

# **Inference for Delay Differential Equations Using Manifold-Constrained Gaussian Processes**

Yuxuan Zhao and Samuel W.K. Wong

*University of Waterloo*

## **Supplementary Material**

This Supplementary Material contains an illustrative figure that highlights the parameter sensitivity and computational complexity of numerical solvers for a simple DDE model (referenced in Section 1 of the main text), expressions of the posterior distribution in MAGI (referenced in Section 2 of the main text), conditions and proofs of the theoretical error bounds of the approximation schemes along with numerical validations (referenced in Section 3.2 of the main text), the investigation of effects of the discretization set (referenced in Section 3.3 of the main text), the implementation details and supplementary results for method comparisons (referenced in Section 4.1 of the main text), summary of parameter values and implementation details for the *lac* operon model along with a comparison to the NLS approach (referenced in Section 4.2 of the main text), and data processing steps with implementation details for the real data application and prediction using the time-delayed SIRD model (referenced in Section 5 of the main text).

## S1 DDE Example: Effects of the Time-delay Parameter on Oscillation Behavior and Numerical Solver Complexity

We take Hutchinson's equation,

$$dP(t)/dt = 0.8 \cdot P(t) [1 - P(t - \tau)/2000],$$

as used for the simulation study in Section 4.1 of the main text, to illustrate the effects of the time-delay parameter  $\tau$ . Figure 1 shows that the DDE trajectories are sensitive to small variations in  $\tau$ : the oscillation behavior of the numerical solution varies significantly among  $\tau = \{0, 1.5, 3, 4\}$ . Note that setting  $\tau = 0$  reduces the equation to an ODE, and is associated with a trajectory that stabilizes to a steady state without any oscillation. In contrast, the oscillation behavior of the DDE trajectories for  $\tau = \{1.5, 3, 4\}$  suggests that a numerical solver will require finer discretization time steps to maintain accuracy in the numerical solution, compared to  $\tau = 0$ . This intuition is confirmed by benchmarking: running the ODE solver (for  $\tau = 0$ ) takes approximately 0.9 milliseconds per repetition, while running the DDE solver (for  $\tau = \{1.5, 3, 4\}$ ) takes approximately 8 milliseconds per repetition. Hence, the DDE numerical solver is associated with an increased computational cost, taking approximately 9 times longer than solving the corresponding ODE. It has been previously noted that optimization-based

S1. DDE EXAMPLE: EFFECTS OF THE TIME-DELAY PARAMETER ON  
OSCILLATION BEHAVIOR AND NUMERICAL SOLVER COMPLEXITY<sup>3</sup>

---

algorithms using a numerical solver (e.g., nonlinear least squares, or NLS) may only converge to local optima due to the sensitivity of the numerical solution to the parameters and initial conditions (Liang and Wu, 2008). This example further illustrates that for DDEs, the convergence difficulties of NLS can be even more pronounced due to the numerical solution’s sensitivity to the time delay parameter; a detailed investigation using NLS and numerical solvers for this inference problem is presented in Section S8.3 of the Supplementary Material.

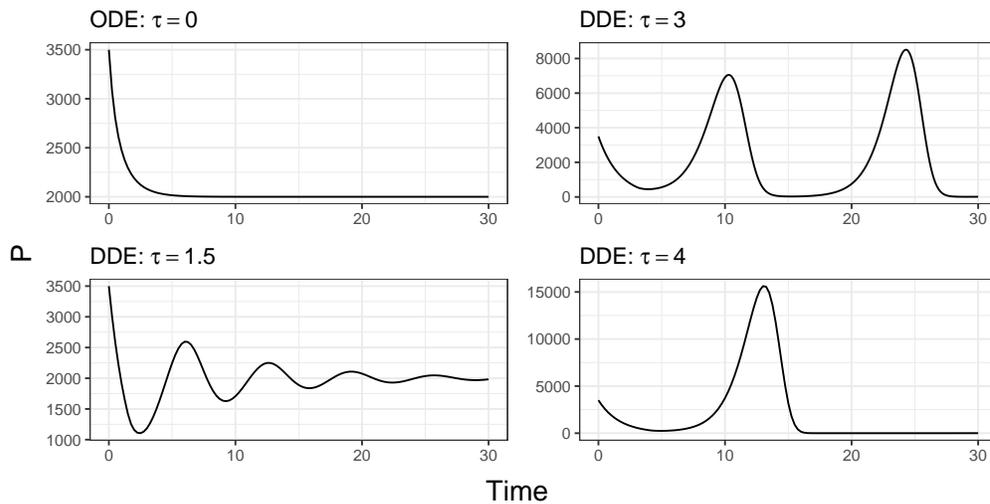


Figure 1: The effect of time-delay parameter  $\tau$  on the oscillation behavior of the system trajectory in Hutchinson’s equation:  $dP(t)/dt = 0.8 \cdot P(t) [1 - P(t - \tau)/2000]$ .

## S2 Detailed Expression of Posterior Distribution in MAGI

This section reviews the details of the original MAGI posterior distribution in Yang et al. (2021). According to Bayes' rule, the posterior distribution of  $\boldsymbol{\theta}$  and  $\mathbf{x}(\mathbf{I})$  given  $W_{\mathbf{I}} = 0$  and  $\mathbf{y}(\boldsymbol{\gamma})$  is

$$p(\boldsymbol{\theta}, \mathbf{x}(\mathbf{I}) | W_{\mathbf{I}} = 0, \mathbf{y}(\boldsymbol{\gamma})) \propto p(\boldsymbol{\theta}, \mathbf{x}(\mathbf{I}), W_{\mathbf{I}} = 0, \mathbf{y}(\boldsymbol{\gamma})). \quad (\text{S2.1})$$

Factorizing Equation (S2.1) yields the four terms,

$$p(\boldsymbol{\theta}, \mathbf{x}(\mathbf{I}), W_{\mathbf{I}} = 0, \mathbf{y}(\boldsymbol{\gamma})) = \underbrace{\pi(\boldsymbol{\theta})}_{(1)} \times \underbrace{p(\mathbf{x}(\mathbf{I}) | \boldsymbol{\theta})}_{(2)} \times \underbrace{p(\mathbf{y}(\boldsymbol{\gamma}) | \mathbf{x}(\mathbf{I}), \boldsymbol{\theta})}_{(3)} \times \underbrace{p(W_{\mathbf{I}} = 0 | \mathbf{y}(\boldsymbol{\gamma}), \mathbf{x}(\mathbf{I}), \boldsymbol{\theta})}_{(4)}.$$

The first term is the prior distribution of the model parameters. Since  $\boldsymbol{\theta}$  is independent of  $\mathbf{x}(\mathbf{I})$  from the GP,  $p(\mathbf{x}(\mathbf{I}) | \boldsymbol{\theta})$  can be simplified to  $p(\mathbf{x}(\mathbf{I}))$ . Similarly, due to the independence between the noisy observations and model parameters, the condition on the parameters can be also be suppressed, yielding  $p(\mathbf{y}(\boldsymbol{\gamma}) | \mathbf{x}(\mathbf{I}))$  for the third term.

We apply the definition of  $W_{\mathbf{I}} = 0$  in the fourth term and note that the distribution of the GP derivative  $\mathbf{x}'(\mathbf{I})$  given  $\mathbf{x}(\mathbf{I})$  is conditionally independent of both the model parameters and the noisy observations; hence the fourth term can be rewritten as  $p(\mathbf{x}'(\mathbf{I}) = \mathbf{f}(\mathbf{x}(\mathbf{I}), \boldsymbol{\theta}, \mathbf{I}) | \mathbf{x}(\mathbf{I}))$ , i.e., evaluating the density of  $\mathbf{x}'(\mathbf{I})$  at  $\mathbf{x}'(\mathbf{I}) = \mathbf{f}(\mathbf{x}(\mathbf{I}), \boldsymbol{\theta}, \mathbf{I})$ .

Combining the above with the fact that the second, third, and fourth terms

are all multivariate Gaussian, this leads to the following expression of the posterior:

$$\begin{aligned}
& \pi(\boldsymbol{\theta}) \times p(\mathbf{x}(\mathbf{I})) \times p(\mathbf{y}(\boldsymbol{\gamma})|\mathbf{x}(\mathbf{I})) \times p(W_{\mathbf{I}} = 0|\mathbf{x}(\mathbf{I})) \\
& \propto \underbrace{\pi(\boldsymbol{\theta})}_{(1)} \exp \left\{ -\frac{1}{2} \sum_{i=1}^m \left[ \underbrace{|\mathbf{I}| \log(2\pi) + \log |\mathbf{C}_i| + \|\mathbf{x}_i(\mathbf{I}) - \boldsymbol{\mu}_i(\mathbf{I})\|_{\mathbf{C}_i^{-1}}^2}_{(2)} \right. \right. \\
& \quad + \underbrace{N_i \log(2\pi\sigma_i^2) + \|\mathbf{x}_i(\boldsymbol{\gamma}_i) - \mathbf{y}_i(\boldsymbol{\gamma}_i)\|_{\sigma_i^{-2}}^2}_{(3)} \\
& \quad \left. \left. + \underbrace{|\mathbf{I}| \log(2\pi) + \log |\boldsymbol{\zeta}_i| + \left\| \mathbf{f}_{i,\mathbf{I}}^{\mathbf{x},\boldsymbol{\theta}} - \boldsymbol{\mu}'_i(\mathbf{I}) - \mathbf{m}_i \{\mathbf{x}_i(\mathbf{I}) - \boldsymbol{\mu}_i(\mathbf{I})\} \right\|_{\boldsymbol{\zeta}_i^{-1}}^2}_{(4)} \right] \right\},
\end{aligned}$$

where  $\|\mathbf{v}\|_A^2 = \mathbf{v}^\top A \mathbf{v}$ ,  $|\mathbf{I}|$  is the cardinality of  $\mathbf{I}$ , and  $\boldsymbol{\mu}_i(\mathbf{I})$  is the mean function of the  $i$ -th component. Moreover,  $\sigma_i$  denotes the noise level for component  $i$ ,  $N_i$  denotes the number of observations for the  $i$ -th component, and  $\mathbf{f}_{i,\mathbf{I}}^{\mathbf{x},\boldsymbol{\theta}}$  represents the  $i$ -th component of  $\mathbf{f}(\mathbf{x}(\mathbf{I}), \boldsymbol{\theta}, \mathbf{I})$ . For each component  $i$ , the multivariate Gaussian covariance matrices  $\mathbf{C}_i$  and  $\boldsymbol{\zeta}_i$  are computed via

$$\left\{ \begin{array}{l} \mathbf{C}_i = \mathcal{K}_i(\mathbf{I}, \mathbf{I}) \\ \mathbf{m}_i = {}' \mathcal{K}_i(\mathbf{I}, \mathbf{I}) \mathcal{K}_i(\mathbf{I}, \mathbf{I})^{-1} \\ \boldsymbol{\zeta}_i = \mathcal{K}_i''(\mathbf{I}, \mathbf{I}) - {}' \mathcal{K}_i(\mathbf{I}, \mathbf{I}) \mathcal{K}_i(\mathbf{I}, \mathbf{I})^{-1} \mathcal{K}_i'(\mathbf{I}, \mathbf{I}) \end{array} \right.,$$

where  $'\mathcal{K}_i = \frac{\partial}{\partial s} \mathcal{K}_i(s, t)$ ,  $\mathcal{K}_i' = \frac{\partial}{\partial t} \mathcal{K}_i(s, t)$ , and  $\mathcal{K}_i'' = \frac{\partial^2}{\partial s \partial t} \mathcal{K}_i(s, t)$ .

### S3 Conditions of Theorems 1 and 2

We state the required differentiability assumptions in Conditions 1–3.

**Condition 1.** For any  $i \in \{1, \dots, m\}$  and  $t \in \mathbf{I}$ ,  $f_i(\mathbf{x}(t), \mathbf{x}(t - \boldsymbol{\tau}), \boldsymbol{\theta}, t)$  is continuously differentiable with respect to  $\mathbf{x}(t - \boldsymbol{\tau})$  in a neighborhood of  $\mathbf{x}(t - \boldsymbol{\tau})$ .

