^{(n)}(\tau) = e^{a\tau^2}\big(\cos(b\tau)P_c^n(\tau) + \sin(b\tau)P_s^n(\tau)\big)$$

where $P_s^n(\tau)$ and $P_c^n(\tau)$ are polynomials functions of $\tau$ which also involve $a$ and $b$. From this form we can define these polynomials recursively:

$$P_c^{n+1} = 2a\tau P_c^n + \frac{d}{d\tau}P_c^n + bP_s^n, \quad P_s^{n+1} = 2a\tau P_s^n + \frac{d}{d\tau}P_s^n - bP_c^n$$

We compute the first eight such polynomials in order to analytically represent all eight derivatives of $k$ for the KS system:

$$P_c^0(\tau) = 1$$
$$P_s^0(\tau) = 0$$

$$P_c^1(\tau) = 2a\tau$$
$$P_s^1(\tau) = -b$$

$$P_c^2(\tau) = 4a^2\tau^2 + (2a - b^2)$$
$$P_s^2(\tau) = -4ab\tau$$

$$P_c^3(\tau) = 8a^3\tau^3 + (12a^2 - 6ab^2)\tau$$
$$P_s^3(\tau) = -12a^2b\tau^2 + (-6ab + b^3)$$

$$P_c^4(\tau) = 16a^4\tau^4 + (48a^3 - 24a^2b^2)\tau^2 + (12a^2 - 12ab^2 + b^4)$$
$$P_s^4(\tau) = -32a^3b\tau^3 + (-48a^2b + 8ab^3)\tau$$

$$P_c^5(\tau) = 32a^5\tau^5 + (160a^4 - 80a^3b^2)\tau^3 + (120a^3 - 120a^2b^2 + 10ab^4)\tau$$
$$P_s^5(\tau) = -80a^4b\tau^4 + (-240a^3b + 40a^2b^3)\tau^2 + (-60a^2b + 20ab^3 - b^5)$$

$$P_c^6(\tau) = 64a^6\tau^6 + (480a^5 - 240a^4b^2)\tau^4 + (720a^4 - 720a^3b^2 + 60a^2b^4)\tau^2 + \cdots$$
$$+ (120a^3 - 180a^2b^2 + 30ab^4 - b^6)$$
$$P_s^6(\tau) = -192a^5b\tau^5 + (-960a^4b + 160a^3b^3)\tau^3 + (-960a^3b + 240a^2b^3 - 12ab^5)\tau$$

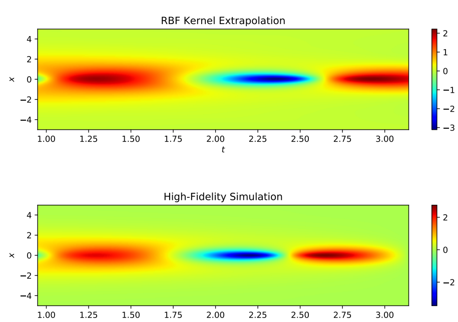
$$
\begin{aligned}
P_c^7(\tau) &= 128a^7\tau^7 + (1344a^6 - 672a^5b^2)\tau^5 + (3360a^5 - 3360a^4b^2 + 280a^3b^4)\tau^3 + \cdots \\
&\qquad + (1680a^4 - 2760a^3b^2 + 420a^2b^4 - 14ab^6)\tau \\
P_s^7(\tau) &= -488a^6b\tau^6 + (-3360a^5b + 560a^4b^3)\tau^4 + (-5520a^4b + 1680a^3b^3 - 84a^2b^5)\tau^2 + \cdots \\
&\qquad + (-1080a^3b + 420a^2b^3 - 42ab^5 + b^7)
\end{aligned}
$$

$$
\begin{aligned}
P_c^8(\tau) &= 256a^8\tau^8 + (3584a^7 - 1832a^6b^2)\tau^6 + (13440a^6 - 13440a^5b^2 + 1120a^4b^4)\tau^4 + \cdots \\
&\qquad + (13440a^5 - 21120a^4b^2 + 3360a^3b^4 - 112a^2b^6)\tau^2 + \cdots \\
&\qquad + (1680a^4 - 3840a^3b^2 + 840a^2b^4 - 56ab^6 + b^8) \\
P_s^8(\tau) &= -1104a^7b\tau^7 + (-10992a^6b + 1792a^5b^3)\tau^5 + (-27840a^5b + 8960a^4b^3 - 448a^3b^5)\tau^3 + \cdots \\
&\qquad + (-14880a^4b + 6960a^3b^3 - 672a^2b^5 + 16ab^7)\tau
\end{aligned}
$$

# D Additional Experiment for the RBF Time Extrapolation Model

In addition to the KS and KdV systems, we examined the Nonlinear Schrödinger (NLS) equation with the RBF model. However due to time constraints and a qualitatively different nonlinearity we were not able to implement the SM model for this system. Results are given in figure 1 below. All results are measured on the real part of the solution for space considerations. Note that the governing equation for the NLS system is

$$
i\frac{\partial u}{\partial t} = -\lambda_1 \frac{\partial^2 u}{\partial x^2} - \lambda_2 |u|^2 u,\ u \in \mathbb{C},\ \lambda_1 = \frac{1}{2},\ \lambda_2 = 1
$$




| | RBF Model | |
|---|---|---|
| Data | Clean | Noisy |
| log(noise) | -17.11 | -10.41 |
| $\lambda_1$ | 0.51 | 0.48 |
| $\lambda_2$ | 1.00 | 0.97 |
| Peak Error | 3.86 | 2.67 |
| Average Error | 1.17 | 1.00 |

Figure 1: Results of an example NLS simulation for RBF kernel with clean data (left), and statistics on error and discovered coefficients (right). Here "Peak Error" refers to the largest error magnitude across the simulation, and "Average Error" refers to the average relative error in 2-norm over all simulated timesteps.

As can be seen in figure 1, the NLS system's dynamics were well recaptured by the GP model, although due to sensitivity to hyperparameters the periodicity of the system was not perfectly captured, leading to error estimates which appear large but in fact are simply due to a deviation in synchronization between the simulation and extrapolated system. Once again the RBF model accurately recovers the desired coefficients, and once again qualitatively reproduces key dynamics. In figure 2, we see that as described in the figure 1 table the value of the total error in simulation is in fact quite high due to the desynchronized behavior of the extrapolation as compared to the simulation. However the relative error statistics are inflated by the fact that the NLS system occasionally has zero signal, hence inflating relative error as compared to absolute error.

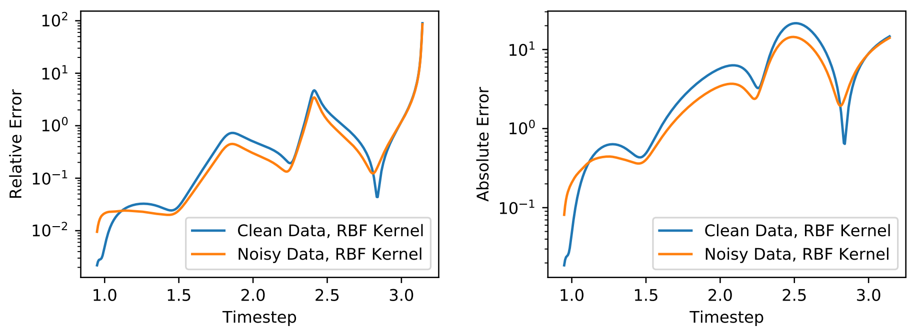



Figure 2: Relative error (left) and absolute error (right) vs timestep for NLS system.