

AutoRepar: A method to obtain identifiable and observable reparameterizations of dynamic models with mechanistic insights

Gemma Massonis¹ | Julio R. Banga¹ | Alejandro F. Villaverde² 

¹BioProcess Engineering Group, IIM-CSIC, Vigo, Galicia, Spain

²Department of Systems Engineering and Control, Universidade de Vigo, Vigo, Galicia, Spain

Correspondence

Alejandro F. Villaverde, Department of Systems Engineering and Control, Universidade de Vigo, Vigo, Galicia, Spain.
Email: afvillaverde@uvigo.gal

Funding information

Consejo Superior de Investigaciones Científicas, Grant/Award Number: PIE 202070E062; Consellería de Cultura, Educación e Ordenación Universitaria, Xunta de Galicia, Grant/Award Number: ED431F 2021/003; MCIN/AEI, Grant/Award Numbers: PID2020-113992RA-I00, PID2020-117271RB-C22; MCIN/AEI and ERDF A way of making Europe, Grant/Award Number: DPI2017-82896-C2-2-R; MCIN/AEI and ESF Investing in your future, Grant/Award Number: RYC-2019-027537-I

Abstract

Mechanistic dynamic models of biological systems allow for a quantitative and systematic interpretation of data and the generation of testable hypotheses. However, these models are often over-parameterized, leading to nonidentifiability and nonobservability, that is, the impossibility of inferring their parameters and state variables. The lack of structural identifiability and observability (SIO) compromises a model's ability to make predictions and provide insight. Here we present a methodology, AutoRepar, that corrects SIO deficiencies of nonlinear ODE models automatically, yielding reparameterized models that are structurally identifiable and observable. The reparameterization preserves the mechanistic meaning of selected variables, and has the exact same dynamics and input-output mapping as the original model. We implement AutoRepar as an extension of the STRIKE-GOLDD software toolbox for SIO analysis, applying it to several models from the literature to demonstrate its ability to repair their structural deficiencies. AutoRepar increases the applicability of mechanistic models, enabling them to provide reliable information about their parameters and dynamics.

KEYWORDS

computational methods, nonlinear observability, structural identifiability, systems biology

1 | INTRODUCTION

Dynamic mathematical models are used for understanding, analyzing, and making quantitative predictions about the behavior of a system over time. They are routinely used in the physical sciences and engineering, and they have also become a fundamental tool in the life sciences,^{1,2} with applications ranging from pharmacology³ to medicine⁴ and biotechnology.^{5,6} The dynamics of many biological systems or processes of interest can often be accurately captured by ordinary differential equations (ODEs) that are generally nonlinear. In particular, kinetic models are based on first-principles and provide a mechanistic description of the underlying physicochemical processes. Due to their

Abbreviations: FISPO, full input-state-parameter observability; ODE, ordinary differential equations; SGI, structurally globally identifiable; SIO, structural identifiability and observability; SLI, structurally locally identifiable; SU, structurally unidentifiable.

This is an open access article under the terms of the Creative Commons Attribution-NonCommercial-NoDerivs License, which permits use and distribution in any medium, provided the original work is properly cited, the use is non-commercial and no modifications or adaptations are made.

© 2021 The Authors. *International Journal of Robust and Nonlinear Control* published by John Wiley & Sons Ltd.

mechanistic nature, they have a significant predictive power,^{7–9} and provide complementary strengths to machine learning approaches.¹⁰ However, they usually contain unknown parameters, which must be estimated by optimizing the model fit to experimental measurements of the model output.¹¹

This task, known as model calibration,^{12–14} can only be performed successfully if the optimization problem has a unique solution (or at most, a finite number of them, as will be discussed in the next section), that is, if the model is identifiable. Unidentifiable models can lead to false interpretations and misguided interventions.^{15–19} The impossibility of fulfilling the identifiability condition may be due to an inadequate structure of the model equations (structural unidentifiability) or to insufficiently informative data (practical unidentifiability).²⁰ The first type of cause should be analyzed theoretically before attempting to calibrate the model in order to characterize the sources of unidentifiability correctly.^{1,11} The second type is dealt with using statistical methods, and it can only be assessed numerically from experimental data.²¹ It is important to note that, although checking practical identifiability can be easier than structural identifiability, the latter cannot be deduced from the former.¹⁶

Many methods have been proposed to study structural identifiability, starting with the Laplace transformation²² and including power series,²³ direct tests,^{24,25} generating series,²⁶ differential algebra,^{27,28} observability-based rank tests,^{29–31} and similarity transformations.^{32,33} The analysis of this property can be very complex and in most cases must be carried out computationally. To this end, a number of software tools that implement the above mentioned algorithms have been developed, including ObservabilityTest,²⁹ DAISY,²⁷ GenSSI,^{34,35} EAR,³⁰ COMBOS,³⁶ STRIKE-GOLDD,³¹ and SIAN.³⁷

Structural identifiability (the theoretical possibility of estimating the values of the parameters from the model output) is closely related to another property, observability, which refers to the possibility of inferring the state of the system from the given observations.³⁸ If a model is not observable, it cannot be used to predict the time-course of its nonobservable states. We discuss the implications of this scenario in Figure 1, where we illustrate the issues with an example from the literature.

The essential point here is that lack of structural identifiability or observability leads to wrong insights and predictions, drastically compromising the usefulness and reliability of the model. In this scenario, the model user has several options, which can be classified as follows:

1. Continue using the model regardless of its lack of identifiability. This is the easiest yet the least recommendable option, since it may yield modeling artifacts, as shown in Figure 1D.
2. Modify the model output, that is, measure more of their states or functions. An adequate choice of the set of outputs of a model may improve its identifiability.⁴¹ However, it is not always possible or practical to modify the experimental setup, due to the associated effort and costs or the impossibility of measuring all the necessary outputs.
3. Reduce the dimensions of the model by removing some states or other expressions from the equations. The most common techniques are lumping, time scale separation, sensitivity-based analysis, and singular value decomposition; a recent overview is provided in Reference 42. These methods are generally not designed to achieve identifiability or observability. Furthermore, most of them are based on previous knowledge of the dynamics, or on the assumption that the evolution of the system can be characterized in different time scales. In addition, they usually yield approximations of the system behavior that are not exact.
4. Set certain parameters to a known value found in the literature. This practice makes it possible to reduce the number of parameters to be estimated, and can in practice seemingly improve the identifiability of the model. However, it must be done with caution, since there is no guarantee that the values to which the parameters are fixed are adequate (bear in mind that if we were reasonably confident about the value of a parameter, we would probably consider it a known constant instead of attempting to estimate it from data).
5. Reduce the dimension of the model, reparameterizing it to remove the redundant parameters. By eliminating the redundancies, the symmetries in the equations disappear, and the model no longer has the flexibility to compensate for deviations in one parameter with changes in another variable. That is, we obtain a reformulated model that is structurally identifiable and observable.

The ideal solution is to reformulate the model, modifying its dynamic equations in order to remove redundant parameters (i.e., option 5 in the above list). In this way, we hope to obtain a reparameterized model that is both structurally identifiable and observable. Unfortunately, achieving this goal is far from trivial, for two reasons:

1. Even if we determine which parameter combinations are identifiable, this knowledge does not directly lead to an automatic reformulation of the model. For example, in the example in Figure 1, $p \times s_i$ is an identifiable combination, but this product does not appear in the model and it is therefore not straightforward to replace it.

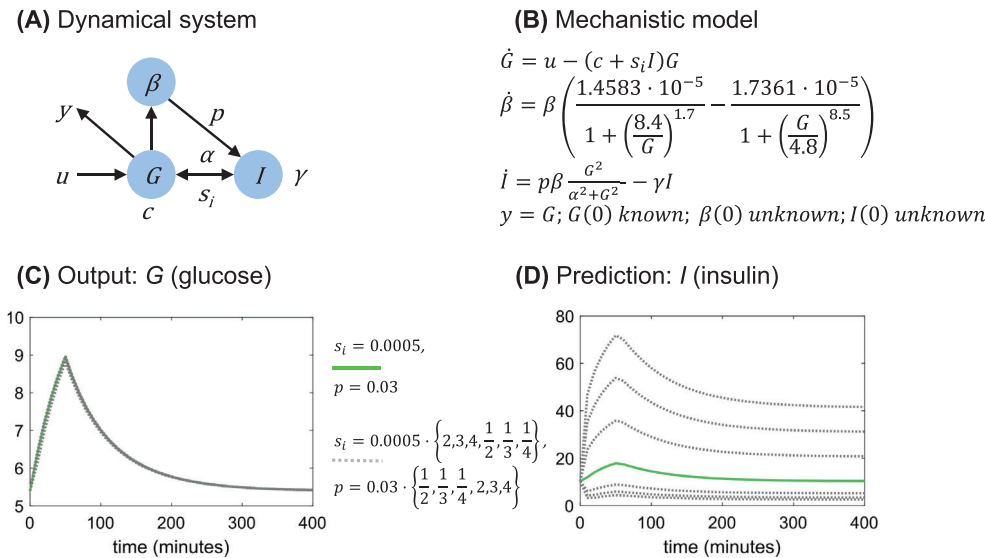


FIGURE 1 Relevance of structural identifiability and observability. Illustration of how a lack of structural identifiability or observability can lead to wrong insights and predictions. The βIG model shown in diagram (A) represents the glucose-insulin regulation circuit, which maintains plasma glucose concentration within admissible levels.³⁹ The glucose uptake (u) increases glucose concentration (G), which in turn affects the beta cells' mass (β). Beta cells secrete insulin (I) at a rate p , and insulin decreases glucose concentration at a rate s_i . This dynamic system can be modeled with the set of ordinary differential equations shown in panel (B), which contain five unknown parameters ($p, s_i, c, \alpha, \gamma$). The purpose of this model could be for example, to estimate insulin concentration in a diabetes management system using a continuous glucose monitor, possibly in a closed-loop control setting.⁴⁰ However, if we only measure the glucose concentration over time (i.e., glucose is the model output, $y = G$), two of these parameters (p, s_i) are structurally unidentifiable. In other words, there is an infinite number of pairs of (p, s_i) values that generate the same time-course of glucose. Plot (C) shows that the model simulations of G for several such pairs are identical. Hence, it is not possible to infer the correct values ($s_i = 0.0005, p = 0.03$) from glucose measurements. Furthermore, this lack of identifiability is linked to a lack of observability of the unmeasured state variables (I, β). As a consequence, predictions of the time-courses of (I, β) made by a model calibrated from experimental data are sure to be wrong. Plot (D) shows the time-courses of the insulin concentration corresponding to the (p, s_i) pairs in panel (C), which can differ considerably from the true one (shown in green) even for reasonable values of (p, s_i). The cause of this ambiguity is structural: it is rooted in the existence of symmetries in the model equations. These symmetries are due to over-parameterization: the model contains more parameters that can be estimated from the available output measurements, and this excessive flexibility hampers its predictive capabilities. This model is analyzed and reparameterized in Section 3.3

2. Assuming that we manage to reformulate the model so that it becomes identifiable and observable, the transformed variables may lose their mechanistic meaning—in which case the model cannot provide any insight into them. For the example in Box 1, if the reparameterized model combines insulin (I) and p in a new variable $\tilde{I} = I/p$, the new variable may be observable but the model cannot be used to directly monitor insulin concentration.