**Condition 2.** For any  $i, d \in \{1, \dots, m\}$ , any  $x_{d,0}$  lying in a neighborhood of  $x_d(t - \tau_d)$ , and  $t \in \mathbf{I}$ , there exists  $C_1 > 0$ , such that

$$\left| \frac{\partial f_i(\mathbf{x}(t), \mathbf{x}(t - \boldsymbol{\tau}), \boldsymbol{\theta}, t)}{\partial x_d(t - \tau_d)} \Big|_{x_d(t - \tau_d) = x_{d,0}} \right| < C_1.$$

**Condition 3.** For any  $i, d, k \in \{1, \dots, m\}$ , any  $x_{k,0}$  lying in a neighbourhood of  $x_k(t - \tau_k)$ , and  $t \in \mathbf{I}$ , there exists  $C_2 > 0$ , such that

$$\left| \frac{\partial^2 f_i(\mathbf{x}(t), \mathbf{x}(t - \boldsymbol{\tau}), \boldsymbol{\theta}, t)}{\partial x_d(t - \tau_d) \partial x_k(t - \tau_k)} \Big|_{\substack{x_d(t - \tau_d) = x_{d,0} \\ x_k(t - \tau_k) = x_{k,0}}} \right| < C_2.$$

Condition 1 guarantees the differentiability of  $f_i(\mathbf{x}(t), \mathbf{x}(t - \boldsymbol{\tau}), \boldsymbol{\theta}, t)$  with respect to  $\mathbf{x}(t - \boldsymbol{\tau})$ . Conditions 2 and 3 ensure the boundedness of the first and second derivatives in a neighborhood of  $\mathbf{x}(t - \boldsymbol{\tau})$ . Under these three conditions, we have Theorems 1 and 2, which respectively provide the asymptotic error bounds for  $f_i(\mathbf{x}(t), \hat{\mathbf{x}}(t - \boldsymbol{\tau}), \boldsymbol{\theta}, t)$  using the conditional expectation and linear interpolation schemes.

## S4 Proof of Theorem 1

Consider  $\Omega = [0, T]$  as the time interval of interest. Without loss of generality, assume  $T = 1$  and suppose the discretization set  $\mathbf{I} = \{t_1, \dots, t_n\}$  follows a uniform space-filling design, such that  $t_j = \frac{j}{n}$ , for  $j = 1, \dots, n$ .

Using conditional expectation to approximate the historical output  $x_i(t - \tau_i)$  if  $t > \tau_i$ , we may write  $\hat{x}_i(t - \tau_i) = E[x_i(t - \tau_i)|x_i(\mathbf{I})]$  for  $i = 1, 2, \dots, m$ . We denote  $\hat{\mathbf{x}}(t - \boldsymbol{\tau}) = (\hat{x}_1(t - \tau_1), \dots, \hat{x}_m(t - \tau_m))$ . Recall that if  $t < \tau_i$ , we set  $\hat{x}_i(t - \tau_i) = x(0)$  based on the history function.

Define the demeaned Gaussian process  $z_i(t) = x_i(t) - \mu_i(t)$ , then  $z_i(t) \sim \mathcal{GP}(\mathbf{0}, \mathcal{K}_i)$ . Then according to Example 1 from Wang et al. (2020),

$$\sup_{r \in [0, 1]} \left| E[z_i(r)|z_i(\mathbf{I})] - z_i(r) \right| = O_P \left( n^{-\nu} (\log(n))^{\frac{1}{2}} \right).$$

Substituting the definition of  $z_i(t)$ , we obtain

$$\begin{aligned} \sup_{r \in [0, 1]} \left| E[z_i(r)|z_i(\mathbf{I})] - z_i(r) \right| &= \sup_{r \in [0, 1]} \left| E[x_i(r) - \mu_i(t)|z_i(\mathbf{I})] - x_i(r) + \mu_i(t) \right| \\ &= \sup_{r \in [0, 1]} \left| \hat{x}_i(r) - x_i(r) \right| \\ &= O_P \left( n^{-\nu} (\log(n))^{\frac{1}{2}} \right). \end{aligned}$$

Clearly, when  $\nu = 2.01$  or  $2.5$ ,  $n^{-\nu} \log^{1/2}(n) \rightarrow 0$  as  $n \rightarrow \infty$ , so that

$$\sup_{r \in [0, 1]} \left| E[x_i(r)|x_i(\mathbf{I})] - x_i(r) \right| = o_p(1).$$

Next, consider the uniform deviation

$$\max_{t \in I} \left| f_i(\mathbf{x}(t), \mathbf{x}(t - \tau), \boldsymbol{\theta}, t) - f_i(\mathbf{x}(t), \hat{\mathbf{x}}(t - \tau), \boldsymbol{\theta}, t) \right|.$$

Expanding  $f_i(\mathbf{x}(t), \mathbf{x}(t - \tau), \boldsymbol{\theta}, t)$  around  $\mathbf{x}(t - \tau) = \hat{\mathbf{x}}(t - \tau)$ ,

$$\begin{aligned} f_i(\mathbf{x}(t), \mathbf{x}(t - \tau), \boldsymbol{\theta}, t) &= f_i(\mathbf{x}(t), \hat{\mathbf{x}}(t - \tau), \boldsymbol{\theta}, t) \\ &+ \sum_{d=1}^m \frac{\partial f_i(\mathbf{x}(t), \mathbf{x}(t - \tau), \boldsymbol{\theta}, t)}{\partial x_d(t - \tau_d)} \Big|_{x_d(t - \tau_d) = \hat{x}_d(t - \tau_d)} \cdot (x_d(t - \tau_d) - \hat{x}_d(t - \tau_d)) + err \end{aligned}$$

Since  $n^{-\nu} \log^{1/2}(n) \rightarrow 0$  as  $n \rightarrow \infty$ , we have  $|\hat{x}_i(r) - x_i(r)| = o_p(1)$ . By the mean value theorem, there exists a  $\xi_d$  lying between  $x_d(t - \tau_d)$  and  $\hat{x}_d(t - \tau_d)$  for  $d = 1, \dots, m$ , such that

$$\begin{aligned} |err| &= \left| \frac{1}{2} \sum_{d=1}^m \sum_{k=1}^m \frac{\partial^2 f_i(\mathbf{x}(t), \mathbf{x}(t - \tau), \boldsymbol{\theta}, t)}{\partial x_d(t - \tau_d) \partial x_k(t - \tau_k)} \Big|_{x_d(t - \tau_d) = \xi_d, x_k(t - \tau_k) = \xi_k} \right. \\ &\quad \left. \cdot (x_d(t - \tau_d) - \hat{x}_d(t - \tau_d)) \cdot (x_k(t - \tau_k) - \hat{x}_k(t - \tau_k)) \right|. \end{aligned}$$

Using Condition 3 yields

$$\begin{aligned} |err| &< \frac{1}{2} m^2 \cdot C_2 \cdot \sup_{(t - \tau_d) \in [0, 1]} |(x_d(t - \tau_d) - \hat{x}_d(t - \tau_d))| \cdot \sup_{(t - \tau_k) \in [0, 1]} |(x_k(t - \tau_k) - \hat{x}_k(t - \tau_k))| \\ &= O_P(n^{-2\nu} \log(n)). \end{aligned}$$

Similarly, for the first-order term, we use Condition 2, obtaining

$$\begin{aligned} &\left| \sum_{d=1}^m \frac{\partial f_i(\mathbf{x}(t), \mathbf{x}(t - \tau), \boldsymbol{\theta}, t)}{\partial x_d(t - \tau_d)} \Big|_{x_d(t - \tau_d) = \hat{x}_d(t - \tau_d)} \cdot (x_d(t - \tau_d) - \hat{x}_d(t - \tau_d)) \right| \\ &= O_P(n^{-\nu} (\log(n))^{\frac{1}{2}}). \end{aligned}$$

Therefore, by the triangle inequality,

$$\begin{aligned} & \max_{t \in \mathbf{I}} \left| f_i(\mathbf{x}(t), \mathbf{x}(t - \boldsymbol{\tau}), \boldsymbol{\theta}, t) - f_i(\mathbf{x}(t), \hat{\mathbf{x}}(t - \boldsymbol{\tau}), \boldsymbol{\theta}, t) \right| \\ &= O_P \left( n^{-\nu} (\log(n))^{\frac{1}{2}} \right) + O_P \left( n^{-2\nu} \log(n) \right) \\ &= O_P \left( n^{-\nu} (\log(n))^{\frac{1}{2}} \right). \end{aligned}$$

## S5 Proof of Theorem 2

Take the settings of the space-filling design to be same as in S4. We use linear interpolation to approximate the historical output  $x_i(t - \tau_i)$  if  $t > \tau_i$  for  $i = 1, 2, \dots, m$  as described in Section 3.2. Denote  $\hat{x}_i(t)$  as the linear interpolated values at time points  $t$  and  $\hat{\mathbf{x}}(t - \boldsymbol{\tau}) = (\hat{x}_1(t - \tau_1), \dots, \hat{x}_m(t - \tau_m))$ . Let  $z_i(t) = x_i(t) - \mu_i(t)$  denote the demeaned Gaussian process. Let  $\delta_i(t) = x_i(t) - \hat{x}_i(t) = z_i(t) - \hat{z}_i(t)$  be the deviation process that quantifies the linear interpolation error, and  $\max_{t \in [0, 1]} |\delta_i(t)|$  denotes the uniform deviation. We begin by verifying two lemmas.

**Lemma 1.** *For any  $i \in 1, 2, \dots, m$ , there exists  $L > 0$ ,  $2 \geq \alpha > 0$ , such that*

$$E \left[ (z'_i(t) - z'_i(s))^2 \right] \leq L^2 |t - s|^\alpha.$$

*Proof.* Since increasing  $\nu$  will increase the smoothness of the Matern covariance function, to have the lemma hold for  $\nu = 2.01$  and  $\nu = 2.5$ , it suffices to verify the lemma for the Matern covariance with any smaller  $\nu$ . We will prove this lemma under a rougher covariance function, namely the Matern with  $\nu = 3/2$ ,

which is twice-differentiable so that  $E [(z'_i(t) - z'_i(s))^2]$  is well-defined. Note that the Matern covariance function is  $2m$  times differentiable if and only if  $\nu > m$ ,  $m \in \mathbb{N}^+$  (Stein, 2012).

Suppose  $z_i(t)$  is a zero mean Gaussian process, with a Matern covariance of  $\nu = 3/2$  that can be written as

$$\mathcal{K}_i(d) = \sigma^2 \left( 1 + \frac{\sqrt{3}d}{\rho} \right) \exp \left( -\frac{\sqrt{3}d}{\rho} \right),$$

where  $d = |t - s|$ , and  $0 \leq d \leq 1$ . Consider

$$E [(z'_i(t) - z'_i(s))^2] = 2 \cdot [\mathcal{K}'_i(0) - \mathcal{K}'_i(d)],$$

where  $\mathcal{K}'_i(d)$  is the covariance function for  $z'_i(t)$ . Since this Matern covariance function is twice-differentiable, the covariance function for  $z'_i$  is  $\frac{\partial^2 \mathcal{K}_i(s,t)}{\partial s \partial t} = \frac{\partial^2 \mathcal{K}_i(d)}{\partial d^2}(d) \cdot \frac{\partial d}{\partial s} \cdot \frac{\partial d}{\partial t} = -\frac{\partial^2 \mathcal{K}_i(d)}{\partial d^2}$ .

