entropy

MDPI

Article

Data-Driven Corrections of Partial Lotka–Volterra Models

Rebecca E. Morrison

Department of Computer Science, University of Colorado Boulder, 1111 Engineering Drive,
Boulder, CO 80309, USA; rebeccam@colorado.edu

Received: 2 October 2020; Accepted: 16 November 2020; Published: 18 November 2020

Abstract: In many applications of interacting systems, we are only interested in the dynamic behavior of a subset of all possible active species. For example, this is true in combustion models (many transient chemical species are not of interest in a given reaction) and in epidemiological models (only certain subpopulations are consequential). Thus, it is common to use greatly reduced or partial models in which only the interactions among the species of interest are known. In this work, we explore the use of an embedded, sparse, and data-driven discrepancy operator to augment these partial interaction models. Preliminary results show that the model error caused by severe reductions—e.g., elimination of hundreds of terms—can be captured with sparse operators, built with only a small fraction of that number. The operator is embedded within the differential equations of the model, which allows the action of the operator to be interpretable. Moreover, it is constrained by available physical information and calibrated over many scenarios. These qualities of the discrepancy model—interpretability, physical consistency, and robustness to different scenarios—are intended to support reliable predictions under extrapolative conditions.

Keywords: model error; Lotka–Volterra equations; partial models; data-driven model correction; Bayesian calibration and validation

1. Introduction

In the realm of computational modeling today, scientists, mathematicians, and engineers investigate, design, optimize, and make predictions and decisions about an incredible multitude of real-world systems. In general, a computational model implements a mathematical model; the mathematical model represents the actual system in question using abstraction and simplification. In this paper, we investigate what happens when common simplifications go too far—resulting in an overly reduced or partial model—and how to account for the discrepancy between this model and the true system of interest. At the same time, these partial models still contain significant deterministic information, and they should not be thrown out entirely. Instead, we augment the partial models with a data-driven correction: we use what we know, and learn the rest.

Partial models are especially common in the context of the generalized Lotka–Volterra (GLV) equations. These equations describe the interactive behavior of any number S of different species. The concentration of each species is represented by a variable $x_i, i = 1, \dots S$; there is one differential equation for each x_i whose right-hand side (RHS) includes a linear growth rate term and nonlinear interaction terms. This framework, also called the quasipolynomial form, is canonical in dynamical systems [1] and is used to describe many types of physical systems, including reaction models for chemical kinetics [2], ecological models [3], and epidemiological models [4]. In these applied fields, modelers commonly build a partial model with only $s < S$ species. For example, there are over 50 chemical species thought to be involved in methane combustion [5]; in practice, often merely five to ten species are included [6]. Surprisingly, Kourdis and Belan found that the removal of 90%–99%

of chemical species can still lead to reliable output [7,8]. SEIR-type epidemiology models include subpopulations of Susceptible, Exposed, Infected, and Recovered (hence the name) humans and the disease carriers (e.g., mosquitoes), while omitting many others such as asymptomatic or hospitalized humans [9], or even cattle or nonhuman primates [10].

Validation is the process by which we check that the mathematical model faithfully represents the system in question. While all models should undergo validation, this is especially important when we know our models are incomplete. (*Verification* is the process by which we check that any computation correctly solves the mathematical problems. For example, proper verification procedures include code documentation, unit and regression testing, and solution comparisons against a posteriori error estimates, to name a few. While critical to the success of computational modeling, verification is not a concern of this paper; we assume all computational implementations are correctly documented, implemented, and executed. For more information about verification, see, e.g., [11–13].) In its most basic form, validation checks that the model output is consistent with all relevant observations. Statistical techniques that do not require knowledge of the model include, for example, goodness of fit (computing R^2 values), analysis of residuals between model output and data, and k-fold cross-validation [14]. However, a validation process may require a more nuanced procedure, depending on what one plans to do with the model. In [15], Oliver et al. described a sophisticated approach to model validation for predictions of unobservable quantities. Their framework relies on knowledge of the model and system under study, and it takes the behavior of the model over different scenarios into account. In [16], Bayarri et al. described a comprehensive framework for the validation of computer experiments. Their framework includes detailed processes such as determining appropriate domains of model inputs, guarding against overfitting, and accounting for bias in the simulation output. In [17], Farrell-Maupin and Oden described an adaptive method for model calibration and validation using increasingly complex models. In that method, additional richness is only introduced to the model after the simpler version is shown to be invalid. Finally, Jiang et al. implemented a sequential approach to model calibration, validation, and experimental design with the goal of reducing model bias by misspecified and reduced models [18].

Note that all validation procedures rely on access to observations, which should (hopefully) include a description of the associated measurement error. If there is some mismatch between the model and the observations, the source of the discrepancy could either be the model or the observations, or both. Reliable experimental practices and proper data reduction techniques ensure that all observations are reported correctly with quantified measurement uncertainty. In this paper, we assume that any discrepancy between the model and observations is not caused by faulty experimental procedures or reporting. In this way, we may focus on what to do when the model itself causes the discrepancy.

Once validation reveals a discrepancy that cannot be reasonably explained by measurement error, we must improve the model. Consider that the discrepancy is revealed by comparing some set of model output to the corresponding set of observations. If a bias is perceived between the two, a natural first step would be to attach a discrepancy function or stochastic process to the model output, which can then be calibrated to correct the model. In fact, this type of discrepancy model, which we call a response discrepancy model, has been duly investigated, starting with the fundamental work of Kennedy and O'Hagan in 2001 [19]. Since then, the response discrepancy model has been adapted into fields as diverse as climate modeling [20], hydrology [21], and cardiology [22], among many others. Lewis et al. developed an information-theoretic approach to calibrate a low-fidelity model, including a response discrepancy function, from high-fidelity model outputs [23]. This approach is based on Bayesian experimental design to both minimize uncertainty in the low-fidelity model parameters and reduce the number of needed high-fidelity model runs. A response discrepancy model can be useful and relatively quick to develop when one only needs to interpolate between data points.

Recall, however, our goals for computational modeling: to investigate, design, optimize, and make predictions and decisions. To achieve these goals, we must be able to trust the model output beyond a specific calibration scenario; otherwise, we could just rely on observations without need for a model. To this end, we aim to represent the model discrepancy with a discrepancy operator embedded within the model itself, i.e., an embedded discrepancy operator. There are several advantages of an embedded discrepancy operator. First, the operator can be constrained by physical information such as conservation laws, symmetries, fractional concentrations, nonnegativity constraints, and so on. Second, as a function of state variables or other existing model variables, the action of the discrepancy operator is physically interpretable. Third, the operator can be calibrated over many different scenarios, such as initial conditions, boundary conditions, or simulation geometries. Because of these qualities—physical consistency, interpretability, and robustness in different scenarios—an embedded discrepancy model could be valid for extrapolative predictions.

Embedded, or intrusive, approaches have been previously investigated. In [24], Sargsyan et al. allowed for model error by endowing model parameters with random variable expansions. As an approach to model discrepancy, this does not break physical constraints, and the random parameters can be calibrated over many scenarios. However, not all model error can by captured in this way. With complex computational models, missing physics or misspecified physics sometimes causes the discrepancy. This is the situation considered in the current paper. Thus, the discrepancy model becomes part of the model itself, yielding an augmented or enriched model, in which case the specific form of the discrepancy model depends on the modeling context. In [25], the authors investigated this type of inadequacy operator in the context of chemical kinetics for combustion. In this work, we propose and analyze a class of embedded discrepancy operators in the context of the generalized Lotka–Volterra equations, and we show that the error of highly reduced models can be captured by sparse linear operators. For example, a detailed model with 20 species includes 420 parameters where 400 correspond to nonlinear terms, a partial model of four species includes 20 of these, and our discrepancy operator introduces only eight new linear terms.

Although the context is different, the work here is perhaps most similar in philosophy and techniques to that which examines the closure problem of reduced-order fluid dynamics models such as RANS (Reynolds-averaged Navier–Stokes) and LES (large-eddy simulation). For example, Portone et al. developed an embedded operator in [26] for porous media flow models of contaminant transport. Pan and Duraisamy constructed general data-driven closure models for both linear and chaotic systems, including canonical ordinary differential equations (ODEs) and the one-dimensional Burgers equation [27]. Other works describe data-driven closure models using proper orthogonal decomposition [28], Mori–Zwanzig techniques [29], and linear approximations to closure models for the Kuramoto–Sivashinsky equation [30]. In a sense, the current paper presents a type of data-driven "closure model" for partial Lotka–Volterra equations.

The paper is organized as follows. A brief review of the generalized Lotka–Volterra (GLV) equations along with a description of the detailed and partial models is given in Section 2. In Section 3, a class of embedded discrepancy operators and a method for enforcing physics-based constraints are proposed. The details of calibration and validation for the enriched models are given in Section 4. Numerical results are listed in Section 5, and the paper concludes with a discussion in Section 6. Existing techniques to manipulate ordinary differential equations that motivate the proposed discrepancy operators are reviewed in Appendix A.

2. Generalized Lotka–Volterra Equations

The generalized Lotka–Volterra equations are coupled ordinary differential equations, used to model the time dynamics of any number of interacting quantities. In particular, the Lotka–Volterra framework allows for linear (growth rate) and quadratic (interaction) terms.