To the best of our knowledge, there is currently no method that addresses the two aforementioned challenges satisfactorily. This shortcoming is a major obstacle for the exploitation of mechanistic dynamic models. Partial results have been obtained with a number of approaches, including the search for scaling symmetries,⁴³ Gröbner bases,³⁶ Taylor Series,⁴⁴ similarity transformations,⁴⁵ and Lie Point Symmetries.^{46,47} Although some of the resulting methods have computational implementations, none of them can produce identifiable reparameterizations automatically. The methods based on Lie Point Symmetries are the ones that have come closest to achieving this. Software tools that can be applied to dynamic systems include a Maple implementation²⁹ that uses a probabilistic seminumerical algorithm with the computation of Hermite–Padé series, a Python code⁴⁶ that implements a deterministic method limiting the type of transformations obtained from the variables, and a MATLAB tool⁴⁷ that extends the previous method to any type of symmetry and computes the transformations using Lie series. The final three of the above tools can find transformations that break the symmetries in the model equations. However, they introduce an additional parameter that leads to the model variables losing their mechanistic meaning.

Here we present a novel method and an associated software tool (AutoRepar) that together achieve the two aforementioned goals: obtaining an identifiable and observable model reparameterization in a fully automatic way, while

preserving the mechanistic meaning of the variables of interest. More specifically, we consider structural *local* versions of identifiability and observability. That is, we want to be able to distinguish the true values of states and parameters from other values in their neighborhood. This is often sufficient for practical purposes. The methodology begins by analyzing a model given by a set of deterministic nonlinear ODEs and an input/output mapping, using the algorithm in Reference 48 to characterize the structural identifiability of each of its parameters and the observability of each of its state variables. Then, as an intermediate step, it applies the procedure presented in Reference 47 to search for the Lie symmetries that cause the lack of identifiability and/or observability. This search goes beyond scaling symmetries, making it possible to find all types of symmetries and to take into account initial conditions and unknown inputs. Our method then builds on this knowledge to compute all possible reparameterizations. Since typically several transformations are possible, the user can select the one that best suits the needs of the application, that is, choosing which parameter(s) can be removed from the model and which ones should be kept. Sometimes it is necessary to apply more than one transformation, in which case the user is consulted at each step. The final result is a reparameterized model that is structurally locally identifiable and observable. It is provided as a set of equations along with the corresponding changes of variables.

This method yields an exact—not approximate—model reformulation without requiring any assumptions about the system behavior. AutoRepar is implemented in MATLAB and integrated in a new version of the widely used STRIKE-GOLDD toolbox.⁴⁸ The method and its computational implementation are the first and, to date, only ones that meet the desired specifications. In the next section, we provide the methodological details. Then we illustrate the capabilities of the method by applying it to nonlinear models of different biological disciplines (physiology, cell signaling, pharmacokinetics). We conclude the paper by discussing the main implications of the results.

2 | METHODOLOGY

2.1 | Notation and definitions

We consider models of deterministic ordinary differential equations (ODEs) of the form:

$$\mathcal{M} = \begin{cases} \dot{x}(t) = f(x(t), \theta, u(t), w(t)), & (1) \\ y(t) = g(x(t), \theta, u(t), w(t)), & (2) \\ x^0 = x(t_0, \theta), & (3) \end{cases}$$

where f and g are rational functions of the states $x(t) \in \mathbb{R}^{n_x}$, known inputs $u(t) \in \mathbb{R}^{n_u}$, unknown constant parameters $\theta \in \mathbb{R}^{n_\theta}$, and unknown inputs $w(t) \in \mathbb{R}^{n_w}$. The output $y(t) \in \mathbb{R}^{n_y}$ represents the measurable functions of model variables. Both inputs, $u(t)$ and $w(t)$, are in general assumed to be infinitely differentiable (“smooth”) functions. It would also be possible to consider piecewise infinitely differentiable inputs by adding to the model one replicate of $x(t)$, $y(t)$, $u(t)$, and $w(t)$ for each interval in the piecewise function, although this would entail an approximation (for details, see Reference 48). Impulsive inputs are not considered in this approach.

Definition 1 (Structural local identifiability). A parameter θ_i of model \mathcal{M} is *structurally locally identifiable* (SLI) if for almost any parameter vector $\theta^* \in \mathbb{R}^{n_\theta}$ there is a neighborhood $\mathcal{N}(\theta^*)$ in which the following relationship holds:¹

$$\hat{\theta} \in \mathcal{N}(\theta^*) \text{ and } y(t, \hat{\theta}) = y(t, \theta^*) \Rightarrow \hat{\theta}_i = \theta_i^*. \quad (4)$$

If the above implication is not fulfilled, θ_i is *structurally unidentifiable* (SU). If all model parameters are SLI the model is SLI. If there is at least one SU parameter, the model is SU.

Similarly, a state $x_i(\tau)$ is *observable* if it can be distinguished from any other states in a neighborhood from observations of the model output $y(t)$ and input $u(t)$ in the interval $t_0 \leq \tau \leq t_f$, for a finite t_f . Otherwise, $x_i(\tau)$ is *unobservable*. A model is called observable if all its states are observable. Finally, \mathcal{M} is *input observable* if it is possible to infer its unknown input $w(t)$, which is said to be observable or reconstructible.

It should be noted that the properties analyzed by our method are *local*, that is, they describe the possibility of distinguishing the true value of a variable—be it a parameter or state—from other values in its neighborhood, but not necessarily from distant values. In contrast, the *global* versions of these properties apply in the whole variable space:

Definition 2 (Structural global identifiability). A parameter θ_i of model \mathcal{M} is *structurally globally identifiable* (SGI) if the relationship given by (4) holds for every possible neighborhood $\mathcal{N}(\theta^*)$.

Thus, while a globally identifiable parameter is uniquely determined, for a locally identifiable parameter there may exist a finite number of indistinguishable solutions. However, in many practical applications it is sufficient to guarantee local identifiability and observability.

Identifiability and input observability can be considered particular instances of a general observability property. To reflect this view, the acronym FISPO, which stands for full input-state-parameter observability (or Observable when used as an adjective) has been introduced.⁴⁸ Its definition is as follows.

Definition 3 (Full Input-State-Parameter Observability, FISPO). Consider a model \mathcal{M} given by equations (1), (2), (3). Let $z(t) = (x(t), \theta, w(t))$ be the vector of its states, parameters, and unknown inputs. \mathcal{M} has the FISPO property if, for every t_0 in the time interval $[t_0, t_f] \subset \cdot$, every model unknown $z_i(\tau)$ can be inferred from $y(t)$ and $u(t)$. Thus, \mathcal{M} is FISPO if, for every $z(t_0)$, for almost any vector $z^*(t_0)$, there is a neighborhood $\mathcal{N}(z^*(t_0))$ such that, for all $\hat{z}(t_0) \in \mathcal{N}(z^*(t_0))$, the following condition holds:

$$y(t, \hat{z}(t_0)) = y(t, z^*(t_0)) \Rightarrow \hat{z}_i(t_0) = z_i^*(t_0), \quad 1 \leq i \leq n_x + n_\theta + n_w.$$

2.2 | Assessing the property FISPO with a differential geometry approach

Structural identifiability can be studied in conjunction with observability by considering the unknown parameters θ as state variables with zero dynamics, which leads to an augmented state vector $\tilde{x} = (x^T, \theta^T)^T$.⁴⁹ The reconstructibility of unknown inputs $w(t)$ can also be seen as a particular case of observability, although in this case their derivatives may be nonzero. To this end we augment the state vector further with w as additional states, as well as their derivatives up to some non-negative integer l :

$$\tilde{x}(t) = \left(x(t)^T \quad \theta^T \quad w(t)^T \quad \dots \quad w(t)^{(l)T} \right)^T, \tag{5}$$

The l -augmented dynamics are given by:

$$\dot{\tilde{x}}(t) = f^l(\tilde{x}(t), u(t)) = \left(f(x(t), \theta, u(t), w(t))^T \quad 0_{1 \times n_\theta} \quad \dot{w}(t)^T \quad \dots \quad w^{(l+1)}(t)^T \right)^T,$$

leading to the l -augmented system:

$$\mathcal{M}^l = \begin{cases} \dot{\tilde{x}}(t) = f^l(\tilde{x}(t), u(t)), \\ y(t) = g(\tilde{x}(t), u(t)). \end{cases} \tag{6}$$

