Inference of dynamic systems from noisy and sparse data via manifold-constrained Gaussian processes

Shihao Yang[a], **Samuel W. K. Wong**[b], **and S. C. Kou**[c,1]

[a]H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, USA; [b]Department of Statistics and Actuarial Science, University of Waterloo, Canada; [c]Department of Statistics, Harvard University, USA

This manuscript was compiled on February 21, 2021

Parameter estimation for nonlinear dynamic system models, represented by ordinary differential equations (ODEs), using noisy and sparse data is a vital task in many fields. We propose a fast and accurate method, MAGI (MAnifold-constrained Gaussian process Inference), for this task. MAGI uses a Gaussian process model over time-series data, explicitly conditioned on the manifold constraint that derivatives of the Gaussian process must satisfy the ODE system. By doing so, we completely bypass the need for numerical integration and achieve substantial savings in computational time. MAGI is also suitable for inference with unobserved system components, which often occur in real experiments. MAGI is distinct from existing approaches as we provide a principled statistical construction under a Bayesian framework, which incorporates the ODE system through the manifold constraint. We demonstrate the accuracy and speed of MAGI using realistic examples based on physical experiments.

Parameter estimation | Ordinary differential equations | Posterior sampling | Inverse problem

Dynamic systems, represented as a set of ordinary differential equations (ODEs), are commonly used to model behaviors in scientific domains, such as gene regulation (1), biological rhythms (2), spread of disease (3), ecology (4), etc. We focus on models specified by a set of ODEs

$$\dot{\boldsymbol{x}}(t) = \frac{d\boldsymbol{x}(t)}{dt} = \mathbf{f}(\boldsymbol{x}(t), \boldsymbol{\theta}, t), \quad t \in [0, T], \qquad [1]$$

where the vector $\boldsymbol{x}(t)$ contains the system outputs that evolve over time t, and $\boldsymbol{\theta}$ is the vector of model parameters to be estimated from experimental/observational data. When $\mathbf{f}$ is nonlinear, solving $\boldsymbol{x}(t)$ given initial conditions $\boldsymbol{x}(0)$ and $\boldsymbol{\theta}$ generally requires a numerical integration method, such as Runge-Kutta.

Historically, ODEs have mainly been used for conceptual or theoretical understanding rather than data fitting as experimental data were limited. Advances in experimental and data-collection techniques have increased the capacity to follow dynamic systems closer to real-time. Such data will generally be recorded at discrete times and subject to measurement error. Thus, we assume that we observe $\boldsymbol{y}(\boldsymbol{\tau}) = \boldsymbol{x}(\boldsymbol{\tau}) + \boldsymbol{\epsilon}(\boldsymbol{\tau})$ at a set of observation time points $\boldsymbol{\tau}$ with error $\boldsymbol{\epsilon}$ governed by noise level σ. Our focus here is inference of $\boldsymbol{\theta}$ given $\boldsymbol{y}(\boldsymbol{\tau})$, with emphasis on nonlinear $\mathbf{f}$ where specialized methods that exploit a linear structure, e.g. (5, 6), are not generally applicable. We shall present a coherent, statistically principled framework for dynamic system inference with the help of Gaussian processes (GPs). The key of our method is to restrict the GPs on a manifold that satisfies the ODE system: thus we name our method MAGI (MAnifold-constrained Gaussian process Inference). Placing a GP on $\boldsymbol{x}(t)$ facilitates inference of $\boldsymbol{\theta}$ without numerical integration, and our explicit manifold constraint is the key novel idea that addresses the conceptual incompatibility between the GP and the specification of the ODE model, as we shall discuss shortly when overviewing our method. We show that the resulting parameter inference is computationally efficient, statistically principled, and effective in a variety of practical scenarios. MAGI particularly works in the cases when some system component(s) is/are unobserved. To the best of our knowledge, none of the current available software packages that do not use numerical integration can analyze systems with unobserved component(s).

Overview of our method. Following the Bayesian paradigm, we view the D-dimensional system $\boldsymbol{x}(t)$ to be a realization of the stochastic process $\boldsymbol{X}(t) = (X_1(t), \ldots, X_D(t))$, and the model parameters $\boldsymbol{\theta}$ a realization of the random variable $\boldsymbol{\Theta}$. In Bayesian statistics, the basis of inference is the posterior distribution, obtained by combining the likelihood function with a chosen prior distribution on the unknown parameters and stochastic processes. Specifically, we impose a general prior distribution $\pi(\cdot)$ on $\boldsymbol{\theta}$ and independent GP prior distributions on each component $X_d(t)$ so that $X_d(t) \sim \mathcal{GP}(\mu_d, \mathcal{K}_d)$, $t \in [0, T]$, where $\mathcal{K}_d : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ is a positive definite covariance kernel for the GP and $\mu_d : \mathbb{R} \to \mathbb{R}$ is

Significance Statement

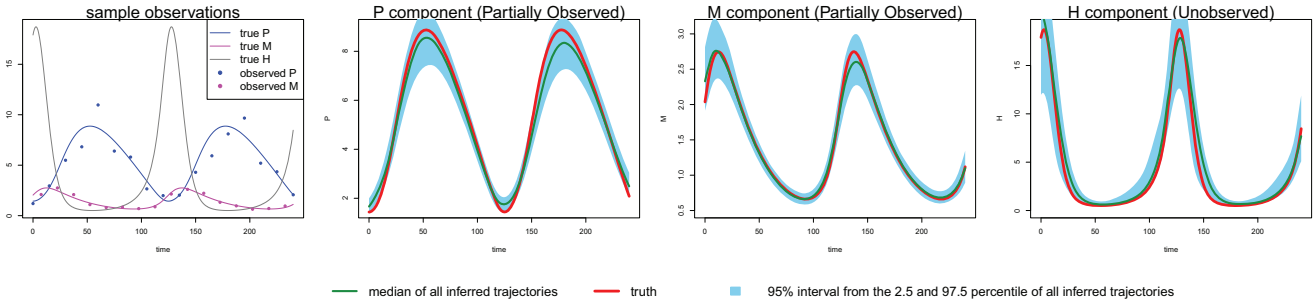
Ordinary differential equations are a ubiquitous tool for modeling behaviors in science, such as gene regulation, biological rhythms, epidemics and ecology. An important problem is to infer and characterize the uncertainty of parameters that govern the equations. Here we present an accurate and fast inference method using manifold-constrained Gaussian processes, such that the derivatives of the Gaussian process must satisfy the dynamics of the differential equations. Our method completely avoids the use of numerical integration and is thus fast to compute. Our construction is embedded in a principled statistical framework and is demonstrated to yield fast and reliable inference in a variety of practical problems. Our method works even when some system component(s) is/are unobserved, which is a significant challenge for previous methods.

Author contributions: S.Y., S.W.K.W., and S.C.K. designed research; S.Y., S.W.K.W., and S.C.K. performed research; S.Y. and S.W.K.W. contributed new reagents/analytic tools; S.Y. and S.W.K.W. analyzed data; and S.Y., S.W.K.W., and S.C.K. wrote the paper.

The authors declare no conflict of interest.

[1]To whom correspondence should be addressed. E-mail: kou@stat.harvard.edu

Fig. 1. Inference by MAGI for Hes1 partially observed asynchronous system on 2000 simulated datasets. The red curve is the truth. MAGI recovers the system well, without the usage of any numerical solver: the green curve shows the median of the inferred trajectories among the 2000 simulated datasets, and a 95% interval from the 2.5% and 97.5% of all inferred trajectories is shown via the blue dashed area.



the mean function. Then for any finite set of time points $\boldsymbol{\tau}_d$, $X_d(\boldsymbol{\tau}_d)$ has a multivariate Gaussian distribution with mean vector $\mu_d(\boldsymbol{\tau}_d)$ and covariance matrix $\mathcal{K}_d(\boldsymbol{\tau}_d, \boldsymbol{\tau}_d)$. Denote the observations by $\boldsymbol{y}(\boldsymbol{\tau}) = (\boldsymbol{y}_1(\boldsymbol{\tau}_1), \ldots, \boldsymbol{y}_D(\boldsymbol{\tau}_D))$, where $\boldsymbol{\tau} = (\boldsymbol{\tau}_1, \boldsymbol{\tau}_2, \ldots, \boldsymbol{\tau}_D)$ is the collection of all observation time points and each component X_d can have its own set of observation times $\boldsymbol{\tau}_d = (\tau_{d,1}, \ldots, \tau_{d,N_d})$. If the d-th component is not observed, then $N_d = 0$, and $\boldsymbol{\tau}_d = \emptyset$. $N = N_1 + \cdots + N_D$ is the total number of observations. We note that for the remainder of the paper, the notation t shall refer to time generically, while τ shall refer specifically to the observation time points.

