

Article

Nonlinear Observability Algorithms with Known and Unknown Inputs: Analysis and Implementation

Nerea Martínez ^{1,2,3} and Alejandro F. Villaverde ^{1,*} ¹ BioProcess Engineering Group, IIM-CSIC, 36208 Vigo, Galicia, Spain; nerea.martinez.lopez0@usc.es² Department of Applied Mathematics II, University of Vigo, 36310 Vigo, Galicia, Spain³ Department of Applied Mathematics, University of Santiago de Compostela, 15782 Santiago de Compostela, Galicia, Spain

* Correspondence: afvillaverde@iim.csic.es

Received: 30 September 2020; Accepted: 22 October 2020; Published: 29 October 2020



Abstract: The observability of a dynamical system is affected by the presence of external inputs, either known (such as control actions) or unknown (disturbances). Inputs of unknown magnitude are especially detrimental for observability, and they also complicate its analysis. Hence, the availability of computational tools capable of analysing the observability of nonlinear systems with unknown inputs has been limited until lately. Two symbolic algorithms based on differential geometry, ORC-DF and FISPO, have been recently proposed for this task, but their critical analysis and comparison is still lacking. Here we perform an analytical comparison of both algorithms and evaluate their performance on a set of problems, while discussing their strengths and limitations. Additionally, we use these analyses to provide insights about certain aspects of the relationship between inputs and observability. We found that, while ORC-DF and FISPO follow a similar approach, they differ in key aspects that can have a substantial influence on their applicability and computational cost. The FISPO algorithm is more generally applicable, since it can analyse any nonlinear ODE model. The ORC-DF algorithm analyses models that are affine in the inputs, and if those models have known inputs it is sometimes more efficient. Thus, the optimal choice of a method depends on the characteristics of the problem under consideration. To facilitate the use of both algorithms, we implemented the ORC-DF condition in a new version of STRIKE-GOLDD, a MATLAB toolbox for structural identifiability and observability analysis. Since this software tool already had an implementation of the FISPO algorithm, the new release allows modellers and model users the convenience of choosing between different algorithms in a single tool, without changing the coding of their model.

Keywords: observability; identifiability; nonlinear systems; control theory; differential geometry; software

1. Introduction

Mathematical models of ordinary differential equations (ODEs) are used in all areas of science and technology for describing nonlinear systems. The ODEs define the evolution of the system state variables $x(t)$ with respect to time, while the measurable quantities $y(t)$ are defined by the output function. The model equations (ODEs and output function) may contain unknown parameters θ , and external inputs that may be known ($u(t)$) or unknown ($w(t)$). The structure of the model equations determines whether it is possible to estimate the model unknowns from the outputs. The theoretical possibility of inferring the states (respectively parameters) from the outputs is called observability (respectively structural identifiability) [1,2]. Since a parameter can be considered as a state variable with time derivative equal to zero, structural identifiability can be considered as a particular case of observability. Additionally, the possibility of recovering the unknown inputs is called reconstructibility

or input observability. For simplicity, in this manuscript we use the word *observability* for all model unknowns, that is, to refer to the possibility of determining states, parameters, and/or inputs from the output. We assume that the model structure—the ODEs and output function—is known and correct. The assessment of the possible existence of alternative structures capable of generating exactly the same output is a different albeit related problem [3] that is not addressed in the present article.

The concept of observability arose in systems and control theory. It was initially defined for linear models and extended to the nonlinear case afterwards [4]. The concept of structural identifiability, on the other hand, was motivated by the analysis of biological models [5], due to the specific challenges that parameter identification poses in mathematical biology and other biosciences. Hence, many observability analysis methods developed in that context aimed at analysing structural identifiability and were named accordingly, even though they could be applied or adapted to the more general task of analysing observability. Examples of software tools include DAISY [6], COMBOS [7], IdentifiabilityAnalysis [8], STRIKE-GOLDD [9], GenSSI [10], and SIAN [11].

The existence of external inputs affects the observability of a model and determines which methods can be applied for its analysis. A key distinction is between known and unknown inputs, where "known" is interpreted as "quantified"; thus, we are aware of the existence of an unknown input but not of its magnitude. A known input that can be manipulated is also called a control input, or simply a control. An unknown input can be considered as an unmeasured disturbance or as a time-varying parameter. Some techniques are applicable specifically to uncontrolled systems [12], while others allow for the existence of known inputs. To the best of our knowledge, the first works describing algorithms capable of handling *unknown* inputs were presented by [13,14]. These methods are not applicable to systems in which the outputs are direct functions of the inputs, and do not analyse the observability of the unknown input itself.

To address these issues, two differential geometry algorithms called ORC-DF—observability rank criterion with direct feedthrough [15]—and FISPO—full input, state, and parameter observability [16]—have been recently presented. Both methods are capable of determining the observability of states and parameters, and unknown inputs of nonlinear ODE models. ORC-DF is applicable to affine-in-the-inputs systems, while FISPO does not have this requirement. The latter algorithm is already implemented in the STRIKE-GOLDD toolbox [9].

In the present paper we perform a critical examination of the ORC-DF and FISPO algorithms. First we provide the necessary background on observability analysis and differential geometry in Section 2. Then we perform a theoretical analysis of the two methods in Section 3, showing that they are equivalent for systems without known inputs, and that they differ for other classes of models, as a result of building different observability matrices. Realising the convenience of having both algorithms available in the same software environment, we provide their implementations in a new version of the MATLAB toolbox STRIKE-GOLDD, which is described in Section 4. The new release includes an implementation of ORC-DF, and a seamless integration with the already existing FISPO. Furthermore, it enables the automatic analysis of multi-experiment observability throughout any of the two implemented algorithms. Since ORC-DF and FISPO are symbolic algorithms that can be computationally expensive, in Section 5 we evaluate their performances by applying them to a number of modelling problems of different domains, from mechanical engineering to biology, and report their applicability and computational costs. The analysis of the selected case studies is also helpful for obtaining detailed insights about the inner workings of the algorithms. Finally, we conclude with a discussion of the results in Section 6.

2. Materials and Methods

2.1. Notation and Model Classes

We are interested in studying the observability of nonlinear systems of the following (fairly general) form:

$$\Sigma = \dot{x} = f(x, \theta, u, w), \quad (1)$$

$$y = h(x, \theta, u, w), \quad (2)$$

where f and h are nonlinear and analytical (infinitely differentiable) functions of the model variables (which depend on time $t \in I = [0, T)$, $T > 0$), i.e., system states $x(t) \in \mathbb{R}^{n_x}$, a set of inputs, both known $u(t) \in \mathbb{R}^{n_u}$, and unknown $w(t) \in \mathbb{R}^{n_w}$, and unmeasured parameters $\theta \in \mathbb{R}^{n_\theta}$. On the other hand, the system output vector $y(t) \in \mathbb{R}^m$ consists of measured functions of model variables. We assume that $n_x, m \geq 1$, while n_θ, n_u , and n_w are non-negative integers. Note that the dependence of model variables over time is omitted from the Equations (1)–(2) for simplicity of notation.

As a special case of Σ we also study affine-in-inputs systems, which are given by the expressions:

$$\Sigma_A = \dot{x} = f_0(x, \theta) + \sum_{i=1}^{n_u} f_{u_i}(x, \theta) u_i + \sum_{i=1}^{n_w} f_{w_i}(x, \theta) w_i, \quad (3)$$

$$y = h_0(x, \theta) + \sum_{i=1}^{n_u} h_{u_i}(x, \theta) u_i + \sum_{i=1}^{n_w} h_{w_i}(x, \theta) w_i, \quad (4)$$

where f_0, f_{u_i}, f_{w_j} and h_0, h_{u_i}, h_{w_j} are—possibly nonlinear—analytical functions for $1 \leq i \leq n_u$, $1 \leq j \leq n_w$, and:

$$f = f_{xw} + \sum_{i=1}^{n_u} f_{u_i} u_i, \quad h = h_{xw} + \sum_{i=1}^{n_u} h_{u_i} u_i, \quad (5)$$

where, following [15]:

$$f_{xw} = f_0 + \sum_{i=1}^{n_w} f_{w_i} w_i, \quad h_{xw} = h_0 + \sum_{i=1}^{n_w} h_{w_i} w_i. \quad (6)$$

Note that, as in Equations (3) and (4), the functions $f_0, f_{u_i}, f_{w_i}, h_0, h_{u_i}, h_{w_i}$ can depend on (x, θ) , but we omit this dependency in the expressions above and elsewhere when convenient.