2.1. Detailed Models

The objects in the detailed models will be denoted with a $\hat{}$ symbol. Let the vector $\hat{x} \in \mathbb{R}^S$ represent species concentrations. Here, the units of a particular $\hat{x}_i$ refer to the number of specimens per unit area, but specific units are omitted in this paper. The GLV equations of the detailed model $\mathcal{D}$ are written succinctly as:

$$\frac{d\hat{\boldsymbol{x}}}{dt} = \mathcal{D}(\hat{\boldsymbol{x}}) = \operatorname{diag}(\hat{\boldsymbol{x}})(\hat{\boldsymbol{r}} + \hat{A}\hat{\boldsymbol{x}}), \tag{1}$$

where the vector $\hat{\boldsymbol{r}} \in \mathbb{R}^S$ represents the intrinsic growth rates, and the matrix $\hat{A} \in \mathbb{R}^{S\times S}$ collects the interaction rates—that is, the *ij*th entry of $\hat{A}$, a_{ij}, indicates how species j affects the concentration of species i. The equilibrium solution is $\hat{\boldsymbol{x}}_{eq} = -\hat{A}^{-1}\hat{\boldsymbol{r}}$.

Since this model is completely determined by the vector $\hat{\boldsymbol{r}}$ and the matrix $\hat{A}$ (modulo initial conditions), we also say that $D = \{\hat{A}, \hat{\boldsymbol{r}}\}$. The species included in the detailed model are called the detailed set. The term intraspecific refers to interactive behavior within a particular species (the $\hat{a}_{ii}$ are intraspecific terms), while the term interspecific refers to the behavior between two different species (the $a_{ij}, i \neq j$, are interspecific terms).

2.2. Partial Models

The partial model is comprised of all terms involving the s species of interest, i.e., by subsampling the detailed one. For example, suppose $S = 3$ and $s = 2$. Then, the detailed model, written out, is

$$\frac{d\hat{x}_1}{dt} = r_1\hat{x}_1 + (a_{11}\hat{x}_1 + a_{12}\hat{x}_2 + a_{13}\hat{x}_3)\hat{x}_1 \tag{2a}$$

$$\frac{d\hat{x}_2}{dt} = r_2\hat{x}_2 + (a_{21}\hat{x}_1 + a_{22}\hat{x}_2 + a_{23}\hat{x}_3)\hat{x}_2 \tag{2b}$$

$$\frac{d\hat{x}_3}{dt} = r_3\hat{x}_3 + (a_{31}\hat{x}_1 + a_{32}\hat{x}_2 + a_{33}\hat{x}_3)\hat{x}_3 \tag{2c}$$

and the partial model is (now without the ˆ)

$$\frac{dx_1}{dt} = r_1x_1 + (a_{11}x_1 + a_{12}x_2)x_1 \tag{3a}$$

$$\frac{dx_2}{dt} = r_2x_2 + (a_{21}x_1 + a_{22}x_2)x_2. \tag{3b}$$

Likewise, the partial model is referred to as $\mathcal{P}$, so

$$\frac{d\boldsymbol{x}}{dt} = \mathcal{P}(\boldsymbol{x}) = \operatorname{diag}(\boldsymbol{x})(\boldsymbol{r} + A\boldsymbol{x}), \tag{4}$$

and $P = \{A, \boldsymbol{r}\}$. Here, the equilibrium solution is $\boldsymbol{x}_{eq} = -A^{-1}\boldsymbol{r}$.

In this example, the growth rate vectors and interaction matrices are

$$\hat{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}, \quad \hat{\boldsymbol{r}} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix}, \quad A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}, \quad \boldsymbol{r} = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix}. \tag{5}$$

The species included in the partial model are called the partial set and sometimes also the remaining species—that is, remaining after a reduction process.

2.3. Defining the Scope

As the objective of this paper is to understand model discrepancy in the context of partial GLV models, we must define the scope of this context. There are a few considerations to keep in mind. First, note that, as explained above, the partial model considered here follows immediately from the detailed model. Thus, when determining the scope of models under investigation, it suffices to determine the detailed model(s). Then, given a detailed model, we investigate all possible partial models from $s = 1$ to $s = S - 1$.

Second, the GLV equations encompass an infinite number of specific models, or model realizations, as S can be any integer ≥ 2, and the entries of $\hat{A}$ and $\hat{\boldsymbol{r}}$ can be, in theory, any real numbers. Moreover, any two GLV models, determined by a specific $\hat{A}$ and $\hat{\boldsymbol{r}}$, may behave differently from one another. At the most specific extreme, all model parameters are fixed, yielding a single fixed pair of detailed and partial models, and we could then investigate the model discrepancy therein. At the most general extreme, many models are supplied via highly unconstrained realizations of the model parameters, and we could hope to thus discover highly general results about the model discrepancy. In this paper, by aiming somewhere in between these two extremes, we examine a moderately general random class of LV models. This class is determined by specifying appropriate distributions for the entries of $\hat{A}$ and $\hat{\boldsymbol{r}}$.

Third, since this is an initial exploration into representing model discrepancy in the GLV context, let us narrow the scope in order to examine well-behaved models, i.e., those with stable equilibria. We focus on symmetric interaction matrices $\hat{A}$ with negative entries. This constraint says that all interactions between and within species are competitive, not cooperative. Then, the matrix $\hat{A}$ can be stabilized by making its diagonal entries larger in magnitude than the sum of off-diagonal entries in the same row (or column), ensuring diagonal dominance and thus negative eigenvalues. Here, we consider models whose interaction matrices are symmetric, diagonally dominant, and have negative entries. These restrictions (or similar, e.g., that $\hat{A}$ be negative definite) are common assumptions in mathematical ecology studies; see, e.g., [3,31,32]. The distributional form characterizing entries of $\hat{A}$ and $\hat{\boldsymbol{r}}$ are given in the following subsection. Then, in Section 5, specific models are sampled and analyzed through numerical examples.

2.4. Creating the GLV Detailed and Partial Models

In this subsection, the information above is summarized and refined algorithmically. Algorithm 1 generates a realization of a detailed model, and Algorithm 2 provides the corresponding partial model. Recall that we use ˆ to differentiate between the two and denote a quantity of the detailed model.

Algorithm 1 Generating a realization of the detailed model.

```
Initialize $S$
Sample $B_{ij} \sim \log\mathcal{N}(0, \sigma_B^2), 1 \leq i < j \leq S$
Set $B_{ji} = B_{ij}$
Sample $C_{ii} \sim \log\mathcal{N}(0, \sigma_C^2) + \sum_{k \neq i} B_{ki}, 1 \leq i \leq S$
Set interaction matrix $\hat{A} = -(B + C)$
Set growth rate vector $\hat{\boldsymbol{r}} = \max\{C\}\mathbf{1}_S$
return $D = \{\hat{A}, \hat{\boldsymbol{r}}\}$
```

Algorithm 2 Subsampling the partial model.

```
Initialize $s < S, D$
Set $A$ as submatrix: $A = \hat{A}_{1:s,1:s}$
Set $r$ as subvector: $\boldsymbol{r} = \hat{\boldsymbol{r}}_{1:s}$
return $P = \{A, \boldsymbol{r}\}$
```

Without loss of generality, we simply choose the first s species in Algorithm 2, as the detailed set follows no special or implicit ordering. Note that existence of a stable equilibrium of the partial model follows directly from that of the detailed model.

3. Enriched GLV Model

Previous work shows how a set of S coupled Lotka–Volterra equations can be converted to a set of s equations, $s < S$, using algebraic substitutions and/or integration [33]. The resulting equations will either need to depend on higher derivatives of the remaining species or on their complete time history—that is, the exact dynamics from the detailed model may be written only in terms of the partial set:

$$\frac{d\boldsymbol{x}}{dt} = \mathcal{F}(\boldsymbol{x}, \dot{\boldsymbol{x}}, \ddot{\boldsymbol{x}}, \dots, K(\boldsymbol{x})), \tag{6}$$

where $\dot{\boldsymbol{x}}$ is the first derivative of $\boldsymbol{x}$, $\ddot{\boldsymbol{x}}$ the second derivative, and so on, and $K(\boldsymbol{x})$ represents some memory kernel. Two such manipulations are reviewed in Appendix A. This motivates an approximation of $\mathcal{F}$ with the available partial model $\mathcal{P}$ and a discrepancy model Δ that is a function of either the derivatives or memories of the remaining variables—that is, we seek a model for the partial set of variables as:

$$\frac{d\boldsymbol{x}}{dt} \approx \mathcal{P}(\boldsymbol{x}) + \Delta(\boldsymbol{x}, \dot{\boldsymbol{x}}, \ddot{\boldsymbol{x}}, K(\boldsymbol{x})). \tag{7}$$

The above may be reminiscent of Takens's theorem [34], in which a dynamical system is reconstructed from (delayed) observations of the system. However, the two approaches differ fundamentally: here, the LHS derivatives are restricted to those of the partial set, i.e., a subset of the original variables, but that is not true in a delay embedding.

We now propose a particular form of Δ.