As an illustrative example, consider the dynamic system in (1) that governs the oscillation of Hes1 mRNA (M) and Hes1 protein (P) levels in cultured cells, where it is postulated that a Hes1-interacting (H) factor contributes to a stable oscillation, a manifestation of biological rhythm (2). The ODEs of the three-component system $X = (P, M, H)$ are

$$\mathbf{f}(X, \boldsymbol{\theta}, t) = \begin{pmatrix} -aPH + bM - cP \\ -dM + \frac{e}{1+P^2} \\ -aPH + \frac{f}{1+P^2} - gH \end{pmatrix},$$

where $\boldsymbol{\theta} = (a, b, c, d, e, f, g)$ are the associated parameters. In Fig 1 (left panel) we show noise-contaminated data generated from the system, which closely mimics the experimental setup described in (1): P and M are observed at 15-minute intervals for 4 hours but H is never observed. In addition, P and M observations are asynchronous: starting at time 0, every 15 minutes we observe P; starting at 7.5 minutes, every 15 minutes we observe M; P and M are never observed at the same time. It can be seen that the mRNA and protein levels exhibit the behavior of regulation via negative feedback.

The goal here is to infer the seven parameters of the system: a, b govern the rate of protein synthesis in the presence of the interacting factor; c, d, g are the rates of decomposition; and e, f are inhibition rates. The unobserved H component poses a challenge for most existing methods that do not use numerical integration, but is capably handled by MAGI: the P and M panels of Fig 1 show that our inferred trajectories provide good fits to the observed data, and the H panel shows that the dynamics of the entirely unobserved H component are largely recovered as well. We emphasize that these trajectories are inferred without any use of numerical solvers. We shall return to the Hes1 example in detail in the Results section.

Intuitively, the GP prior on $\boldsymbol{X}(t)$ facilitates computation as GP provides closed analytical forms for $\dot{\boldsymbol{X}}(t)$ and $\boldsymbol{X}(t)$, which could bypass the need for numerical integration. In particular, with a GP prior on $\boldsymbol{X}(t)$, the conditional distribution of $\dot{\boldsymbol{X}}(t)$ given $\boldsymbol{X}(t)$ is also a GP with its mean function and covariance kernel completely specified. This GP specification for the derivatives $\dot{\boldsymbol{x}}(t)$, however, is inherently incompatible with the ODE model because Eq. (1) also completely specifies $\dot{\boldsymbol{x}}(t)$ given $\boldsymbol{x}(t)$ (via the function $\mathbf{f}$). As a key novel contribution of our method, MAGI addresses this conceptual incompatibility by constraining the GP to satisfy the ODE model in Eq. (1). To do so, we first define a random variable W quantifying the difference between stochastic process $\boldsymbol{X}(t)$ and the ODE structure with a given value of the parameter $\boldsymbol{\theta}$:

$$W = \sup_{t \in [0,T], d \in \{1,\ldots,D\}} |\dot{X}_d(t) - \mathbf{f}(\boldsymbol{X}(t), \boldsymbol{\theta}, t)_d|. \qquad [2]$$

$W \equiv 0$ if and only if ODEs with parameter $\boldsymbol{\theta}$ are satisfied by $\boldsymbol{X}(t)$. Therefore, ideally the posterior distribution for $\boldsymbol{X}(t)$ and $\boldsymbol{\theta}$ given the observations $\boldsymbol{y}(\boldsymbol{\tau})$ and the ODE constraint, $W \equiv 0$, is (informally)

$$p_{\boldsymbol{\Theta}, \boldsymbol{X}(t) | W, \boldsymbol{Y}(\boldsymbol{\tau})}(\boldsymbol{\theta}, \boldsymbol{x}(t) | W = 0, \boldsymbol{Y}(\boldsymbol{\tau}) = \boldsymbol{y}(\boldsymbol{\tau})). \qquad [3]$$

While Eq. (3) is the ideal posterior, in reality W is not generally computable. In practice we approximate W by finite discretization on the set $\boldsymbol{I} = (t_1, t_2, \ldots, t_n)$ such that $\boldsymbol{\tau} \subset \boldsymbol{I} \subset [0, T]$ and similarly define $W_{\boldsymbol{I}}$ as

$$W_{\boldsymbol{I}} = \max_{t \in \boldsymbol{I}, d \in \{1,\ldots,D\}} |\dot{X}_d(t) - \mathbf{f}(\boldsymbol{X}(t), \boldsymbol{\theta}, t)_d|. \qquad [4]$$

Note that $W_{\boldsymbol{I}}$ is the maximum of a finite set, and $W_{\boldsymbol{I}} \to W$ monotonically as $\boldsymbol{I}$ becomes dense in $[0, T]$. Therefore, the practically computable posterior distribution is

$$p_{\boldsymbol{\Theta}, \boldsymbol{X}(\boldsymbol{I}) | W_{\boldsymbol{I}}, \boldsymbol{Y}(\boldsymbol{\tau})}(\boldsymbol{\theta}, \boldsymbol{x}(\boldsymbol{I}) | W_{\boldsymbol{I}} = 0, \boldsymbol{Y}(\boldsymbol{\tau}) = \boldsymbol{y}(\boldsymbol{\tau})),$$

which is the joint conditional distribution of $\boldsymbol{\theta}$ and $\boldsymbol{X}(\boldsymbol{I})$ together. Thus, effectively, we simultaneously infer both the parameters and the unobserved trajectory $\boldsymbol{X}(\boldsymbol{I})$ from the noisy observations $\boldsymbol{y}(\boldsymbol{\tau})$.

Under Bayes' rule, we have

$$\begin{aligned} &p_{\boldsymbol{\Theta}, \boldsymbol{X}(\boldsymbol{I}) | W_{\boldsymbol{I}}, \boldsymbol{Y}(\boldsymbol{\tau})}(\boldsymbol{\theta}, \boldsymbol{x}(\boldsymbol{I}) | W_{\boldsymbol{I}} = 0, \boldsymbol{Y}(\boldsymbol{\tau}) = \boldsymbol{y}(\boldsymbol{\tau})) \\ &\propto P(\boldsymbol{\Theta} = \boldsymbol{\theta}, \boldsymbol{X}(\boldsymbol{I}) = \boldsymbol{x}(\boldsymbol{I}), W_{\boldsymbol{I}} = 0, \boldsymbol{Y}(\boldsymbol{\tau}) = \boldsymbol{y}(\boldsymbol{\tau})), \end{aligned}$$

where the right hand side can be decomposed as

$$
\begin{aligned}
&P(\boldsymbol{\Theta}=\boldsymbol{\theta}, \boldsymbol{X}(\boldsymbol{I})=\boldsymbol{x}(\boldsymbol{I}), W_{\boldsymbol{I}}=0, \boldsymbol{Y}(\boldsymbol{\tau})=\boldsymbol{y}(\boldsymbol{\tau})) \\
&= \pi_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) \times \underbrace{P(\boldsymbol{X}(\boldsymbol{I})=\boldsymbol{x}(\boldsymbol{I})|\boldsymbol{\Theta}=\boldsymbol{\theta})}_{(1)} \\
&\quad \times \underbrace{P(\boldsymbol{Y}(\boldsymbol{\tau})=\boldsymbol{y}(\boldsymbol{\tau})|\boldsymbol{X}(\boldsymbol{I})=\boldsymbol{x}(\boldsymbol{I}), \boldsymbol{\Theta}=\boldsymbol{\theta})}_{(2)} \\
&\quad \times \underbrace{P(W_{\boldsymbol{I}}=0|\boldsymbol{Y}(\boldsymbol{\tau})=\boldsymbol{y}(\boldsymbol{\tau}), \boldsymbol{X}(\boldsymbol{I})=\boldsymbol{x}(\boldsymbol{I}), \boldsymbol{\Theta}=\boldsymbol{\theta})}_{(3)}.
\end{aligned}
$$