In what follows, a vector $v \in \mathbb{R}^n$ is assumed to be a one column matrix and v^T its transpose. The Jacobian matrix of a function $\phi = (\phi_1, \dots, \phi_s)$ with respect to a vector field $v = (v_1, \dots, v_n)$, will be denoted as:

$$\frac{\partial \phi}{\partial v} = \left[\frac{\partial \phi_i}{\partial v_j} \right]_{ij}, \quad 1 \leq i \leq s, \quad 1 \leq j \leq n.$$

2.2. Background

2.2.1. Structural Identifiability, Observability, and Differential Geometry

Roughly speaking, a nonlinear system Σ is structurally observable if it is possible to distinguish between its state trajectories from the data provided by its output, and structurally reconstructible (or input observable) if its disturbances can be tracked from the aforementioned measurements. Similarly, Σ is said to be structurally identifiable if it is possible to infer the values of its unknown parameters from the output. In practice, it is often not necessary distinguish between every pair of unmeasured states in the phase mapping of Σ —a property called structural global observability—and it is sufficient to distinguish neighbouring states—a property called local "weak" observability in some texts [4]. In this work we will not further make this distinction, and local "weak" structural

observability will be simply called observability. Likewise, we will refer to structural local "weak" identifiability simply as identifiability.

Structural identifiability can be addressed as a particular case of observability. This is because any unknown parameter θ_i of Σ can be considered as a zero-dynamics state, that is, the equation $\dot{\theta}_i = 0$ holds for $1 \leq i \leq n_p$. It is then possible to augment the system state as:

$$\bar{x} = \begin{pmatrix} x & \theta \end{pmatrix}^T, \quad (7)$$

which consists of $n_{\bar{x}} = n_x + n_{\theta}$ components and follows the augmented dynamics:

$$\dot{\bar{x}} = \begin{pmatrix} f(x, \theta, u, w)^T & 0_{1 \times n_p} \end{pmatrix}^T = \bar{f}(\bar{x}, u, w). \quad (8)$$

Thus, the identifiability and observability of Σ can be studied as the observability of the augmented system (totally equivalent to the original one) with state vector (7), dynamics (8), and output (2), the same as Σ .

The algorithms analysed in this work adopt a differential geometry approach, which relies on the use of the concept of Lie derivative to bring out algebraic (and in some sense, geometric) tests suitable to address the study of observability of analytic systems Σ . Let us consider first the simpler case in which Σ is not dependent on unknown inputs:

$$\Sigma' \begin{cases} \dot{x} = f(x, \theta, u), \\ y = h(x, \theta, u). \end{cases}$$

Definition 1 (Lie derivative [17]). Consider the system Σ' with augmented state vector (7) and augmented dynamics (8). Fixing the control variables u , the Lie derivative of the output function $h(\cdot, u)$ in \bar{x} along the tangent vector field $\bar{f} = \bar{f}(\cdot, u)$ is:

$$L_{\bar{f}} h(\bar{x}, u) = \frac{\partial h}{\partial \bar{x}}(\bar{x}, u) \bar{f}(\bar{x}, u).$$

Furthermore, setting $L_{\bar{f}}^0 h = h$, the i -order Lie derivative of $h(\cdot, u)$ (along the vector field $\bar{f} = \bar{f}(\cdot, u)$) can be recursively computed in \bar{x} as:

$$L_{\bar{f}}^i h(\bar{x}, u) = L_{\bar{f}} \left(L_{\bar{f}}^{i-1} h(\bar{x}, u) \right), \quad i \geq 1.$$

The above definition does not capture the effect of possible time variations in the control variables. To account for it, this notion can be extended as follows:

Definition 2 (Extended Lie derivative [8]). Consider the system Σ' with augmented state vector (7), augmented dynamics (8), and assume that the controls $u = (u_1, \dots, u_{n_u})$ are analytical functions on $I = [0, T)$, $T > 0$. The extended Lie derivative of the output function h in \bar{x} by the tangent vector field $\bar{f} = \bar{f}(\cdot, u)$ is:

$$L_{\bar{f}}^e h(\bar{x}, u) = \frac{\partial h}{\partial \bar{x}}(\bar{x}, u) \bar{f}(\bar{x}, u) + \frac{\partial h}{\partial u}(\bar{x}, u) \dot{u}.$$

Moreover, setting $L_{\bar{f}}^{e,0} h = h$, the i -order extended Lie derivative of h can be recursively computed in \bar{x} as:

$$L_{\bar{f}}^{e,i} h(\bar{x}, u) = \frac{\partial L_{\bar{f}}^{e,i-1} h}{\partial \bar{x}}(\bar{x}, u) \bar{f}(\bar{x}, u) + \sum_{j=0}^{i-1} \frac{\partial L_{\bar{f}}^{e,i-1} h}{\partial u^j}(\bar{x}, u) u^{j+1}, \quad i \geq 1, \quad (9)$$

where u^{j+1} ($j \geq 0$) stands for the $(j+1)$ -order time derivative of u .

Remark 1. For any fixed controls $u(t)$, assuming Σ is complete and denoting $\bar{x}_0^u(t)$ the state trajectory such that:

$$\begin{cases} \dot{\bar{x}}_0^u(t) = \bar{f}(\bar{x}_0^u(t), u(t)), & t \in I, \\ \bar{x}_0^u(0) = \bar{x}_0, \end{cases}$$

then, in the case of constant inputs $u(t) = u_0 \in \mathbb{R}^{n_u}$, the Lie derivative introduced in Definition (1) verifies:

$$\begin{aligned} y(0) &= h(\bar{x}_0^u(t), u_0)(0) = L_{\bar{f}}^0 h(\bar{x}_0, u_0), \\ y'(0) &= \frac{d}{dt} h(\bar{x}_0^u(t), u_0)(0) = \frac{\partial h}{\partial \bar{x}}(\bar{x}_0^u(t), u_0) \dot{\bar{x}}_0^u(t) \Big|_{t=0} = \frac{\partial h}{\partial \bar{x}}(\bar{x}_0, u_0) \bar{f}(\bar{x}_0, u_0) = L_{\bar{f}} h(\bar{x}_0, u_0), \\ y''(0) &= \frac{d}{dt} L_{\bar{f}} h(\bar{x}_0^u(t), u_0)(0) = \frac{\partial L_{\bar{f}} h}{\partial \bar{x}}(\bar{x}_0^u(t), u_0) \dot{\bar{x}}_0^u(t) \Big|_{t=0} = \frac{\partial L_{\bar{f}} h}{\partial \bar{x}}(\bar{x}_0, u_0) \bar{f}(\bar{x}_0, u_0) = L_{\bar{f}}^2 h(\bar{x}_0, u_0), \\ &\vdots \\ y^{(i)}(0) &= \frac{d}{dt} L_{\bar{f}}^{i-1} h(\bar{x}_0^u(t), u_0)(0) = L_{\bar{f}}(L_{\bar{f}}^{i-1} h(\bar{x}_0^u(t), u_0)) \Big|_{t=0} = L_{\bar{f}}^i h(\bar{x}_0, u_0), \quad i \geq 0, \end{aligned}$$

which can be proven by using repeatedly the chain rule. Likewise, if $u(t)$ are time-varying and $u(0) = u_0$, the extended Lie derivative of Definition (2) verifies:

$$y^{(i)}(0) = L_{\bar{f}}^{e,i} h(\bar{x}_0, u_0), \quad i \geq 0.$$

Given a nonlinear system Σ' with augmented state (7) and analytical inputs, it is possible to use the extended Lie derivatives of the output to build (symbolically) the following $mn_{\bar{x}} \times n_{\bar{x}}$ matrix:

$$\mathcal{O}_I(\bar{x}, u) = \frac{\partial}{\partial \bar{x}} \left(L_{\bar{f}}^0 h(\bar{x}, u)^T \quad L_{\bar{f}} h(\bar{x}, u)^T \quad L_{\bar{f}}^2 h(\bar{x}, u)^T \quad \dots \quad L_{\bar{f}}^{n_{\bar{x}}-1} h(\bar{x}, u)^T \right)^T, \quad (10)$$

which is the observability-identifiability matrix of Σ' . Note that the assumption of analytic controls is conservative, since it suffices that they are differentiable up to order $n_d = n_{\bar{x}} - 1$. By calculating the rank of the above matrix, it is possible to establish the observability and identifiability of Σ' using the following condition.