3.1. Linear Embedded Discrepancy Operator

Recall that the detailed model is

$$\frac{d\hat{\boldsymbol{x}}}{dt} = \mathcal{D}(\hat{\boldsymbol{x}}) = \mathrm{diag}(\hat{\boldsymbol{x}})(\hat{\boldsymbol{r}} + \hat{A}\hat{\boldsymbol{x}}). \tag{8}$$

and the partial model is

$$\frac{d\boldsymbol{x}}{dt} = \mathcal{P}(\boldsymbol{x}) = \mathrm{diag}(\boldsymbol{x})(\boldsymbol{r} + A\boldsymbol{x}). \tag{9}$$

We initially propose an enriched model $\mathcal{E}$, linear in $(\boldsymbol{x}, \dot{\boldsymbol{x}})$, of the form

$$\frac{d\boldsymbol{x}}{dt} = \mathcal{E}(\boldsymbol{x}, \dot{\boldsymbol{x}}) \tag{10a}$$

$$= \mathcal{P}(\boldsymbol{x}) + \mathrm{diag}(\boldsymbol{x})\delta_0 + \mathrm{diag}(\dot{\boldsymbol{x}})\delta_1 \tag{10b}$$

$$= \mathcal{P}(\boldsymbol{x}) + \Delta(\boldsymbol{x}, \dot{\boldsymbol{x}}), \tag{10c}$$

where $\delta_0 = (\delta_{10}, \delta_{20}, \dots, \delta_{s0})^T$ and $\delta_1 = (\delta_{11}, \delta_{21}, \dots, \delta_{s1})^T$. The subscripts on each δ_{ij} are chosen so that i indicates that this coefficient appears in the RHS of the variable x_i, and j indicates that this coefficient is multiplying the jth derivative of x_i.

A major advantage of an embedded operator, as opposed to a response discrepancy model, is that the operator can be constrained by any available information about the physical system. In this simple example, we do have some information about the system that implies constraints on the introduced discrepancy parameters δ_0, δ_1. First, we make the modeling ansatz that these discrepancy parameters should not depend explicitly on time. A result of this ansatz is then that the parameters be constrained independently. We also (assume that we) know that all interspecific interactions are competitive. In particular, note that $a_{ij}x_ix_j < 0$ because $a_{ij} < 0$ and $x_i, x_j \geq 0$. Thus, we enforce that $\Delta_i(x_i, \dot{x}_i) \leq 0$. Thus, specific information about the high-fidelity physical system implies the following constraints:

- We know $x_i \geq 0$ which implies $\delta_{i0} \leq 0$.

- The constraint on δ_{i1} is slightly less clear since the sign of $\dot{x}_i$ could be positive or negative. Thus, we could set $\delta_{i1} = \tilde{\delta}_{i1}\mathrm{sgn}(\dot{x}_i)$, where $\tilde{\delta}_{i1} \leq 0$. Equivalently, we can write the discrepancy as

$$\Delta_i(x_i, \dot{x}_i) = \delta_{i0}x_i + \delta_{i1}\,|\dot{x}_i|\,. \tag{11}$$

Then, set $\delta_{i1} \leq 0$ and the constraint is satisfied.

Because of this final constraint, the discrepancy operator is no longer linear in $\dot{\boldsymbol{x}}$, but rather in $|\dot{\boldsymbol{x}}|$. We still refer to such a formulation as linear (precedence for this use of linear is found in [35]). Thus, we amend the above enriched model in lines (10a)–(10c) as

$$\begin{aligned} \frac{d\boldsymbol{x}}{dt} &= \mathcal{E}(\boldsymbol{x}, |\dot{\boldsymbol{x}}|) && \text{(12a)}\\ &= \mathcal{P}(\boldsymbol{x}) + \mathrm{diag}(\boldsymbol{x})\boldsymbol{\delta}_0 + \mathrm{diag}(|\dot{\boldsymbol{x}}|)\boldsymbol{\delta}_1 && \text{(12b)}\\ &= \mathcal{P}(\boldsymbol{x}) + \Delta(\boldsymbol{x}, |\dot{\boldsymbol{x}}|). && \text{(12c)} \end{aligned}$$

Finally, the introduced discrepancy parameters $\boldsymbol{\delta}_0, \boldsymbol{\delta}_1$ are calibrated, using observations of species concentrations generated by the detailed model. Indeed, the strength of the embedded operator approach stems from two properties: (1) the ability to constrain the formulation by available physical information, and (2) the ability to leverage information from the detailed system by calibrating the model discrepancy parameters. Moreover, we calibrate over a range of initial conditions, denoted as $\boldsymbol{\phi}_i, i = 1, \ldots, n_{\phi_c}$. Note that we also validate over a range of n_{ϕ_v} initial conditions $\boldsymbol{\phi}_i, i = n_{\phi_c} + 1, \ldots, n_\phi$ so that $n_{\phi_c} + n_{\phi_v} = n_\phi$. Each $\boldsymbol{\phi}_i$ specifies the species initial concentrations:

$$\boldsymbol{\phi}_i = (x_1(0), x_2(0), \ldots, x_s(0)), \quad i = 1, \ldots, n_\phi. \tag{13}$$

By calibrating with observations from all n_{ϕ_c} scenarios, the goal is to build a more robust discrepancy model that is valid over several scenarios instead of only calibrated to a very specific dataset. This property of the model discrepancy construction further allows for the possibility, at least, that such an enriched model could be used in extrapolative conditions, such as a prediction in time, or in scenarios given by different initial conditions.

Note that the actual observations used to calibrate the parameters are specified in Section 4, along with the particulars of the calibration itself. We have tried to separate what is essential to the formation of the discrepancy operator from the calibration details, which could reasonably change based on the example at hand.

3.2. Equilibrium and Stability of the Enriched Models

The equilibrium solution of the enriched model is

$$\boldsymbol{x}_{eq} = -A^{-1}(\boldsymbol{r} + \boldsymbol{\delta}_0). \tag{14}$$

Thus, the parameters $\boldsymbol{\delta}_0$, which act linearly on the state $\boldsymbol{x}$, directly control the equilibrium solution. The stability of the enriched model is less obvious because of the absolute values; here, we conjecture that the models are indeed stable.

To see why, first consider an example enriched system of just one variable x:

$$\dot{x} = x - x^2 - \frac{1}{2}|\dot{x}|. \tag{15}$$

Depending on the sign of $\dot{x}$, this becomes one of the following logistic equations

$$\dot{x} = 2x(1-x), \quad \dot{x} < 0 \tag{16a}$$

$$\dot{x} = \frac{2}{3}x(1-x), \quad \dot{x} > 0. \tag{16b}$$

The logistic equations admit solutions of the qualitative nature shown in Figure 1.

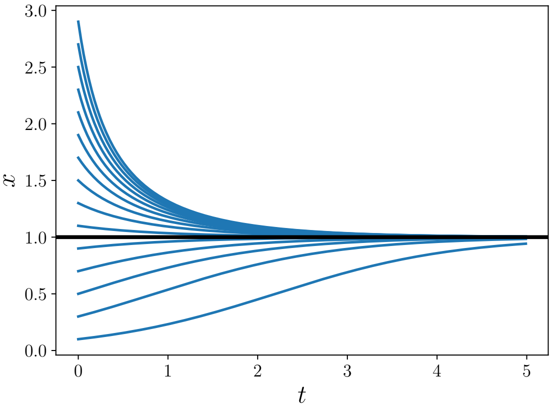


Figure 1. Solutions to the logistic equation. The thick black line shows the $x = 1$ phase line.

Importantly, the sign of $\dot{x}$ never changes over any solution curve. Given the initial condition, we could in fact solve the same system without the absolute value by choosing either (16a) or (16b); both reach stable equilibrium. Thus, the presence of the absolute value in this example does not affect the stability of the system.

In general, the signs of the derivatives may change. However, we conjecture that the derivatives of all species do not change sign after a given point in time, say t^* (as seen in the numerical examples in Section 5). In this case, the enriched differential equation for $x_i(t), t > t^*$ is

$$\dot{x}_i = \lambda_i \left((r_i + \delta_{i0})x_i + \sum_{j=1}^{s} a_{ij}x_i x_j \right) \tag{17}$$

where

$$\lambda_i = \begin{cases} \frac{1}{1+\delta_{i1}}, & \dot{x}_i(t) < 0 \ \forall t > t^* \\ \frac{1}{1-\delta_{i1}}, & \dot{x}_i(t) > 0 \ \forall t > t^*. \end{cases} \tag{18}$$

The above does reach a stable equilibrium, as λ_i simply scales the overall dynamics and the interaction matrix A (still) determines the stability of the system.

A more rigorous analysis of stability for these systems will be addressed in future work. For now, we note that differential equations with absolute value terms have been treated in the literature. In particular, Khan and Barton showed that, for ODEs whose RHS are a composition of analytic and absolute value functions of the state variables, the arguments of the absolute values change sign finitely many times in any finite duration [36], while Barton et al. provided a theoretical and computational framework for evaluating nonsmooth derivatives called lexicographic directional derivatives [37]. Finally, Oakley demonstrated that certain second-order differential equations with absolute values admit solutions of sets of related linear differential equations [35].