The first term (1) can be simplified as $P(\boldsymbol{X}(\boldsymbol{I})=\boldsymbol{x}(\boldsymbol{I})|\boldsymbol{\Theta}=\boldsymbol{\theta}) = P(\boldsymbol{X}(\boldsymbol{I})=\boldsymbol{x}(\boldsymbol{I}))$ due to the prior independence of $\boldsymbol{X}(\boldsymbol{I})$ and $\boldsymbol{\Theta}$; it corresponds to the GP prior on $\boldsymbol{X}$. The second term (2) corresponds to the noisy observations. The third term (3) can be simplified as

$$
\begin{aligned}
&P(W_{\boldsymbol{I}}=0|\boldsymbol{Y}(\boldsymbol{\tau})=\boldsymbol{y}(\boldsymbol{\tau}), \boldsymbol{X}(\boldsymbol{I})=\boldsymbol{x}(\boldsymbol{I}), \boldsymbol{\Theta}=\boldsymbol{\theta}) \\
&= P(\dot{\boldsymbol{X}}(\boldsymbol{I}) - \mathbf{f}(\boldsymbol{x}(\boldsymbol{I}), \boldsymbol{\theta}, t_{\boldsymbol{I}}) = \mathbf{0}|\boldsymbol{Y}(\boldsymbol{\tau})=\boldsymbol{y}(\boldsymbol{\tau}), \boldsymbol{X}(\boldsymbol{I})=\boldsymbol{x}(\boldsymbol{I}), \boldsymbol{\Theta}=\boldsymbol{\theta}) \\
&= P(\dot{\boldsymbol{X}}(\boldsymbol{I}) - \mathbf{f}(\boldsymbol{x}(\boldsymbol{I}), \boldsymbol{\theta}, t_{\boldsymbol{I}}) = \mathbf{0}|\boldsymbol{X}(\boldsymbol{I})=\boldsymbol{x}(\boldsymbol{I})) \\
&= P(\dot{\boldsymbol{X}}(\boldsymbol{I}) = \mathbf{f}(\boldsymbol{x}(\boldsymbol{I}), \boldsymbol{\theta}, t_{\boldsymbol{I}})|\boldsymbol{X}(\boldsymbol{I})=\boldsymbol{x}(\boldsymbol{I})),
\end{aligned}
$$

which is the conditional density of $\dot{\boldsymbol{X}}(\boldsymbol{I})$ given $\boldsymbol{X}(\boldsymbol{I})$ evaluated at $\mathbf{f}(\boldsymbol{x}(\boldsymbol{I}), \boldsymbol{\theta}, t_{\boldsymbol{I}})$. All three terms are multivariate Gaussian: the third term is Gaussian because $\dot{\boldsymbol{X}}(\boldsymbol{I})$ given $\boldsymbol{X}(\boldsymbol{I})$ has a multivariate Gaussian distribution as long as the kernel $\mathcal{K}$ is twice differentiable.

Therefore, the practically computable posterior distribution simplifies to

$$
\begin{aligned}
&p_{\boldsymbol{\Theta}, \boldsymbol{X}(\boldsymbol{I})|W_{\boldsymbol{I}}, \boldsymbol{Y}(\boldsymbol{\tau})}(\boldsymbol{\theta}, \boldsymbol{x}(\boldsymbol{I})|W_{\boldsymbol{I}}=0, \boldsymbol{Y}(\boldsymbol{\tau})=\boldsymbol{y}(\boldsymbol{\tau})) \qquad [5] \\
&\propto \pi_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) \exp\Bigg\{ -\frac{1}{2}\sum_{d=1}^{D}\Bigg[\\
&\underbrace{+|\boldsymbol{I}|\log(2\pi) + \log|C_d| + \|x_d(\boldsymbol{I}) - \mu_d(\boldsymbol{I})\|^2_{C_d^{-1}}}_{(1)} \\
&\underbrace{+|\boldsymbol{I}|\log(2\pi) + \log|K_d| + \left\|\mathbf{f}_{d,\boldsymbol{I}}^{\boldsymbol{x},\boldsymbol{\theta}} - \dot{\mu}_d(\boldsymbol{I}) - m_d\{x_d(\boldsymbol{I}) - \mu_d(\boldsymbol{I})\}\right\|^2_{K_d^{-1}}}_{(3)} \\
&\underbrace{+N_d\log(2\pi\sigma_d^2) + \|x_d(\boldsymbol{\tau}_d) - y_d(\boldsymbol{\tau}_d)\|^2_{\sigma_d^{-2}}}_{(2)}\Bigg]\Bigg\}
\end{aligned}
$$

where $\|\boldsymbol{v}\|_A^2 = \boldsymbol{v}^\intercal A\boldsymbol{v}$, $|\boldsymbol{I}|$ is the cardinality of $\boldsymbol{I}$, $\mathbf{f}_{d,\boldsymbol{I}}^{\boldsymbol{x},\boldsymbol{\theta}}$ is short for the d-th component of $\mathbf{f}(\boldsymbol{x}(\boldsymbol{I}), \boldsymbol{\theta}, t_{\boldsymbol{I}})$, and the multivariate Gaussian covariance matrix C_d and the matrix K_d can be derived as follows for each component d:

$$
\begin{cases}
C &= \mathcal{K}(\boldsymbol{I}, \boldsymbol{I}) \\
m &= {}'\mathcal{K}(\boldsymbol{I}, \boldsymbol{I})\mathcal{K}(\boldsymbol{I}, \boldsymbol{I})^{-1} \\
K &= \mathcal{K}''(\boldsymbol{I}, \boldsymbol{I}) - {}'\mathcal{K}(\boldsymbol{I}, \boldsymbol{I})\mathcal{K}(\boldsymbol{I}, \boldsymbol{I})^{-1}\mathcal{K}'(\boldsymbol{I}, \boldsymbol{I})
\end{cases} \qquad [6]
$$

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

In practice we choose the Matern kernel $\mathcal{K}(s,t) = \phi_1 \frac{2^{1-\nu}}{\Gamma(\nu)}\left(\sqrt{2\nu}\frac{l}{\phi_2}\right)^\nu B_\nu\left(\sqrt{2\nu}\frac{l}{\phi_2}\right)$ where $l = |s-t|$, Γ is the Gamma function and B_ν is the modified Bessel function of the second kind, and the degree of freedom ν is set to be 2.01 to ensure that the kernel is twice differentiable. $\mathcal{K}$ has two hyper-parameters ϕ_1 and ϕ_2. Their meaning and specification are discussed in the *Materials and Methods* section.

With the posterior distribution specified in Eq. (5), we use Hamiltonian Monte Carlo (HMC) (7) to obtain samples of $\boldsymbol{X}_{\boldsymbol{I}}$ and the parameters together. At the completion of HMC sampling, we take the posterior mean of $\boldsymbol{X}_{\boldsymbol{I}}$ as the inferred trajectory, and the posterior means of the sampled parameters as the parameter estimates. Throughout the MAGI computation, no numerical integration is ever needed.

Review of related work. The problem of dynamic system inference has been studied in the literature, which we now briefly review. We first note that a simple approach to constructing the 'ideal' likelihood function is according to $p(\boldsymbol{y}(\boldsymbol{t})|\hat{\boldsymbol{x}}(\boldsymbol{t}, \boldsymbol{\theta}, \boldsymbol{x}(0)), \sigma)$, where $\hat{\boldsymbol{x}}(\boldsymbol{t}, \boldsymbol{\theta}, \boldsymbol{x}(0))$ is the numerical solution of the ODE obtained by numerical integration given $\boldsymbol{\theta}$ and the initial conditions. This approach suffers from a high computational burden: numerical integration is required for every $\boldsymbol{\theta}$ sampled in an optimization or Markov chain Monte Carlo (MCMC) routine (8). Smoothing methods have been useful for eliminating the dependence on numerical ODE solutions, and an innovative penalized likelihood approach (9) uses a B-spline basis for constructing estimated functions to simultaneously satisfy the ODE system and fit the observed data. While in principle the method in (9) can handle an unobserved system component, substantive manual input is required as we show in the Results, which contrasts with the ready-made solution that MAGI provides.