If  $t = s$ , the inequality  $L^2|t - s|^\alpha \geq 0$  clearly always holds. Now considering  $t \neq s$ ,  $\mathcal{K}'_i(0) = \frac{\sigma^2}{\rho^2} \cdot \frac{\nu}{\nu-1} |_{\nu=1.5} = 3\frac{\sigma^2}{\rho^2}$ , and

$$\begin{aligned} 2 \cdot [\mathcal{K}'_i(0) - \mathcal{K}'_i(d)] &= 2 \cdot \left[ 3\frac{\sigma^2}{\rho^2} + \frac{\partial^2 \mathcal{K}}{\partial d^2} \right] \\ &= 2 \cdot \left[ 3\frac{\sigma^2}{\rho^2} - \frac{3\sigma^2 e^{-\frac{\sqrt{3}d}{\rho}} (\rho - \sqrt{3}d)}{\rho^3} \right] \\ &= 6\frac{\sigma^2}{\rho^2} \left[ 1 - e^{-\frac{\sqrt{3}d}{\rho}} \left( 1 - \frac{\sqrt{3}d}{\rho} \right) \right]. \end{aligned}$$

Let  $\frac{\sqrt{3}d}{\rho} = q$ , then  $0 < q \leq \frac{\sqrt{3}}{\rho}$ , and

$$6\frac{\sigma^2}{\rho^2} \left[ 1 - e^{-\frac{\sqrt{3}d}{\rho}} \left( 1 - \frac{\sqrt{3}d}{\rho} \right) \right] = 6\frac{\sigma^2}{\rho^2} [1 - e^{-q} (1 - q)]. \quad (\text{S5.2})$$

Now let  $w(q) = 1 - e^{-q}(1 - q) - 4 \cdot q$ .  $w'(q) = e^{-q}(2 - q) - 4 < e^0(2) - 4 < 0$ . Then  $w(q)$  is monotonically decreasing when  $q > 0$  and  $w(q) < w(0) = 0$ . Thus,  $1 - e^{-q}(1 - q) < 4 \cdot q$  when  $q > 0$ . Returning to Equation S5.2, we have  $6 \frac{\sigma^2}{\rho^2} \cdot w(q) < 0$ , or equivalently,

$$6 \frac{\sigma^2}{\rho^2} [1 - e^{-q}(1 - q)] < 24 \frac{\sigma^2}{\rho^2} \cdot q.$$

Therefore, we can choose  $\alpha = 1$  and  $L = \sqrt{\frac{24\sqrt{3}\sigma^2}{\rho^3}}$  to satisfy

$$E [(z'_i(t) - z'_i(s))^2] = 2 \cdot [\mathcal{K}'_i(0) - \mathcal{K}'_i(d)] \leq L^2 |t - s|^\alpha,$$

which completes the proof. □

**Lemma 2.** *When Lemma 1 holds,  $\max_{t \in [0,1]} |\delta_i(t)| = O_P \left( (\log(n))^{\frac{1}{2}} n^{-1-\alpha/2} \right)$ .*

*Proof.* Let  $u = C(2 \log(n))^{\frac{1}{2}}$ . According to Theorem 4 from Seleznev (1996) and Lemma 1, for any  $u > 2$ , we will have that

$$P \left( \max_{t \in [0,1]} |\delta_i(t)| > \epsilon(n) \right) \leq C^2 K_\alpha C_\alpha^2 n (2 \log(n))^{1/\alpha} \Psi(u)$$

where  $B_\alpha = 2^{-\alpha/2}$ ,  $C_\alpha = 2^{1+\alpha/2}$ , and  $K_\alpha = 2 \exp(5 \cdot 2/\alpha) (2\alpha)^{1/\alpha}$ .  $\Psi(u) = \phi(u)/u$ , where  $\phi(u)$  is the standard normal density function.

By setting  $\epsilon(n) = C B_\alpha L (2 \log(n))^{\frac{1}{2}} n^{-1+\alpha/2}$  (Seleznev, 1996), where  $C$  is an arbitrary constant larger than 1,

$$\begin{aligned} P \left( \max_{t \in [0,1]} |\delta_i(t)| > \epsilon(n) \right) &\leq C^2 K_\alpha C_\alpha^2 n (2 \log(n))^{1/\alpha} \Psi(u) \\ \Leftrightarrow P \left( \frac{\max_{t \in [0,1]} |\delta_i(t)|}{(\log(n))^{\frac{1}{2}} n^{-1-\alpha/2}} > \sqrt{2} C B_\alpha L \right) &\leq C^2 K_\alpha C_\alpha^2 n (2 \log(n))^{1/\alpha} \Psi(u), \end{aligned} \tag{S5.3}$$

and we further investigate the asymptotic behavior of  $C^2 K_\alpha C_\alpha^2 n(\log(n))^{1/\alpha} \Psi(u)$ , or simply  $n(\log(n))^{1/\alpha} \Psi(u)$  after dropping the constant terms.

Note that  $\Psi(u) = \phi(u)/u$  is asymptotically equivalent to  $\Phi(u)$ , where  $\Phi(u)$  is the standard normal distribution function, according to the asymptotic behavior of Mill's ratio of the standard normal distribution (Wainwright, 2019). Thus, we alternatively explore the asymptotic behavior of  $n(\log(n))^{1/\alpha}(1 - \Phi(u))$ . Using the approximation of  $\Phi(u)$  from Bowling et al. (2009) yields

$$\begin{aligned} n(\log(n))^{1/\alpha}(1 - \Phi(u)) &\approx n \log(n) \left( 1 - \frac{1}{1 + \exp(-1.5976u - 0.07056u^3)} \right) \\ &= \frac{n(\log(n))^{1/\alpha}}{1 + \exp(1.5976u + 0.07056u^3)}. \end{aligned}$$

Replacing  $u = C(2 \log(n))^{\frac{1}{2}}$  the equation above becomes

$$\frac{n(\log(n))^{1/\alpha}}{1 + \exp\left(1.5976C(2 \log(n))^{\frac{1}{2}} + 0.07056C(2 \log(n))^{\frac{3}{2}}\right)}$$

By L'Hospital's rule and some algebra,

$$\begin{aligned} &\lim_{n \rightarrow \infty} \frac{n(\log(n))^{1/\alpha}}{1 + \exp\left(1.5976C(2 \log(n))^{\frac{1}{2}} + 0.07056C(2 \log(n))^{\frac{3}{2}}\right)} \\ &= \frac{6250\sqrt{2}n(\alpha \log(n) + 1)}{C\alpha \log(n)^{\frac{\alpha-2}{2\alpha}} \exp\left(\frac{\sqrt{2}C\sqrt{\log(n)}(882\log(n)+9985)}{6250}\right) (2646 \log(n) + 9985)} \\ &= \frac{6250\sqrt{2}(\alpha \log(n) + 1)}{C\alpha \log(n)^{\frac{\alpha-2}{2\alpha}} \exp\left(\frac{9985}{6250}\right) n^{\sqrt{2}C\frac{882}{6250}\sqrt{\log(n)}-1} (2646 \log(n) + 9985)} \\ &= 0. \end{aligned}$$

As  $n \rightarrow \infty$ ,  $n(\log(n))^{1/\alpha}(1 - \Phi(u)) \rightarrow 0$ , and  $C^2 K_\alpha C_\alpha^2 n(2 \log(n))^{1/\alpha} \Psi(u) \rightarrow 0$ . Then, for any  $\epsilon > 0$ , there exists an  $N_1$ , such that for all  $n > N_1$ , we

have  $C^2 K_\alpha C_\alpha^2 n (2 \log(n))^{1/\alpha} \Psi(u) < \epsilon$ . Denote  $M = \sqrt{2} C B_\alpha L < \infty$  and  $a_n = (\log(n))^{\frac{1}{2}} n^{-1-\alpha/2}$ . For all  $n > N_1$ ,

$$P \left( \left| \frac{\max_{t \in [0,1]} |\delta_i(t)|}{a_n} \right| > M \right) \leq C^2 K_\alpha C_\alpha^2 n (2 \log(n))^{1/\alpha} \Psi(u) < \epsilon.$$

Therefore, by definition,  $\max_{t \in [0,1]} |\delta_i(t)| = O_P \left( (\log(n))^{\frac{1}{2}} n^{-1-\alpha/2} \right)$ , which completes the proof of this part. □

Using Lemma 1 and 2, we can now show that Theorem 2 holds.

*Proof.* Consider the uniform deviation

$$\max_{t \in I} \left| f_i(\mathbf{x}(t), \mathbf{x}(t - \boldsymbol{\tau}), \boldsymbol{\theta}, t) - f_i(\mathbf{x}(t), \hat{\mathbf{x}}(t - \boldsymbol{\tau}), \boldsymbol{\theta}, t) \right|.$$

Expanding  $f_i(\mathbf{x}(t), \mathbf{x}(t - \boldsymbol{\tau}), \boldsymbol{\theta}, t)$  around  $\mathbf{x}(t - \boldsymbol{\tau}) = \hat{\mathbf{x}}(t - \boldsymbol{\tau})$ , we obtain

$$\begin{aligned} f_i(\mathbf{x}(t), \mathbf{x}(t - \boldsymbol{\tau}), \boldsymbol{\theta}, t) &= f_i(\mathbf{x}(t), \hat{\mathbf{x}}(t - \boldsymbol{\tau}), \boldsymbol{\theta}, t) \\ &\quad + \sum_{d=1}^m \frac{\partial f_i(\mathbf{x}(t), \mathbf{x}(t - \boldsymbol{\tau}), \boldsymbol{\theta}, t)}{\partial x_d(t - \boldsymbol{\tau})} \Big|_{x_d(t - \boldsymbol{\tau}) = \hat{x}_d(t - \boldsymbol{\tau})} \cdot (x_d(t - \boldsymbol{\tau}) - \hat{x}_d(t - \boldsymbol{\tau})) + err \end{aligned}$$

Clearly, as  $n \rightarrow \infty$ ,  $(\log(n))^{\frac{1}{2}} n^{-1-\alpha/2} \rightarrow 0$ . For any  $t \in [0, 1]$ ,  $|x_i(t) - \hat{x}_i(t)| \leq \max_{t \in [0,1]} |\delta_i(t)|$ , and  $|x_i(t) - \hat{x}_i(t)| = o_p(1)$ .

By the mean value theorem, there exists a  $\xi_d$  for  $d = 1, 2, \dots, m$ , lying

between  $x_d(t - \boldsymbol{\tau}_d)$  and  $\hat{x}_d(t - \boldsymbol{\tau}_d)$  such that

$$\begin{aligned} |err| &= \left| \frac{1}{2} \sum_{d=1}^m \sum_{k=1}^m \frac{\partial^2 f_i(\mathbf{x}(t), \mathbf{x}(t - \boldsymbol{\tau}), \boldsymbol{\theta}, t)}{\partial x_d(t - \boldsymbol{\tau}_d) \partial x_k(t - \boldsymbol{\tau}_k)} \Big|_{\substack{x_d(t - \boldsymbol{\tau}) = \xi_d, \\ x_k(t - \boldsymbol{\tau}) = \xi_k}} \cdot (x_d(t - \boldsymbol{\tau}_d) - \hat{x}_d(t - \boldsymbol{\tau}_d))(x_k(t - \boldsymbol{\tau}_k) - \hat{x}_k(t - \boldsymbol{\tau}_k)) \right| \\ &\leq \left| \frac{1}{2} \sum_{d=1}^m \sum_{k=1}^m \frac{\partial^2 f_i(\mathbf{x}(t), \mathbf{x}(t - \boldsymbol{\tau}), \boldsymbol{\theta}, t)}{\partial x_d(t - \boldsymbol{\tau}_d) \partial x_k(t - \boldsymbol{\tau}_k)} \Big|_{\substack{x_d(t - \boldsymbol{\tau}_d) = \xi_d, \\ x_k(t - \boldsymbol{\tau}_k) = \xi_k}} \right| \cdot \max_{t - \boldsymbol{\tau}_d \in [0,1]} |\delta_d(t - \boldsymbol{\tau}_d)| \cdot \max_{t - \boldsymbol{\tau}_k \in [0,1]} |\delta_k(t - \boldsymbol{\tau}_k)| \end{aligned}$$