3.3. Proof of Concept: Linear Embedded Discrepancy Operator, $S = 2, s = 1$

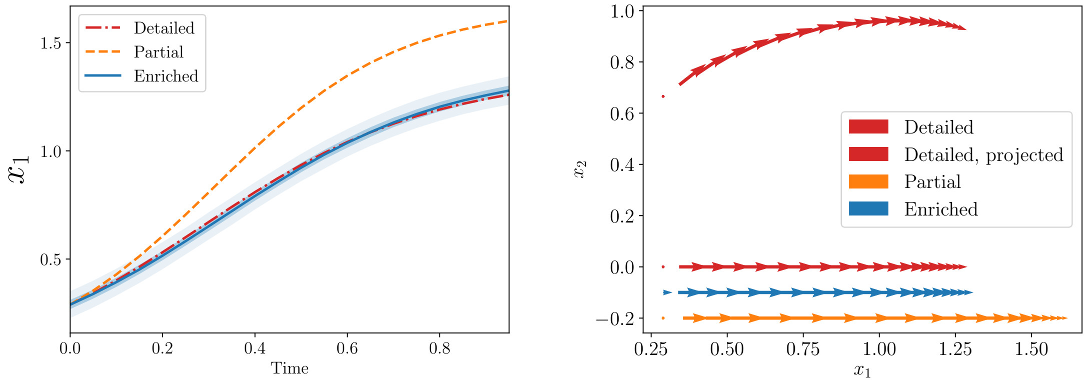
As an initial proof of concept, consider the $S = 2, s = 1$ case. The detailed model is

$$\mathcal{D} = (\hat{A}, \hat{\boldsymbol{r}}) = \left(\begin{bmatrix} -3 & -1 \\ -1 & -2 \end{bmatrix}, \begin{bmatrix} 5 \\ 3 \end{bmatrix} \right) \tag{19}$$

and the partial model is simply $\mathcal{P} = (A, \boldsymbol{r}) = (a_{11}, r_1) = (-3, 5)$. In this case, the exact discrepancy is $-x_2 x_1$, and we aim to approximate the effect of this term with

$$\Delta_1(x_1, \dot{x}_1) = \delta_{10} x_1 + \delta_{11} |\dot{x}_1|. \tag{20}$$

Calibration yields posterior mean values of $\bar{\delta}_{10} \approx -0.837$ and $\bar{\delta}_{11} \approx -0.0224$; further calibration details are deferred to the next section. The three models—detailed, partial, and enriched—are shown in Figure 2a; excellent agreement between the detailed and enriched models is achieved.



(**a**) (**b**)

Figure 2. (**a**) Trajectories of x_1 from detailed, partial, and enriched models for a simple example with $S = 2, s = 1$. (**b**) Phase diagrams from the same three models. The 1D derivatives from the enriched model approximately match the projection from the 2D detailed case onto the x_1-axis. Note that the 1D enriched and partial derivatives are plotted below the projected detailed case for visualization purposes (along lines $x_2 = -0.1, -0.2$).

We also show the phase diagram of the three models in Figure 2b including the 2D phase diagram from the detailed model projected onto the x_1-axis. The recovered derivatives of the enriched model approximately match this projection quite well. Analogous plots are difficult to visualize in higher dimensions, but in this low-dimensional case, this projection may provide some intuition about why the enriched model behaves like the detailed model.

3.4. Other Possible Formulations

There are a number of related possible formulations of the model discrepancy. Some options are the following:

1. An affine expression up to the Nth derivative:

$$\Delta_i = \mu_i + \sum_{j=0}^{N} \delta_{ij} \frac{d^j}{dt^j}(x_i). \tag{21}$$

2. A quadratic expression up to the Nth derivative. Let

$$\boldsymbol{q} = \left\{ \frac{d^0}{dt^0}(x_1), \dots, \frac{d^N}{dt}(x_1), \dots, \frac{d^0}{dt^0}(x_s), \dots, \frac{d^N}{dt^N}(x_s) \right\}.$$

Then

$$\Delta_i = \mu_i + \sum_{i,j}^{s(N+1)} \delta_{ij}(q_i q_j). \tag{22}$$

3. A memory expression, such as:

$$\Delta_i(t) = \mu_i + \beta_i \int_{s=0}^{t} x_i(s)ds \tag{23}$$

for some $\beta_i \in \mathbb{R}$.

Each of the above formulations includes an affine term μ_i. Whether or not such a constant term would be advantageous when all the missing dynamics terms are state-dependent is not immediately clear.

Of course, one could also propose some combination of the above formulations as an embedded discrepancy operator. Investigating the numerical advantages and limitations of many such discrepancy operators is beyond the scope of the current paper. For now, numerical results are presented in Section 5 about the proposed linear embedded discrepancy operator, as described in Section 3.1.

4. Calibration and Validation

This section contains all relevant details about the calibration and validation processes. First, for both of these, it is necessary to know what observations are available.

4.1. The Observations

The datasets used to calibrate and validate the discrepancy model include observations from the detailed model trajectories of the s species included in the partial model. From each trajectory, T observations are taken, and there is a new trajectory for each initial condition ϕ, so that the observations can be summarized as

$$\mathcal{O} = \{y_{ijk}\}, i = 1, \ldots, s; j = 1, \ldots, T; k = 1, \ldots, n_\phi \tag{24}$$

where y_{ijk} is the observation of $x_i(t_j)$ given the initial condition ϕ_k. This observed value y^* is given by the true value y^t with additive measurement error ϵ:

$$y^* = y^t + \epsilon, \tag{25}$$

where the distribution of measurement error is normal: $p_\epsilon = \mathcal{N}(0, \sigma_\epsilon^2)$.

Finally, this set of observations is partitioned into two sets, one for calibration and the other for validation. Let us partition as follows:

$$\text{Calibration data:} \quad \mathcal{O}_c = \{y_{ijk}\}, i = 1, \ldots, s; j = 1, \ldots, T; k = 1, \ldots, n_{\phi_c} \tag{26}$$

$$\text{Validation data:} \quad \mathcal{O}_v = \{y_{ijk}\}, i = 1, \ldots, s; j = 1, \ldots, T; k = n_{\phi_c} + 1, \ldots, n_\phi. \tag{27}$$

That is, n_{ϕ_c} initial conditions are used for calibration, and the remaining n_{ϕ_v} are designated for validation, where $n_{\phi_c} + n_{\phi_v} = n_\phi$.

4.2. Calibration Details

The calibration is done using a Bayesian approach, and the details of the calibration problem are as follows.

- **Prior:** We set uniform prior distributions on the discrepancy parameters θ:

$$p(\theta) = \prod_{\substack{i=1,\dots s \\ j=0,1}} p(\delta_{ij}), \tag{28}$$

where

$$p(\delta_{ij}) = \mathcal{U}(-100, 0) \quad i = 1, \dots, s; j = 0, 1. \tag{29}$$

(One might expect a negative lognormal distribution for these priors, and this was in fact the first choice. However, the uniform priors performed much better during the sampling process, and all of the parameter chains in Markov Chain Monte Carlo simulations were well-contained by the uniform bounds. Why the lognormal priors led to poor mixing will be investigated further in future work.)

- **Likelihood:** The likelihood is determined by the measurement error:

$$p(\mathcal{O}_c|\theta) = \prod_{l=1,\dots,|\mathcal{O}_c|} p_\epsilon(y_l - y_{l,\mathcal{E}}) \tag{30}$$

where the observations have been re-indexed from 1 to $|\mathcal{O}_c|$ (to avoid triple subscripts here) and $y_{l,\mathcal{E}}$ is the corresponding model output from the enriched model $\mathcal{E}$.

- **Posterior:** Given the prior and likelihood distributions above, the posterior distribution follows as:

$$p(\theta|\mathcal{O}_c) \propto p(\mathcal{O}_c|\theta)p(\theta). \tag{31}$$

Specifically, the calibration is performed according to the Delayed Rejection Adaptive Metropolis (DRAM) method, introduced in [38] and implemented in the statistical library QUESO [39].

4.3. Validation Metric

Next, we must define an appropriate quantitative validation metric. First, we quantify the consistency between the enriched model output and the corresponding observation. We compute how probable the observation is as a realization of the model output. The probability of observing some y^*, given the data $\mathcal{O}_c$, is

$$p(y^*|\mathcal{O}_c) = \int_{y^t} p_\epsilon(y^t - y^*) \left(\int_\theta p(y^t|\theta)p(\theta|\mathcal{O}_c)d\theta \right) dy^t. \tag{32}$$

We can compare this probability to the rest of possible model outputs. In particular, we are interested in how much of the distribution corresponds to model outputs less likely than the one above in (32). This amount is exactly given by the $\gamma-$value, as defined in [15]:

$$\gamma_{y^*} = \int_{y \in \mathcal{S}} p(y|\mathcal{O}_c)dy \tag{33}$$

where $\mathcal{S} = \{y : p(y|\mathcal{O}_c) \leq p(y^*|\mathcal{O}_c)\}$. Note that a low γ-value implies that the observation is less probably an outcome of this model than most possible outcomes. In contrast, values that are not low demonstrate consistency between the model and observation. In this work, we compute the fraction of γ-values below a given threshold τ. For a more thorough introduction to γ-values and discussion of their use in model validation, see [15], and for another example of this used in practice as a validation metric, see [25].

An example of the area corresponding to this integral is given in Figure 3. In this work, we compute the integrals with a Monte Carlo approach [40].

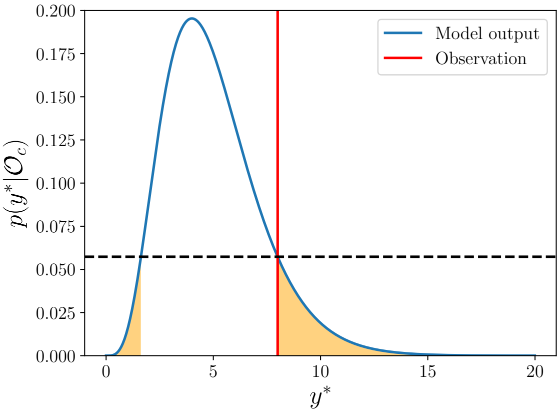


Figure 3. The γ-value corresponds to the shaded area. The dashed horizontal line simply shows the y-axis value where the observation crosses the model output density, so that we integrate over the set for which the density is lower than this value.

5. Numerical Results

We now present the numerical performance of the proposed linear embedded discrepancy operator described in Section 3.1. All code—to run forward and inverse problems, generate data, and postprocess—is available here: `github.com/rebeccaem/enriched-glv` [41].