As an alternative to the penalized likelihood approach, GPs are a natural candidate for fulfilling the smoothing role in a Bayesian paradigm due to their flexibility and analytic tractability (10). The use of GPs to approximate the dynamic system and facilitate computation has been previously studied by a number of authors (8, 11–15). The basic idea is to specify a joint GP over $\boldsymbol{y}, \boldsymbol{x}, \dot{\boldsymbol{x}}$ with hyperparameters ϕ, and then provide a factorization of the joint density $p(\boldsymbol{y}, \boldsymbol{x}, \dot{\boldsymbol{x}}, \boldsymbol{\theta}, \phi, \sigma)$ that is suitable for inference. The main challenge is to find a coherent way to combine information from two distinct sources: the approximation to the system by the GP governed by hyperparameters ϕ, and the actual dynamic system equations governed by parameters $\boldsymbol{\theta}$. In (8, 11), the factorization proposed is $p(\boldsymbol{y}, \boldsymbol{x}, \dot{\boldsymbol{x}}, \boldsymbol{\theta}, \phi, \sigma) = p(\boldsymbol{y}|\boldsymbol{x}, \sigma)p(\dot{\boldsymbol{x}}|\boldsymbol{x}, \boldsymbol{\theta}, \phi)p(\boldsymbol{x}|\phi)p(\phi)p(\boldsymbol{\theta})$, where $p(\boldsymbol{y}|\boldsymbol{x}, \sigma)$ comes from the observation model and $p(\boldsymbol{x}|\phi)$ comes from the GP prior as in our approach. However, there are significant conceptual difficulties in specifying $p(\dot{\boldsymbol{x}}|\boldsymbol{x}, \boldsymbol{\theta}, \phi)$: on one hand, the distribution of $\dot{\boldsymbol{x}}$ is completely determined by the GP given $\boldsymbol{x}$, while on the other hand $\dot{\boldsymbol{x}}$ is completely specified by the ODE system $\dot{\boldsymbol{x}} = \mathbf{f}(\boldsymbol{x}, \boldsymbol{\theta}, t)$; these two are incompatible. Previous authors have attempted to circumvent this incompatibility of the GP and ODE system: (8, 11) use a product-of-experts heuristic by letting $p(\dot{\boldsymbol{x}}|\boldsymbol{x}, \boldsymbol{\theta}, \phi) \propto p(\dot{\boldsymbol{x}}|\boldsymbol{x}, \phi)p(\dot{\boldsymbol{x}}|\boldsymbol{x}, \boldsymbol{\theta})$, where the two distributions in the product come from the GP and a noisy version of the ODE, respectively. In (15), the authors arrive at the same posterior as (8, 11) by assuming an alternative graphical model that bypasses the product of experts heuristic; nonetheless, the method requires working with an artificial noisy version of the ODE. In (12), the authors start with a different factorization: $p(\boldsymbol{y}, \boldsymbol{x}, \dot{\boldsymbol{x}}, \boldsymbol{\theta}, \phi, \sigma) = p(\boldsymbol{y}|\dot{\boldsymbol{x}}, \phi, \sigma)p(\dot{\boldsymbol{x}}|\boldsymbol{x}, \boldsymbol{\theta})p(\boldsymbol{x}|\phi)p(\phi)p(\boldsymbol{\theta})$, where $p(\boldsymbol{y}|\dot{\boldsymbol{x}}, \phi)$ and $p(\boldsymbol{x}|\phi)$ are given by the GP and $p(\dot{\boldsymbol{x}}|\boldsymbol{x}, \boldsymbol{\theta})$ is a Dirac delta distribution given by the ODE. However, this factorization is incompatible with the observation model $p(\boldsymbol{y}|\boldsymbol{x}, \sigma)$ as discussed in detail in (16). There is other related work that uses GPs in an ad hoc partial fashion to aid inference. In (13), GP regression is used to obtain the means of $\boldsymbol{x}$ and $\dot{\boldsymbol{x}}$

for embedding within an Approximate Bayesian Computation estimation procedure. In (14), GP smoothing is used during an initial burn-in phase as a proxy for the likelihood, before switching to the 'ideal' likelihood to obtain final MCMC samples. While empirical results from the aforementioned studies are promising, a principled statistical framework for inference that addresses the previously noted conceptual incompatibility between the GP and ODE specifications is lacking. Our work presents one such principled statistical framework through the explicit manifold constraint. MAGI is therefore distinct from recent GP-based approaches (11, 15) or any other Bayesian analogs of (9).

In addition to the conceptual incompatibility, none of the existing methods that do not use numerical integration offer a practical solution for a system with unobserved component(s), which highlights another unique and important contribution of our approach.

Results

We apply MAGI to three systems. We begin with an illustration that demonstrates the effectiveness of MAGI in practical problems with unobserved system component(s). Then, we make comparisons with other current methods on two benchmark systems, which show that our proposed method provides more accurate inference while having much faster runtime.

Illustration: Hes1 model. The Hes1 model described in the Introduction demonstrates inference on a system with an unobserved component and asynchronous observation times. This section continues the inference of this model. Ref (1) studied the theoretical oscillation behavior using parameter values $a = 0.022$, $b = 0.3$, $c = 0.031$, $d = 0.028$; $e = 0.5$, $f = 20$, $g = 0.3$, which leads to one oscillation cycle approximately every 2 hours. Ref (1) also set the initial condition at the lowest value of P when the system is in oscillation equilibrium (1): $P = 1.439$, $M = 2.037$, $H = 17.904$. The noise level in our simulation is derived from (1) where the standard error based on repeated measures are reported to be around 15% of the P (protein) level and M (mRNA) level, so we set the simulation noise to be multiplicative following a log-normal distribution with standard deviation 0.15, and throughout this example we assume the noise level σ is known to be 0.15 from repeated measures reported in (1). The H component is never observed. Owing to the multiplicative error on the strictly positive system, we apply our method to the log-transformed ODEs, so that the resulting error distributions are Gaussian. To the best of our knowledge, MAGI is the only one that provides a practical and complete solution for handling unobserved component cases like this example.

We generate 2000 simulated datasets based on the above setup for the Hes1 system. The left-most panel in Fig 1 shows one example dataset. For each dataset, we use MAGI to infer the trajectories and estimate the parameters. We use the posterior mean of $X_t = (P, M, H)_t$ as the inferred trajectories for the system components, which are generated by MAGI without using any numerical solver. Fig 1 summarizes the inferred trajectories across the 2000 simulated datasets, showing the median of the inferred trajectories of X_t together with the 95% interval of the inferred trajectories represented by the 2.5% and 97.5% percentiles. The posterior mean of $\boldsymbol{\theta} = (a, b, c, d, f, e, g)$ is our estimate of the parameters. Table 1 summarizes the parameter estimates across the 2000 simulated datasets, by showing their means and standard deviations. Fig 1 shows that MAGI recovers the system well, including the completely unobserved H component. Table 1 shows that MAGI also recovers the system parameters well, except for the parameters that only appear in the equation for the unobserved H component, which we will discuss shortly. Together, Fig 1 and Table 1 demonstrate that MAGI can recover the entire system without any usage of a numerical solver, even in the presence of unobserved component(s).

Metrics for assessing the quality of system recovery. To further assess the quality of the parameter estimates and the system recovery, we consider two metrics. First, as shown in Table 1, we examine the accuracy of the parameter estimates by directly calculating the root mean squared error (RMSE) of the parameter estimates to the true parameter value. We call this measure the *parameter RMSE* metric. Second, it is possible that a system might be insensitive to some of the parameters; in the extreme case, some parameters may not be fully identifiable given only the observed data and components. In these situations, it is possible that the system trajectories implied by quite distinct parameter values are similar to each other (or even close to the true trajectory). We thus consider an additional *trajectory RMSE* metric to account for possible parameter insensitivity, and measure how well the system components are recovered given the parameter and initial condition estimates. The trajectory RMSE is obtained by treating the numerical ODE solution based on the true parameter value as the ground truth: first, the numerical solver is used to reconstruct the trajectory based on the estimates of the parameter and initial condition (from a given method); then, we calculate the RMSE of this reconstructed trajectory to the true trajectory at the observation time points. We emphasize that the trajectory RMSE metric is only for evaluation purpose to assess (and compare across methods) how well a method recovers the trajectories of the system components, and that throughout MAGI no numerical solver is ever needed.

We summarize the trajectory RMSEs of MAGI in Table 2 for the Hes1 system.

We compare MAGI with the benchmark provided by the B-spline-based penalization approach of Ref (9). To the best of our knowledge, among all the existing methods that do not use numerical integration, Ref (9) is the only one with a software package that can be manually adapted to handle an unobserved component. We note, however, this package itself is not ready-made for this problem: it requires substantial manual input as it does not have default or built-in setup of its hyper-parameters for the unobserved component. None of the other benchmark methods, including Ref (11, 15), provide software that is equipped to handle an unobserved component. Table 1 compares our estimates against those given by Ref (9) based on the parameter RMSE, which shows that the parameter RMSEs for MAGI are substantially smaller than (9). Fig 1 shows that the inferred trajectories from MAGI are very close to the truth. On the contrary, the method in (9) is not able to recover the unobserved component H nor the associated parameter f and g; see Fig S1 in the SI for the plots. Table 2 compares the trajectory RMSE of the two methods. It is seen that the trajectory RMSE of MAGI is substantially smaller than that of (9). Further implementation details and

comparison are provided in the SI.