After using Condition 3, the error term will be

$$\begin{aligned} |err| &< \frac{1}{2} \cdot m^2 \cdot C_2 \cdot \max_{t \in [0,1]} |\delta_d(t)| \cdot \max_{t \in [0,1]} |\delta_k(t)| \\ &= O_P(\log(n) \cdot n^{-2-\alpha}). \end{aligned}$$

Similarly, for the first-order term, we apply Condition 2,

$$\begin{aligned} &\left| \sum_{d=1}^m \frac{\partial f_i(\mathbf{x}(t), \mathbf{x}(t-\tau), \boldsymbol{\theta}, t)}{\partial x_d(t-\tau_d)} \Big|_{x_d(t-\tau_d)=\hat{x}_d(t-\tau_d)} \cdot (x_d(t-\tau_d) - \hat{x}_d(t-\tau_d)) \right| \\ &\leq \left| \sum_{d=1}^m \frac{\partial f_i(\mathbf{x}(t), \mathbf{x}(t-\tau), \boldsymbol{\theta}, t)}{\partial x_d(t-\tau_d)} \Big|_{x_d(t-\tau)=\hat{x}_d(t-\tau_d)} \cdot \max_{t-\tau_d \in [0,1]} |\delta_d(t-\tau_d)| \right| \\ &= O_P\left((\log(n))^{\frac{1}{2}} n^{-1-\alpha/2}\right). \end{aligned}$$

Therefore, for any  $i \in \{1, 2, \dots, m\}$ , by the triangle inequality,

$$\begin{aligned} &\max_{t \in \mathbf{I}} \left| f_i(\mathbf{x}(t), \mathbf{x}(t-\tau), \boldsymbol{\theta}, t) - f_i(\mathbf{x}(t), \hat{\mathbf{x}}(t-\tau), \boldsymbol{\theta}, t) \right| \\ &= O_P\left((\log(n))^{\frac{1}{2}} n^{-1-\alpha/2}\right). \end{aligned}$$

Finally, since we have shown that taking  $\alpha = 1$  satisfies Lemma 1, the proof is complete.  $\square$

## S6 Numerical Validation of Approximation Schemes

The theoretical results of Section 3.2 indicate that MAGIDDE with approximation schemes for the historical outputs in Equations (9) and (10) is reasonable provided that the discretization set  $\mathbf{I}$  is sufficiently dense. This section provides numerical validation, by examining the effect of  $|\mathbf{I}|$  on the parameter estimates, inferred trajectories, and computation time under the three schemes for histori-

cal outputs described in Section 3: (i) fully Bayesian framework, (ii) conditional expectation, (iii) linear interpolation.

We generate 100 simulated datasets based on the setup for the log-transformed Hutchinson’s equation, given by

$$dN(t)/dt = r [1 - \exp(N(t - \tau))/(1000 \cdot K)].$$

More details about this benchmark system are provided in Section 4.1. For numerical validation, we consider the scenario with 16 sparse and equally-spaced observation time points  $\gamma = \{0, 2, 4, \dots, 30\}$ . We vary the denseness of  $|\mathbf{I}|$  by adding 0, 1, 3, 7 equally-spaced discretization points between each pair of adjacent observation time points, so that the cardinalities of the corresponding discretization sets are  $|\mathbf{I}| = \{16, 31, 61, 121\}$ . For the purpose of creating a simple experiment to compare schemes and discretization sets, in this section only we treat  $\sigma^2$  as known, set the starting values for  $r, K, \tau, \mathbf{x}(\mathbf{I})$  at the truth, and choose  $\nu = 2.5$  for the Matern covariance for faster computation. Uniform priors over  $[0, 5]$  are placed on  $r, K$ , and  $\tau$ . We set the Matern covariance hyper-parameters  $\tilde{\phi}_{1,1}$  and  $\tilde{\phi}_{1,2}$  as described in Section 3.3.

For this experiment, the posterior distributions for the fully Bayesian framework, linear interpolation scheme and conditional expectation scheme are all implemented in *Stan* (Carpenter et al., 2017), which uses NUTS (No-U-Turn sampler, Hoffman et al. (2014)) as the default sampling algorithm. For each

scheme, we run 10,000 MCMC iterations with the first 5,000 discarded as burn-in.

Table 1 summarizes the parameter estimates from the three schemes, across the 100 simulated datasets. We visualize the parameter estimates, and RMSEs between the inferred trajectories and the truth, for each simulation run using side-by-side boxplots in Figures 2, 3. First, considering the effect of  $|\mathbf{I}|$ , the results from all three schemes show a clear decreasing trend in both RMSEs of parameters and inferred trajectories as  $|\mathbf{I}|$  increases from 16 to 61. At  $|\mathbf{I}| = 61$ , all three schemes provide similarly low RMSEs of the parameters and inferred trajectories. A further increase to  $|\mathbf{I}| = 121$  yields largely similar parameter estimates as  $|\mathbf{I}| = 61$  in Figure 2. These results indicate that all the schemes, including fully Bayesian, require  $|\mathbf{I}| = 61$  as a sufficiently dense discretization set to achieve stable inference for this problem. Second, we consider the effect of the approximation schemes for the historical outputs. When  $|\mathbf{I}| = 16$ , the fully Bayesian framework outperforms the two approximation schemes to recover the system, and linear interpolation shows a significant deviation from the other schemes. When  $|\mathbf{I}| = 31$ , the conditional expectation scheme already performs similarly to fully Bayesian, while linear interpolation still has slightly higher error. This suggests the convergence rate of conditional expectation is faster than linear interpolation, as supported by the results of Theorems 1 and 2. Once

$|\mathbf{I}|$  reaches 61, the inferences from the three schemes become substantively identical. This simulation result corroborates the theory, that an increase in the number of discretization points will lead to a more accurate approximation of  $f_i(\mathbf{x}(t), \hat{\mathbf{x}}(t - \tau), \boldsymbol{\theta}, t)$ , and in turn the posterior distributions of  $r, K, \tau, \mathbf{x}(\mathbf{I})$  also become largely indistinguishable between the fully Bayesian and approximate schemes. Third, computation times follow an expected pattern: linear interpolation scales most favorably with  $|\mathbf{I}|$ , while the other schemes become increasingly slow for larger  $|\mathbf{I}|$ . While approximation via conditional expectation is about one to two orders of magnitude faster than the fully Bayesian framework, both are hindered by the dense matrix operations required. For  $|\mathbf{I}| = 61$ , which would be recommended for this problem to achieve stable inference (regardless of the scheme used), linear interpolation is about 200 times faster than fully Bayesian and 4 times faster than conditional expectation. This suggests the significantly faster computation speed of linear interpolation is a worthwhile trade-off in practice, even though conditional expectation has a faster theoretical convergence rate. Therefore, we recommend MAGIDDE with the linear interpolation scheme for practical implementation of the method.

Table 1: Average parameter estimates (with RMSEs over 100 simulated datasets in parentheses) and average running time (in minutes, on a single CPU core) for the log-transformed Hutchinson’s equation, comparing the three schemes for historical outputs. Bold highlights the best scheme with respect to the lowest values of parameter RMSEs and shortest running time.

$ \mathcal{I} $	Scheme	$r$	$K$	$\tau$	$N(0)$	Runtime
16	Fully Bayesian	0.769( <b>0.033</b> )	1.990( <b>0.068</b> )	2.976( <b>0.044</b> )	8.128( <b>0.085</b> )	0.61
	Conditional Expectation	0.770(0.033)	2.046(0.082)	2.878(0.132)	8.125(0.087)	0.06
	Linear Interpolation	0.850(0.051)	1.741(0.265)	3.171(0.174)	8.033(0.144)	<b>0.03</b>
31	Fully Bayesian	0.795(0.008)	1.977(0.067)	3.014( <b>0.025</b> )	8.117( <b>0.063</b> )	66.7
	Conditional Expectation	0.795(0.008)	1.977( <b>0.067</b> )	3.015(0.026)	8.116(0.064)	0.29
	Linear Interpolation	0.798( <b>0.007</b> )	1.967(0.071)	3.027(0.035)	8.109(0.069)	<b>0.11</b>
61	Fully Bayesian	0.799(0.004)	1.982(0.064)	3.005( <b>0.013</b> )	8.140(0.045)	251.9
	Conditional Expectation	0.799(0.004)	1.982( <b>0.064</b> )	3.005(0.013)	8.139( <b>0.045</b> )	4.50
	Linear Interpolation	0.800( <b>0.004</b> )	1.979(0.064)	3.007(0.014)	8.137(0.045)	<b>1.14</b>
121	Fully Bayesian	0.799(0.004)	1.986(0.064)	3.003( <b>0.012</b> )	8.148(0.036)	1295.3
	Conditional Expectation	0.799(0.004)	1.985(0.062)	3.003(0.012)	8.148(0.036)	42.2
	Linear Interpolation	0.800( <b>0.004</b> )	1.985( <b>0.062</b> )	3.004(0.012)	8.147( <b>0.036</b> )	<b>8.65</b>

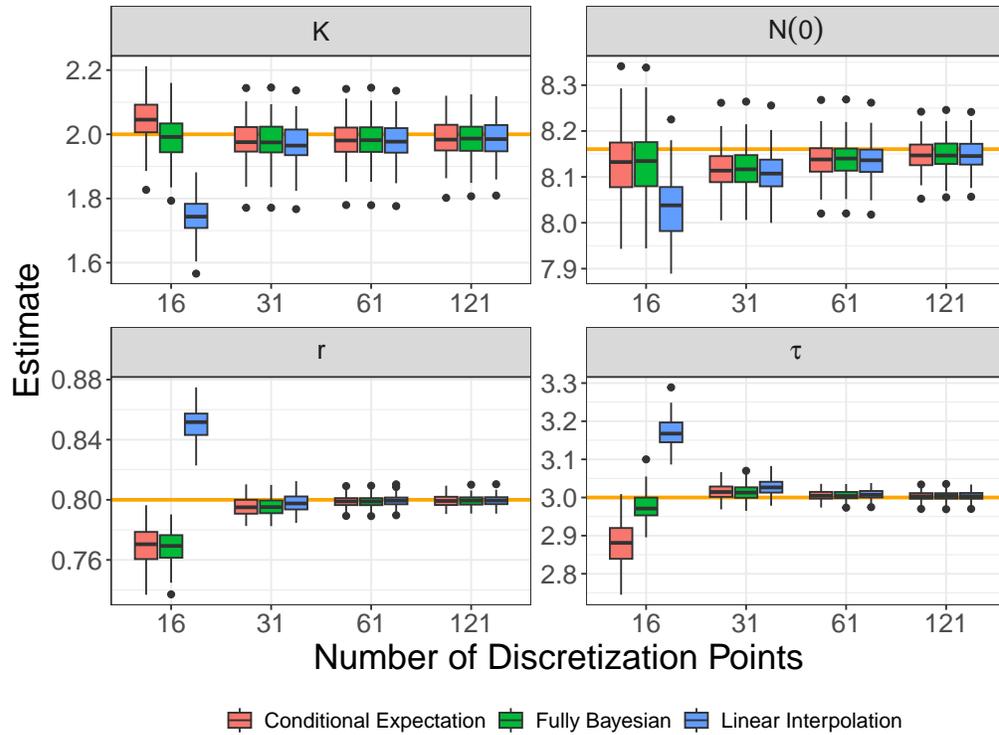


Figure 2: Boxplots of parameter estimates for 100 simulated datasets from the log-transformed Hutchinson's equation, comparing the three schemes for historical outputs. The horizontal orange line indicates the true value of the corresponding parameter.