Theorem 1 (Observability-identifiability condition, OIC [8]). *If the identifiability-observability matrix (10) of a model Σ' satisfies $\text{rank}(\mathcal{O}_I(\bar{x}_0, u)) = n_{\bar{x}}$, with \bar{x}_0 being a (possibly generic) point in the augmented state space (8) of Σ' , then the system is structurally locally observable and structurally locally identifiable.*

Remark 2. The rank of (10) is constant except for a zero-measurement subset in the augmented state space (8) of Σ' where the rank is smaller, as a consequence of the system being analytical [18]. Thus, to verify the condition of the Theorem 1 it is sufficient to calculate the rank of (10) at any non-singular point of the phase space.

2.2.2. FISPO

Unknown inputs w can be taken into account by further augmenting the phase mapping of Σ , including w as unmeasured states. Thus, for any non-negative integer l we have the l -augmented state vector:

$$x^l = \left(x, \theta, w, \dots, w^{(l)} \right), \quad (11)$$

which follows the l -augmented dynamics:

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

leading to the l -augmented system:

$$\Sigma^l \begin{cases} \dot{x}^l = f^l(x^l, u, w^{l+1}), \\ y = h(x^0, u). \end{cases} \quad (12)$$

An analogous extension for affine systems Σ_A exists. Using the notation given in (5) and (6), the l -augmented system described above takes the form [13]:

$$\Sigma_A^l \begin{cases} \dot{x}^l = f_{xw}^l(x^l, w^{l+1}) + \sum_{i=1}^{n_u} f_{u_i}^l(x, \theta) u_i, \\ y = h_{xw}(x^0) + \sum_{i=1}^{n_u} h_{u_i}(x, \theta) u_i, \end{cases}$$

where the l -augmented dynamics is decomposed as follows:

$$f_{xw}^l(x^l, w^{l+1}) = \left(f_{xw}(x^0)^T \quad 0_{1 \times n_p} \quad \dot{w}^T \quad \dots \quad (w^{l+1})^T \right)^T, \quad (13)$$

$$f_{u_i}^l(x, \theta) = \left(f_{u_i}(x, \theta) \quad 0_{1 \times n_p} \quad 0_{1 \times (l+1)n_w} \right)^T, \quad 1 \leq i \leq n_u. \quad (14)$$

We note that, in order to build l -augmented systems Σ^l and Σ_A^l , it must be possible to calculate the $l + 1$ -time derivative of disturbances $w(t)$, and therefore, they will be considered as analytical functions from now on (again, this hypothesis is more restrictive than necessary). We also note that the l -augmented form of Σ and Σ_A is totally equivalent to the original system, and consists of $n^l = n_x + n_\theta + (l + 1)n_w$ states, n_u controls, n_w disturbances (the $l + 1$ -order time derivatives of w) and m outputs, that have not changed due to state augmentation [13].

As an additional hypothesis we assume that a non-negative integer s exists (possibly $s = +\infty$) such that $w^{(s)}(t) \neq 0$ and $w^{(i)}(t) = 0$ for all $i > s$. In principle, this assumption introduces a restriction on the type of allowed inputs, and it is equivalent to assuming that the disturbances are (s order) polynomial functions of time. Nevertheless, in practice, the method may still provide relevant information about the analytical case, as is discussed in [16].

In what follows, if a vector function $\phi = (\phi_1, \dots, \phi_r)$, $r \geq 1$, depends on variables x^l we denote:

$$d^l \phi(x^l) = \frac{\partial \phi}{\partial x^l}(x^l) = \left[\frac{\partial \phi_i}{\partial x_j^l}(x^l) \right]_{ij}, \quad 1 \leq i \leq r, \quad 1 \leq j \leq n^l, \\ L_{f^l} \phi(x^l) = d^l \phi(x^l) f^l(x^l, w^{l+1}),$$

and if $\phi = \phi(\cdot, u)$ (where controls u are considered to be analytical on the time interval I), then:

$$L_{f^l}^e \phi(x^l, u) = d^l \phi(x^l, u) f^l(x^l, u, w^{l+1}) + \frac{\partial \phi}{\partial u}(x^l, u) \dot{u}.$$

Definition 3 (Full input-state-parameter observability, FISPO [16]). Consider the system Σ and the augmented state vector $z(t) = (x(t), \theta, w(t))$. We say that Σ has the FISPO property if, for every $t_0 \in I$ and $1 \leq i \leq n^0$, $z_i(t_0)$ can be locally determined from the output $y(t)$ and the known inputs $u(t) = (u_1(t), \dots, u_{n_u}(t))$ in a finite time interval $[t_0, t_f] \subset I$. Thus, a system Σ is FISPO if, for every $z(t_0)$ and

for almost any vector $z^*(t_0)$, there is a neighbourhood $\mathcal{N}(z^*(t_0))$ such that, for all $\hat{z}(t_0) \in \mathcal{N}(z^*(t_0))$, the following condition holds:

$$y(t, \hat{z}(t_0)) = y(t, z^*(t_0)) \Rightarrow \hat{z}_i(t_0) = z_i^*(t_0), 1 \leq i \leq n^0.$$

Remark 3. The original definition of the term FISPO reproduced above refers to a model property. Here we also use it to refer to the algorithm presented for its evaluation by [16].

Using the system augmentation (12) and taking the unique $l = s$ such that $w^s(t) \neq 0$ and $w^i(t) = 0$ for all $i > s$, it is possible to build the following matrix,

$$\mathcal{O}_l^s(x^s, u) = d^s \left(L_{fs}^0 h(x^s, u)^T \quad L_{fs} h(x^s, u)^T \quad L_{fs}^2 h(x^s, u)^T \quad \dots \quad L_{fs}^{n^s-1} h(x^s, u)^T \right)^T, \quad (15)$$

which is the generalised observability matrix of Σ . Note that (15) coincides with the observability matrix (10) of Σ^s without disturbances. Thus, the rank of \mathcal{O}_l^s provides a condition for assessing the observability of Σ as follows:

Theorem 2 (FISPO condition [16]). A nonlinear system Σ given by (1-2) with analytic inputs is FISPO if, for x_0^s being a (possibly generic) point in the state space of the s -augmented system Σ^s , the generalised observability matrix (15) verifies $\text{rank}(\mathcal{O}_l^s(x_0^s, u)) = n^s$.

Remark 4. For $1 \leq i \leq n^s$, the observability of the i -th state of x^s can also be studied using the matrix (15). Thus, if $\mathcal{O}_l^{s,i}(x_0^s, u)$ denotes the matrix obtained from $\mathcal{O}_l^s(x_0^s, u)$ after removing its i -th column, state x_i is observable if $\text{rank}(\mathcal{O}_l^{s,i}(x_0^s, u)) < \text{rank}(\mathcal{O}_l^s(x_0^s, u))$ for almost any x_0^s in the phase space of Σ^s .

2.2.3. ORC-DF

The observability of affine systems Σ_A with bounded measurable controls $u(t)$ can also be analysed by building a different observability matrix, as explained below. For a full description of the procedure, see [15].

Definition 4 (Observability rank criterion for systems with direct feedthrough, ORC-DF). A system Σ_A is classified as k -row observable if almost any initial state $x^k(t_0)$, $t_0 \in I$, in the state space of the k -augmented system Σ_A^k can be separated locally from its neighbours based on the output at $k+1$ consecutive times t_0, t_1, \dots, t_k . If there exists $k \geq 1$ such that Σ_A is k -row observable, it is said that Σ_A satisfies the ORC-DF.