From now on we omit the dependency of the time-varying variables on time, to simplify the notation. The property FISPO is assessed by calculating the rank of a generalized observability matrix constructed with “extended” Lie derivatives, which are given by:⁴⁸

$$L_f^g(\tilde{x}, u) = \frac{\partial g}{\partial \tilde{x}}(\tilde{x}, u) \tilde{f}(\tilde{x}, u) + \frac{\partial g}{\partial u}(\tilde{x}, u) \dot{u}.$$

The zero-order Lie derivative is $L_f^0 g = g$, and the i -order extended Lie derivatives can be recursively calculated as:

$$L_f^i g(\tilde{x}, u) = \frac{\partial L_f^{i-1} g}{\partial \tilde{x}}(\tilde{x}, u) \tilde{f}(\tilde{x}, u) + \sum_{j=0}^{i-1} \frac{\partial L_f^{i-1} g}{\partial u^j}(\tilde{x}, u) u^{j+1}, \quad i \geq 1.$$

The observability-identifiability matrix of \mathcal{M} (6) is:

$$\mathcal{O}_I(\tilde{x}, u) = \frac{\partial}{\partial \tilde{x}} \left(L_f^0 g(\tilde{x}, u)^T \quad L_f g(\tilde{x}, u)^T \quad L_f^2 g(\tilde{x}, u)^T \quad \dots \quad L_f^{n_x-1} g(\tilde{x}, u)^T \right)^T, \tag{8}$$

We can now state the following condition:

Theorem 1 (Observability-Identifiability Condition (OIC)⁴⁸). *A model \mathcal{M} defined by (6) and (7) is FISPO around a (possibly generic) point in the augmented state space \tilde{x}_0 if the rank of its observability-identifiability matrix (8) is: $\text{rank}(\mathcal{O}_I(\tilde{x}_0, u)) = n_{\tilde{x}} = n_x + n_\theta + n_w$.*

If the observability-identifiability matrix is rank-deficient the model is over-parameterized, that is, there are parameters that must be eliminated to achieve the property FISPO. The following theorem gives the number of parameters that must be eliminated from the model:

Theorem 2 (Existence of reparameterization⁵⁰). *Let $Y(\theta_1, \dots, \theta_p, t)$ be a function whose Taylor series expansion gives rise to the coefficients $g_1(\theta), \dots, g_p(\theta)$, where $g_i : A \subset \mathcal{R}^p \Rightarrow \mathcal{R}$ for $i = 1, \dots, p$. Define $G = (h_1, \dots, h_p)$. If the Jacobian Matrix $DG(\theta)$ with respect to θ has rank $q (< p)$ for all θ in a neighborhood of $\theta^0 \in A$, then the function Y may be locally reparameterized in terms of a set of q of the Taylor series coefficients,*

$$\{\phi_1 = g_{j_1}, \dots, \phi_q = g_{j_q}\}, \quad (9)$$

that is,

$$G(\theta_1, \dots, \theta_p) = \tilde{G}(\phi_1, \dots, \phi_q). \quad (10)$$

The reparameterized system is locally identifiable.

The identifiability of each parameter is assessed by comparing the rank of the observability-identifiability matrix before and after eliminating the column corresponding to that parameter. If the rank remains constant the parameter is unidentifiable, while if it decreases it is identifiable. The same procedure is applied to assess the observability of each state.

2.3 | Finding the roots of nonobservability: Lie symmetries

The lack of structural identifiability and/or observability means that the model output does not vary if some parameters or states are modified in a certain way. The presence of Lie symmetries in the model equations amounts to the existence of similarity transformations that allow the parameters and states to be transformed while leaving the output invariant.³³ Thus, the presence of transformations of this type in the model can be used to characterize the relationships that cause the lack of structural identifiability and/or observability. In what follows we provide a brief introduction to the study of Lie symmetries; more details can be found elsewhere, for example in Reference 51.

A one-parameter Lie group of transformations is a morphism that maps a solution of the differential equations onto itself in terms of state variables, $x^* = X(x; \varepsilon)$. Its expansion in some neighborhood of $\varepsilon = 0$ is:

$$x^* = x + \varepsilon \left(\frac{\partial X(x; \varepsilon)}{\partial \varepsilon} \Big|_{\varepsilon=0} \right) + \frac{1}{2} \varepsilon^2 \left(\frac{\partial^2 X(x; \varepsilon)}{\partial \varepsilon^2} \Big|_{\varepsilon=0} \right) + \dots = x + \varepsilon \left(\frac{\partial X(x; \varepsilon)}{\partial \varepsilon} \Big|_{\varepsilon=0} \right) + O(\varepsilon^2), \quad (11)$$

The *infinitesimal* of the Lie group of transformations (11) is:

$$\eta(x) = \frac{\partial X(x; \varepsilon)}{\partial \varepsilon} \Big|_{\varepsilon=0}, \quad (12)$$

and $x + \varepsilon \eta(x)$ is the *infinitesimal transformation*.

The *infinitesimal generator* is the differential operator:

$$X = X(x) = \eta(x) \cdot \nabla = \sum_{i=1}^n \eta_i(x) \frac{\partial}{\partial x_i}, \quad (13)$$

where ∇ defines the gradient

$$\nabla = \left(\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}, \dots, \frac{\partial}{\partial x_n} \right). \quad (14)$$

The one-parameter Lie group of transformations (11) is equivalent to:

$$\begin{aligned} x^* &= \exp[\varepsilon X]x = x + \varepsilon Xx + \frac{1}{2}\varepsilon^2 X^2x + \dots = \\ &= \left(1 + \varepsilon X + \frac{1}{2}\varepsilon^2 X^2 + \dots\right)x = \sum_{k=0}^{\infty} \frac{\varepsilon^k}{k!} X^k x, \end{aligned} \tag{15}$$

where X is given by (13) and $X^k = XX^{k-1}$, $k = 1, 2 \dots$ with $X^0x = x$.

To find the Lie symmetries of a dynamic model \mathcal{M} (6), we augment its state vector as in (5), with dimension $n_{\bar{x}} = n_{\theta} + n_x + n_w$:

$$\begin{aligned} \dot{x}_i(t) &= f_i(x(t), u(t)), \quad i = 1, \dots, n_x, \\ x_i(t) &= \theta, \quad i = m + 1, \dots, n_x + n_{\theta}, \\ x_i(t) &= w_i(t), \quad i = n_x + n_{\theta} + 1, \dots, n_{\bar{x}}. \end{aligned}$$

Then, the following criterion is applied:

Theorem 3.⁵¹ *The system of ordinary differential equations defined by (6) and (7) admits a one-parameter Lie group of transformations defined by the infinitesimal generator (13) if and only if:*

$$X' \cdot (\dot{x}_k - f_k(x)) = 0, \quad k = 1, \dots, n_x, \tag{16}$$

$$X \cdot (y_l - g_l(x)) = 0, \quad l = 1, \dots, n. \tag{17}$$

Then, an admitted Lie symmetry is a continuous group of transformations X such that the observed data is unchanged:

$$g(x^*(t), u^*(t)) = g(x(t), u(t)).$$

Thus, the output map should not be modified.

The first step to finding Lie symmetries is the creation of (12). Different expressions will be considered depending on the complexity of the combinations of states, parameters and/or unknown input functions. These expressions are *Ansatz polynomials* where the coefficients are the abovementioned combinations and its variables are unknown constants to determine. Then, the polynomials are incorporated in Theorem (3) leading to an over-determined linear system of equations with numeric entries, whose kernel is the infinitesimal generators. The last step is to compute the one-parameter Lie group of transformations with Lie series (15). For a more detailed explanation, see Reference 47.

2.4 | Obtaining identifiable and observable reparameterizations with mechanistic interpretations

To make a model observable and identifiable the number of its variables must be equal to the rank of its observability-identifiability matrix. When a transformation removes a nonidentifiable parameter, the rank of the matrix remains the same while the number of variables decreases, shortening the distance between both values. The application of the one-parameter Lie groups of transformations described in the preceding paragraph can render a model structurally identifiable and observable, that is, FISPO. However, it also modifies the parameter(s) and/or state(s) involved in the transformation(s), introducing an artificial parameter ε in their equations that makes them lose their mechanistic meaning.⁴⁷ To obtain a reformulation of the model that preserves its mechanistic character, AutoRepar expresses ε in terms of other parameters, that is, it normalizes one of the transformed parameters to the unit value and removes ε from the expression.

We illustrate how this procedure works with the example from Vajda et al..⁵² In the analysis of this model, shown in Section 3.1, we remove the parameter θ_2 by introducing it into the unknown input function w . The one-parameter transformations are:

$$w^* = x_1 x_2 \varepsilon + w, \quad \theta_2^* = -\varepsilon + \theta_2.$$

Since we want to remove θ_2 , we find an expression for ε for which the parameter $\theta_2^* = 1$. The rationale behind this assignment is that it removes the parameter from the model while keeping its equations unaltered. This expression is given by:

$$\varepsilon = \theta_2 - 1,$$

which provides the following reparameterization:

$$w^* = w + x_1 x_2 (\theta_2 - 1), \quad \theta_2^* = 1;$$

Another example used in Section 3.2, the pharmacokinetic (PK) model, requires several transformations in order for the model to become FISPO. In the first one, the one-parameter Lie groups are:

$$\begin{aligned} x_1^* &= x_1 \exp(\varepsilon), \\ u^* &= x_1(k_1 + k_2) + \exp(\varepsilon)(u - x_1(k_1 + k_2)), \\ k_1^* &= k_1 \exp(-\varepsilon), \\ k_2^* &= k_2 \exp(-\varepsilon). \end{aligned}$$