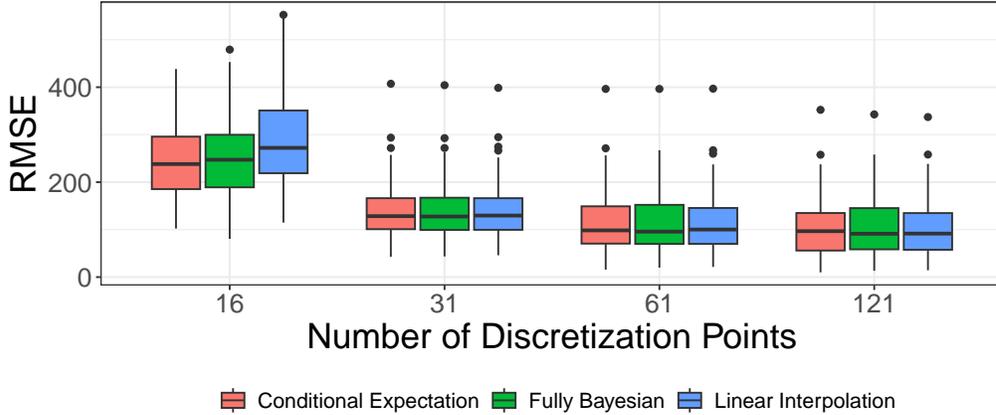


Figure 3: Boxplots of RMSEs between the inferred trajectories and the truth, comparing the three schemes for historical outputs over 100 simulated datasets from Hutchinson’s equation.

## S7 Effect of Denseness of Discretization Points

As noted in Section 3.3 of the main text, MAGIDDE requires choosing a discretization set  $\mathbf{I}$  for practical computation. This section illustrates how this choice impacts computation time and inference accuracy in the context of Hutchinson’s equation.

We consider the sparse 16 observation scenario with  $\gamma = \{0, 2, 4, \dots, 30\}$ . We then vary the denseness of  $\mathbf{I}$  by taking  $|\mathbf{I}| = \{16, 31, 61, 121\}$ , which we obtain by respectively inserting 0, 1, 3, or 7 equally-spaced discretization points between each pair of adjacent observation time points. Otherwise, the settings are the same as in the simulation study in Section 4.1 of the main text. To

run MAGIDDE, we obtain the starting values for HMC sampling and estimate the GP hyperparameters as described in Section 3.3 of the main text. Uniform priors over the positive real numbers are assigned to  $K$ ,  $r$ ,  $\tau$ , and  $\sigma$ . For each  $|\mathbf{I}|$ , we run 40,000 HMC iterations (each with 20 leapfrog steps) and the first 20,000 samples are discarded as warm-up. Table 2 summarizes the parameter inference results and Figure 4 visualizes the changes in the inferred trajectory under varying denseness of discretization sets.

Figure 4 illustrates how the qualitative “rule-of-thumb” for choosing  $\mathbf{I}$  would be applied. When  $|\mathbf{I}| = 16$  or  $31$ , the inferred trajectory looks “bumpy”, i.e., there are not enough points to estimate a smooth curve for the system, and the approximation of the manifold constraint may also be too crude. In contrast,  $|\mathbf{I}| = 61$  can give a relatively smooth trajectory and so would be a reasonable choice based on this qualitative guideline.

The results in Table 2 further illustrate the idea of taking increasingly dense discretization sets (starting from the smallest  $\mathbf{I}$  that includes the observation time points) until the inference results are stable. As we increase  $|\mathbf{I}|$  from 16 to 31 to 61, the parameter estimates shift and the width of the 95% credible interval decreases notably for each parameter. In contrast, a further increase of  $|\mathbf{I}|$  to 121 has minimal impact on the parameter estimates, which suggests  $|\mathbf{I}| = 61$  is sufficient. Indeed, the parameter RMSEs also stabilize at  $|\mathbf{I}| = 61$ , while the

trajectory RMSE for  $|\mathbf{I}| = 121$  only improves slightly compared to  $|\mathbf{I}| = 61$  (but increases the computation time to  $\sim 0.93$  min). Using our MAGIDDE R package, the computation time scales approximately linearly in the number of discretization points.

Table 2: Average parameter estimates (with RMSEs in parentheses), average 95% credible interval (CI) width and average trajectory RMSEs over 300 simulated data sets, with varying number of discretization points  $\{16, 31, 61, 121\}$  equally spaced over time  $[0, 30]$ .

The last column gives the average runtime (in minutes, on a single CPU core).

$ \mathbf{I} $	$r$		$K$		$\tau$		$\sigma$		$N(0)$		Trajectory	Runtime (min.)
	Estimate	CI Width	Estimate	CI Width	Estimate	CI Width	Estimate	CI Width	Estimate	CI Width		
16	0.86(0.06)	0.20	1.76(0.24)	0.44	3.16(0.16)	0.32	0.14(0.06)	0.25	8.01(0.16)	0.39	2975.65	0.10
31	0.80(<0.01)	0.07	1.99(0.07)	0.30	3.01(0.03)	0.15	0.09(0.03)	0.12	8.13(0.06)	0.22	282.85	0.17
61	0.80(<0.01)	0.03	2.00(0.07)	0.28	3.00(0.01)	0.08	0.11(0.02)	0.10	8.16(0.04)	0.20	154.33	0.43
121	0.80(<0.01)	0.02	2.01(0.07)	0.24	3.00(0.01)	0.05	0.11(0.02)	0.10	8.16(0.04)	0.16	123.97	0.93

## S8 Method Comparison Details

We provide the implementation details of MAGIDDE, deBInfer, and SMCDE as described in Section 4.1 for Hutchinson’s equation. We follow the setup in Wang and Cao (2012) and take the time interval of interest as  $t \in [0, 30]$ . Simulations in previous studies focused on a relatively dense observation set; however, estimation performance under more challenging scenarios with sparser observations also deserves attention. Therefore, we consider scenarios with  $|\gamma| = 16, 31, 61,$  and 121 equally-spaced observations over  $[0, 30]$ . To visualize the performance of each

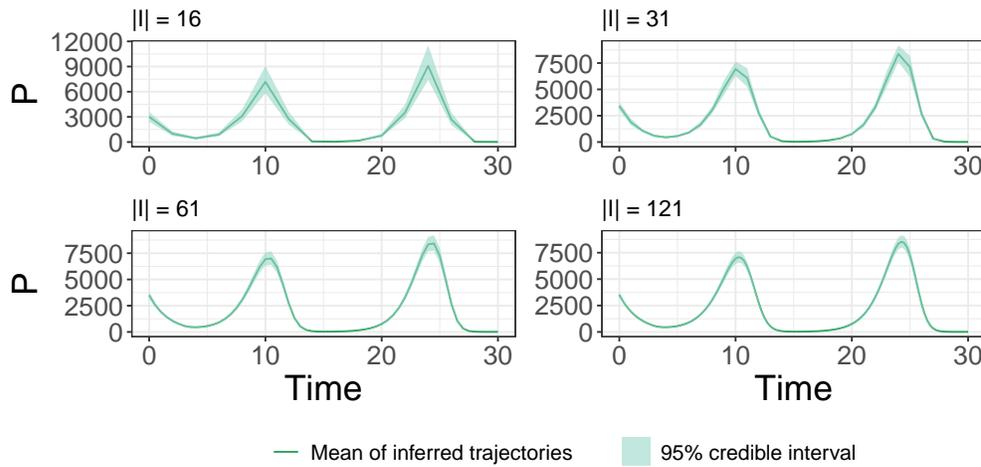


Figure 4: Inferred trajectory of Hutchinson's equation from 16 observations using MAGIDDE under varying denseness of discretization points. The green solid line represents the mean inferred trajectory across 300 simulated datasets. The green shaded area is the 95% pointwise credible interval, constructed by taking the average 0.025 and 0.975 quantiles of the inferred trajectories over the simulated datasets.

method, we present side-by-side boxplots of parameter estimates and trajectory RMSEs. Finally, we also examine the efficacy of applying the NLS approach with a numerical solver to this inference problem.

### S8.1 Method Implementations

We compare MAGIDDE with two other representative methods for DDE inference: the ‘deBInfer’ R package and the semiparametric Bayesian collocation method (SMCDE, Wang et al., 2022). deBInfer uses MCMC to conduct parameter inference with the help of a numerical solver. SMCDE represents the solution trajectory via a basis function expansion and uses SMC to sample the parameters and associated trajectories. For each method, we choose similar flat priors for the model parameters to ensure fair comparison, with detailed implementation steps provided as follows.

For MAGIDDE, we obtain the starting values for HMC sampling and estimate the GP hyperparameters as described in Section 3.3. Uniform priors over the positive real numbers are assigned to  $K$ ,  $r$ ,  $\tau$ , and  $\sigma$ . We run 40,000 HMC iterations (each with 20 leapfrog steps), with the first 20,000 samples discarded as burn-in.

For deBInfer, we construct a normal likelihood function for noisy observations on the log-scale. While we use uniform priors, we restrict their ranges as

follows:  $K$ ,  $r$ , and  $\tau$  are uniform on the interval  $[0, 5]$ ,  $N(0)$  is uniform on  $[5, 10]$ , and  $\sigma$  is uniform on  $[0, 0.2]$ . Without such restrictions, deBInfer frequently cannot converge to a reasonable result; deBInfer employs a variant of the random-walk Metropolis algorithm with a “reflection” mechanism to handle the boundaries of the parameter space. The proposal variance of each parameter is tuned to ensure that convergence is generally achieved in the simulation runs, and we run 40,000 MCMC iterations with the first 20,000 samples discarded as burn-in.

For SMCDE, we follow the authors’ recommendations when running the code. Specifically, one knot is placed at every two time units, the normalized conditional effective sample size threshold is set as 0.9, and resampling threshold is set as 0.5. We use the priors suggested by the authors when they resemble flat priors:  $K$  and  $r$  are  $N(0, 5^2)$  restricted to be positive, and  $\tau$  is uniform on  $[0, 50]$ . For fair comparison, we adjusted the authors’ default inverse-gamma(1, 1) prior for  $\sigma^2$  to be more uniform, with shape parameter 0.001 and scale parameter 0.001. We found that this flatter prior actually led to more accurate parameter estimates, compared to the authors’ default. Note that SMCDE expects conjugate priors for some parameters, and these cannot be easily adjusted to be strictly uniform.

## S8.2 Supplementary Results for Method Comparison

Figures 5 and 6 respectively display the side-by-side boxplots of parameter estimates and trajectory RMSEs for 300 simulated datasets from Hutchison’s equation, as obtained from MAGIDDE, deBInfer, and SMCDE. Among these methods, MAGIDDE has the most favorable performance in estimating the parameters and reconstructing the trajectory. Both MAGIDDE and SMCDE exhibit a similar pattern, that increasing the number of observations leads to more accurate parameter estimates; however, deBInfer does not follow this pattern since the stochastic moves of the random-walk MCMC sampler can be inefficient for exploring the parameter space, regardless of the number of observations. deBInfer is therefore hindered by a number of outliers where its MCMC sampler did not converge to the correct parameter values, despite tuning the proposal distribution; its trajectory RMSEs also have large outliers as a result. SMCDE exhibits some amount of bias in recovering the system, notably in its estimates for  $r$  and  $\tau$  when the observations are sparse. Overall, SMCDE has relatively small parameter RMSEs and the inferred trajectories provided in SMCDE’s output (via its spline approximations) have reasonable RMSEs to the ground truth: using 16, 31, 61, and 121 observations, the average RMSEs of SMCDE’s inferred trajectories are 290.78, 238.51, 193.94, and 152.18, respectively. However, these values exhibit notable differences compared to the trajectory RMSEs presented

in Table 1 of the main text due to the sensitivity of the numerical solution to the parameters and initial conditions. Figure 7 shows an example of SMCDE's inferred trajectory, which aligns reasonably well with the true trajectory. However, despite having relatively accurate parameter estimates, the corresponding reconstructed trajectory has a large RMSE and fails to align with the truth because the solution trajectory is highly sensitive to the parameter values and initial condition.