Lemma 1. Consider the system Σ_A , and the vector field Ω_k , which is recursively defined for $k \geq 0$ by:

$$\begin{aligned} \Omega_0 &= \left(h_{xw}^T \quad h_{u_1}^T \quad \dots \quad h_{u_{n_u}}^T \right)^T, \quad \Delta\Omega_0 = \Omega_0, \\ \Delta\Omega_{k+1} &= \left(L_{fxw}^k (\Delta\Omega_k)^T \quad L_{fu_1}^k (\Delta\Omega_k)^T \quad \dots \quad L_{fu_{n_u}}^k (\Delta\Omega_k)^T \right)^T, \\ \Omega_{k+1} &= \left(\Omega_k^T \quad \Delta\Omega_{k+1}^T \right)^T, \end{aligned}$$

then Σ_A is k -row observable if $\text{rank}(d^k \Omega_k(x_0^k)) = n^k$ for almost any x_0^k in the phase space of Σ_A^k .

Lemma 2. If Σ_A satisfies the ORC-DF, then Σ_A is observable in the presence of unmeasured inputs.

Corollary 1. Let $d^k \Omega_k^i$ denote the matrix that is obtained after removing the i -th column from $d^k \Omega_k$. The i -th state of x^k is k -row observable if and only if $\text{rank}(d^k \Omega_k^i(x_0^k)) < \text{rank}(d^k \Omega_k(x_0^k))$ for almost any x_0^k in the phase space of Σ^k .

3. Theory: Analysis of the FISPO and ORC-DF Algorithms

In this section we discuss the similarities and differences between FISPO and ORC-DF, whose pseudo-code is provided in Algorithms 1 and 2.

3.1. Preliminary Remarks

We begin by recalling three facts that are relevant for the analysis: (i) The FISPO algorithm does not always require building the full matrix (15) (a full matrix can be obtained for undisturbed systems Σ' or assuming $w^{(j)} = 0$ for $j > s$, where s is a non-negative integer), (ii) Both ORC-DF and FISPO can be inconclusive for certain models, and (iii) ORC-DF and FISPO can handle different types of inputs, although both methods allow the study of observability in the presence of generic (measurable) controls.

Remark 5 (The FISPO algorithm does not always require building the full matrix). *In each iteration $k \geq 1$ the FISPO algorithm builds the matrix $\mathcal{O}_l^k(x^k, u)$, composed by extended Lie derivatives of output up to order k , and then calculates its rank and partial ranks, instead of directly building the full matrix (15). The algorithm is programmed in this way because the matrix $\mathcal{O}_l^k(x^k, u)$ can reach full rank for some $k \leq n_s - 1$, and if the number of states increases indefinitely ($s = \infty$), (15) can never be built in practice. In addition, the above procedure may classify some states as observable before obtaining the full matrix (15), since any observable state in the k -augmented system Σ^k remains observable in Σ^l , for $l \geq k$ [13]. Moreover, if the system does not have unknown inputs it is possible to classify it as unobservable or unidentifiable using fewer than $n_{\bar{x}} - 1$ Lie derivatives [19].*

Remark 6 (Both ORC-DF and FISPO can be inconclusive for certain models). *If the model under study has unknown inputs and their time derivatives $w^{(s)}(t)$ do not vanish for any non-negative integer $s < +\infty$, both FISPO and ORC-DF algorithms can be inconclusive. This happens when the rank of the observability matrices grows at each iteration without reaching a value equal to the number of states (which also increases with each iteration). Therefore, a computational implementation of both algorithms should include shutdown conditions based on computation time or number of iterations. This impediment can also be avoided by assuming that $s < \infty$, which implies:*

$$\begin{aligned} x^k &= x^s, \\ f^k &= \left((f^s)^T \quad 0_{1 \times (k-s)n_w} \right)^T, \end{aligned}$$

for any $k > s$, so the observability matrix constructed by FISPO at k -th iteration is:

$$\mathcal{O}^k(x^k, u) = d^k \left(L_{f^k}^0 h(x^k, u) \quad \dots \quad L_{f^k}^k h(x^k, u) \right) = d^s \left(L_{f^s}^0 h(x^s, u) \quad \dots \quad L_{f^s}^k h(x^s, u) \right),$$

while, in the case of affine-in-inputs systems, ORC-DF can be modified at each iteration $k > s$ to account for the above assumption, by taking:

$$\begin{aligned} x^k &= x^s, \\ f_{xw}^k &= \left((f_{xw}^s)^T \quad 0_{1 \times (k-s)n_w} \right)^T, \\ d^k \Omega_k &= d^s \Omega_k. \end{aligned}$$

Actually, ORC-DF is designed to work with unknown inputs such that $w^{(s)}(t)$ are piecewise constant for $s = \infty$ but, obviously, if $w^{(s)}(t) = 0$ for a finite integer s , this condition is trivially verified.

Algorithm 1 The FISPO algorithm [16].**Result:** Observable and unobservable states, parameters and disturbances.

$$k = 0, x^0 = \begin{pmatrix} x^T & \theta^T & w^T \end{pmatrix}^T, n^0 = n_x + n_\theta + n_w, \Lambda^0 = \{1, 2, \dots, n^0\};$$

$$f^0 = \begin{pmatrix} f^T & 0_{1 \times n_\theta} & \dot{w}^T \end{pmatrix}^T;$$

$$\mathcal{O}_I^0 = d^0 h;$$
while $\text{rank}(\mathcal{O}_I^k) < n^k$ **do**
 $k = k + 1;$
 $x^k = \begin{pmatrix} x^{k-1T} & w^k \end{pmatrix}^T, n^k = n^{k-1} + n_w, \Lambda^k = \Lambda^{k-1} \cup \{n^{k-1} + 1, n^{k-1} + 2, \dots, n^k\};$
 $\mathcal{O}_I^k = \begin{pmatrix} \mathcal{O}_I^{k-1} & 0 \\ d^k L_{f^{k-1}}^{e,k} h \end{pmatrix};$
 for $i \in \Lambda^k$ **do**
 $\mathcal{O}_I^{k,i} = \mathcal{O}_I^k - \{i \text{ column}\};$
 if $\text{rank}(\mathcal{O}_I^{k,i}) < \text{rank}(\mathcal{O}_I^k)$ **then**
 x_i is observable;
 $\Lambda^k = \Lambda^k - \{i\};$
 end
 end
 $f^k = \begin{pmatrix} f^{k-1T} & w^{k+1T} \end{pmatrix}^T;$
end

Algorithm 2 The ORC-DF algorithm [15].**Result:** Observable and unobservable states, parameters and disturbances.

$$k = 0, x^0 = \begin{pmatrix} x^T & \theta^T & w^T \end{pmatrix}^T, n^0 = n_x + n_\theta + n_w, \Lambda^0 = \{1, 2, \dots, n^0\};$$

$$f_{xw}^0 = \begin{pmatrix} f_{xw}^T & 0_{1 \times n_\theta} & \dot{w}^T \end{pmatrix}^T, f_{u_i}^0 = \begin{pmatrix} f_{u_i}^T & 0_{1 \times n_\theta} & 0_{1 \times n_w} \end{pmatrix}^T, 1 \leq i \leq n_u;$$

$$\Delta\Omega_0 = \begin{pmatrix} h_{xw}^T & h_{u_1}^T & \dots & h_{u_{n_u}}^T \end{pmatrix}^T, \Omega_0 = \Delta\Omega_0;$$
while $\text{rank}(d^k \Omega_k) < n^k$ **do**
 $k = k + 1;$
 $\Delta\Omega_k = \begin{pmatrix} L_{f_{xw}^{k-1}}(\Omega_{k-1})^T & L_{f_{u_1}^{k-1}}(\Omega_{k-1})^T & \dots & L_{f_{u_{n_u}}^{k-1}}(\Omega_{k-1})^T \end{pmatrix}^T, \Omega_k = \begin{pmatrix} \Omega_{k-1}^T & \Delta\Omega_k^T \end{pmatrix}^T;$
 $x^k = \begin{pmatrix} x^{k-1T} & w^k \end{pmatrix}^T, n^k = n^{k-1} + n_w, \Lambda^k = \Lambda^{k-1} \cup \{n^{k-1} + 1, n^{k-1} + 2, \dots, n^k\};$
 $d^k \Omega_k = \begin{pmatrix} d^{k-1} \Omega_{k-1} & 0 \\ d^k \Delta\Omega_k \end{pmatrix};$
 for $i \in \Lambda^k$ **do**
 $d^k \Omega_k^i = d^k \Omega_k - \{i \text{ column}\};$
 if $\text{rank}(d^k \Omega_k^i) < \text{rank}(d^k \Omega_k)$ **then**
 x_i is k -row observable;
 $\Lambda^k = \Lambda^k - \{i\};$
 end
 end
 $f_{xw}^k = \begin{pmatrix} f_{xw}^{k-1T} & w^k \end{pmatrix}^T, f_{u_i}^k = \begin{pmatrix} f_{u_i}^{k-1T} & 0_{1 \times n_w} \end{pmatrix}^T, 1 \leq i \leq n_u;$
end

Remark 7 (ORC-DF and FISPO can handle different types of inputs). Both ORC-DF and FISPO construct an observation space generated by Lie derivatives of the output; its dimension determines observability. FISPO builds an observation space spanned by extended Lie derivatives (2) considering analytical inputs, while ORC-DF assumes piecewise constant inputs and exploits certain properties specific to affine systems in order to build a different observation space. If an affine system is classified as observable by ORC-DF or FISPO, it is observable when a generic measurable input is considered [15,19].