Finally, we note that MAGI recovers the unobserved component H almost as well as the observed components of P and M, as measured by the trajectory RMSEs. In comparison, for the result of (9) in Table 2, the trajectory RMSE of the unobserved H component is orders of magnitude worse than those of P and M. The numerical results thus illustrate the effectiveness of MAGI in borrowing information from the observed components to infer the unobserved component, which is made possible by explicitly conditioning on the ODE structure. The self-regulating parameter g and inhibition rate parameter f for the unobserved component appear to have high inference variation across the simulated datasets despite the small trajectory RMSEs. This suggests that the system itself could be insensitive to f and g when the H component is unobserved.

Table 1. Parameter inference in the Hes1 partially observed asynchronous system based on 2000 simulation datasets. Average parameter estimates based on MAGI and Ref (9) across the 2000 simulated datasets are reported together with the standard deviation. Parameter RMSEs are reported in the following column. The boldface highlights the best method in terms of parameter RMSE for each parameter.

		MAGI		Ref (9)	
$\boldsymbol{\theta}$	Truth	Estimate	RMSE	Estimate	RMSE
a	0.022	0.021 ± 0.003	**0.003**	0.027 ± 0.026	0.026
b	0.3	0.329 ± 0.051	**0.059**	0.302 ± 0.086	0.086
c	0.031	0.035 ± 0.006	**0.007**	0.031 ± 0.010	0.010
d	0.028	0.029 ± 0.002	**0.003**	0.028 ± 0.003	**0.003**
e	0.5	0.552 ± 0.074	0.090	0.498 ± 0.088	**0.088**
f	20	13.759 ± 3.026	**6.936**	604.9 ± 5084.8	5117.0
g	0.3	0.141 ± 0.026	**0.162**	1.442 ± 9.452	9.519

Table 2. Trajectory RMSEs of the individual components in the Hes1 system, comparing the average trajectory RMSEs of MAGI and Ref (9) over the 2000 simulated datasets. The best trajectory RMSE for each system component is shown in boldface.

Method	P	M	H
MAGI	**0.97**	**0.21**	**2.57**
Ref (9)	1.30	0.40	59.47

Comparison with previous methods based on GPs. To further assess MAGI, we compare with two methods: Adaptive Gradient Matching (AGM) of Ref (11) and Fast Gaussian process based Gradient Matching (FGPGM) of Ref (15), representing the state-of-the-art of inference methods based on GPs. For fair comparison, we use the same benchmark systems, scripts and software provided by the authors for performance assessment, and run the software using the settings recommended by the authors. The benchmark systems include the FitzHugh-Nagumo (FN) equations (17) and a protein transduction model (18).

FN model. The FitzHugh-Nagumo (FN) equations are a classic Ion channel model that describes spike potentials. The system consists of $X = (V, R)$, where V is the variable defining the voltage of the neuron membrane potential and R is the recovery variable from neuron currents, satisfying the ODE

$$\mathbf{f}(X, \boldsymbol{\theta}, t) = \begin{pmatrix} c(V - \frac{V^3}{3} + R) \\ -\frac{1}{c}(V - a + bR) \end{pmatrix}$$

where $\boldsymbol{\theta} = (a, b, c)$ are the associated parameters. As in (11, 15), the true parameters are set to $a = 0.2, b = 0.2, c = 3$, and we generate the true trajectories for this model using a numerical solver with initial conditions $V = -1$, $R = 1$.

Fig. 2. Inferred trajectories by MAGI for each component of the FN system over 100 simulated datasets. The blue shaded area represents the 95% interval.

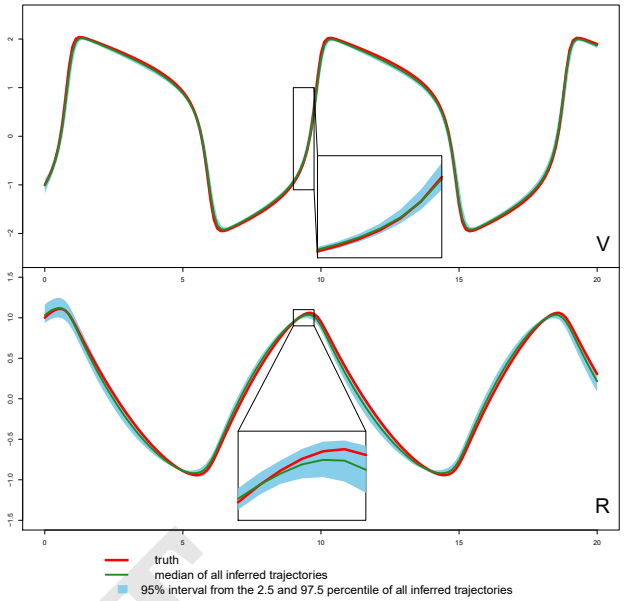


Table 3. Parameter inference in the FN model based on 100 simulated datasets. For each method, average parameter estimates are reported together with standard deviation; parameter RMSEs across simulations are also reported. The boldface highlights the best method in terms of parameter RMSE for each parameter.

	MAGI		FGPGM (15)		AGM (11)	
$\boldsymbol{\theta}$	Estimate	RMSE	Estimate	RMSE	Estimate	RMSE
a	0.19 ± 0.02	**0.02**	0.22 ± 0.04	0.05	0.30 ± 0.03	0.10
b	0.35 ± 0.09	**0.17**	0.32 ± 0.13	0.18	0.36 ± 0.06	**0.17**
c	2.89 ± 0.06	**0.13**	2.85 ± 0.15	0.21	2.04 ± 0.14	0.97

To compare MAGI with FGPGM of Ref (15) and AGM of Ref (11), we simulated 100 datasets under the noise setting of $\sigma_V = \sigma_R = 0.2$ with 41 observations. The noise level is chosen to be on similar magnitude with that of (15), and the noise level is set to be the same across the two components as the implementation of (11) can only handle equal-variance noise. The number of repetitions (i.e., 100) is set to be the same as (15) due to the high computing time of these alternative methods.

The parameter estimation results from the three methods are summarized in Table 3, where MAGI has the lowest parameter RMSEs among the three. Fig 2 shows the inferred trajectories obtained by our method: MAGI recovers the system well, and the 95% interval band is so narrow around the truth that we can only see the band clearly after magnification

(as shown in the figure inset). The SI provides visual comparison of the inferred trajectories of different methods, where MAGI gives the most consistent results across the simulations. Furthermore, to assess how well the methods recover the system components, we calculated the trajectory RMSEs, and the results are summarized in Table 4, where MAGI significantly outperforms the others, reducing the trajectory RMSE over the best alternative method for 60% in V and 25% in R. We note that compared to the true parameter value, all three methods show some bias in the parameter estimates, which is partly due to the GP prior as discussed in (15), and MAGI appears to have the smallest bias.

For computing cost, the average runtime of MAGI for this system over the repetitions is 3 minutes, which is 145 times faster than FGPGM (15) and 90 times faster than AGM (11) on the same CPU (we follow the authors' recommendation for running their methods, see SI for details).

Table 4. Trajectory RMSEs of each component in the FN system, comparing the average trajectory RMSE of the three methods over 100 simulated datasets. The best trajectory RMSE for each system component is shown in boldface. MAGI reduces the RMSE for 60% in component V and 25% in component R over the best alternative method.

Method	V	R
MAGI	**0.103**	**0.070**
FGPGM (15)	0.257	0.094
AGM (11)	1.177	0.662

Protein transduction model. This protein transduction example is based on systems biology where components S and S_d represent a signaling protein and its degraded form, respectively. In the biochemical reaction S binds to protein R to form the complex S_R, which enables the activation of R into R_{pp}. $X = (S, S_d, R, S_R, R_{pp})$ satisfies the ODE

$$\mathbf{f}(X, \boldsymbol{\theta}, t) = \begin{pmatrix} -k_1 \cdot S - k_2 \cdot S \cdot R + k_3 \cdot S_R \\ k_1 \cdot S \\ -k_2 \cdot S \cdot R + k_3 \cdot S_R + \frac{V \cdot R_{pp}}{K_m + R_{pp}} \\ k_2 \cdot S \cdot R - k_3 \cdot S_R - k_4 \cdot S_R \\ k_4 \cdot S_R - \frac{V \cdot R_{pp}}{K_m + R_{pp}} \end{pmatrix},$$

where $\boldsymbol{\theta} = (k_1, k_2, k_3, k_4, V, K_m)$ are the associated rate parameters.