In this step it is possible to remove either k_1 or k_2 . The easiest way to remove the artificial parameter is not by isolating ε but $\exp(\varepsilon)$. For example, if we want to remove k_1 we could do:

$$k_1^* = k_1 \exp(-\varepsilon) = 1 \rightarrow \exp(\varepsilon) = k_1.$$

After reparameterizing the model in this way we perform a second symmetry search, obtaining the following one-parameter Lie-groups of transformations:

$$\begin{aligned} \tilde{x}_3 &= x_3 \exp(\varepsilon), \\ \tilde{k}_2 &= k_2^* \exp(\varepsilon), \\ \tilde{k}_3 &= k_3 \exp(\varepsilon), \\ \tilde{k}_7 &= k_3(1 - \exp(\varepsilon)) + k_7, \\ \tilde{s}_3 &= s_3 \exp(-\varepsilon), \\ \tilde{u} &= \tilde{k}_2 x_1^*(\exp(\varepsilon) - 1) + u^*. \end{aligned}$$

We remove the parameter s_3 by isolating $\exp(\varepsilon)$ from the above expression:

$$\tilde{s}_3 = s_3 \exp(-\varepsilon) = 1 \rightarrow \exp(\varepsilon) = s_3.$$

The reparameterized model is:

$$\begin{aligned} \tilde{x}_3 &= x_3 s_3, \\ \tilde{k}_2 &= k_2^* s_3, \\ \tilde{k}_3 &= k_3 s_3, \\ \tilde{k}_7 &= k_3(1 - s_3) + k_7, \\ \tilde{s}_3 &= \frac{s_3}{s_3} = 1, \\ \tilde{u} &= \tilde{k}_2 x_1^*(s_3 - 1) + u^*. \end{aligned}$$

This model has undergone two stages of reparameterizations to become FISPO. The second one involves two parameters that already appeared in the first one, k_2 and u . We can express these two variables as functions of the original ones

by recursively replacing the transformations in reverse order:

$$\begin{aligned}\tilde{k}_2 &= k_2^* s_3 = \frac{k_2}{k_1} s_3, \\ \tilde{u} &= \tilde{k}_2 x_1^* (s_3 - 1) + u^* = \frac{k_2}{k_1} s_3 x_1 k_1 (s_3 - 1) + x_1 (k_1 + k_2) + k_1 (u - x_1 (k_1 + k_2)) = \\ &= k_2 s_3 x_1 (s_3 - 1) + x_1 (k_1 + k_2) + k_1 (u - x_1 (k_1 + k_2)).\end{aligned}$$

It should be noted that reparameterization based on Lie symmetries has connections to nondimensionalization. More specifically, model reduction based on nondimensionalization amounts to a rescaling of model variables. Scaling symmetries are one of the types of symmetries encountered in dynamical models;⁵³ a further explanation of the relation between scaling symmetries and nondimensionalization can be found in Reference 54.

2.5 | Computational implementation: A new version of the STRIKE-GOLDD toolbox

We have implemented AutoRepar in a new release (v3.0) of the STRIKE-GOLDD toolbox, which is available at: <https://github.com/afvillaverde/strike-goldd/releases/tag/v3.0>. STRIKE-GOLDD is an open source Matlab toolbox for analyzing nonlinear ODE models. It determines the structural identifiability of their parameters and the observability of their states and unknown inputs, and it finds the Lie symmetries that cause the lack of these properties. The new version is also capable of providing fully observable reparameterizations. All the analyses reported in this paper have been carried out using STRIKE-GOLDD v3.0.

The combination of SIO analysis, Lie symmetries analysis, and automatic reparameterization yields an integrated computational pipeline that is depicted in Figure 2.

3 | RESULTS

In this section we present the main features of the methodology, applying it to four nonlinear models. More detailed information about the analysis of the case studies, including different possible transformations, is given in the Supplementary Information, which can also be found in the doc folder of STRIKE-GOLDD v3.0 (<https://github.com/afvillaverde/strike-goldd/releases/tag/v3.0>).

3.1 | Making a model identifiable and observable while preserving its mechanistic meaning

We begin with a deliberately simple case study to illustrate the basic functioning of our method. We revisit the first example presented in Reference 52 by Vajda et al., which is a nonlinear model with an unknown input ($w(t)$), two states $(x_1(t), x_2(t))^T$, and four unknown parameters $(\theta_1, \theta_2, \theta_3, \theta_4)$. The measured output is the first state, $y(t) = x_1(t)$, and the initial conditions are assumed to be zero, $x_1(0) = x_2(0) = 0$. The dynamics is given by the following equations:

$$\begin{aligned}\dot{x}_1(t) &= w(t) + \theta_1 x_1(t)^2 + \theta_2 x_1(t) x_2(t), \\ \dot{x}_2(t) &= \theta_3 x_1(t)^2 + \theta_4 x_2(t) x_1(t),\end{aligned}$$

We apply our methodology following the workflow shown in Figure 2.

First we analyze the SIO of the model, finding that it is nonidentifiable and nonobservable. Specifically, θ_2 and θ_3 are unidentifiable, and $x_2(t)$ is nonobservable.

We then look for the Lie symmetries that cause the lack of identifiability and observability. This analysis finds three infinitesimal generators and their corresponding symmetry-breaking transformations (see Methods for a detailed explanation). The first infinitesimal generator involves the unknown input w and the parameter θ_2 . By choosing this generator and applying the corresponding transformation we remove the unidentifiable parameter θ_2 from the model, obtaining

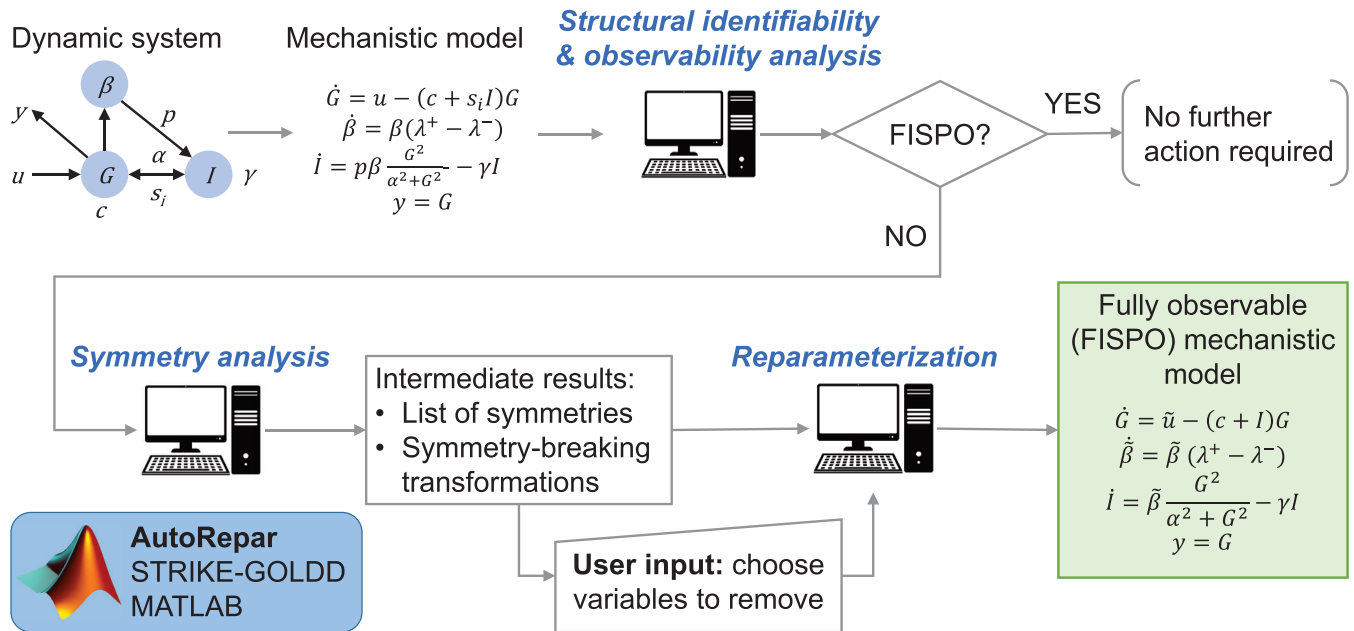


FIGURE 2 AutoRepar: Automatic reparameterization to obtain identifiable and observable models. The workflow consists of three main steps. The starting point is an ODE model of a dynamical system. The first step is to perform a *structural identifiability and observability analysis*. If the model is structurally identifiable and observable, no further action is required. (Following,⁴⁸ we will refer to the property of full input, state, and parameter observability as FISPO.) If the model is not FISPO, the next step is to perform a *symmetry analysis* to find the existing Lie symmetries and the variable transformations that can be performed to remove them. The third and final step is to perform a model *reparameterization* by applying one or more transformations. This procedure obtains a fully observable version of the original model. The procedure is implemented in MATLAB and included in the STRIKE-GOLDD toolbox. It is automated and requires minimal intervention from the user: when multiple reformulations are possible, the user can choose which parameters or state variables should be removed and which ones should be kept in the model

the following reformulated equations:

$$\begin{aligned}\dot{x}_1(t) &= w^*(t) + \theta_1 x_1(t)^2 + x_1(t)x_2(t), \\ \dot{x}_2(t) &= \theta_3 x_1(t)^2 + \theta_4 x_2(t)x_1(t),\end{aligned}$$

where the transformed variable is

$$w^*(t) = w(t) + x_1(t)x_2(t)(\theta_2 - 1)$$

The reparameterized model shown above is FISPO, that is, its parameters are structurally identifiable and its unmeasured state and input are observable. We note that all computations are performed automatically, and the user simply has to choose the preferred transformation from those proposed by the program.

Albeit simple, this case illustrates two interesting features of the method. The first is that *it is capable of finding nonelementary transformations*. The second is that it manages to *render a variable observable without transforming it*, thus making it possible to infer its value while preserving its mechanistic meaning. This is the case of the state variable $x_2(t)$ and of parameter θ_3 , which are observable/identifiable in the final model, unlike in the initial one.

Figure 3 shows structural diagrams of the original and reformulated models. These diagrams are useful for visualizing which variables are included in each model, along with the relationships among them, and which parts have been simplified.