5.1. Results for One Realization of the Detailed Model

First, let us examine results for a single detailed and partial model. The detailed model is generated according to Algorithm 1, with the following values:

$$S = 10, \quad \sigma_B^2 = 1, \quad \sigma_C^2 = 1. \tag{34}$$

Then, the partial model is generated according to Algorithm 2 with $s = 4$. In this example, the observations from the detailed model are taken so that $n_{\phi_c} = 3$, $n_{\phi_v} = 3$, $T = 10$, and $\sigma_\epsilon^2 = 0.001$. The entries of each initial condition vector $\boldsymbol{\phi}_i$ are generated randomly from a lognormal distribution, $\log \mathcal{N}(0, 1)$. Note that 90 parameters are omitted during reduction, while only eight are introduced during enrichment.

Figure 4 shows trajectories for calibration scenarios from the three models: detailed, partial, and enriched. The 50% and 95% quantiles are plotted for the enriched model output. There is an obvious discrepancy between the output from the detailed and partial models, and the enriched model is able to capture the bulk of this discrepancy. Nearly all of the observations from the detailed model are contained within the model output bounds from the enriched model.

Figure 5 shows the same results, but for validation scenarios. Recall that these observations have not been used to calibrate the discrepancy operator. The output of the enriched model, at least to the eye, appears decent. The enriched model is greatly improved in comparison to the partial model alone and, similarly to the calibration scenarios, captures the bulk behavior of the detailed model in the validation scenarios.

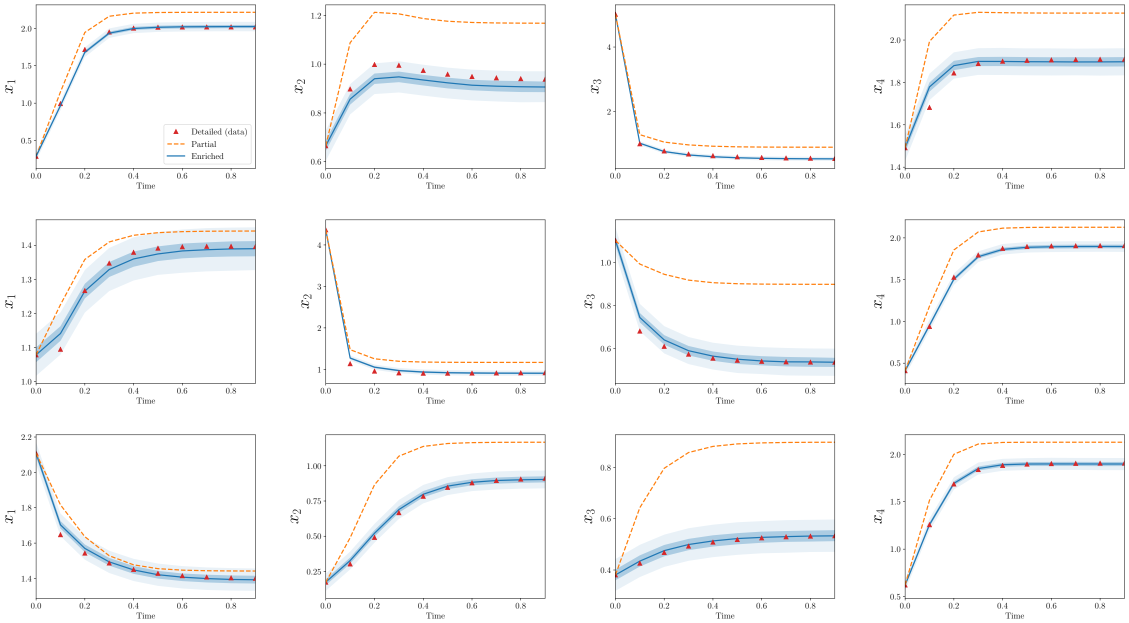


Figure 4. Partial and enriched models, compared to observations, over three calibration scenarios, $S = 10, s = 4$. Note that 90 parameters are omitted during reduction, while only eight are introduced during enrichment.

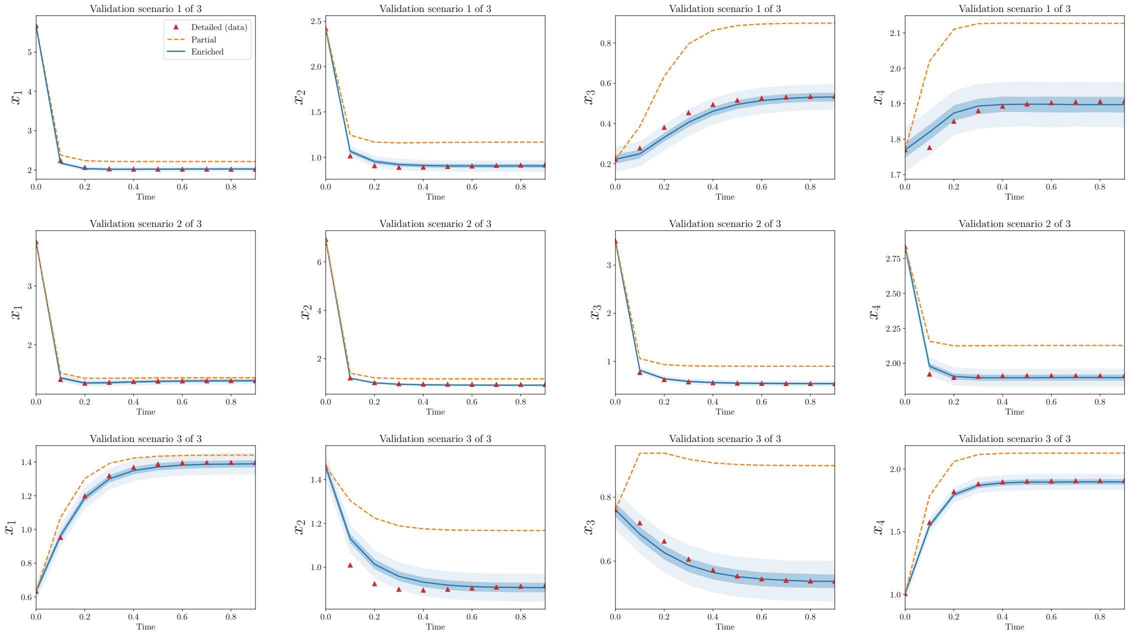


Figure 5. Partial and enriched models, compared to observations, over three validation scenarios. $S = 10, s \ = 4$.

Figures 6 and 7 show analogous plots for $S = 20$, $s = 4$. In this case, 400 parameters are omitted during reduction, while only eight are introduced during enrichment.

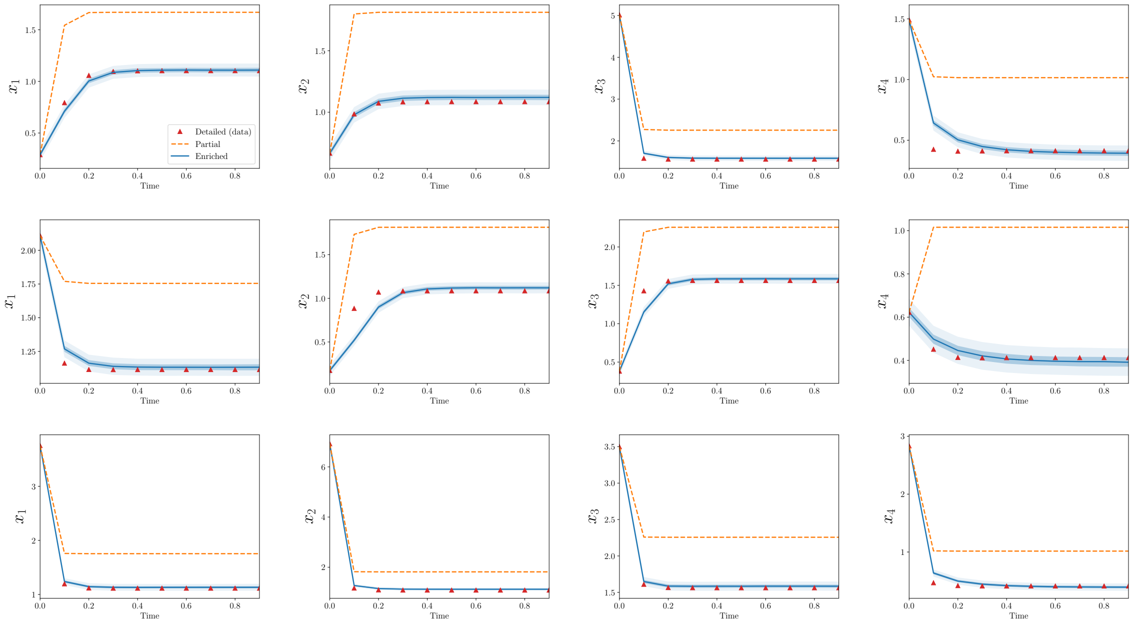


Figure 6. Partial and enriched models, compared to observations, over three calibration scenarios. $S = 20, s = 4$. Note that 400 parameters are omitted during reduction, while only eight are introduced during enrichment.