We follow the same simulation setup as (11, 15), by taking $t = \{0, 1, 2, 4, 5, 7, 10, 15, 20, 30, 40, 50, 60, 80, 100\}$ as the observation times, $X(0) = (1, 0, 1, 0, 0)$ as the initial values, and $\boldsymbol{\theta} = (0.07, 0.6, 0.05, 0.3, 0.017, 0.3)$ as the true parameter values. Two scenarios for additive observation noise are considered: $\sigma = 0.001$ (low noise) and $\sigma = 0.01$ (high noise). Note that the observation times are unequally spaced, with only a sparse number of observations from $t = 20$ to $t = 100$. Further, inference for this system has been noted to be challenging due to the non-identifiability of the parameters, in particular K_m and V (15). Therefore, the parameter RMSE is not meaningful for this system, and we focus our comparison on the trajectory RMSE.

We compare MAGI with FGPGM of Ref (15) and AGM of Ref (11) on 100 simulated datasets for each noise setting (see the SI for method and implementation details). We plot the inferred trajectories of MAGI in the high noise setting in Fig 3, which closely recover the system. The 95% interval band from MAGI is quite narrow that for most of the inferred components we need magnifications (as shown in the figure insets) to clearly see the 95% band. We then calculated the trajectory RMSEs, and the results are summarized in Table 5 for each system component. In both noise settings, MAGI produces trajectory RMSEs that are uniformly smaller than both FGPGM (15) and AGM (11) for all system components. In the low noise setting, the advantage of MAGI is especially apparent for components S, R, S_R, and R_{pp}, with trajectory RMSEs less than half of the closest comparison method. For the high noise setting, MAGI reduces trajectory RMSE the most for S_d and R_{pp} (∼50%). AGM (11) struggles with this example at both noise settings. To visually compare the trajectory RMSEs in Table 5, plots of the corresponding reconstructed trajectories by different methods at both noise settings are given in the SI.

The runtime of MAGI for this system averaged over the repetitions is 18 minutes, which is 12 times faster than FGPGM (15) and 18 times faster than AGM (11) on the same CPU (we follow the authors' recommendation for running their methods, see SI for details).

Table 5. Trajectory RMSEs of the individual components in the protein transduction system, by comparing the average RMSEs of the three methods over 100 simulated datasets. The method achieving the best RMSE for each system component is shown in boldface.

Low noise case, $\sigma = 0.001$					
Method	S	S_d	R	S_R	R_{pp}
MAGI	**0.0020**	**0.0013**	**0.0040**	**0.0017**	**0.0036**
FGPGM (15)	0.0049	0.0016	0.0156	0.0036	0.0149
AGM (11)	0.0476	0.2881	0.3992	0.0826	0.2807
High noise case, $\sigma = 0.01$					
Method	S	S_d	R	S_R	R_{pp}
MAGI	**0.0122**	**0.0043**	**0.0167**	**0.0135**	**0.0136**
FGPGM (15)	0.0128	0.0089	0.0210	0.0136	0.0309
AGM (11)	0.0671	0.3125	0.4138	0.0980	0.2973

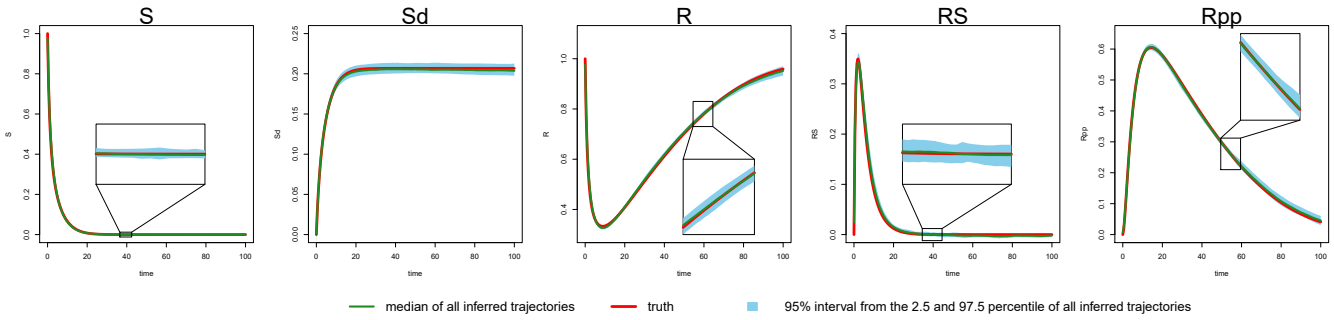
Discussion

We have presented a novel methodology for the inference of dynamic systems, using manifold-constrained Gaussian processes. A key feature that distinguishes our work from the previous approaches is that it provides a principled statistical framework, firmly grounded on the Bayesian paradigm. Our method also outperformed currently available GP-based approaches in the accuracy of inference on benchmark examples. Furthermore, the computation time for our method is much faster. Our method is robust and able to handle a variety of challenging systems, including unobserved components, asynchronous observations, and parameter non-identifiability.

A robust software implementation is provided, with user interfaces available for R, MATLAB, and Python, as described in the SI. The user may specify custom ODE systems in any of these languages for inference with our package, by following the syntax in the examples that accompany this article. In practice, inference with MAGI using our software can be carried out with relatively few user interventions. The setting of hyperparameters and initial values is fully automatic, though may be overridden by the user.

The main setting that requires some tuning is the number of discretization points in $\boldsymbol{I}$. In our examples, this was determined by gradually increasing the denseness of the points with

Fig. 3. Inferred trajectories by MAGI for each component of the protein transduction system in the high noise setting. The red line is the truth, and the green line is the median inferred trajectory over 100 simulated datasets. The blue shaded area represents the 95% interval. The inset plots magnify the corresponding segment.



short sampler runs, until the results become indistinguishable. Note that further increasing the denseness of $\boldsymbol{I}$ has no ill effect, apart from increasing the computational time. To illustrate the effect of the denseness of $\boldsymbol{I}$ on MAGI inference results, an empirical study is included in the SI "varying number of discretization" section, where we examined the results of the FN model with the discretization set $\boldsymbol{I}$ taken to be 41, 81, 161, and 321 equally spaced points, respectively. The results confirm that our proposal of gradually increasing the denseness of $\boldsymbol{I}$ works well. The inference results improve as we increase $\boldsymbol{I}$ from 41 to 161 points, and at 161 points the results are stabilized. If we further increase the discretization to 321 points, that doubles the compute time with only a slight gain in accuracy compared to 161 points in terms of trajectory RMSEs. This empirical study also indicates that as $W_{\boldsymbol{I}}$ becomes an increasingly dense approximation of W, an inference limit would be expected. A theoretical study is a natural future direction of investigation.

We also investigated the stability of MAGI when the observation time points are farther apart. This empirical study, based on the FN model with 21 observations, is included in the SI "FN model with fewer observations" section. The inferred trajectories from the 21 observations are still close to the truth, while the interval bands become wider, which is expected as we have less information in this case. We also found that the denseness of the discretization needs to be increased (to 321 time points in this case) to compensate for the sparser 21 observations[*].

An inherent feature of the GP approximation is the tendency to favor smoother curves. This limitation has been previously acknowledged (11, 15). As a consequence, two potential forms of bias can exist. First, estimates derived from the posterior distributions of the parameters may have some statistical bias. Second, the trajectories reconstructed by a numerical solver based on the estimated parameters may differ slightly from the inferred trajectories. MAGI, which is built on a GP framework, does not entirely eliminate these forms of bias. However, as seen in the benchmark systems, the magnitude of our bias in both respects is significantly smaller than the current state-of-the-art in (11, 15).

We considered the inference of dynamic systems specified by ODEs in this article. Such deterministic ODE models are often adequate to describe dynamics at the aggregate or population level (19). However, when the goal is to describe the behavior of individuals (e.g., individual molecules (20, 21)), models such as stochastic differential equations (SDEs) and continuous-time Markov processes, which explicitly incorporate intrinsic (stochastic) noise, often become the model of choice. Extending our method to the inference of SDEs and continuous-time Markov models is a future direction we plan to investigate. Finally, recent developments in deep learning have shown connections between deep neural networks and GPs (22, 23). It could thus also be interesting to explore the application of neural networks to model the ODE system outputs $\boldsymbol{x}(t)$ in conjunction with GPs.