### S8.3 Parameter Estimation for Hutchinson's Equation using NLS

To apply the NLS approach with a DDE numerical solver, we construct the objective function using the error sum of squares (SSE) and choose L-BFGS-B as the optimization algorithm. We used 300 simulated datasets to study its performance. We set the lower and upper bounds for optimization relatively close to the truth for all of the model parameters:  $K$ ,  $r$ , and  $\tau$  are bounded by  $[0,5]$ . To initialize NLS, we randomly draw starting parameter values from uniform distributions with these bounds, and the starting value for  $N(0)$  is set to be the noisy observation at  $t = 0$ . With a single run of NLS, the optimizer converged to local optima with large SSE values for over 90% of simulated data sets, and these correspond to large trajectory RMSEs that are above 1500; the remaining trajectory RMSEs (i.e., for <10% of simulated datasets) fall below 800, indicating

Figure 5: Boxplots of parameter estimates for 300 simulated datasets from the log-transformed Hutchinson's equation, comparing the MAGIDDE, deBInfer, and SMCDE methods. The horizontal orange line indicates the true value of the corresponding parameter. The horizontal orange line represents the corresponding true values.

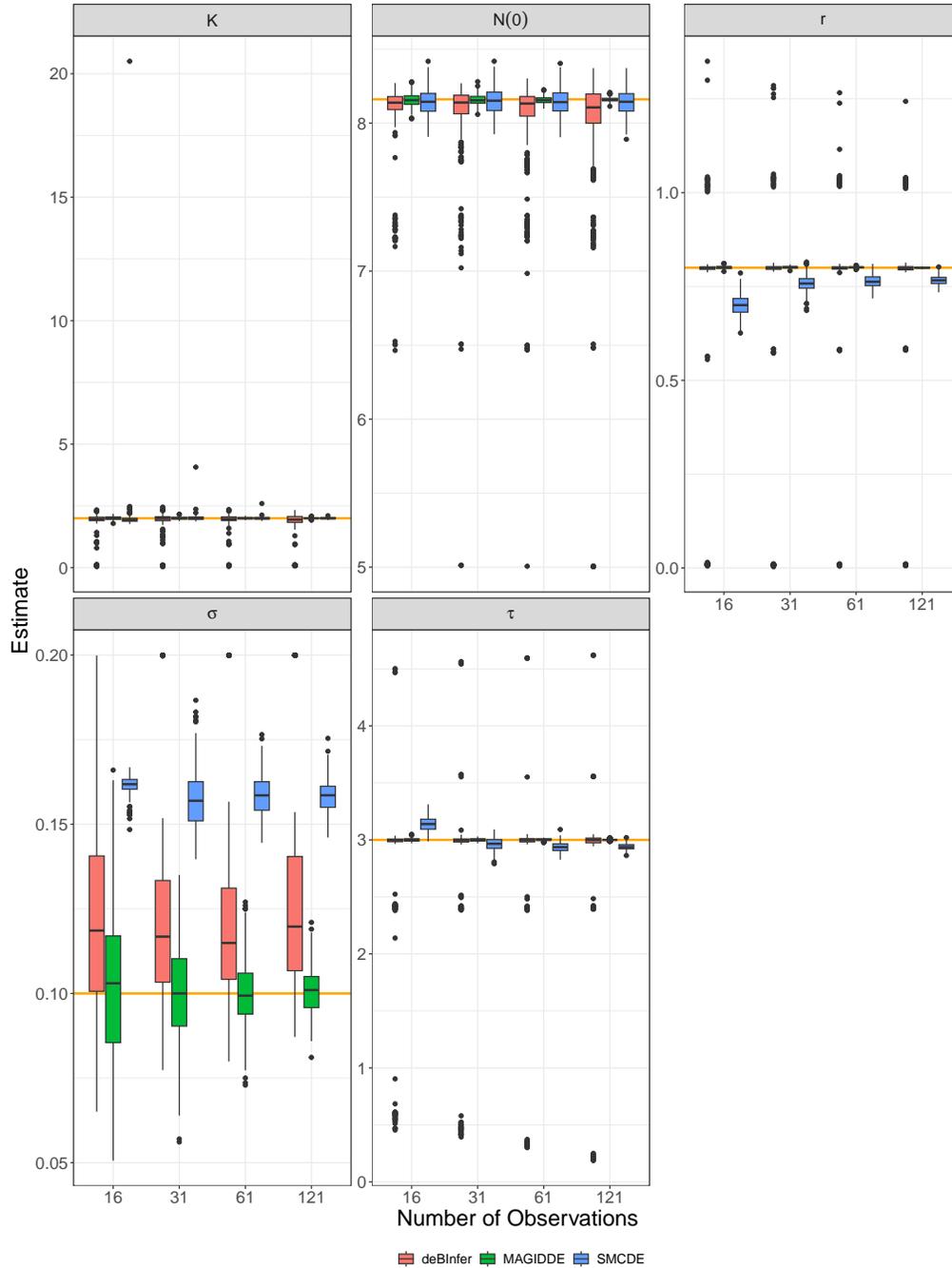
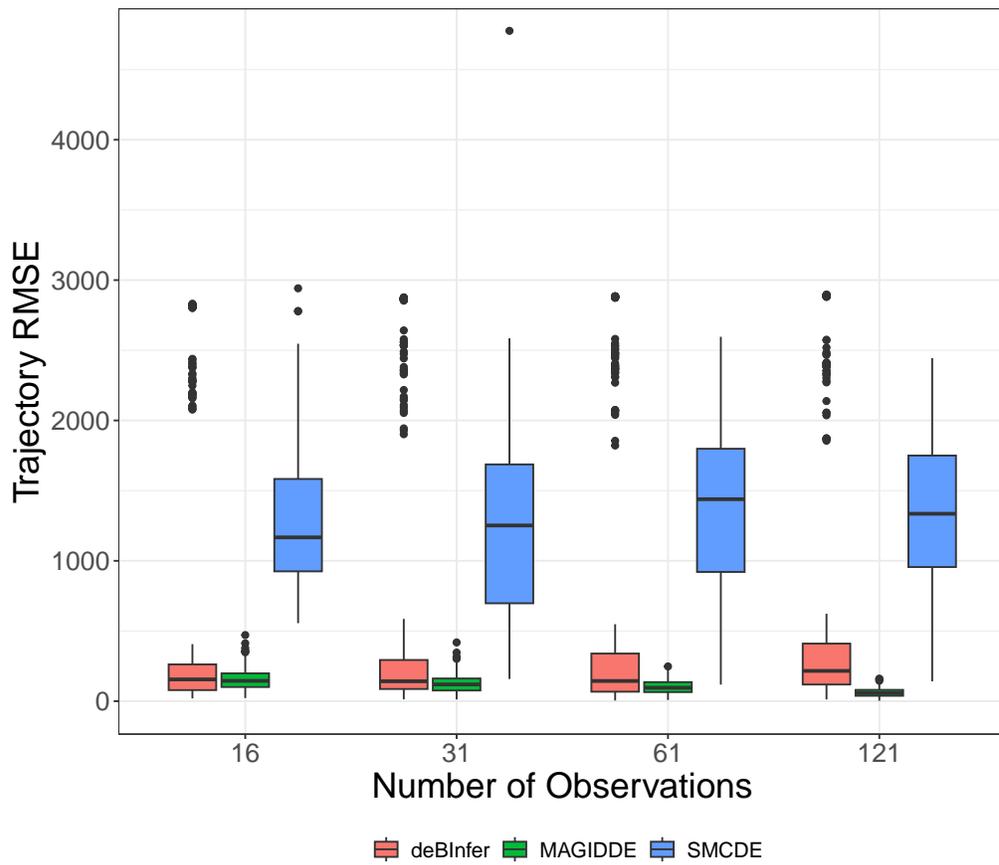


Figure 6: Boxplots of trajectory RMSEs for 300 simulated data sets from Hutchinson's equation, comparing the MAGIDDE, deBInfer, and SMCDE methods. One outlier from SMCDE method using 16 observations with a value of 26225.95 was removed.



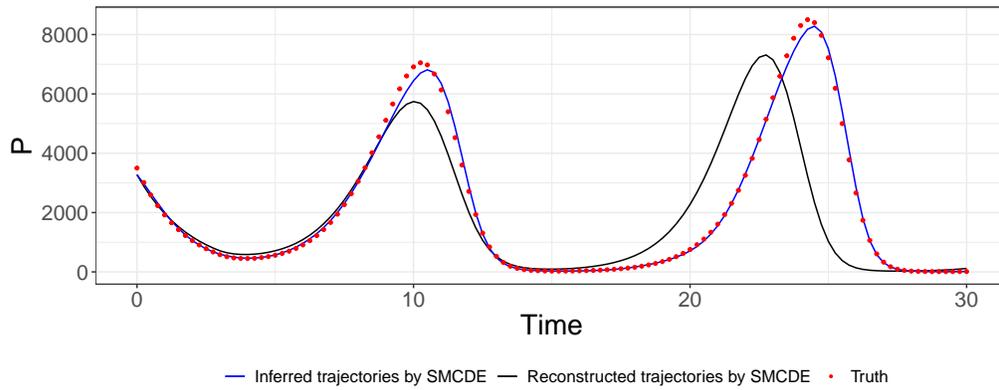


Figure 7: Inferred trajectory and reconstructed trajectory of Hutchinson's equation from 121 observations using SMCDE based on one simulated data set. The blue line represents the inferred trajectory from the spline approximation using Wang et al. (2022)'s method, and the red line represents the reconstructed trajectory obtained by numerically solving Hutchinson's equation using SMCDE's estimated parameters  $((\hat{r}, \hat{K}, \hat{\tau}, \hat{\sigma}) = (0.77, 1.97, 2.92, 0.16))$  and initial condition  $(\hat{N}(0) = 8.09)$ .

good convergence to the expected solution. Running multiple tries of NLS from different starting parameter values can help mitigate the convergence issue: with 100 NLS tries, the best attempt converged to reasonable parameter estimates in approximately 90-95% of simulated datasets; however, the remaining datasets still had very large SSEs. Figure 8 displays the reconstructed trajectory for one simulated dataset where the best NLS attempt (out of 100) had a trajectory RMSE  $\approx 1500$ . Other simulated datasets that exhibit poor convergence tend to yield reconstructed trajectories that resemble this example or worse. Furthermore, the total computational cost of running 100 NLS attempts is expensive: it ranges from  $\sim 2.5$  min (for the case of 16 observations) to  $\sim 5.4$  min (for the case of 121 observations), which is approximately 5 times longer than running MAGIDDE. Therefore, this example illustrates that the NLS approach may not be recommended even for inferring simple DDE systems.

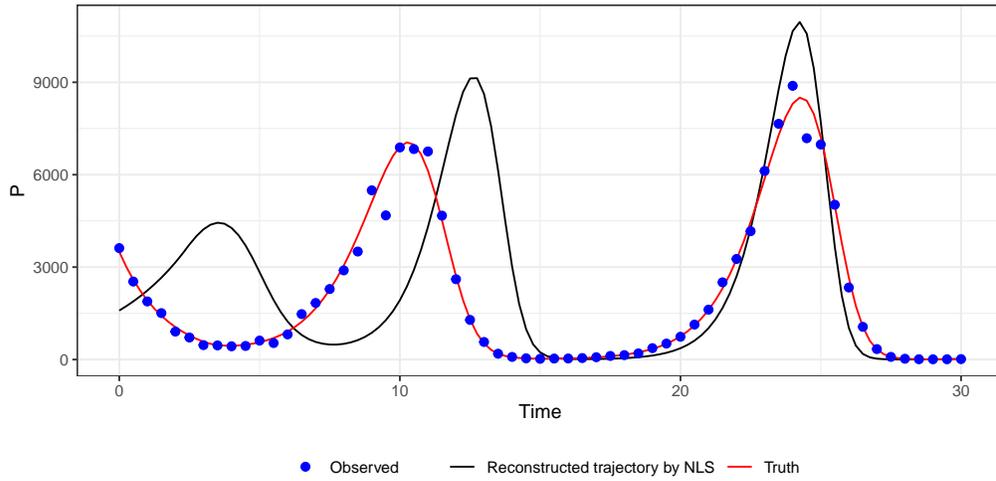


Figure 8: Example of reconstructed trajectory for Hutchinson's equation based on a 61 observation dataset and the NLS approach. The blue dots represent the noisy observations. The red line represents the true trajectory and the blue line represents the reconstructed trajectory using the best parameter estimates from 100 NLS attempts. The parameter estimates are  $(\hat{r}, \hat{K}, \hat{\tau}, \hat{N}(0)) = (1.04, 2.37, 2.38, 7.37)$ .