It should also be noted that the number of transformations needed to make a model identifiable and observable is given by the difference between the dimension of the augmented state vector and the rank of the observability-identifiability matrix.^{44,50} The original formulation of this model has two states, four parameters, one unknown input, and the input derivative, making its dimension equal to eight. (As explained in the Methods section, we assume that the unknown inputs

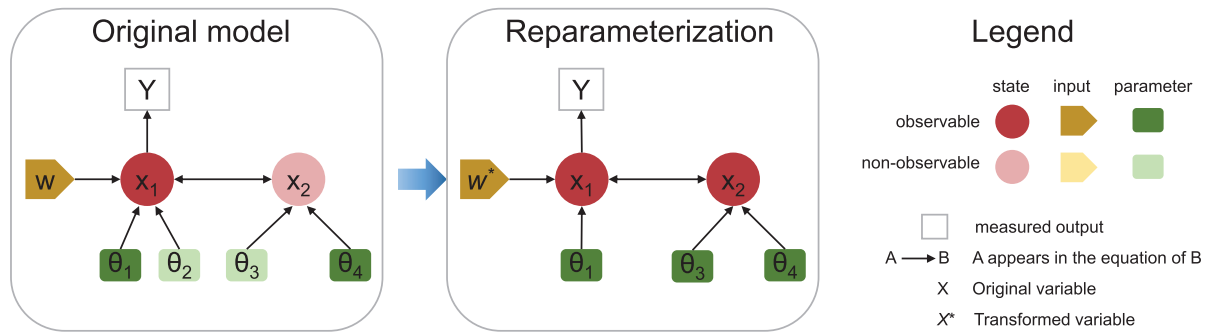


FIGURE 3 The model of Vajda et al.⁵² The image on the left shows the original model (two states, four parameters, one output, and one input function) while the one on the right shows the transformed model (two states, three parameters, one output, and one input function). States are colored in red, inputs in yellow, parameters in green, and outputs in white. The intensity of the colors (darker or lighter) of the first three variables symbolizes whether they are structurally observable or not, respectively. Self-interaction loops (e.g., x_1 to x_1) are not shown

have a finite number of nonzero derivatives. For simplicity we set this number to one; we obtained identical results when setting it to two and to three.) Since the initial SIO analysis reported that the rank of its observability-identifiability matrix is seven, in this case one transformation was sufficient. Next, we show an example that requires several transformations.

3.2 | A model may require several transformations to become identifiable and observable

The following example is a pharmacokinetic (PK) model, which describes the time-course of the concentration of a drug after it has been ingested.²⁹ It consists of four states (x_1 – x_4), nine parameters (k_1 – k_7 , s_2 , s_3), and an unknown input function (u); two scaled states are measured ($y_1 = s_2x_2$, $y_2 = s_3x_3$):

$$\begin{aligned}\dot{x}_1 &= u - (k_1 + k_2)x_1, \\ \dot{x}_2 &= k_1x_1 - (k_3 + k_6 + k_7)x_2 + k_5x_4, \\ \dot{x}_3 &= k_2x_1 + k_3x_2 - k_4x_3, \\ \dot{x}_4 &= k_6x_2 - k_5x_4, \\ y_1 &= s_2x_2, \\ y_2 &= s_3x_3.\end{aligned}$$

Note that we have omitted the dependency of x and u on time to simplify the notation. The initial conditions are considered unknown. The initial SIO study classifies only three parameters as identifiable: k_4 , k_5 , and k_6 . Since the augmented state vector has dimension 15 and the rank of the observability-identifiability matrix is 13, it is necessary to apply two transformations to make the model fully identifiable and observable.

For the first transformation the user can choose from four different infinitesimal generators (full details are provided in the Supplementary Information). All the unidentifiable parameters appear in at least one of the symmetries, so any of them can be removed. In this example we choose k_1 ; other choices would be equally possible.

After the first transformation, the second SIO analysis establishes that two parameters have become identifiable, k_3 and k_7 . The subsequent symmetry search returns three possible infinitesimal generators; again, all the remaining unidentifiable parameters are involved in at least one of them. The transformation given by the first generator is useful because it does not require transformation of all of the states and parameters. This transformation allows us to choose between k_2 and s_3 ; we select s_3 as the parameter to remove.

This procedure yields the following reformulated model:

$$\begin{aligned}\dot{x}_1^* &= \tilde{u} - \tilde{k}_2x_1^*, \\ \dot{x}_2 &= x_1^* - (\tilde{k}_3 + k_6 + \tilde{k}_7)x_2 + k_5x_4,\end{aligned}$$

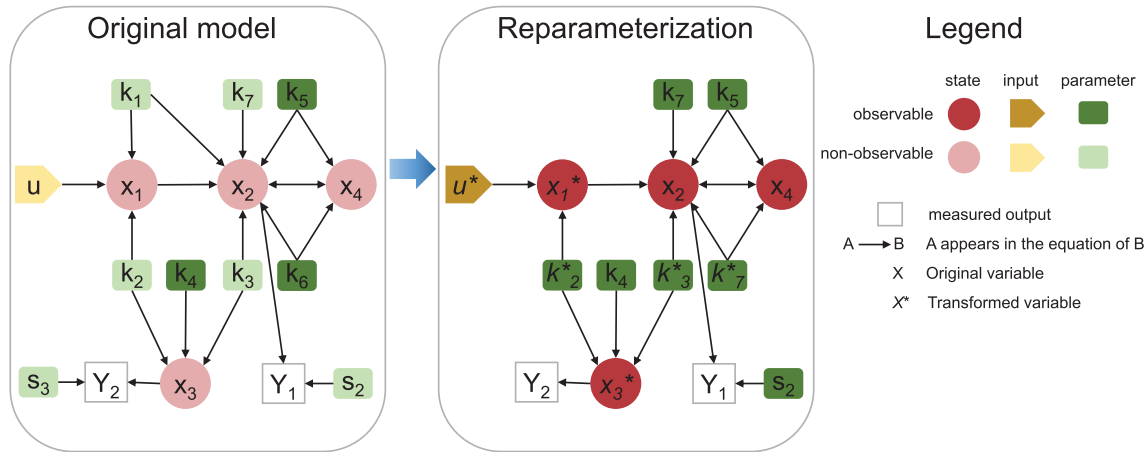


FIGURE 4 Pharmacokinetic model, PK.²⁹ The image on the left shows the original model (four states, nine parameters, two outputs, and one input function) while the one on the right shows the transformed model (four states, seven parameters, two output, and one input function). States are colored in red, inputs in yellow, parameters in green, and outputs in white. The intensity of the colors (darker or lighter) of the first three variables symbolizes whether they are structurally observable or not, respectively. Self-interaction loops (e.g., x_1 to x_1) are not shown

$$\begin{aligned}\tilde{x}_3 &= \tilde{k}_2 x_1^* + \tilde{k}_3 x_2 - k_4 \tilde{x}_3, \\ x_4 &= k_6 x_2 - k_5 x_4, \\ y_1 &= s_2 x_2, \\ y_2 &= \tilde{x}_3.\end{aligned}$$

The corresponding diagrams are shown in Figure 4.

The transformed variables are:

$$\begin{aligned}x_1^* &= x_1 k_1, \\ u^* &= x_1(k_1 + k_2) + k_1(u - x_1(k_1 + k_2)), \\ k_2^* &= \frac{k_2}{k_1}, \\ \tilde{x}_3 &= x_3 s_3, \\ \tilde{k}_2 &= k_2^* s_3 = \frac{k_2 s_3}{k_1}, \\ \tilde{k}_3 &= k_3 s_3, \\ \tilde{k}_7 &= k_3(1 - s_3) + k_7, \\ \tilde{u} &= \tilde{k}_2 x_1^*(s_3 - 1) + u^* = k_2 s_3 x_1(s_3 - 1) + x_1(k_1 + k_2) + k_1(u - x_1(k_1 + k_2)),\end{aligned}$$

where we have distinguished with x_1^* the changes made in the first transformation and with \tilde{x}_i those of the second one. Note that the parameter k_2 and the input function $u(t)$ have been transformed twice.

This reformulation eliminates two parameters, k_1 and s_3 , and combines them with k_2 , k_3 , k_7 , x_1 , and x_3 . The parameter s_2 and the states x_2 and x_4 become identifiable and observable, respectively, without undergoing any transformation. Thus, we obtain a fully identifiable and observable model without the need to transform all the states and/or parameters.

3.3 | Unknown inputs provide flexibility

Let us now examine the βIG model that was introduced as an example in Box 1. It is composed of three states (β, I, G), the latter of which is the measured output (i.e., $y = G$), five parameters ($p, s_i, c, \alpha, \gamma$), and one input, the glucose uptake $u(t)$,

which we will start by assuming to be known:

$$\begin{aligned} \dot{G}(t) &= u(t) - (c + s_i I(t))G(t), \\ \dot{\beta}(t) &= \beta(t)(\lambda^+ - \lambda^-), \\ \dot{I}(t) &= p\beta(t)\rho_G - \gamma I(t), \\ y(t) &= G(t), \\ \rho_G &= \frac{G(t)^2}{\alpha^2 + G(t)^2}, \\ \lambda^+ &= \frac{\mu^+}{1 + \left(\frac{8.4}{G(t)}\right)^{1.7}}, \\ \lambda^- &= \frac{\mu^-}{1 + \left(\frac{G(t)}{4.8}\right)^{8.5}}, \\ \mu^+ &= \frac{0.021}{24 \cdot 60}, \\ \mu^- &= \frac{0.025}{24 \cdot 60}. \end{aligned}$$

This model was presented in Reference 39 and its SIO was analyzed in Reference 17. The latter study discussed how to make it identifiable and observable by directly measuring additional outputs or parameters. Here we illustrate how to achieve the same goal without requiring additional measurements. The initial conditions are considered unknown, except for $G(0)$.