Materials and Methods

For notational simplicity, we drop the dimension index d in this section when the meaning is clear.

Algorithm overview. We begin by summarizing the computational scheme of MAGI. Overall, we use Hamiltonian Monte Carlo (HMC) (7) to obtain samples of $\boldsymbol{X}_{\boldsymbol{I}}$ and the parameters from their joint posterior distribution. Details of the HMC sampling are included in the SI section 'Hamiltonian Monte Carlo'. At each iteration of HMC, $\boldsymbol{X}_{\boldsymbol{I}}$ and the parameters[†] are updated together with a joint gradient, with leapfrog step sizes automatically tuned during the burn-in period to achieve an acceptance rate between 60-90%. At the completion of HMC sampling (and after discarding an appropriate burn-in period for convergence), we take the posterior means of $\boldsymbol{X}_{\boldsymbol{I}}$ as the inferred trajectories, and the posterior means of the sampled parameters as the parameter estimates. The techniques we use to temper the posterior and speed up the computations are discussed in the following 'Prior tempering' subsection and 'Techniques for computational efficiency' in the SI.

Several steps are taken to initialize the HMC sampler. First, we apply a GP fitting procedure to obtain values of $\boldsymbol{\phi}$ and σ for the observed components; the computed values of the hyper-parameters $\boldsymbol{\phi}$ are subsequently held fixed during the HMC sampling, while the computed value of σ is used as the starting value in the HMC sampler. (If σ is known, the GP fitting procedure is used to obtain values of $\boldsymbol{\phi}$ only.) Second, starting values of $\boldsymbol{X}_{\boldsymbol{I}}$ for the observed components are obtained by linearly interpolating between the observation time points. Third, starting values for the remaining quantities – $\boldsymbol{\theta}$ and $(\boldsymbol{X}_{\boldsymbol{I}}, \boldsymbol{\phi})$ for any unobserved component(s) – are obtained by optimization of the posterior as described below.

Setting hyper-parameters ϕ for observed components. The GP prior $X_d(t) \sim \mathcal{GP}(\mu_d, \mathcal{K}_d)$, $t \in [0, T]$, is on each component $X_d(t)$ separately. The Gaussian process Matern kernel $\mathcal{K}(l) = \phi_1 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu}\frac{l}{\phi_2}\right)^{\nu} B_{\nu}\left(\sqrt{2\nu}\frac{l}{\phi_2}\right)$ has two hyper-parameters that are held fixed during sampling: ϕ_1 controls overall variance level of the

[*] This finding echos the classical understanding that stiff systems require denser discretization (observations farther apart make the system appear relatively more stiff).

[†] The parameters here refer to $\boldsymbol{\theta}$ and σ. If the noise level σ is known a priori, the parameters then refer to $\boldsymbol{\theta}$ only.

GP, while ϕ_2 controls the bandwidth for how much neighboring points of the GP affect each other.

When the observation noise level σ is unknown, values of (ϕ_1, ϕ_2, σ) are obtained jointly by maximizing GP fitting without conditioning on any ODE information, namely:

$$\begin{aligned}(\tilde{\boldsymbol{\phi}}, \tilde{\sigma}) &= \arg\max_{\boldsymbol{\phi},\sigma} p(\boldsymbol{\phi}, \sigma^2 | \boldsymbol{y}_{\boldsymbol{I_0}}) \\ &= \arg\max_{\boldsymbol{\phi},\sigma} \pi_{\Phi_1}(\phi_1)\pi_{\Phi_2}(\phi_2)\pi_\sigma(\sigma^2) p(\boldsymbol{y}_{\boldsymbol{I_0}} | \boldsymbol{\phi}, \sigma^2) \end{aligned} \quad [7]$$

where $\boldsymbol{y}_{\boldsymbol{I_0}} | \boldsymbol{\phi}, \sigma \sim \mathcal{N}(0, \mathcal{K}_{\boldsymbol{\phi}} + \sigma^2)$. The index set I_0 is the smallest evenly spaced set such that all observation time points in this component are in I_0, i.e., $\boldsymbol{\tau} \subseteq I_0$. The priors $\pi_{\Phi_1}(\phi_1)$ and $\pi_\sigma(\sigma^2)$ for the variance parameter ϕ_1 and σ are set to be flat. The prior $\pi_{\Phi_2}(\phi_2)$ for the bandwidth parameter ϕ_2 is set to be a Gaussian distribution: (a) the mean μ_{Φ_2} is set to be half of the period corresponding to the frequency that is the weighted average of all the frequencies in the Fourier transform of y on I_0 (the values of y on I_0 are linearly interpolated from the observations at $\boldsymbol{\tau}$), where the weight on a given frequency is the squared modulus of the Fourier transform with that frequency, and (b) the standard deviation is set such that T is three standard deviations away from μ_{Φ_2}. This Gaussian prior on ϕ_2 serves to prevent it from being too extreme. In the subsequent sampling of $(\boldsymbol{\theta}, \boldsymbol{X}_{\boldsymbol{\tau}}, \sigma^2)$, the hyper-parameters $\boldsymbol{\phi}$ are fixed at $\tilde{\boldsymbol{\phi}}$ while $\tilde{\sigma}$ gives the starting value of σ in the HMC sampler.

If σ is known, then values of (ϕ_1, ϕ_2) are obtained by maximizing

$$\tilde{\boldsymbol{\phi}} = \arg\max_{\boldsymbol{\phi}} p(\boldsymbol{\phi} | \boldsymbol{y}_{\boldsymbol{I_0}}, \sigma^2) = \arg\max_{\boldsymbol{\phi}} \pi_{\Phi_1}(\phi_1)\pi_{\Phi_2}(\phi_2) p(\boldsymbol{y}_{\boldsymbol{I_0}} | \boldsymbol{\phi}, \sigma^2) \quad [8]$$

and held fixed at $\tilde{\boldsymbol{\phi}}$ in the subsequent HMC sampling of $(\boldsymbol{\theta}, \boldsymbol{X}_{\boldsymbol{\tau}})$. The priors for ϕ_1 and ϕ_2 are the same as previously defined.

Initialization of $\boldsymbol{X}_{\boldsymbol{I}}$ for the observed components. To provide starting values of $\boldsymbol{X}_{\boldsymbol{I}}$ for the HMC sampler, we use the values of $\boldsymbol{Y}_{\boldsymbol{\tau}}$ at the observation time points and linearly interpolate the remaining points in $\boldsymbol{I}$.

Initialization of the parameter vector θ when all system components are observed. To provide starting values of $\boldsymbol{\theta}$ for the HMC sampler, we optimize the posterior Eq. (5) as a function of $\boldsymbol{\theta}$ alone, holding $\boldsymbol{X}_{\boldsymbol{I}}$ and σ unchanged at their starting values, when there is no unobserved component(s). The optimized $\boldsymbol{\theta}$ is then used as the starting value for the HMC sampler in this case.

Settings in the presence of unobserved system components: setting ϕ, initializing X_I for unobserved components, and initializing θ. Separate treatment is needed for the setting of $\boldsymbol{\phi}$ and initialization of $(\boldsymbol{\theta}, \boldsymbol{X}_{\boldsymbol{I}})$ for the unobserved component(s). We use an optimization procedure that seeks to maximize the full posterior in Eq. (5) as a function of $\boldsymbol{\theta}$ together with $\boldsymbol{\phi}$ and the whole curve of $\boldsymbol{X}_{\boldsymbol{I}}$ for unobserved components, while holding the σ, $\boldsymbol{\phi}$ and $\boldsymbol{X}_{\boldsymbol{I}}$ for the observed components unchanged at their initial value discussed above. We thereby set $\boldsymbol{\phi}$ for the unobserved component, and the starting values of $\boldsymbol{\theta}$ and $\boldsymbol{X}_{\boldsymbol{I}}$ for unobserved components at the optimized value. In the subsequent sampling, the hyper-parameters are fixed at the optimized $\boldsymbol{\phi}$, while the HMC sampling starts at the $\boldsymbol{\theta}$ and the $\boldsymbol{X}_{\boldsymbol{I}}$ obtained by this optimization.