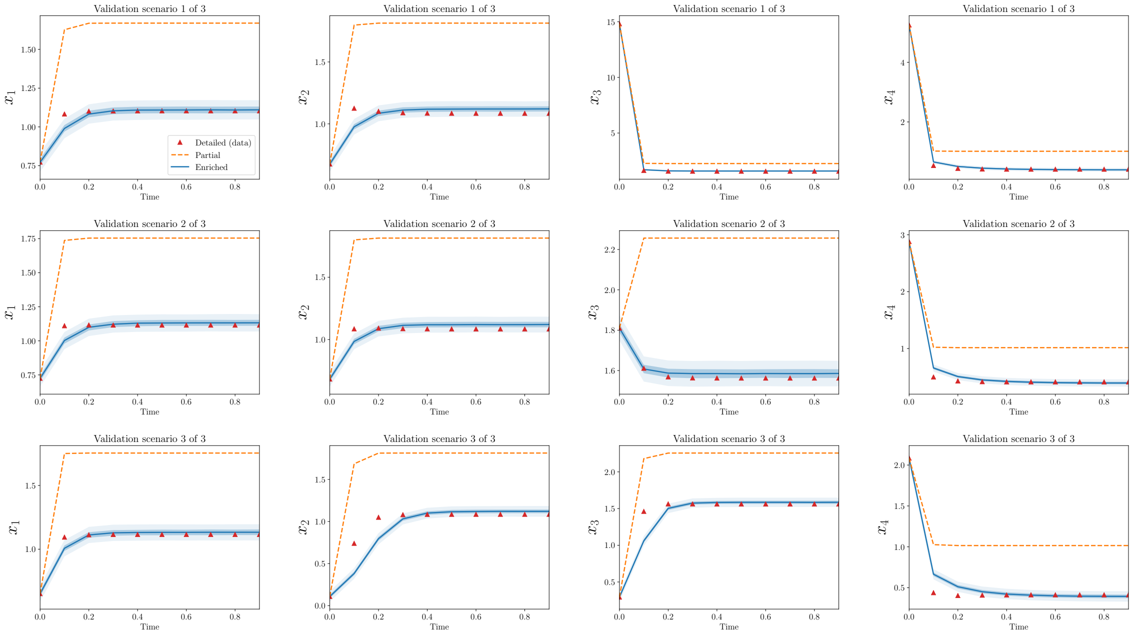


Figure 7. Partial and enriched models, compared to observations, over three validation scenarios. $S = 20, s \ = 4$.

In the above cases, there are a few observations which lie outside the predicted bounds of the enriched model. This problem must be addressed more carefully with a quantitative validation process as described in Section 4. Additionally, these results only show the performance of the discrepancy operator for a particular S and s and a single realization of (D, P). The agreement between trajectories from detailed and partial models for different choices of (D, P) are qualitatively similar, but some interesting differences appear by varying s with respect to S. In the next subsections, these statements are made more precise.

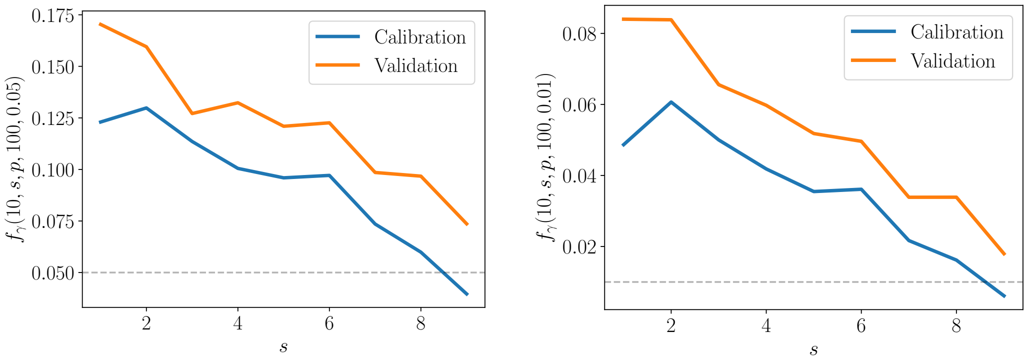
5.2. Results for Many Realizations of the Detailed Model

We examine the performance of the proposed discrepancy model in the context of random forward models. To this end, three relevant concepts are detailed below.

1. We quantify the average performance of the discrepancy models. In this sense, we compute these γ-values for trajectories from n_M realizations of detailed models, where $n_M \gg 1$.
2. Note γ-values are computed with two types of data: calibration and validation data. To refer to these two types of data, we will use the variable $p = \{c, v\}$, so that $p = c$ denotes calibration data and $p = v$ denotes validation data. We must check how well the enriched model performs both in terms of the data that has been used to calibrate it, and also in terms of data that has not. Both types are shown in Figures 8 and 9.
3. Finally, let us examine how well the discrepancy operators perform for different pairs (S, s). We fix S, s, p, n_M and then compute $\gamma-$values for all type p observations over n_M models, for a particular pair (S, s). Call this set of γ-values $\Gamma(S, s, p, n_M)$. Now let $Q(S, s, p, n_M, \tau) = \{\gamma_i : \gamma_i < \tau, \gamma_i \in \Gamma(S, s, p, n_M)\}$. Then, the fraction of γ-values below the threshold τ is:

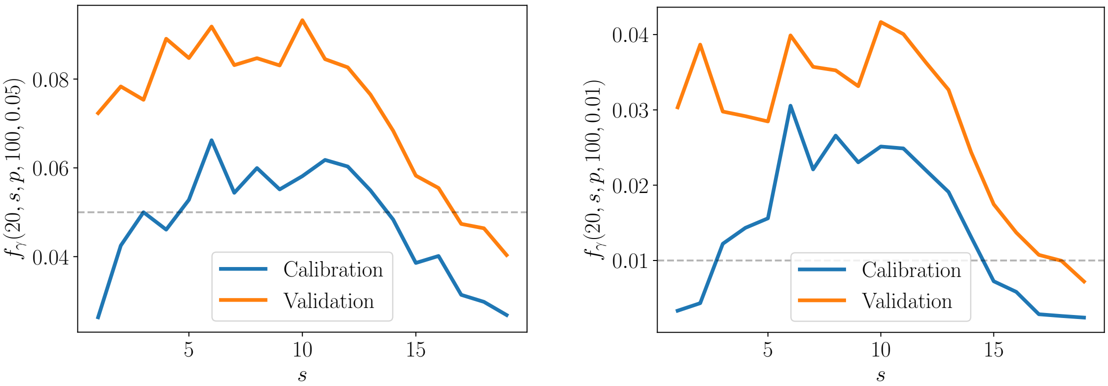
$$f_\gamma(S, s, p, n_M, \tau) = \frac{|Q|}{|\Gamma|}. \tag{35}$$

For example, if we want to compute f_γ for all calibration data over n_M model realizations, the denominator above is $|\Gamma| = sTn_{\phi_c}n_M$. The value f_γ is plotted in Figures 8 and 9, and S is fixed at 10 and 20, respectively. Along the x-axis, s ranges from 1 to $S - 1$. The results for two values of τ—0.05 (shown in Figures 8a and 9a) and 0.01 (shown in Figures 8b and 9b)—are also shown.



(**a**) $\tau = 0.05$ (**b**) $\tau = 0.01$

Figure 8. Average fraction of γ-values below given threshold. $S = 10$.



(**a**) $\tau = 0.05$ (**b**) $\tau = 0.01$

Figure 9. Average fraction of γ-values below given threshold. $S = 20$.

Let $\alpha = s/S$. In the case that the model truly does represent the data-generating process and in the limit of infinite observations, then this fraction of γ-values below the threshold is equal to the threshold itself—that is,

$$\lim_{n_M \to \infty} f_\gamma(S, s, p, n_M, \tau) = \tau, \tag{36}$$

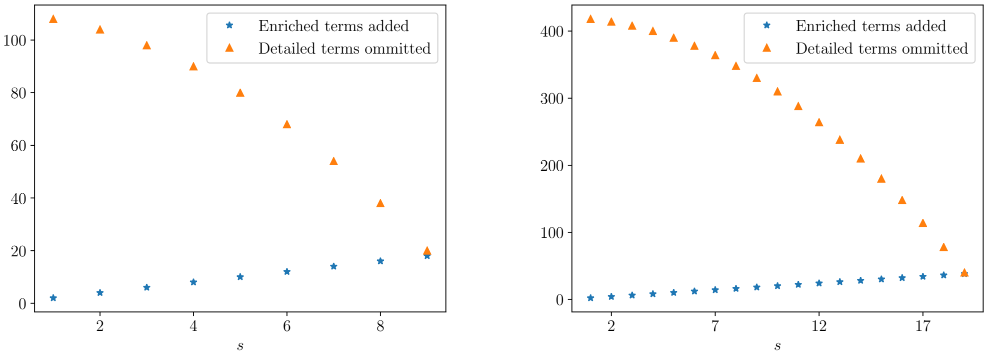
when the model is a true match to the data-generating process. Indeed, $f_\gamma(10, s, c, 100, \tau)$ approaches τ as α approaches 1 (Figure 8). This suggests that the enriched model is better able to capture the behavior of the detailed one as more species are included in the partial model, as one might expect.

Interestingly, in the $S = 20$ case, f_τ peaks somewhere in the middle of the plot, when $\alpha \approx 0.5$ (Figure 9). In other words, the enriched model is poorest for moderate α, and performs best as α approaches 1. Consider that when α is low, only a few species are included in the partial model relative to the detailed one, but also consider that the discrepancy model has only those few species to modify. When α is close to one, the partial model already includes much of the detailed model, and the discrepancy model must only fill a small gap between the two. For moderate α, however, there are neither of these advantages—the discrepancy model must account for the behavior of a large enough number of species, but the partial model is still significantly lacking compared to the detailed model.