The initial SIO analysis yields that two parameters are unidentifiable, p and s_i , and two states are unobservable, $\beta(t)$ and $I(t)$. Since the rank of the observability-identifiability matrix is 6 and the total number of parameters and states is 8, two transformations are required for eliminating all redundancies. By removing s_i in the first transformation and p in the second one, we obtain the following reparameterized model:

$$\begin{aligned} \dot{G}(t) &= u(t) - (c + \tilde{I}(t))G(t), \\ \dot{\tilde{\beta}}(t) &= \tilde{\beta}(t)(\lambda^+ - \lambda^-), \\ \dot{\tilde{I}}(t) &= \tilde{\beta}(t)\rho_G - \gamma \tilde{I}(t), \\ y(t) &= G(t), \end{aligned}$$

where

$$\begin{aligned} \beta^*(t) &= \beta(t)p, \\ \tilde{\beta}(t) &= \beta(t)^* s_i = \beta(t)s_i p, \\ \tilde{I}(t) &= I(t)s_i. \end{aligned}$$

As in the previous example, we write the variables transformed in the first reparameterization with $*$ and those of the second one with \sim .

We note that in this example it has been necessary to transform all the unobservable states and unidentifiable parameters. Thus, while we obtain a fully identifiable and observable model, we cannot use it to infer the variables that were not observable in the original formulation. However, there is a way of achieving this goal: the idea is to exploit the additional flexibility provided by the assumption that the external input is unknown. Considering unknown input functions has recently been shown to be useful in the context of estimation, in order to account for structural model errors.⁵⁵ Assuming that $u(t)$ is unknown, we obtain a reparameterization that makes the insulin concentration, $I(t)$, observable without undergoing any transformation:

$$\begin{aligned} \dot{G}(t) &= u^*(t) - (c + I(t))G(t), \\ \dot{\tilde{\beta}}(t) &= \tilde{\beta}(t)(\lambda^+ - \lambda^-), \end{aligned}$$

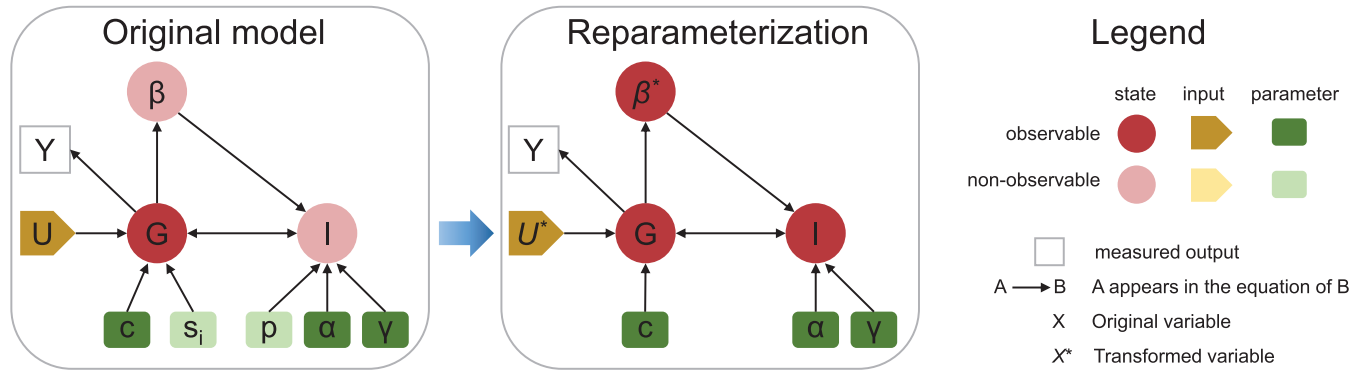


FIGURE 5 β IG model. The image on the left shows the original model (three states, five parameters, one output, and one input function) while the one on the right shows the transformed model (three states, three parameters, one output, and one input function). States are colored in red, inputs in yellow, parameters in green, and outputs in white. The intensity of the colors (darker or lighter) of the first three variables symbolizes whether they are structurally observable or not, respectively. Self-interaction loops (e.g., x_1 to x_1) are not shown

$$\begin{aligned}\dot{I}(t) &= \tilde{\beta}(t)\rho_G - \gamma I(t), \\ y(t) &= G(t).\end{aligned}$$

Since the input is unknown, we can include one of the transformed terms in it; reparameterizing the input in this way, along with the β -cell mass, also eliminates the requirement of transforming the insulin state:

$$\begin{aligned}u^*(t) &= u(t) + G(t)I(t)(-s_i + 1), \\ \tilde{\beta}(t) &= \beta(t)p.\end{aligned}$$

The original and reformulated model diagrams are shown in Figure 5.

The unknown input can be estimated from the experimental data, along with the unknown parameters. To this end, it is necessary to extend the parameter estimation procedure. A possible solution was described and demonstrated in Reference 48.

3.4 | AutoRepar can reparameterize models with multiple states and parameters

With this last example we show the applicability of the methodology to models of larger dimension. To this end we consider a model of the transcription factor NF- κ B signaling with ten states, twenty parameters, four output functions, and one input that we consider unknown.⁴⁶

$$\begin{aligned}\dot{x}_1 &= k_{11}x_{10} - \left(\frac{k_1 u}{1 + k_0 u} + k_{1p} \right) x_1, \\ \dot{x}_2 &= \left(\frac{k_1 u}{1 + k_0 u} + k_{1p} \right) x_1 - k_2 x_2, \\ \dot{x}_3 &= k_2 x_2 - k_3 x_3, \\ \dot{x}_4 &= k_2 x_2 - k_4 x_4, \\ \dot{x}_5 &= k_3 \rho_{vol} x_3 - k_5 x_5, \\ \dot{x}_6 &= k_5 x_5 - k_{10} x_9 x_6, \\ \dot{x}_7 &= k_6 x_6 - k_7 x_7, \\ \dot{x}_8 &= k_8 x_7 - k_9 x_8, \\ \dot{x}_9 &= k_9 \rho_{vol} x_8 - k_{10} x_9 x_6, \\ \dot{x}_{10} &= k_{10} x_9 x_6 - k_{11} \rho_{vol} x_{10},\end{aligned}$$

$$\begin{aligned}
 y_1 &= s_1(x_1 + x_2 + x_3) + I_{0_{\text{cvt}}}, \\
 y_2 &= s_2(x_{10} + x_5 + x_6) + I_{0_{\text{nuc}}}, \\
 y_3 &= s_3(x_2 + x_3), \\
 y_4 &= s_4(x_2 + x_4).
 \end{aligned}$$

The initial conditions are parameterized as follows:

$$\begin{aligned}
 x_1(0) &= x_1, \\
 x_2(0) &= \frac{k_{1p}x_1(0)}{k_2}, \\
 x_3(0) &= \frac{k_{1p}x_1(0)}{k_3}, \\
 x_4(0) &= \frac{k_{1p}x_1(0)}{k_4}, \\
 x_5(0) &= \frac{k_{1p}\rho_{\text{vol}}x_1(0)}{k_5}, \\
 x_6(0) &= \frac{k_{1p}x_1(0)}{k_9}, \\
 x_7(0) &= \frac{k_6k_{1p}x_1(0)}{k_7k_9}, \\
 x_8(0) &= \frac{k_{1p}x_1(0)}{k_9}, \\
 x_9(0) &= \frac{k_9\rho_{\text{vol}}}{k_{10}}, \\
 x_{10}(0) &= \frac{k_{1p}x_1}{k_{11}}.
 \end{aligned}$$

In this initial formulation only 11 parameters are structurally identifiable (k_{1p} , k_2 , k_3 , k_4 , k_5 , k_7 , k_9 , k_{11} , ρ_{vol} , $I_{0_{\text{nuc}}}$, and $I_{0_{\text{cvt}}}$). The model requires three rounds of transformations to achieve full identifiability and observability. At each step it is possible to choose from several parameters for removal; here we report one possible solution, but others could be selected.

In the first transformation the user can select from $\{k_0, k_1\}$; we choose to remove k_0 . Next, from $\{k_6, k_8\}$ we remove k_6 , and finally, from $\{k_{10}, s_1, s_2, s_3, s_4\}$ we eliminate k_{10} .

These choices lead to the following model reformulation, the diagram of which is shown in Figure 6:

$$\begin{aligned}
 \dot{\bar{x}}_1 &= k_{11}\bar{x}_{10} - \left(\frac{k_1^*u^*}{1+u^*} + k_{1p} \right) \bar{x}_1, \\
 \dot{\bar{x}}_2 &= \left(\frac{k_1^*u^*}{1+u^*} + k_{1p} \right) \bar{x}_1 - k_2\bar{x}_2, \\
 \dot{\bar{x}}_3 &= k_2\bar{x}_2 - k_3\bar{x}_3, \\
 \dot{\bar{x}}_4 &= k_2\bar{x}_2 - k_4\bar{x}_4, \\
 \dot{\bar{x}}_5 &= k_3\rho_{\text{vol}}\bar{x}_3 - k_5\bar{x}_5, \\
 \dot{\bar{x}}_6 &= k_5\bar{x}_5 - \bar{x}_9\bar{x}_6, \\
 \dot{\bar{x}}_7 &= k_6\bar{x}_6 - k_7\bar{x}_7, \\
 \dot{\bar{x}}_8 &= \tilde{k}_8\bar{x}_7 - k_9\bar{x}_8, \\
 \dot{\bar{x}}_9 &= k_9\rho_{\text{vol}}\bar{x}_8 - \bar{x}_9\bar{x}_6, \\
 \dot{\bar{x}}_{10} &= \bar{x}_9\bar{x}_6 - k_{11}\rho_{\text{vol}}\bar{x}_{10}, \\
 y_1 &= s_1(\bar{x}_1 + \bar{x}_2 + \bar{x}_3) + I_{0_{\text{cvt}}},
 \end{aligned}$$