Prior tempering. After $\boldsymbol{\phi}$ is set, we use a tempering scheme to control the influence of the GP prior relative to the likelihood during HMC sampling. Note that Eq. (5) can be written as

$$\begin{aligned} &p_{\boldsymbol{\Theta},\boldsymbol{X}(\boldsymbol{I})|\boldsymbol{Y}(\boldsymbol{\tau}),W_I}(\boldsymbol{\theta}, \boldsymbol{x}(\boldsymbol{I}) | \boldsymbol{y}(\boldsymbol{\tau}), W_{\boldsymbol{I}} = 0) \\ &\propto p_{\boldsymbol{\Theta},\boldsymbol{X}(\boldsymbol{I})|W_I}(\boldsymbol{\theta}, \boldsymbol{x}(\boldsymbol{I}) | W_{\boldsymbol{I}} = 0) p_{\boldsymbol{Y}(\boldsymbol{\tau})|\boldsymbol{X}(\boldsymbol{\tau})}(\boldsymbol{y}(\boldsymbol{\tau}) | \boldsymbol{x}(\boldsymbol{\tau})). \end{aligned} \quad [9]$$

As the cardinality of $|\boldsymbol{I}|$ increases with more discretization points, the prior part $p_{\boldsymbol{\Theta},\boldsymbol{X}(\boldsymbol{I})|W_I}(\boldsymbol{\theta}, \boldsymbol{x}(\boldsymbol{I})|W_{\boldsymbol{I}} = 0)$ grows, while the likelihood part $p_{\boldsymbol{Y}(\boldsymbol{\tau})|\boldsymbol{X}(\boldsymbol{\tau})}(\boldsymbol{y}(\boldsymbol{\tau})|\boldsymbol{x}(\boldsymbol{\tau}))$ stays unchanged. Thus, to balance the influence of the prior, we introduce a tempering hyper-parameter β with the corresponding posterior

$$\begin{aligned} &p^{(\beta)}_{\boldsymbol{\Theta},\boldsymbol{X}_{\boldsymbol{I}}|W_I,\boldsymbol{Y}_{\boldsymbol{\tau}}}(\boldsymbol{\theta}, \boldsymbol{x}_{\boldsymbol{I}} | 0, \boldsymbol{y}_{\boldsymbol{\tau}}) \\ &\propto p_{\boldsymbol{\Theta},\boldsymbol{X}(\boldsymbol{I})|W_I}(\boldsymbol{\theta}, \boldsymbol{x}(\boldsymbol{I}) | W_{\boldsymbol{I}} = 0)^{1/\beta} p_{\boldsymbol{Y}(\boldsymbol{\tau})|\boldsymbol{X}(\boldsymbol{I})}(\boldsymbol{y}(\boldsymbol{\tau})|\boldsymbol{x}(\boldsymbol{I})) \\ &\propto \pi_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) \exp\Bigg\{ -\frac{1}{2} \sum_{d=1}^{D} \Bigg[N_d \log(2\pi\sigma_d^2) + \|(x_d(\boldsymbol{\tau}_d) - y_d(\boldsymbol{\tau}_d))\|^2_{\sigma_d^{-2}} \\ &\quad + \frac{1}{\beta} \Bigg(\|x_d(\boldsymbol{I}) - \mu_d(\boldsymbol{I})\|^2_{C_d^{-1}} \\ &\quad + \left\| \mathbf{f}^{\boldsymbol{x},\boldsymbol{\theta}}_{d,\boldsymbol{I}} - \dot{\mu}_d(\boldsymbol{I}) - m_d(x_d(\boldsymbol{I}) - \mu_d(\boldsymbol{I})) \right\|^2_{K_d^{-1}} \Bigg) \Bigg] \Bigg\} \end{aligned}$$

A useful setting that we recommend is $\beta = D|\boldsymbol{I}|/N$, where D is the number of system components, $|\boldsymbol{I}|$ is the number of discretization time points, and $N = \sum_{d=1}^{D} N_d$ is the total number of observations. This setting aims to balance the likelihood contribution from the observations with the total number of discretization points.

Data availability. All of the data used in the article are simulation data. The details, including the models to generate the simulation data, are described in *Results* and the SI. Our software package also includes complete replication scripts for all the data and examples.

ACKNOWLEDGMENTS. The research of S.W.K.W. is supported in part by Discovery Grant RGPIN-2019-04771 from the Natural Sciences and Engineering Research Council of Canada. The research of S.C.K. is supported in part by NSF Grant DMS-1810914.

1. Hirata H, et al. (2002) Oscillatory expression of the bHLH factor Hes1 regulated by a negative feedback loop. *Science* 298(5594):840–843.
2. Forger DB (2017) *Biological clocks, rhythms, and oscillations: the theory of biological timekeeping.* (MIT Press).
3. Miao H, Dykes C, Demeter LM, Wu H (2009) Differential equation modeling of HIV viral fitness experiments: model identification, model selection, and multimodel inference. *Biometrics* 65(1):292–300.
4. Busenberg S (2012) *Differential Equations and Applications in Ecology, Epidemics, and Population Problems.* (Elsevier).
5. Gorbach NS, Bauer S, Buhmann JM (2017) Scalable variational inference for dynamical systems in *Advances in Neural Information Processing Systems*. pp. 4806–4815.
6. Wu L, Qiu X, Yuan Yx, Wu H (2019) Parameter estimation and variable selection for big systems of linear ordinary differential equations: A matrix-based approach. *Journal of the American Statistical Association* 114(526):657–667.
7. Neal RM (2011) MCMC Using Hamiltonian Dynamics in *Handbook of Markov Chain Monte Carlo*, Chapman & Hall/CRC Handbooks of Modern Statistical Methods, eds. Brooks S, Gelman A, Jones G, Meng X. (CRC Press), pp. 113–162.
8. Calderhead B, Girolami M, Lawrence ND (2009) Accelerating Bayesian inference over nonlinear differential equations with Gaussian processes in *Advances in neural information processing systems*. pp. 217–224.
9. Ramsay JO, Hooker G, Campbell D, Cao J (2007) Parameter estimation for differential equations: a generalized smoothing approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 69(5):741–796.
10. Hennig P, Osborne MA, Girolami M (2015) Probabilistic numerics and uncertainty in computations. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 471(2179):20150142.
11. Dondelinger F, Husmeier D, Rogers S, Filippone M (2013) ODE parameter inference using adaptive gradient matching with gaussian processes in *International Conference on Artificial Intelligence and Statistics*. pp. 216–228.
12. Barber D, Wang Y (2014) Gaussian processes for Bayesian estimation in ordinary differential equations in *International Conference on Machine Learning*. pp. 1485–1493.
13. Ghosh S, Dasmahapatra S, Maharatna K (2017) Fast approximate Bayesian computation for estimating parameters in differential equations. *Statistics and Computing* 27(1):19–38.
14. Lazarus A, Husmeier D, Papamarkou T (2018) Multiphase MCMC sampling for parameter inference in nonlinear ordinary differential equations in *International Conference on Artificial Intelligence and Statistics*. pp. 1252–1260.
15. Wenk P, et al. (2019) Fast Gaussian process based gradient matching for parameter identification in systems of nonlinear ODEs in *International Conference on Artificial Intelligence and Statistics*. pp. 1351–1360.
16. Macdonald B, Higham C, Husmeier D (2015) Controversy in mechanistic modelling with Gaussian processes. *Proceedings of Machine Learning Research* 37:1539–1547.
17. FitzHugh R (1961) Impulses and physiological states in theoretical models of nerve membrane. *Biophysical journal* 1(6):445–466.
18. Vyshemirsky V, Girolami MA (2007) Bayesian ranking of biochemical system models. *Bioinformatics* 24(6):833–839.
19. Kurtz TG (1972) The relationship between stochastic and deterministic models for chemical reactions. *The Journal of Chemical Physics* 57(7):2976–2978.
20. Kou SC, Xie XS (2004) Generalized langevin equation with fractional gaussian noise: subdiffusion within a single protein molecule. *Physical review letters* 93(18):180603.
21. Kou SC, Cherayil B, Min W, English B, Xie X (2005) Single-molecule michaelis-menten equations. *The Journal of Physical chemistry. B* 109(41):19068–19081.

22. Jacot A, Gabriel F, Hongler C (2018) Neural tangent kernel: convergence and generalization in neural networks in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*. pp. 8580–8589.
23. Lee J, et al. (2018) Deep neural networks as gaussian processes in *International Conference on Learning Representations*.