In the next subsections we present the main novel insights of our theoretical analysis of the algorithms.

3.2. For Systems without Known Inputs, ORC-DF and FISPO Reduce to the Same Algorithm

Here we prove by induction that, if no inputs u are involved in Σ_A , the FISPO algorithm reduces to ORC-DF (note that Σ_A is a particular case of Σ , so FISPO is directly applicable). Before presenting the result, we remark that, in the case $n_u = 0$, the extended Lie derivative reduces to:

$$L_f^e(\cdot) = L_f(\cdot)$$

and denoting the composition of functions with \circ , the $k + 1$ -order Lie derivative verifies:

$$L_{f^k}^{k+1}(h) = L_{f^k} \circ \overbrace{\dots}^{k+1} \circ L_{f^k}(h) = L_{f^k} \circ L_{f^{k-1}} \circ \dots \circ L_{f^0}(h) = L_{f^k} \left(L_{f^{k-1}}^k(h) \right) \quad k \geq 1, \quad (16)$$

since $L_{f^k}^j h$ depends only on time derivatives $w^i(t)$ for $1 \leq i \leq j \leq k + 1$.

Proof. Setting $n_u = 0$ in (5), the dynamics and output of Σ_A are given by:

$$\begin{aligned} f(x(t), \theta, w(t)) &= f_{xw}(x(t), \theta, w(t)) \\ h(x(t), \theta, w(t)) &= h_{xw}(x(t), \theta, w(t)) \end{aligned}$$

Let $k = 0$. By the recursion given in Lemma (1), it is verified that:

$$\Delta\Omega_0 = h_{xw} = h, \quad (17)$$

so the induction hypothesis holds for $k = 0$:

$$d^0\Omega_0 = d^0\Delta\Omega_0 = d^0h = \mathcal{O}_I^0.$$

Consider now any non-negative integer $k \geq 0$ and suppose that the induction hypothesis holds for $0 \leq j \leq k$, then:

$$d^{k+1}\Omega_{k+1} = d^{k+1} \begin{pmatrix} \Omega_k^T & \Delta\Omega_{k+1}^T \end{pmatrix}^T = \begin{pmatrix} d^k\Omega_k & 0 \\ d^{k+1}\Delta\Omega_{k+1} \end{pmatrix} = \begin{pmatrix} \mathcal{O}_I^k & 0 \\ d^{k+1}\Delta\Omega_{k+1} \end{pmatrix}$$

and the result is proven if for every $k \geq 0$ it holds that:

$$\Delta\Omega_{k+1} = L_{f^k}^{e,k+1}h = L_{f^k}^{k+1}h. \quad (18)$$

The above equality is fulfilled for $k \geq 0$. Indeed, for $k = 0$, using (17) we have:

$$\Delta\Omega_1 = L_{f_{xw}^0}(\Delta\Omega_0) = L_{f_{xw}^0}h = L_{f^0}h,$$

so, if $k \geq 0$ and the condition (18) holds for $0 \leq j \leq k-1$, then:

$$\Delta\Omega_{k+1} = L_{f_{xw}^k}(\Delta\Omega_k) = L_{f^k}(\Delta\Omega_k) = L_{f^k}(L_{f^{k-1}}^k h) = L_{f^k}^{k+1} h$$

where in the last equality we have applied (16). Thus, condition (18) holds for $k \geq 0$. \square

3.3. For Systems with Known Inputs, ORC-DF and FISPO Lead to Different Observability Matrices

Excluding the case $n_u = 0$, an important difference between the observability matrices built by algorithms ORC-DF and FISPO is the number of Lie derivatives (rows) they include in each iteration. Indeed, for $k \geq 0$,

$$\mathcal{O}_I^k(x^k, u) \in m(k+1) \times n^k, \quad d^k \Omega^k(x^k) \in \sum_{i=0}^k m(1+n_u)^{i+1} \times n^k,$$

so the observability matrix constructed by FISPO grows in m rows in each iteration, while the matrix constructed by ORC-DF includes $m(1+n_u)^{k+1}$ new rows in the k -th stage. This fact can be an advantage for ORC-DF, as it makes it possible to reach full rank more rapidly, i.e., with lower order Lie derivatives. However, it may also be a disadvantage if this growth makes the problem dimension increase rapidly while adding little new information. Hence the faster growth may be beneficial or not depending on the form of the mathematical expressions of the dynamics and output functions in which the known inputs are present. For example, suppose that there exists an integer $1 \leq i \leq n_u$ and $n_u - 1$ real numbers λ_j not simultaneously zero, such that:

$$f_{u_i} = \sum_{j \neq i=1}^{n_u} \lambda_j f_{u_j},$$

which, using (13), implies:

$$f_{u_i}^k = \sum_{j \neq i=1}^{n_u} \lambda_j f_{u_j}^k, \quad k \geq 0.$$

Since, by definition, it holds that:

$$d^{k+1} \Omega_{k+1} = d^{k+1} \left(\Omega_k^T \quad \Delta\Omega_{k+1}^T \right)^T = d^{k+1} \left(\Omega_k^T \quad L_{f_{xw}^k}(\Delta\Omega_k)^T \quad L_{f_{u_1}^k}(\Delta\Omega_k)^T \quad \cdots \quad L_{f_{u_{n_u}}^k}(\Delta\Omega_k)^T \right)^T,$$

the matrix built by ORC-DF algorithm in the $(k+1)$ -th iteration includes $m(1+n_u)^{k+1}$ dependent rows; the rows forming $d^{k+1} L_{f_{u_i}^k}(\Delta\Omega_k)$ can be written as a linear combination of the remaining rows. Indeed, by a property of Lie derivative [18] it holds that:

$$L_{f_{u_i}^k}(\Delta\Omega_k) = d^k \Delta\Omega_k f_{u_i}^k = d^k \Delta\Omega_k \left(\sum_{j \neq i=1}^{n_u} \lambda_j f_{u_j}^k \right) = \sum_{j \neq i=1}^{n_u} \lambda_j d^k \Delta\Omega_k f_{u_j}^k = \sum_{j \neq i=1}^{n_u} \lambda_j L_{f_{u_j}^k}(\Delta\Omega_k),$$

so, using linearity of the derivative, the following linear combination has been obtained:

$$d^{k+1} L_{f_{u_i}^k}(\Delta\Omega_k) = d^{k+1} \left(\sum_{j \neq i=0}^{n_u} \lambda_j L_{f_{u_j}^k}(\Delta\Omega_k) \right) = \sum_{j \neq i=0}^{n_u} \lambda_j d^{k+1} L_{f_{u_j}^k}(\Delta\Omega_k) \in m(1+n_u)^{k+1} \times n^k. \quad (19)$$

Likewise, if there exists $1 \leq i \leq n_u$ such that h_{u_i} is linearly dependent on vector fields h_{u_j} for $1 \leq j \leq n_u, j \neq i$, the observability matrix built by ORC-DF includes $m(1+n_u)^{k+1}$ dependent rows in the $k+1$ -th iteration, for $k \geq 0$.

Note that the real values λ_j can be replaced by functions of unknown parameters, $\lambda_j = \lambda_j(\theta)$, as they are constant variables, so the equality (19) holds for this case as well.