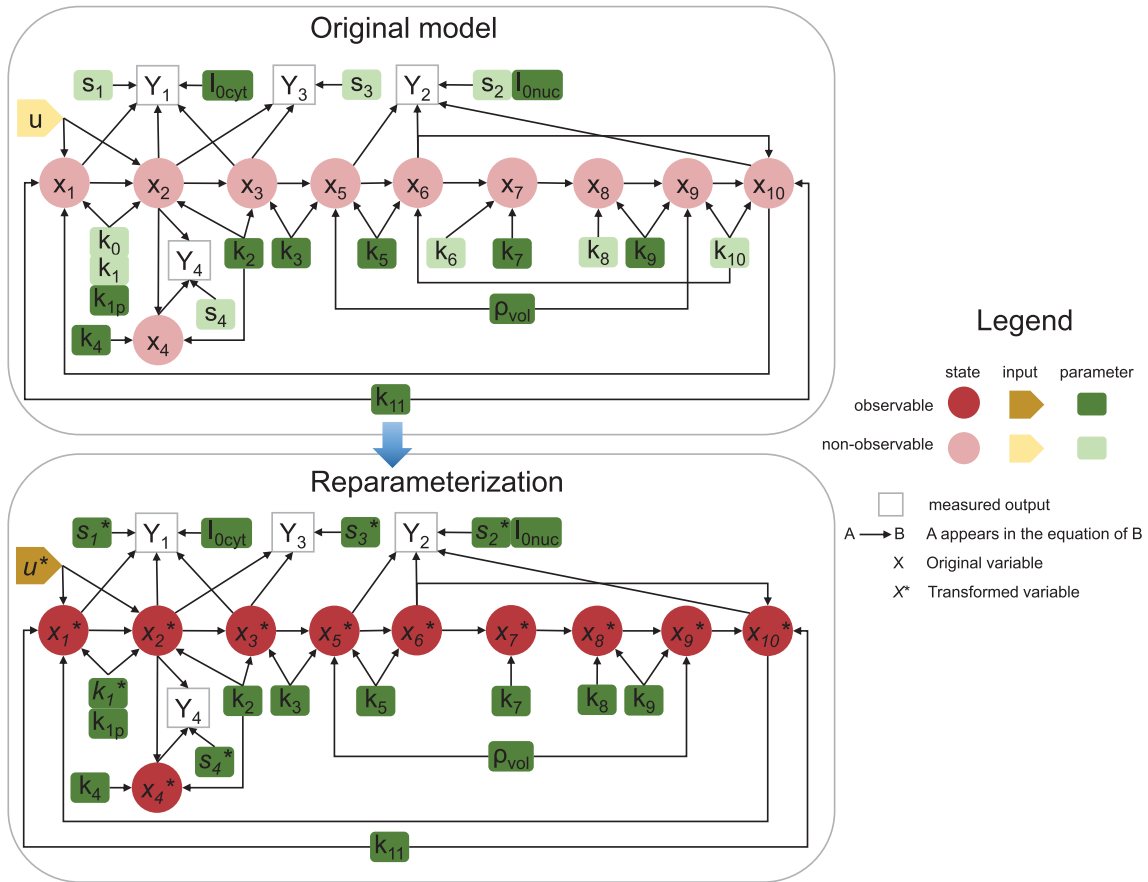


FIGURE 6 NF-κB model. To improve readability, only the relationships arising from the dynamic and output equations are shown; the parametrized initial conditions are omitted. The image on the left shows the original model (nine states, 20 parameters, four outputs, and one input function) while the one on the right shows the transformed model (20 states, 17 parameters, four output, and one input function). States are colored in red, inputs in yellow, parameters in green, and outputs in white. The intensity of the colors (darker or lighter) of the first three variables symbolizes whether they are structurally observable or not, respectively. Self-interaction loops (e.g., x_1 to x_1) are not shown

$$\begin{aligned}
 y_2 &= \bar{s}_2(\bar{x}_{10} + \bar{x}_5 + \bar{x}_6) + I_{0nuc}, \\
 y_3 &= \bar{s}_3(\bar{x}_2 + \bar{x}_3), \\
 y_4 &= \bar{s}_4(\bar{x}_2 + \bar{x}_4).
 \end{aligned}$$

where the variables transformed in the first (x^*), second (\tilde{x}), and third (\bar{x}) reparameterizations are, respectively:

$$\begin{aligned}
 u^* &= uk_0, \quad k_1^* = \frac{k_1}{k_0}; \\
 \tilde{x}_7 &= \frac{x_7}{k_6}, \quad \tilde{k}_8 = k_8 k_6; \\
 \bar{x}_i &= x_i k_{10} \text{ for } i = 1, \dots, 6, 8, 9, 10, \quad \bar{x}_7 = \tilde{x}_7 k_{10} = \frac{x_7 k_{10}}{k_6}, \quad \bar{s}_j = \frac{s_j}{k_{10}} \text{ for } j = 1, \dots, 4;
 \end{aligned}$$

4 | DISCUSSION

A lack of structural identifiability and observability (SIO) can seriously hamper the usefulness of a dynamic model, and a great deal of research over recent decades has focused on addressing this issue, leading to computational tools capable of analyzing SIO ever more efficiently. Yet the answer to a related question has remained elusive: how to fix the identifiability and observability issues detected by these tools systematically and efficiently?

With this in mind, we have presented here a methodology that automatically provides an (at least locally) identifiable and observable reparameterization of an otherwise nonidentifiable and/or nonobservable model. Our method, called AutoRepar, achieves this goal by eliminating those symmetries in the model equations that cause the lack of SIO. Thus, a prerequisite for its application is that the SIO of the model has been analyzed and any existing symmetries have been found. To account for these requirements we have paired the new method with two of our previous developments for (i) analyzing SIO and (ii) finding the Lie symmetries that cause the lack of SIO. The combination of these three techniques yields an integrated procedure that completes the “model curation” process, as far as SIO is concerned. First, the SIO of a model is analyzed; second, if the analysis reports lack of SIO, the Lie symmetries that cause the problem are sought; third, the symmetries are automatically repaired.

AutoRepar is applicable to nonlinear ODE models with rational expressions, which are a very general class of models used widely in many areas of the life sciences. AutoRepar makes these models fully identifiable and observable through a reparameterization that has a number of remarkable features. First, it keeps the output functions invariant, that is, it does not require measuring additional states. Second, it is an exact reformulation, not an approximation, that is, the dynamics of the final model is identical to that of the original one. Third, while the reparameterization typically removes a subset of the parameters, and transforms a number of model variables (parameters and/or states), it usually leaves another subset of variables intact. Thus, these variables that are not transformed preserve their mechanistic meaning. Fourth, the method can be applied to models with unknown inputs. Indeed, we have found that unknown inputs provide flexibility to the reparameterization, and in some cases the consideration of an input as an unknown function can make it possible to keep some variables from being transformed. Fifth, it can consider parameterized initial conditions. Sixth, it can repair symmetries of different types, not only the simplest ones such as scaling. Seventh, it can repair models that need several transformations, each of which may be a nonelementary transformation.

We have implemented the AutoRepar methodology in a new version of our open source MATLAB toolbox, STRIKE-GOLDD, which already included routines for analyzing SIO and finding Lie symmetries. The resulting tool provides an integrated platform that analyzes the structural identifiability and observability of a model, finds the causes that prevent them, and repairs them. The tool is almost fully automated: after entering the model equations, the user is only asked to select which variables (from a given subset) can be removed or transformed, and which ones must be preserved. The possibility of making this choice is a crucial feature of the tool, since it allows for specifying which predictions about parameters or states are to be obtained from the model.

We have illustrated the methodological aspects discussed above using four dynamic models of pharmacokinetics, physiological, and signaling processes. These case studies have also demonstrated the applicability of the computational implementation, showing that the tool can reparameterize models with dozens of variables.

ACKNOWLEDGMENTS

This work was inspired by discussions at the Banff International Research Station workshop on “Model Theory of Differential Equations, Algebraic Geometry, and their Applications to Modeling”, held in 2020. The authors especially thank Gleb Pogudin for helpful conversations.

This research has received funding from MCIN/AEI/ 10.13039/501100011033 through grants PID2020-113992RA-I00 (PREDYCTBIO) and PID2020-117271RB-C22 (BIODYNAMICS); from MCIN/AEI/ 10.13039/501100011033 and “ERDF A way of making Europe” through grant DPI2017-82896-C2-2-R (SYNBIOCONTROL); from MCIN/AEI/ 10.13039/501100011033 and “ESF Investing in your future” through grant RYC-2019-027537-I (Ramón y Cajal); from the Consejo Superior de Investigaciones Científicas through grant PIE 202070E062 (intramural project MOEBIUS); and from the Consellería de Cultura, Educación e Ordenación Universitaria, Xunta de Galicia through grant ED431F 2021/003. The funding bodies played no role in the design of the study, the collection and analysis of data, or in the writing of the manuscript.

AUTHOR CONTRIBUTIONS

Alejandro F. Villaverde designed the study. **Gemma Massonis** wrote the code and performed the computational analyses. All authors discussed the results and wrote the manuscript.

CONFLICT OF INTEREST

The authors declare no potential conflict of interests.

DATA AVAILABILITY STATEMENT

Data sharing is not applicable to this article as no new data were created or analyzed in this study. All the information needed to evaluate the conclusions in the paper is present in the paper and/or the Supplementary Information. Models, codes, and instructions are available from <https://github.com/afvillaverde/strike-goldd/releases/tag/v3.0>.