At the same time, the $S = 10$ plots do not exhibit the above behavior. Note that the $S = 20$ cases appear to reach equilibrium more quickly that those with $S = 10$; the time to equilibrium may influence the shape of curves in Figures 8 and 9. Future work will include extensive numerical testing to better understand these results.

5.3. Relative Model Complexity

A good discrepancy model should not overfit the data, and the best discrepancy model would be rich enough to capture the relevant behavior of the detailed model without adding unnecessary complexity. Although there are different ways one might measure complexity, here, we measure the number of terms introduced in the enriched model ($2s$) compared to those omitted from the detailed model. These omitted terms include the $S^2 - s^2$ interspecific and intraspecific interaction terms, as well as the $(S - s)$ growth rate terms. (Note that the number of terms introduced is equal to the number of enriched model parameters.) For the cases $S = 10, 20$, the absolute values are shown in Figure 10.



(**a**) $S = 10$ (**b**) $S = 20$

Figure 10. Comparison of number of terms added by the enriched model and terms omitted from the detailed model.

In Figure 11, this information is presented as a ratio of terms added relative to terms omitted for various values of S. We call this ratio the relative model complexity.

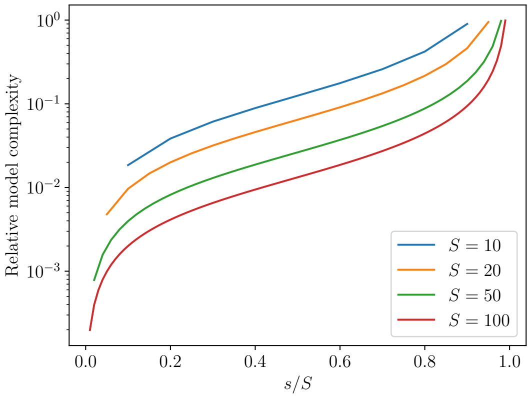


Figure 11. Enriched model complexity, measured as the ratio of enriched model terms added to detailed model terms omitted.

The relative model complexity is plotted as s/S varies from $1/S$ to $(S-1)/S$ for a few different values of S. These include the two cases presented here ($S = 10, 20$). We also show the relative model complexity for two higher values of S, namely $S = 50$ and $S = 100$. One might be interested in how this type of model complexity would scale for much larger systems. Moreover, if one knew a priori the true value of S for some system, one could balance the effectiveness of the enriched model (as measured by f_τ) against its relative model complexity.

Strikingly, the enriched models introduce many fewer terms than what the partial models omit. For example, in the two specific forward models shown in Figures 4–7, the relative model complexity is less than 0.1, yet the enriched model and observations do show surprisingly high consistency.

6. Conclusions

This study is an initial step toward representing model discrepancy in nonlinear dynamical systems of interacting species. The proposed discrepancy model here is a linear operator embedded within the differential equations. The particular form is motivated by circumstances in which a set of differential equations can be converted to a set of fewer equations; in this decoupling process, more information must be introduced about the remaining set, such as memory or higher derivatives. In this work, the discrepancy model is similarly constructed by introducing more information about the partial set, namely as a linear operator which acts on the remaining variables and (the absolute values of) their first derivatives.

We can examine the performance of the enriched models over two regimes: equilibrium and transient dynamics. The introduced parameters δ_0 act on the state variables, directly affect the equilibrium solution, and seem to be sufficient, as the enriched models typically recover equilibria of the detailed models. On the other hand, the parameters δ_1 act on the derivatives of the state, provide a type of overall scaling of the dynamics, and give an improvement but not a total correction; the enriched models recover much of the transient dynamics, but certainly not all of the discrepancy for every combination of (S, s). While the performance in the transient regime could be improved, the linear embedded discrepancy operators show promise as discrepancy models, even in scenarios that extrapolate over initial conditions. The results also bring up many new questions.

For example, what is the effective dimension of the missing dynamics of the partial model? In other words, how many (and which) new random variables need to be introduced to effectively (i.e., within some tolerance) capture the error of the partial model? The initial results here suggest that the discrepancy between the partial and detailed models can, under some conditions, be adequately described with a relatively small number of discrepancy variables and parameters. An outstanding question is whether or not some estimate of this effective discrepancy dimension can be found a priori. Certainly, such an estimate would heavily rely on given knowledge of the detailed and partial models.

Another avenue to explore is the design and analysis of more elaborate discrepancy representations in the generalized Lotka–Volterra setting, including those with second (or higher) derivatives, memory, nonlinear terms, or some combination of these. Of course, a trade-off exists between the richness of the discrepancy representation and the computational expense of both the forward and inverse problems.

Finally, the detailed models (and thus also partial models) investigated here are quite simple; the interaction matrices are negative definite, diagonally dominant, and symmetric, with off-diagonal entries sampled from identical distributions. An immediate next step in this research is to examine the performance of linear embedded discrepancy operators after relaxing these restrictions on the random interaction matrices.

Funding: This research received no external funding.

Acknowledgments: I would like to acknowledge Youssef Marzouk, Prakash Mohan, Bob Moser, and Todd Oliver for many helpful discussions about this work.

Conflicts of Interest: The author declares no conflict of interest.

Appendix A. Model Conversion

A system of S coupled ordinary differential equations can sometimes be converted (decoupled) to a system of s differential equations, where $s < S$, without loss of information. Possible structures of the resultant set, comprising s equations, motivates the functional form of the proposed model discrepancy here. This appendix briefly reviews two methods of model conversion, and what the application of each method yields in the GLV context. For more information about these types of model conversion, or exact model reduction, see [33,42].

Appendix A.1. Algebraic Method

In [43], Harrington and van Gorder present a method to algebraically convert systems of coupled differential equations from one form to another. As an example from that paper, consider the Lorenz system of three ODEs:

$$\dot{x} = a(y - x) \tag{A1a}$$

$$\dot{y} = x(b - z) - y \tag{A1b}$$

$$\dot{z} = xy - cz. \tag{A1c}$$

Through algebraic substitutions, this can be converted to a single third-order nonlinear differential equation in only the variable x and its derivatives. After substituting expressions for z and y in terms of x and its derivatives, we have:

$$\left(\frac{d}{dt} + c\right)\left(b - \frac{1}{ax}(\ddot{x} + (1+a)\dot{x} + ax)\right) - x\left(\frac{\dot{x}}{a} + x\right) = 0. \tag{A2}$$

In this example, variables y and z have been exchanged for derivatives of x.

In the GLV setting, we can perform a similar exchange. For example, consider the following system for x and y:

$$\dot{x} = r_1 x + (a_{11}x + a_{12}y)x \tag{A3a}$$

$$\dot{y} = r_2 y + (a_{21}x + a_{22}y)y. \tag{A3b}$$

This is in fact equivalent to the following single differential equation for x:

$$\left(\frac{d}{dt}\right)\left(\frac{1}{a_{12}}\left(\frac{1}{x}\left(\dot{x}-r_1x\right)-a_{11}x\right)\right)= \\ \frac{r_2}{a_{12}}\left(\frac{1}{x}\left(\dot{x}-r_1x\right)-a_{11}x\right)+\left(a_{21}x+\frac{a_{22}}{a_{12}}\left(\frac{1}{x}\left(\dot{x}-r_1x\right)-a_{11}x\right)\right). \tag{A4}$$

Equation (A4) can also be written more compactly as

$$\dot{z}=r_2z+(a_{21}x+a_{22}z)z, \tag{A5}$$

where

$$z=\frac{1}{a_{12}}\left(\frac{1}{x}\left(\dot{x}-r_1x\right)-a_{11}x\right). \tag{A6}$$

While this single differential equation is quite messy, it is now written entirely in terms of x and its derivatives.

Appendix A.2. Memory Method

Similarly, the Mori–Zwanzig approach to model reduction makes an exchange, but here, variables may be exchanged for time history, or memory, of the remaining variables. Again, a simple example starts with a two-variable system:

$$\frac{dx}{dt}=f(x,y)+\alpha(x,y)\frac{dU}{dt} \tag{A7}$$

$$\frac{dy}{dt}=g(x,y)+\beta(x,y)\frac{dV}{dt}, \tag{A8}$$

where U, V are noise processes. This system of two equations can be converted to one by introducing the memory kernel K:

$$\frac{dx(t)}{dt}=\bar{f}(x(t))+\int_0^t K\left(x(t-s),s\right)ds+n\left(x(0),y(0),t\right) \tag{A9}$$

where $\bar{f}(x(t))$ represents a Markovian term that depends only on the current state of x, the integral $\int_0^t K\left(x(t-s),s\right)ds$ depends on the entire history of x between 0 and t, and the final term $n\left(x(0),y(0),t\right)$ satisfies an auxiliary equation. Further details about this process are provided in [44].

The analogue of a Mori–Zwanzig type process in the GLV setting ($S=2, s=1$) yields:

$$\dot{x}=r_1x+(a_{11}x+a_{12}\chi)x \tag{A10}$$

where

$$\chi=\int_0^t r_2z+(a_{21}x+a_{22}z)z \tag{A11}$$

and z is defined as in the previous subsection. Note that, like the algebraic reduction, we now have a single differential equation in terms of x. In this case, the variable y is exchanged for the memory of x.