4. Implementation

4.1. The STRIKE-GOLDD Software Toolbox

STRIKE-GOLDD (Structural identifiability taken as extended-generalized observability using Lie derivatives and decomposition) is an open source MATLAB toolbox that analyses the observability of nonlinear systems of the form Equations (1) and (2). It is available at <https://sites.google.com/site/strikegolddtoolbox/> and <https://github.com/afvillaverde/strike-goldd/>. STRIKE-GOLDD versions up to v2.1.6 implemented the FISPO algorithm, including a number of additional features that go beyond the core instructions described in Algorithm 1, with the purpose of facilitating the analysis of large models. Furthermore, they also allowed one to indicate a given number of non-zero time derivatives of inputs, both known and unknown.

4.2. Implementation of the ORC-DF Algorithm

We have released a new version of STRIKE-GOLDD (v2.2) that includes an implementation of ORC-DF (Algorithm 2) along with the already existing implementation of FISPO (Algorithm 1). The algorithm is chosen with the newly introduced option `opts.affine` in the `options.m` file (set it to 1 for ORC-DF, and to 0 for FISPO). The `ORC_DF.m` function checks whether a model is indeed affine in the inputs, and if that is the case, converts it to the appropriate form Σ_A (3) and (4) and stores it in a mat-file to avoid repeating this calculation in the future. Thus, the user only needs to enter the model once, using the same format for ORC-DF and FISPO. New specific options for the ORC-DF algorithm include the possibility of setting a maximum number of iterations through the variable `opts.kmax`, limiting the computation time of each stage with `opts.tStage`, plotting graphs of the results obtained by the algorithm with `opts.graphics`, and using the MATLAB Parallel Toolbox by setting the option `opts.parallel` to 1.

4.3. Multiple Experiments and Piecewise Constant Inputs

FISPO analyses the observability of a model for a single experiment with an infinitely differentiable ("smooth") input. However, it is possible to use it to consider multiple experiments with possibly different inputs by applying it to a modified model: if we create as many replicates of the model states, inputs, and outputs as the number of experiments that we want to consider, we obtain a new model whose analysis for a single input has the same observability properties as the original model with multiple inputs [20]. Until now, this feature was only available in the GenSSI 2.0 toolbox [10]. We have included the possibility of carrying out this multi-experiment analysis automatically (using any of the available algorithms) in the new version of STRIKE-GOLDD, by setting the option `opts.multiexp=1`. The number of experiments can be chosen with `opts.numexp`, and a set of initial conditions can be chosen by the user through the `options.m` file.

5. Computational Results and Discussion

We have applied the ORC-DF and FISPO algorithms to a set of illustrative case studies from different areas of science and technology, ranging from civil engineering to different biological disciplines. They are listed in Table 1, along with the computation times of the algorithms.

Table 1. Computation times of the two algorithms for the models analysed in this study. The computation times of case studies with unknown inputs depend on the highest order of the derivatives of the unknown inputs that are assumed to be non-zero, s . Three different cases are shown for those models: $s = \{0, 1, 5\}$. For models without unknown inputs this setting does not apply. Cases in which an algorithm cannot be applied are labelled as N/A. Results were obtained on a personal computer with 16 GB RAM and processor Intel(R) Core(TM) i7-8550U 1.80 GHz.

Model	Section	Reference	#x	#θ	#u	#w	Total Computation Time [s]					
							FISPO			ORC-DF		
							s = 0	s = 1	s = 5	s = 0	s = 1	s = 5
C2M	5.1	[16]	2	4	1	0		0.47			0.41	
Bolie	5.2	[21]	2	5	1	0		1.42			0.59	
2DOF	5.3	[15]	4	3	1	1	0.51	1.26	5.68	0.80	0.89	1.54
HIV	5.4	[22]	3	5	1	0		0.42			0.44	
			3	5	0	1	0.42	0.43	57.8	1.14	1.29	47.1
TS	5.5	[23]	2	10	2	0		99.9			N/A	
			2	6	0	2	1.47	36.2	>10 ⁴		N/A	
JAK-STAT	5.6	[24]	25	26	5	0		Table 2			Table 2	

5.1. An Identifiable and Observable Model with Known Input: "C2M"

Our first case study is a deceptively simple compartmental model [20]:

$$\begin{cases} \dot{x}_1 = -(k_{1e} + k_{12})x_1 + k_{21}x_2 + bu, \\ \dot{x}_2 = k_{12}x_1 - k_{21}x_2, \\ y = x_1, \end{cases}$$

where each state x_i ($i = 1, 2$) corresponds to a compartment, and $\theta = (k_{1e}, k_{12}, k_{21}, b)$ is the unknown parameter vector. The augmented state vector is $\bar{x} = (x_1, x_2, k_{1e}, k_{12}, k_{21}, b)$, with extended dynamics given by:

$$\bar{f}(\bar{x}, u) = f_{xw}(\bar{x}) + f_u(\bar{x})u = \begin{pmatrix} \dot{x}_1 & \dot{x}_2 & 0 & 0 & 0 & 0 \end{pmatrix}^T,$$

with the following vector fields for the affine-in-inputs formulation (5):

$$\begin{aligned} f_{xw}(\bar{x}) &= f_{xw}(x, \theta) = \begin{pmatrix} -(k_{1e} + k_{12})x_1 + k_{21}x_2 & k_{12}x_1 - k_{21}x_2 & 0 & 0 & 0 & 0 \end{pmatrix}^T, \\ f_u(\bar{x}) &= f_u(x, \theta) = \begin{pmatrix} b & 0 & 0 & 0 & 0 & 0 \end{pmatrix}^T. \end{aligned}$$

In addition, the output is given by the function:

$$y = h(x) = h_{xw}(x) = x_1.$$

The results obtained by ORC-DF and FISPO are shown in Figure 1, where the first important difference between both methods can be seen: while FISPO works with analytical, time-varying controls $u(t)$ (which implies that it depends on functions $u(t)$ throughout their time derivatives; see Equations (9) and (10)), ORC-DF assumes piecewise constant inputs. For this reason, the results obtained by FISPO vary according to the number of non-zero $u(t)$ derivatives, while those of ORC-DF do not. Considering different types of inputs can be of interest in biological applications, where it is sometimes experimentally impossible to apply sufficiently exciting signals. Due to its reduced size, this model is well suited for illustrating this difference, so we derive the equations of the Lie derivatives calculated by each algorithm in Appendix A, where we also discuss other aspects and implications.

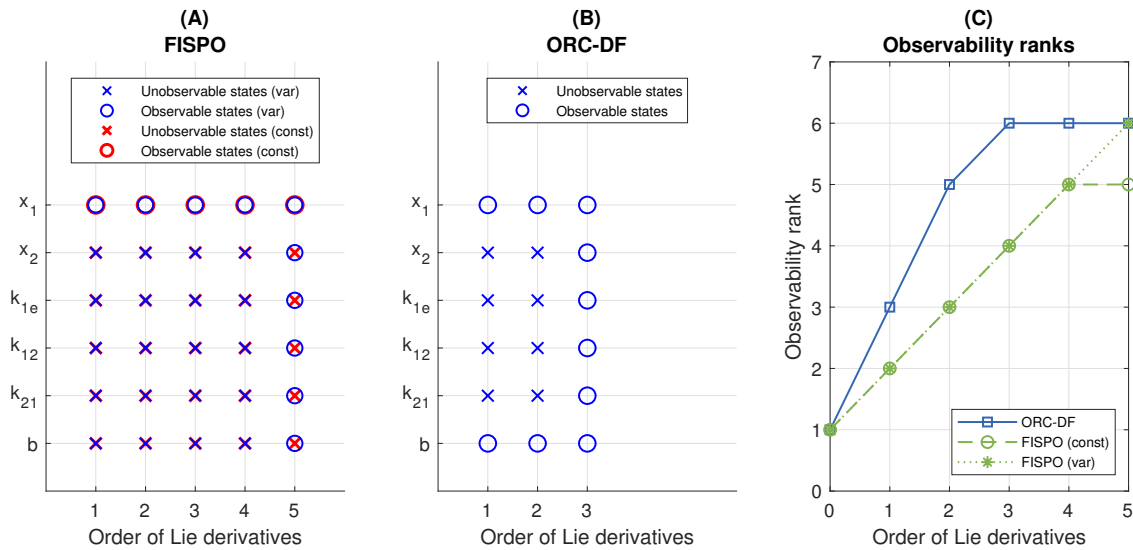


Figure 1. Analysis of the C2M model with the FISPO and ORC-DF algorithm. For FISPO two cases are considered: $\dot{u} = 0$, which is labelled as "(const)", and $\dot{u} \neq 0$, which is labelled as "(var)". In the first case, higher order derivatives are assumed to be zero too. (A,B) Results of the FISPO and ORC-DF algorithms, respectively: the panel shows the states classified as observable or unobservable as a function of the number of Lie derivatives calculated by each algorithm. (C) Observability rank obtained by each algorithm as a function of the number of Lie derivatives. The full rank is equal to the number of states, i.e., six.