## S9 Inferred Trajectory for Hutchison's Equation from 61 Observations using MAGIDDE

Analogous to Figure 1 in the main text, Figure 9 shows that MAGIDDE well-recovers the true underlying trajectory of Hutchison's equation in the 61 observation scenario; it can be seen that the 95% pointwise credible interval is narrower with these denser observations, compared to the scenario with 16 observations in Figure 1 of the main text.

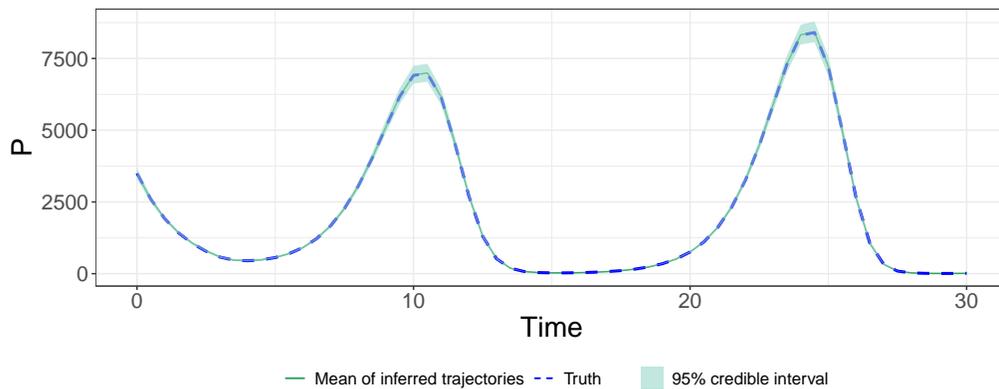


Figure 9: Inferred trajectory of Hutchinson's equation from 61 observations using MAGIDDE. The green solid line represents the mean inferred trajectory over 300 simulated datasets, and the blue dashed line is the truth. The green shaded area is the 95% pointwise credible interval, constructed by taking the average 0.025 and 0.975 quantiles of the inferred trajectories across the simulated datasets.

## S10 *Lac* Operon Model Details

We present the scientific background and parameters of the *lac* operon model. We also provide the implementation details of MAGIDDE described in Section 4.2 of the main text. For comparison, we also show the performance and computational costs of the numerical solver-based NLS approach for this more complex model.

### S10.1 Background and Parameters of the *Lac* Operon Model

The processes of transcription and translation are not instantaneous but require a period of time. Hence, Yildirim and Mackey (2003) incorporated these time delays into their *lac* operon model, to help ensure a satisfactory match between experimental data and the model trajectories. In the system of DDEs presented in the main text, the time-delay parameter  $\tau_M$  represents the time required to produce mRNA from DNA by transcription,  $\tau_P$  denotes the translation time between mRNA and permease, and  $\tau_B$  is the translation time between mRNA and  $\beta$ -galactosidase.

Yildirim and Mackey (2003) derived formulas for  $\alpha_M, \alpha_B, \alpha_P$  according to the steady-state behaviour of the system to calibrate their values;  $\mu$  and  $\gamma_A$  were calibrated by fitting the model to the experimental data for the induction kinetics of  $\beta$ -galactosidase; other parameter values were obtained directly from existing literature. Table 3 summarizes all of the parameters and their corresponding values

from Yildirim and Mackey (2003). Thus, our goal is to estimate the parameters in the first column, namely  $\tau_B, \tau_M, \tau_P, \gamma_A, \alpha_M, \alpha_B, \alpha_P, \mu, M(0), B(0), A(0), L(0), P(0)$ , which include the time-delay parameters, initial conditions, and model parameters calibrated by Yildirim and Mackey (2003); others are treated as known.

Table 3: Parameters in the *lac* operon model.

Parameter	Value	Parameter	Value	Parameter	Value
$\tau_B$	2	$K_1$	$2.52 \times 10^4$	$\beta_{L_1}$	$2.65 \times 10^3$
$\tau_M$	0.1	$K$	7200	$K_{L_1}$	1.81
$\tau_P$	0.83	$n$	2	$\beta_{L_2}$	$1.76 \times 10^4$
$\gamma_A$	0.52	$\Gamma_0$	$7.25 \times 10^{-7}$	$\gamma_L$	0
$\alpha_M$	$9.97 \times 10^{-4}$	$\gamma_M$	0.411	$\gamma_P$	0.65
$\alpha_B$	0.0166	$\gamma_B$	$8.33 \times 10^{-4}$		
$\alpha_P$	10	$\alpha_A$	$1.76 \times 10^4$		
$\mu$	0.0226	$K_L$	0.97		
$M(0)$	$6.26 \times 10^{-4}$	$\beta_A$	$2.15 \times 10^4$		
$B(0)$	0	$K_A$	1.95		
$A(0)$	0.038	$\alpha_L$	2880		
$L(0)$	0.372	$L_e$	0.08		
$P(0)$	0.0149	$K_{L_e}$	0.26		

Some experiments recorded measurements of  $\beta$ -galactosidase every one minute over a 25-minute observation period (Alpers and Tomkins, 1966). Capturing the

transcription delay is still a challenge, as the synthesis of  $\beta$ -galactosidase and initiations of transcription may occur synchronously in 10 seconds (Schwartz et al., 1970). Other researchers took measurements more frequently at the beginning of the experiments, and gradually less frequently afterwards (Alpers and Tomkins, 1966). Inspired by these real experiments, we simulate observations from the system with varying denseness from  $t = 0$  to  $t = 25$ . Specifically, the collection of 23 observation time points for each component is chosen as  $\gamma = \{0, 0.25, 0.5, \dots, 2, 3, 4, \dots, 10, 12, 14, \dots, 20, 25\}$ .

## S10.2 MAGIDDE Implementation

As stated in Section 4.2 of the main text, the observation time points are not equally spaced. The smallest evenly spaced set containing all observation time points is  $\mathbf{I}_0 = \{0, 0.25, \dots, 24.75, 25\}$ . We set  $\mathbf{I} = \mathbf{I}_0$ , which provides stable inference without need for a further increase in discretization points. We place uniform priors over  $(0, \infty)$  for all model parameters and set  $\nu = 2.01$  in the Matern kernel to accommodate rougher system trajectories. We run 50,000 HMC iterations (each with 200 leapfrog steps) with the first 25,000 discarded as burn-in. The computational time required for MAGIDDE on each simulated dataset is  $\sim 30$  minutes on a single CPU core.

### S10.3 Parameter Estimation for *Lac* Operon Model using NLS

To apply the NLS approach with a DDE numerical solver, we construct the objective function using the weighted error sum of squares (SSE), where the weights are set as the reciprocal of the variance for each system component. We used 100 simulated datasets to study its performance. To mitigate the effects of starting values, we try 100 NLS attempts per dataset and take the estimates to be those corresponding to the smallest SSE obtained. We also set the lower and upper bounds for optimization close to the truth for all of the model parameters:  $\tau_B$  is bounded by  $[0, 5]$ ,  $\alpha_P$  is bounded by  $[8, 12]$  and all other parameters are bounded by  $[0, 1]$ . For each NLS attempt, we randomly draw starting parameter values from uniform distributions with these bounds. The starting values for the initial conditions are set to be the observed noisy component values at  $t = 0$ .

For comparison, we compute the RMSEs of the estimated parameters and initial conditions for both NLS and MAGIDDE across the 100 simulated datasets, and these results are summarized in Table 4. The best attempt from NLS notably underperforms MAGIDDE in recovering the parameters, despite the choice of upper and lower bounds for NLS that are close to the truth. Furthermore, the total computation time for 100 NLS attempts on each simulated dataset is  $\sim 118$  minutes on a single CPU core, which is about 4 times longer than that of running MAGIDDE. This example further highlights the challenges of using

numerical solvers to recover a complex DDE system, and hence it is advantageous to consider statistical methods like MAGIDDE that approximate the DDE solution.

Table 4: Average parameter estimates obtained by MAGIDDE and NLS (with RMSEs in parentheses) for the *lac* operon model across 100 simulated datasets.

Parameter	Truth	MAGIDDE	NLS
$\tau_B$	2	2.0024( <b>0.3074</b> )	2.3405(2.4418)
$\tau_M$	0.1	0.2990( <b>0.2363</b> )	0.7662(0.7789)
$\tau_P$	0.83	0.8283( <b>0.3063</b> )	0.6194(0.5173)
$\gamma_A$	0.52	0.4916( <b>0.0382</b> )	0.5168(0.4829)
$10^2 \times \alpha_M$	0.0997	0.0985( <b>0.0034</b> )	6.3861(23.9831)
$\alpha_B$	0.0166	0.0143( <b>0.0025</b> )	0.0045(0.0191)
$\alpha_P$	10	9.6792( <b>0.5053</b> )	8.9830(1.9387)
$\mu$	0.0226	0.0144( <b>0.0109</b> )	0.02083(0.0737)
$10^2 \times M(0)$	0.0626	0.0630( <b>0.0014</b> )	0.0759(0.0255)
$10^3 \times B(0)$	0	0.0018( <b>0.0023</b> )	0.0157(0.0229)
$A(0)$	0.038	0.0319( <b>0.0139</b> )	0.0081(0.0363)
$L(0)$	0.372	0.4018( <b>0.0326</b> )	0.4153(0.4765)
$P(0)$	0.0149	0.0143( <b>0.0007</b> )	0.0085(0.0107)

## S11 Time-delayed SIRD Model Details

In this section, we describe the procedures for processing the Ontario COVID-19 data to obtain noisy observations for the time-delayed SIRD model. We also specifically detail the steps for obtaining the initial observation of each compartment, and provide implementation details. Finally we showcase the predictive performance of MAGIDDE for this SIRD model.

### S11.1 Data Processing Steps

We outline the data processing steps for obtaining noisy observations of the population size in the compartments on each day. These are then converted into proportions by dividing by the Ontario population of 14,999,441 in the first quarter of 2022 (Statistics Canada, 2024). Following Zhao and Wong (2023), we approximate the daily population size in the  $I$  compartment by dividing the number of hospitalized COVID-19 cases (Government of Ontario, 2024) by the overall COVID-19 hospitalization rate. We convert the weekly death counts (Government of Canada, 2024) into daily death counts by linear interpolation; then, the population in the  $D$  compartment on a given day is approximated by the cumulative sum of daily death counts up to that day. The population in the  $S$  compartment on a given day is estimated by subtracting the daily confirmed cases (Government of Ontario, 2022) from the previous day's  $S$  compartment

population. Given these  $S, I, D$  population estimates for each day,  $R$  then consists of the remainder of the population. The data setup further requires the population of each compartment on the initial day of the observation period; this step is detailed in the next subsection.

### S11.2 Initial Noisy Observations

The first wave of COVID-19 driven by the Omicron variant began in Ontario, Canada on January 7, 2022. This wave was marked by a record-high number of hospital admissions (CBC News, 2022). Due to the high reinfection rate of the Omicron variant, previously infected individuals remained susceptible; hence we assume the flow from the  $S$  compartment to the  $I$  compartment starts from January 6 and that there are no deaths related to the Omicron variant prior to January 7. We use the following procedure to approximate the population in the  $S, I, R, D$  compartments on the initial day (January 24) of the observation period under these two assumptions.

A noisy observation for the  $I$  compartment on January 24 can be obtained by dividing the number of patients hospitalized on that day by the hospitalization rate of the Omicron variant (Zhao and Wong, 2023). The trend of  $I$  observations on the few days before January 24 is relatively flat, which aligns well with the history function assumed for the model. Furthermore, based on our assumptions,

a noisy observation for the  $D$  compartment on January 24 is obtained by the cumulative sum of the daily death count from January 7 through January 24.