References

1. Brenig, L. Reducing nonlinear dynamical systems to canonical forms. *Philos. Trans. R. Soc. Math. Phys. Eng. Sci.* **2018**, *376*, 20170384. [CrossRef] [PubMed]
2. Steinfeld, J.I.; Francisco, J.S.; Hase, W.L. *Chemical Kinetics and Dynamics*; Prentice Hall: Englewood Cliffs, NJ, USA, 1989; Volume 3.
3. Barabás, G.; Michalska-Smith, M.J.; Allesina, S. The effect of intra-and interspecific competition on coexistence in multispecies communities. *Am. Nat.* **2016**, *188*, E1–E12. [CrossRef] [PubMed]

4. Dantas, E.; Tosin, M.; Cunha, A., Jr. Calibration of a SEIR–SEI epidemic model to describe the Zika virus outbreak in Brazil. *Appl. Math. Comput.* **2018**, *338*, 249–259. [CrossRef]
5. Smith, G.P.; Golden, D.M.; Frenklach, M.; Moriarty, N.W.; Eiteneer, B.; Goldenberg, M.; Bowman, C.T.; Hanson, R.K.; Song, S.; William, C.G.J.; et al. GRI-Mech v.3.0. Available online: http://www.me.berkeley.edu/gri_mech/ (accessed on 1 September 2020).
6. Bilger, R.; Stårner, S.; Kee, R. On reduced mechanisms for methane-air combustion in nonpremixed flames. *Combust. Flame* **1990**, *80*, 135–149. [CrossRef]
7. Kourdis, P.D.; Bellan, J. High-pressure reduced-kinetics mechanism for n-hexadecane autoignition and oxidation at constant pressure. *Combust. Flame* **2015**, *162*, 571–579. [CrossRef]
8. Kourdis, P.D.; Bellan, J. Highly reduced species mechanisms for iso-cetane using the local self-similarity tabulation method. *Int. J. Chem. Kinet.* **2016**, *48*, 739–752. [CrossRef]
9. Lyra, W.; do Nascimento, J.D.; Belkhiria, J.; de Almeida, L.; Chrispim, P.P.; de Andrade, I. COVID-19 pandemics modeling with SEIR (+ CAQH), social distancing, and age stratification. The effect of vertical confinement and release in Brazil. *medRxiv* **2020**. doi:10.1371/journal.pone.0237627.
10. Childs, M.L.; Nova, N.; Colvin, J.; Mordecai, E.A. Mosquito and primate ecology predict human risk of yellow fever virus spillover in Brazil. *Philos. Trans. R. Soc. B Biol. Sci.* **2019**, *374*. [CrossRef]
11. Oberkampf, W.L.; Roy, C.J. *Verification and Validation in Scientific Computing*; Cambridge University Press: Cambridge, UK, 2010.
12. Prudhomme, S.; Oden, J.T.; Westermann, T.; Bass, J.; Botkin, M.E. Practical methods for a posteriori error estimation in engineering applications. *Int. J. Numer. Methods Eng.* **2003**, *56*, 1193–1224. doi:10.1002/nme.609. Available online: https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.609 (accessed on 1 September 2020).
13. Roache, P.J. Code Verification by the Method of Manufactured Solutions . *J. Fluids Eng.* **2001**, *124*, 4–10. [CrossRef]
14. Bruce, P.; Bruce, A. *Practical Statistics for Data Scientists: 50 Essential Concepts*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2017.
15. Oliver, T.A.; Terejanu, G.; Simmons, C.S.; Moser, R.D. Validating predictions of unobserved quantities. *Comput. Methods Appl. Mech. Eng.* **2015**, *283*, 1310–1335. [CrossRef]
16. Bayarri, M.J.; Berger, J.O.; Paulo, R.; Sacks, J.; Cafeo, J.A.; Cavendish, J.; Lin, C.H.; Tu, J. A framework for validation of computer models. *Technometrics* **2007**, *49*, 138–154. [CrossRef]
17. Farrell-Maupin, K.; Oden, J. Adaptive selection and validation of models of complex systems in the presence of uncertainty. *Res. Math. Sci.* **2017**, *4*, 14. [CrossRef]
18. Jiang, C.; Hu, Z.; Liu, Y.; Mourelatos, Z.P.; Gorsich, D.; Jayakumar, P. A sequential calibration and validation framework for model uncertainty quantification and reduction. *Comput. Methods Appl. Mech. Eng.* **2020**, *368*, 113172. [CrossRef]
19. Kennedy, M.C.; O'Hagan, A. Bayesian calibration of computer models. *J. R. Stat. Soc. Ser. B (Stat. Methodol.)* **2001**, *63*, 425–464. [CrossRef]
20. Stainforth, D.A.; Allen, M.R.; Tredger, E.R.; Smith, L.A. Confidence, uncertainty and decision-support relevance in climate predictions. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* **2007**, *365*, 2145–2161. [CrossRef]
21. Renard, B.; Kavetski, D.; Kuczera, G.; Thyer, M.; Franks, S.W. Understanding predictive uncertainty in hydrologic modeling: The challenge of identifying input and structural errors. *Water Resour. Res.* **2010**, *46*. [CrossRef]
22. Mirams, G.R.; Pathmanathan, P.; Gray, R.A.; Challenor, P.; Clayton, R.H. Uncertainty and variability in computational and mathematical models of cardiac physiology. *J. Physiol.* **2016**, *594*, 6833–6847. [CrossRef]
23. Lewis, A.; Smith, R.; Williams, B.; Figueroa, V. An information theoretic approach to use high-fidelity codes to calibrate low-fidelity codes. *J. Comput. Phys.* **2016**, *324*, 24–43. [CrossRef]
24. Sargsyan, K.; Najm, H.; Ghanem, R. On the statistical calibration of physical models. *Int. J. Chem. Kinet.* **2015**, *47*, 246–276. [CrossRef]
25. Morrison, R.E.; Oliver, T.A.; Moser, R.D. Representing model inadequacy: A stochastic operator approach. *SIAM ASA J. Uncertain. Quantif.* **2018**, *6*, 457–496. [CrossRef]
26. Portone, T.; McDougall, D.; Moser, R.D. A Stochastic Operator Approach to Model Inadequacy with Applications to Contaminant Transport. *arXiv* **2017**, arXiv:1702.07779.

27. Pan, S.; Duraisamy, K. Data-driven discovery of closure models. *SIAM J. Appl. Dyn. Syst.* **2018**, *17*, 2381–2413. [CrossRef]
28. Wang, Z.; Akhtar, I.; Borggaard, J.; Iliescu, T. Proper orthogonal decomposition closure models for turbulent flows: a numerical comparison. *Comput. Methods Appl. Mech. Eng.* **2012**, *237*, 10–26. [CrossRef]
29. Parish, E.J.; Duraisamy, K. Non-Markovian closure models for large eddy simulations using the Mori-Zwanzig formalism. *Phys. Rev. Fluids* **2017**, *2*, 014604. [CrossRef]
30. Lu, F.; Lin, K.K.; Chorin, A.J. Data-based stochastic model reduction for the Kuramoto–Sivashinsky equation. *Phys. D Nonlinear Phenom.* **2017**, *340*, 46–57. [CrossRef]
31. Grilli, J.; Adorisio, M.; Suweis, S.; Barabás, G.; Banavar, J.R.; Allesina, S.; Maritan, A. Feasibility and coexistence of large ecological communities. *Nat. Commun.* **2017**, *8*, 8. [CrossRef]
32. Hernández-García, E.; López, C.; Pigolotti, S.; Andersen, K.H. Species competition: coexistence, exclusion and clustering. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* **2009**, *367*, 3183–3195. [CrossRef]
33. Morrison, R.E. Exact model reduction of the generalized Lotka-Volterra equations. *arXiv* **2019**, arXiv:1909:13837.
34. Takens, F. Detecting strange attractors in turbulence. In *Dynamical Systems and Turbulence, Warwick 1980*; Springer: Berlin/Heidelberg, Germany, 1981; pp. 366–381.
35. Oakley, C.O. Differential equations containing absolute values of derivatives. *Am. J. Math.* **1930**, *52*, 659–672. [CrossRef]
36. Khan, K.A.; Barton, P.I. Switching behavior of solutions of ordinary differential equations with abs-factorable right-hand sides. *Syst. Control Lett.* **2015**, *84*, 27–34. [CrossRef]
37. Barton, P.I.; Khan, K.A.; Stechlinski, P.; Watson, H.A. Computationally relevant generalized derivatives: Theory, evaluation and applications. *Optim. Methods Softw.* **2018**, *33*, 1030–1072. [CrossRef]
38. Haario, H.; Laine, M.; Mira, A.; Saksman, E. DRAM: Efficient adaptive MCMC. *Stat. Comput.* **2006**, *16*, 339–354. [CrossRef]
39. Prudencio, E.E.; Schulz, K.W. The parallel C++ statistical library 'QUESO': Quantification of Uncertainty for Estimation, Simulation and Optimization. In *Euro-Par 2011: Parallel Processing Workshops*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 398–407.
40. Hammersley, J. *Monte Carlo Methods*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2013.
41. Morrison, R.E. rebeccaem/enriched-glv: Initial Release, 2020. Available online: https://zenodo.org/record/3986201 (accessed on 15 September 2020). [CrossRef]
42. Hernández-Bermejo, B.; Fairén, V. Algebraic decoupling of variables for systems of ODEs of quasipolynomial form. *arXiv* **2019**, arXiv:1909:13837.
43. Harrington, H.A.; Van Gorder, R.A. Reduction of dimension for nonlinear dynamical systems. *Nonlinear Dyn.* **2017**, *88*, 715–734. [CrossRef] [PubMed]
44. Givon, D.; Kupferman, R.; Stuart, A. Extracting macroscopic dynamics: model problems and algorithms. *Nonlinearity* **2004**, *17*, R55–R127. [CrossRef]

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.