This model is classified as observable and identifiable by ORC-DF after three iterations. The result yielded by FISPO depends on the number of input derivatives assumed to be zero: the unmeasured variables are classified as unobservable with a constant input, while they become observable in the fifth iteration if the input is any non-constant analytical function. The variables classified as observable by both algorithms at each iteration are illustrated in Figure 1A,B. Figure 1C shows the ranks of the matrices built by both algorithms in each iteration. As can be seen, the observability matrix constructed by ORC-DF reaches full rank after considering Lie derivatives up to order three. The matrix built by FISPO stagnates from the fourth iteration onward with a constant input, while it reaches full rank after five iterations with a non-constant input.

5.2. A Non-Identifiable, Non-Observable Model with Known Inputs: "Bolie"

Our second example is a model with similarities to the previous one, given by [21]:

$$\begin{cases} \dot{q}_1 = p_1 q_1 - p_2 q_2 + u, \\ \dot{q}_2 = p_4 q_1 + p_3 q_2, \\ y = q_1 / V_p, \end{cases}$$

where $x = (q_1, q_2)$ is the state vector, $\theta = (p_1, p_2, p_3, p_4, V_p)$ are the unknown parameters, and u is the control variable. The augmented state vector is $\bar{x} = (q_1, q_2, p_1, p_2, p_3, p_4, V_p)$, and the extended dynamics are given by:

$$\bar{f}(\bar{x}, u) = f_{xw}(\bar{x}) + f_u(\bar{x})u = \begin{pmatrix} \dot{q}_1 & \dot{q}_2 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}^T,$$

and can be separated into the vector fields:

$$f_{xw}(\bar{x}) = f_{xw}(x, \theta) = \begin{pmatrix} p_1 q_1 - p_2 q_2 & p_4 q_1 + p_3 q_2 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}^T,$$

$$f_u(\bar{x}) = f_u(x, \theta) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}^T.$$

The output is a function of the state q_1 and the unknown parameter V_p :

$$h(x, \theta) = h_{xw}(x, \theta) = q_1 / V_p,$$

so, in this case, there are no directly measured states or parameters.

The model is classified as non-identifiable and non-observable by FISPO and ORC-DF, as shown in Figure 2. A detailed analysis of the calculations performed by both algorithms is provided in Appendix B.

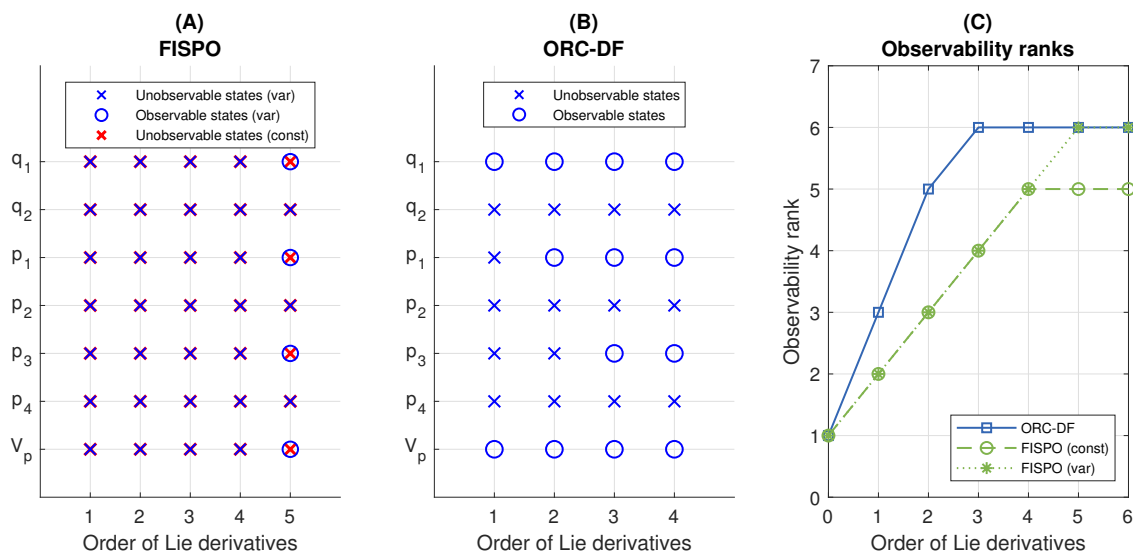


Figure 2. Analysis of the Bolie model with the FISPO and ORC-DF algorithms. For FISPO two cases are considered: $\dot{u} = 0$, which is labelled as “(const)”, and $\dot{u} \neq 0$, which is labelled as “(var)”. In the first case, higher order derivatives are assumed to be zero too. (A,B) Results of the FISPO and ORC-DF algorithms, respectively: the panel shows the states classified as observable or unobservable as a function of the number of Lie derivatives calculated by each algorithm. (C) Observability rank obtained by each algorithm as a function of the number of Lie derivatives. The full rank is equal to the number of states, i.e., seven.

5.3. A Model with Known and Unknown Inputs: “2DOF”

We consider now an affine-in-the-inputs model with a known and an unknown input, proposed by [15]. It describes the behaviour of a mechanical system consisting of two masses connected by a spring. In the form (1) and (2), its dynamics and output functions are given by:

$$f(x, \theta, u, w) = \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ (- (k_1 + \delta k_1 x_1) x_1 + k_2 (x_2 - x_1) - c_1 \dot{x}_1 + c_2 (\dot{x}_2 - \dot{x}_1) + F_1) / m_1 \\ (k_2 (x_1 - x_2) + c_2 (\dot{x}_1 - \dot{x}_2) + F_2) / m_2 \end{pmatrix},$$

$$h(x, \theta, w) = h_{xw}(x, \theta, w) = \begin{pmatrix} x_1 & (k_2 (x_1 - x_2) + c_2 (\dot{x}_1 - \dot{x}_2) + F_2) / m_2 \end{pmatrix}^T.$$

The state vector is $x = (x_1, x_2, \dot{x}_1, \dot{x}_2)$ and the unknown parameters are $\theta = (k_1, \delta k_1, m_2)$. Two external forces act on the system as inputs, one of known magnitude, $u(t) = F_1(t)$, and another of unknown value, $w(t) = F_2(t)$. The remaining parameters k_2, m_1, c_1 and c_2 are known.

Since there is an unknown input acting on the model, it is necessary to include its time derivatives in the extended state vector. The 0-augmented state is $x^0 = (x_1, x_2, \dot{x}_1, \dot{x}_2, k_1, \delta, k_1, m_2, w)$, which follows the dynamics:

$$\dot{x}^0 = f^0(x^0, u, w) = f_{xw}^0(x^0, w) + f_u^0(\bar{x})u = \begin{pmatrix} f(x, \theta, u, w)^T & 0 & 0 & 0 & \dot{w} \end{pmatrix}^T,$$

where the contribution of the known input is:

$$f_u^0(\bar{x}) = \begin{pmatrix} 0 & 0 & 1/m_1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}^T.$$

First we consider the case in which the unknown disturbance $w(t)$ is assumed constant, $\dot{w}(t) = 0$. The results are shown in Figure 3. After calculating three Lie derivatives both algorithms conclude that the system is identifiable, observable and invertible. It should be noted that for this model FISPO always leads to the result shown in Figure 3A, regardless of the number of known input derivatives assumed to be non-zero (note that the figure shows the constant case $\dot{w}(t) = 0$, but this property is fulfilled for any analytical disturbance $w(t)$).