To obtain noisy observations of  $S$  and  $R$  on January 24, we need estimates for all four compartment sizes at the beginning of the Omicron wave (January 6). First, we obtain a noisy observation for  $I$  on January 6 using the aforementioned hospitalization rate method. Second, based on our two assumptions we set  $R$  and  $D$  to be zero on January 6, and so  $S$  on January 6 is estimated by subtracting  $I$  from the total population. Third, a noisy observation of  $S$  on January 24 is obtained by taking the estimated  $S$  on January 6 and subtracting the cumulative case counts from January 6 to January 24. Last, we subtract these estimates of  $S$ ,  $I$ , and  $D$  on January 24 from the total population to obtain a noisy observation of  $R$  on January 24.

### S11.3 Implementation

To set up MAGIDDE for this system, we first consider the discretization set. The system is observed at the equally-spaced time points  $\gamma = \{0, 1, \dots, 29\}$  days. We insert one additional discretization time point, equally-spaced between each pair of adjacent observations, i.e.,  $\mathbf{I} = \{0, 0.5, \dots, 28.5, 29\}$ ; a further increase to  $|\mathbf{I}|$  yielded similar inference. Second, we take the Matern kernel with  $\nu = 2.01$  as a default choice. Third, we set priors for  $\tilde{\beta}$ ,  $\mu_d$ , and  $\lambda$  to be uniform over  $(0, \infty)$ . To

incorporate the *a priori* belief regarding the estimated mean incubation period of 3.5 days for the Omicron variant (Manica et al., 2022), we place an informative prior of  $N(3.5, 1)$  on the time-delay parameter  $h$ . Then, we ran 80,000 HMC iterations (each with 25 leapfrog steps), with the first 40,000 discarded as burn-in.

#### S11.4 Predictive Performance

We utilize the SIRD model to showcase the predictive performance of MAGIDDE: we use the first part of the observations for model fitting and the remainder for assessing model predictions. We consider two scenarios: in the first, we assume the model is correctly specified (i.e., we simulate noisy observations from the SIRD model); in the second, we consider the real Ontario data.

For the first scenario, we set the true values of each model parameter using the values estimated in Table 3 of the main text, numerically solve the SIRD model trajectory over  $[0, 30]$  and add Gaussian noise to simulate a sample set of observations. We use MAGIDDE to fit the model using observations from the first period only ( $t = \{0, 1, \dots, 15\}$ ) following the implementation details in Section S11.3 with  $\mathbf{I} = \{0, 0.5, \dots, 29.5, 30\}$ . We generate predictions for the second period ( $t = \{16, 17, \dots, 30\}$ ) using MAGIDDE's posterior samples, as described in Section 3.3 of the main text. Figure 11 displays the inferred trajectories of

all four components using MAGIDDE. In the prediction period, our inferred trajectory aligns closely with the true trajectory, and the true trajectory is well-contained by the 95% credible intervals. Hence, when the SIRD model is correctly specified, MAGIDDE generates highly accurate predictions.

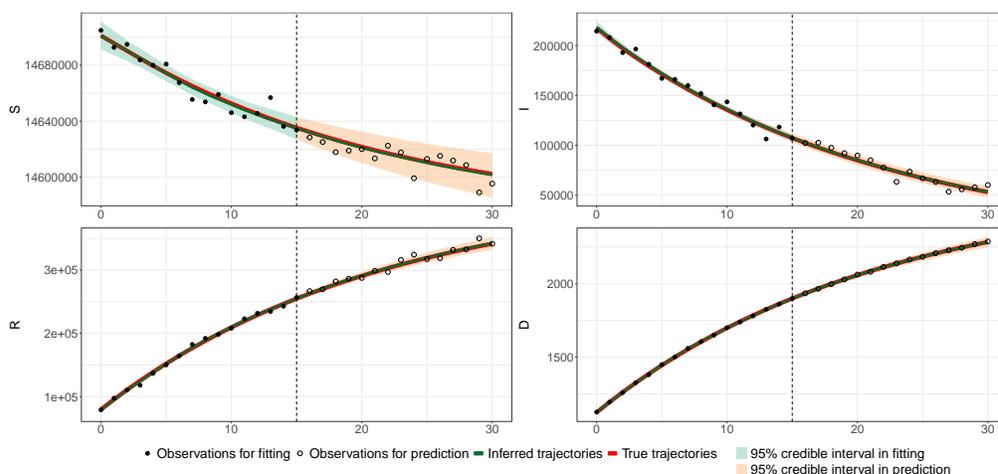


Figure 10: Inferred trajectory of the time-delay SIRD model using a simulated dataset based on the estimated parameters in Table 3 of the main text. The noisy observations consists of two parts: the observations used for fitting ( $t = \{0, 1, \dots, 15\}$ ) are marked with solid black dots and the remaining observations used for prediction ( $t = \{16, 17, \dots, 30\}$ ) are marked with hollow black dots. The vertical dotted line separates the fitting and prediction periods. The red curve is the true trajectory. The green curve is the inferred trajectory. The green shaded area represents the 95% credible interval for the fitting period, and the orange shaded area represents the 95% credible interval for the prediction period.

For the second scenario, we consider the real Ontario data. We use com-

partment observations from only January 24 to February 8 to fit the model, and generate predictions for the remaining 15 days (from February 9 to February 23). We use the same MAGIDDE implementation procedure as in the first scenario, and visualize the inferred trajectory in Figure 11. MAGIDDE's predictions are relatively accurate for the  $S$ ,  $R$  and  $D$  components, and not as accurate for the  $I$  component but still reasonable. This may be because the downward trend in infections (the  $I$  component) accelerates during the second half of the real data, which cannot be captured by fitting the model to the first half only.

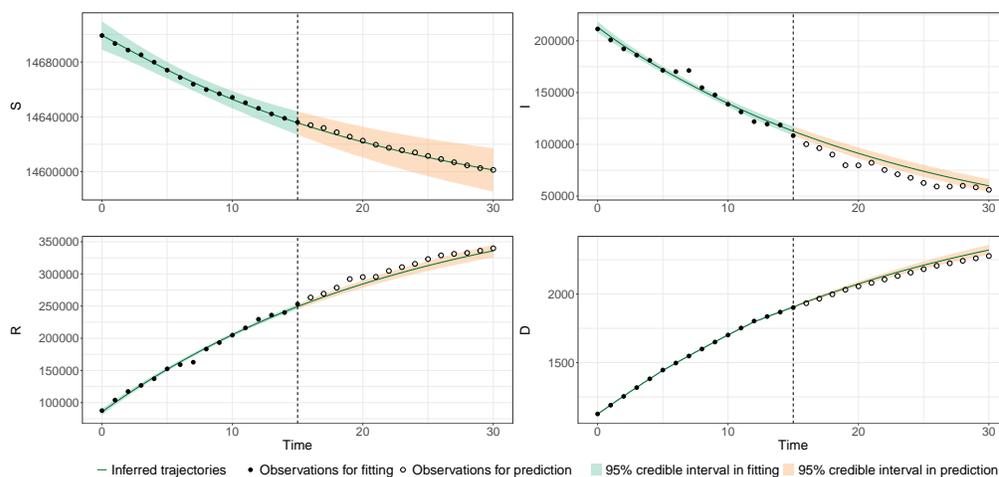


Figure 11: Inferred trajectory of the time-delay SIRD model based on the Ontario dataset. The noisy observations are divided into two parts: the observations from the period January 24 to February 8 ( $t = \{0, 1, \dots, 15\}$ ) are used for fitting and marked with solid black dots, while the remaining observations from February 9 to February 23 ( $t = \{16, 17, \dots, 30\}$ ) are used for prediction and marked with hollow black dots. The vertical dotted line separates the fitting and prediction periods. The green curve shows the inferred trajectory, with the green shaded area representing the 95% credible interval for the fitting period, and the orange shaded area representing the 95% credible interval for the prediction period.

## Bibliography

Alpers, D. H. and G. M. Tomkins (1966). Sequential transcription of the genes of the lactose operon and its regulation by protein synthesis. *Journal of Biological Chemistry* 241(19), 4434–4443.

Bowling, S. R., M. T. Khasawneh, S. Kaewkuekool, and B. R. Cho (2009). A logistic approximation to the cumulative normal distribution. *Journal of industrial engineering and management* 2(1), 114–127.

Carpenter, B., A. Gelman, M. D. Hoffman, D. Lee, B. Goodrich, M. Betancourt, M. A. Brubaker, J. Guo, P. Li, and A. Riddell (2017). Stan: A probabilistic programming language. *Journal of statistical software* 76.

CBC News (2022). Record high number of Ontarians in hospital with COVID-19 as Omicron wave continues. <https://www.cbc.ca/news/canada/toronto/covid-19-ontario-jan-7-2022-hospitalizations-pandemic-high-1.6307179>. Accessed: 2024-04-05.

Government of Canada (2024). COVID-19 epidemiology update: Summary. <https://health-infobase.canada.ca/covid-19/>. Accessed: 2024-04-04.

Government of Ontario (2022). Cases and rates by vaccination status. <https://data.ontario.ca/dataset/covid-19-vaccine-data-in-ontario/resource/eed63cf2-83dd-4598-b337-b288c0a89a16>. Accessed: 2024-04-04.

Government of Ontario (2024). COVID-19 cases in hospital and ICU, by Ontario Health (OH) region. <https://data.ontario.ca/dataset/covid-19-cases-in-hospital-and-icu-by-ontario-health-region>. Accessed: 2024-04-04.

Hoffman, M. D., A. Gelman, et al. (2014). The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo. *J. Mach. Learn. Res.* 15(1), 1593–1623.

Liang, H. and H. Wu (2008). Parameter estimation for differential equation models using a framework of measurement error in regression models. *Journal of the American Statistical Association* 103(484), 1570–1583.

Manica, M., A. De Bellis, G. Guzzetta, P. Mancuso, M. Vicentini, F. Venturelli, et al. (2022). Intrinsic generation time of the SARS-CoV-2 Omicron variant: an observational study of household transmission. *The Lancet Regional Health–Europe* 19, 100446.

Schwartz, T., E. Craig, and D. Kennell (1970). Inactivation and degradation of messenger ribonucleic acid from the lactose operon of *Escherichia coli*. *Journal of Molecular Biology* 54(2), 299–311.

Seleznjev, O. (1996). Large deviations in the piecewise linear approximation of

- Gaussian processes with stationary increments. *Advances in applied probability* 28(2), 481–499.
- Statistics Canada (2024). Population estimates, quarterly. <https://www150.statcan.gc.ca/>. Accessed: 2024-04-02.
- Stein, M. L. (2012). *Interpolation of spatial data: some theory for kriging*. Springer Science & Business Media.
- Wainwright, M. J. (2019). *High-dimensional statistics: A non-asymptotic viewpoint*, Volume 48. Cambridge university press.
- Wang, L. and J. Cao (2012). Estimating parameters in delay differential equation models. *Journal of agricultural, biological, and environmental statistics* 17, 68–83.
- Wang, S., S. Ge, R. Doig, and L. Wang (2022). Adaptive semiparametric Bayesian differential equations via sequential Monte Carlo. *Journal of Computational and Graphical Statistics* 31(2), 600–613.
- Wang, W., R. Tuo, and C. Jeff Wu (2020). On prediction properties of kriging: Uniform error bounds and robustness. *Journal of the American Statistical Association* 115(530), 920–930.
- Yang, S., S. W. Wong, and S. Kou (2021). Inference of dynamic systems from

noisy and sparse data via manifold-constrained Gaussian processes. *Proceedings of the National Academy of Sciences* *118*(15), e2020397118.

Yildirim, N. and M. C. Mackey (2003). Feedback regulation in the lactose operon: a mathematical modeling study and comparison with experimental data. *Biophysical journal* *84*(5), 2841–2851.

Zhao, Y. and S. W. Wong (2023). A comparative study of compartmental models for COVID-19 transmission in Ontario, Canada. *Scientific Reports* *13*(1), 15050.