ORCID

Alejandro F. Villaverde  <https://orcid.org/0000-0001-7401-7380>

REFERENCES

- DiStefano J III. *Dynamic Systems Biology Modeling and Simulation*. Academic Press; 2015.
- Chen WW, Niepel M, Sorger PK. Classic and contemporary approaches to modeling biochemical reactions. *Genes Dev*. 2010;24(17):1861-1875.
- Meibohm B, Derendorf H. Basic concepts of pharmacokinetic/pharmacodynamic (pk/pd) modelling. *Int J Clin Pharmacol Ther*. 1997;35(10):401-413.
- Cobelli C, Carson E. *Introduction to Modeling in Physiology and Medicine*. Academic Press; 2019.
- Almquist J, Cvijovic M, Hatzimanikatis V, Nielsen J, Jirstrand M. Kinetic models in industrial biotechnology—improving cell factory performance. *Metab Eng*. 2014;24:38-60.
- Wiechert W, Noack S. Mechanistic pathway modeling for industrial biotechnology: challenging but worthwhile. *Curr Opin Biotechnol*. 2011;22(5):604-610.
- Tummler K, Klipp E. The discrepancy between data for and expectations on metabolic models: how to match experiments and computational efforts to arrive at quantitative predictions? *Curr Opin Syst Biol*. 2018;8:1-6.
- Alkan O, Schoeberl B, Shah M, et al. Modeling chemotherapy-induced stress to identify rational combination therapies in the DNA damage response pathway. *Sci Signal*. 2018;11(540):eaat0229.
- Becker V, Schilling M, Bachmann J, et al. Covering a broad dynamic range: information processing at the erythropoietin receptor. *Science*. 2010;328(5984):1404-1408.
- Baker RE, Peña J-M, Jayamohan J, Jérusalem A. Mechanistic models versus machine learning, a fight worth fighting for the biological community? *Biol Lett*. 2018;14(5):20170660.
- Walter E, Pronzato L. *Identification of Parametric Models from Experimental Data*. Communications and Control Engineering Series. Springer; 1997.
- Jaqaman K, Danuser G. Linking data to models: data regression. *Nat Rev Mol Cell Biol*. 2006;7(11):813-819.
- Heinemann T, Raue A. Model calibration and uncertainty analysis in signaling networks. *Curr Opin Biotechnol*. 2016;39:143-149.
- Ashyraliyev M, Fomekong-Nanfack Y, Kaandorp J, Blom J. Systems biology: parameter estimation for biochemical models. *FEBS J*. 2009;276(4):886-902.
- Chin S, Chappell M. Structural identifiability and indistinguishability analyses of the minimal model and a euglycemic hyperinsulinemic clamp model for glucose–insulin dynamics. *Comput Methods Programs Biomed*. 2011;104(2):120-134.
- Janzén DL, Bergenholm L, Jirstrand M, et al. Parameter identifiability of fundamental pharmacodynamic models. *Front Physiol*. 2016;7:590.
- Villaverde AF, Banga JR. Dynamical compensation and structural identifiability of biological models: analysis, implications, and reconciliation. *PLoS Comput Biol*. 2017;13(11):e1005878.
- Ryser MD, Gulati R, Eisenberg MC, Shen Y, Hwang ES, Etzioni RB. Identification of the fraction of indolent tumors and associated overdiagnosis in breast cancer screening trials. *Am J Epidemiol*. 2018;188(1):197-205.
- Schmidt PJ, Emelko MB, Thompson ME. Recognizing structural nonidentifiability: when experiments do not provide information about important parameters and misleading models can still have great fit. *Risk Anal*. 2020;40(2):352-369.
- Wieland F-G, Hauber AL, Rosenblatt M, Tönsing C, Timmer J. On structural and practical identifiability. *Curr Opin Syst Biol*. 2021;25:60-69.
- Raue A, Kreutz C, Maiwald T, et al. Structural and practical identifiability analysis of partially observed dynamical models by exploiting the profile likelihood. *Bioinformatics*. 2009;25(15):1923-1929.
- Bellman R, Åström KJ. On structural identifiability. *Math Biosci*. 1970;7(3):329-339.
- Pohjanpallo H. System identifiability based on the power series expansion of the solution. *Math Biosci*. 1978;41(1):21-33.
- Denis-Vidal L, Joly-Blanchard G. An easy to check criterion for (un)identifiability of uncontrolled systems and its applications. *IEEE Trans Automat Contr*. 2000;45:768-771.
- Walter E, Braems I, Jaulin L, Kieffer M. Guaranteed numerical computation as an alternative to computer algebra for testing models for identifiability. *Numerical Software with Result Verification*. Springer; 2004:124-131.
- Walter E, Lecourtier Y. Global approaches to identifiability testing for linear and nonlinear state space models. *Math Comput Simul*. 1982;24(6):472-482.
- Bellu G, Saccomani MP, Audoly S, D'Angio L. DAISY: a new software tool to test global identifiability of biological and physiological systems. *Comput Methods Programs Biomed*. 2007;88(1):52-61.
- Ljung L, Glad T. On global identifiability for arbitrary model parametrizations. *Automatica*. 1994;30(2):265-276.
- Sedoglavic A. A probabilistic algorithm to test local algebraic observability in polynomial time. *J Symb Comput*. 2002;33:735-755.

30. Karlsson J, Anguelova M, Jirstrand M. An efficient method for structural identifiability analysis of large dynamic systems. *IFAC Proc Vol.* 2012;45(16):941-946.
31. Villaverde AF, Barreiro A, Papachristodoulou A. Structural identifiability of dynamic systems biology models. *PLoS Comput Biol.* 2016;12(10):e1005153.
32. Evans ND, Chapman MJ, Chappell MJ, Godfrey KR. Identifiability of uncontrolled nonlinear rational systems. *Automatica.* 2002;38(10):1799-1805.
33. Yates JW, Evans ND, Chappell MJ. Structural identifiability analysis via symmetries of differential equations. *Automatica.* 2009;45(11):2585-2591.
34. Chiş O, Banga JR, Balsa-Canto E. GenSSI: a software toolbox for structural identifiability analysis of biological models. *Bioinformatics.* 2011;27(18):2610-2611.
35. Ligon T, Froehlich F, Chi O, Banga J, Balsa-Canto E, Hasenauer J. Genssi 2.0: multi-experiment structural identifiability analysis of SBML models. *Bioinformatics.* 2018;34(8):1421-1423.
36. Meshkat N, Kuo C E-z, DiStefano J III. On finding and using identifiable parameter combinations in nonlinear dynamic systems biology models and combos: A novel web implementation. *PLoS One.* 2014;9(10):e110261.
37. Hong H, Ovchinnikov A, Yap C, Pogudin G. SIAN: software for structural identifiability analysis of ODE models. *Bioinformatics.* 2019;35(16):2873-2874.
38. Villaverde AF. Observability and structural identifiability of nonlinear biological systems. *Complexity.* 2019;2019:8497093.
39. Karin O, Swisa A, Glaser B, Dor Y, Alon U. Dynamical compensation in physiological circuits. *Mol Syst Biol.* 2016;12(11):886.
40. Kaveh P, Shtessel YB. Blood glucose regulation using higher-order sliding mode control. *Int J Robust Nonlin Control.* 2008;18(4-5):557-569.
41. Anguelova M, Karlsson J, Jirstrand M. Minimal output sets for identifiability. *Math Biosci.* 2012;239(1):139-153.
42. Snowden TJ, van der Graaf PH, Tindall MJ. Methods of model reduction for large-scale biological systems: a survey of current methods and trends. *Bull Math Biol.* 2017;79(7):1449-1486.
43. Meshkat N, Sullivant S. Identifiable reparameterizations of linear compartment models. *J Symb Comput.* 2014;63:46-67.
44. Evans ND, Chappell MJ. Extensions to a procedure for generating locally identifiable reparameterisations of unidentifiable systems. *Math Biosci.* 2000;168(2):137-159.
45. Chappell MJ, Gunn RN. A procedure for generating locally identifiable reparameterisations of unidentifiable non-linear systems by the similarity transformation approach. *Math Biosci.* 1998;148(1):21-41.
46. Merkt B, Timmer J, Kaschek D. Higher-order lie symmetries in identifiability and predictability analysis of dynamic models. *Phys Rev E.* 2015;92:012920.
47. G. Massonis and A. Villaverde, "Finding and breaking lie symmetries: Implications for structural identifiability and observability in biological modelling," *Symmetry*, vol. 12, p. 469, 03 2020.
48. Villaverde AF, Tsiantis N, Banga JR. Full observability and estimation of unknown inputs, states, and parameters of nonlinear biological models. *J R Soc Interf.* 2019;16:20190043.
49. Tunali ET, Tarn T-J. New results for identifiability of nonlinear systems. *IEEE Trans Automat Contr.* 1987;32(2):146-154.
50. Gunn RN, Chappell MJ, Cunningham VJ. Reparameterisation of unidentifiable systems using the Taylor series approach. *IFAC Proc Vol.* 1997;30(2):247-252.
51. Bluman G, Anco S. *Symmetry and Integration Methods for Differential Equations.* Vol 154. Springer Science & Business Media; 2008.
52. Vajda S, Godfrey KR, Rabitz H. Similarity transformation approach to identifiability analysis of nonlinear compartmental models. *Math Biosci.* 1989;93(2):217-248.
53. Villaverde AF, Massonis G. On testing structural identifiability by a simple scaling method: relying on scaling symmetries can be misleading. *PLOS Comput Biol.* 2021;17(10):e1009032.
54. Hubert E, Labalin G. Scaling invariants and symmetry reduction of dynamical systems. *Found Comput Math.* 2013;13(4):479-516.
55. Newmiwaka T, Engelhardt B, Wendland P, Kahl D, Frohlich H, Kschischo M. Seeds: data driven inference of structural model errors and unknown inputs for dynamic systems biology. *Bioinformatics.* 2021;37(9):1330-1331.

SUPPORTING INFORMATION

Additional supporting information may be found online in the Supporting Information section at the end of this article.

How to cite this article: Massonis G, Banga JR, Villaverde AF. AutoRepar: A method to obtain identifiable and observable reparameterizations of dynamic models with mechanistic insights. *Int J Robust Nonlinear Control.* 2021;1-19. doi: 10.1002/rnc.5887