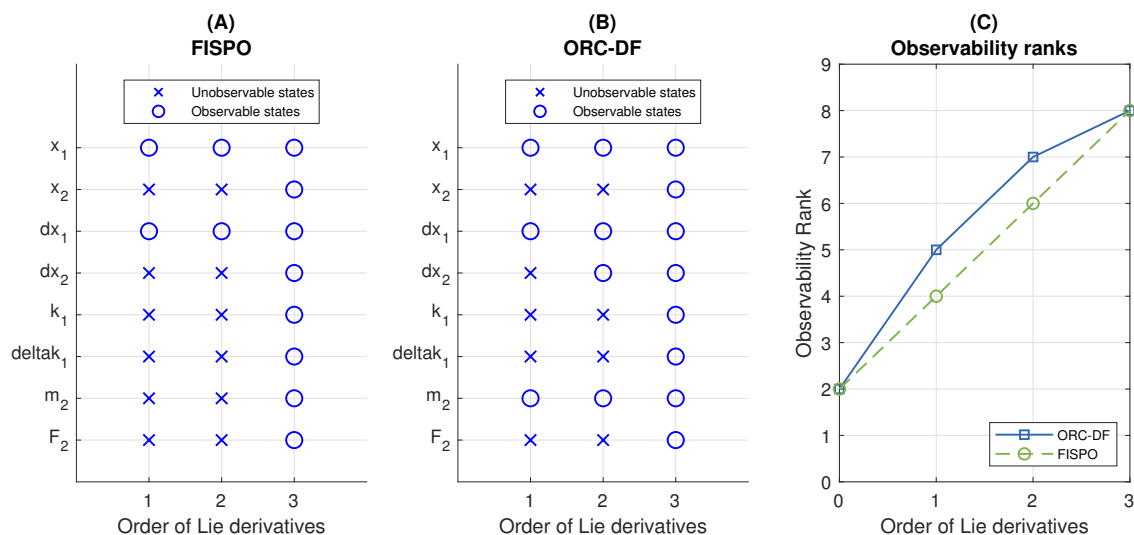


Figure 3. Analysis of the 2DOF model. (A,B) Results of the FISPO and ORC-DF algorithms, respectively, with $\dot{w} = 0$. The panels show the states classified as observable or unobservable as a function of the number of Lie derivatives calculated by each algorithm. (C) Observability rank obtained by each algorithm as a function of the number of Lie derivatives. The full rank is equal to the number of states, i.e., eight.

Next, we consider a time-varying unknown input, assuming that $w^{(s)} = 0$ for some $s > 1$. For this case, the model is again classified as fully observable by both algorithms. However, the paths that they follow to reach that conclusion are different. The number of Lie derivatives required by FISPO to classify the model as observable increases as s grows, due to the number of states in each stage also increasing without reaching full rank, so FISPO does not reach any conclusion when $s = +\infty$. In contrast, ORC-DF ends at most in four iterations, regardless of the value of $s \geq 1$ (including the case $s = \infty$). This situation is illustrated in Figure 4, which shows the number of Lie derivatives required by each algorithm to achieve a result for $0 \leq s \leq 10$.

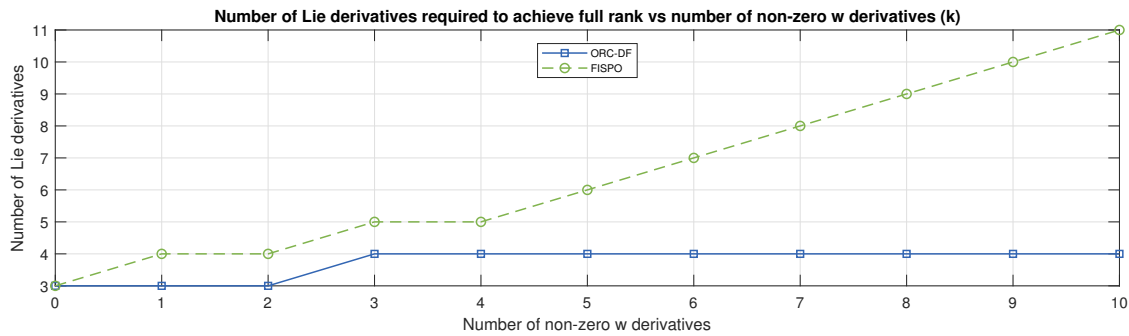


Figure 4. Number of Lie derivatives needed by each algorithm for building the full-rank observability matrix for the 2DOF model as a function of s , the number of non-zero $w(t)$ time derivatives, for $0 \leq s \leq 10$.

A difference between the procedures carried out by both algorithms for this model is that—similarly to the case of parameter b in the C2M example, mentioned in Appendix A—the expressions obtained by ORC-DF determine that the unknown parameter m_2 can be calculated directly from the measurements since the first iteration as:

$$L_{f_u} h(x, \theta, u, w) = c_2 / (m_1 m_2).$$

In the case of FISPO, in contrast, the parameter m_2 is the last to be classified as identifiable, which happens at the same time in which the entry $w(t)$ is classified as invertible for a sufficiently large s ($s \geq 5$).

5.4. A Model with a Known or Unknown Input: "HIV"

Next we consider a model of HIV dynamics in the human body given by [22]:

$$\begin{cases} \dot{T}_U = \lambda - \rho T_U - \eta T_U V, \\ \dot{T}_I = \eta T_U V - \delta T_I, \\ \dot{V} = N \delta T_I - c V, \\ y_1 = V, \\ y_2 = T_I + T_U, \end{cases}$$

where the states are $x = (T_U, T_I, V)$, the unknown parameters vector is $\theta = (\lambda, \rho, \delta, N, c)$, and $\eta(t)$ is a time-varying input, the infection rate.

As was established in Section 3.2, if $\eta(t)$ is unknown, ORC-DF and FISPO become the same algorithm (leaving aside implementation details). If $\eta(t)$ is known and time-varying, they differ.

This model was analysed with FISPO by [16] considering two possibilities, i.e., $\eta(t)$ known and unknown. In both cases the model is classified as observable and identifiable by FISPO. In the latter case, the number of Lie derivatives necessary to achieve this conclusion grows with the number of derivatives of $\eta(t)$ assumed to be non-zero, as happened with the 2DOF model analysed in Section 5.3.

It is possible to analyse the HIV model with the ORC-DF algorithm, since it is affine in input. With the infection rate considered known, i.e., $u(t) = \eta(t)$, the functions of the affine formulation (13) and (14) are written as:

$$\begin{aligned} f_{xw}(x, \theta) &= \begin{pmatrix} \lambda - \rho T_U & -\delta T_I & N \delta T_I - c V \end{pmatrix}^T, \\ f_u(x, \theta) &= \begin{pmatrix} -T_U V & T_U V & 0 \end{pmatrix}^T, \\ h_{xw}(x, \theta) = h(x, \theta) &= \begin{pmatrix} V & T_I + T_U \end{pmatrix}^T, \end{aligned}$$

while if it is unknown, $w(t) = \eta(t)$, we have:

$$\begin{aligned} f_{xw}(x, \theta, w) &= f(x, \theta, w), \\ h_{xw}(x, \theta, w) &= h(x, \theta, w). \end{aligned}$$

If the infection rate is considered known, both algorithms classify the model as observable and identifiable, regardless of the number of input derivatives assumed non-zero by FISPO. Figure 5 illustrates this fact, where the results obtained by FISPO have been represented for a generic analytical input.

The observability matrix calculated by ORC-DF has full rank after including Lie derivatives up to second order, so the number of its rows is $\sum_{i=0}^2 m(1+n_u)^{i+1} = \sum_{i=0}^2 2^{i+2} = 26$ (actually, 14 rows after excluding dependent rows arising from the equality $h_u = 0$) while the matrix constructed by FISPO needs to include Lie derivatives up to order three to achieve full rank, so it has $m(k+1) = 8$ rows.

This is an example of a model for both algorithms perform similarly; although FISPO needs to calculate one more Lie derivative to classify the system as observable, ORC-DF calculates ranks of matrices of greater dimension, resulting in similar computational cost of the calculations involved in each algorithm. As can be seen in Figure 5C, the ranks of both observability matrices coincide up to the first iteration, as a consequence of:

$$h_u(x, \theta) = L_{f_u} h(x, \theta) = \begin{pmatrix} 0 & 0 \end{pmatrix}^T.$$

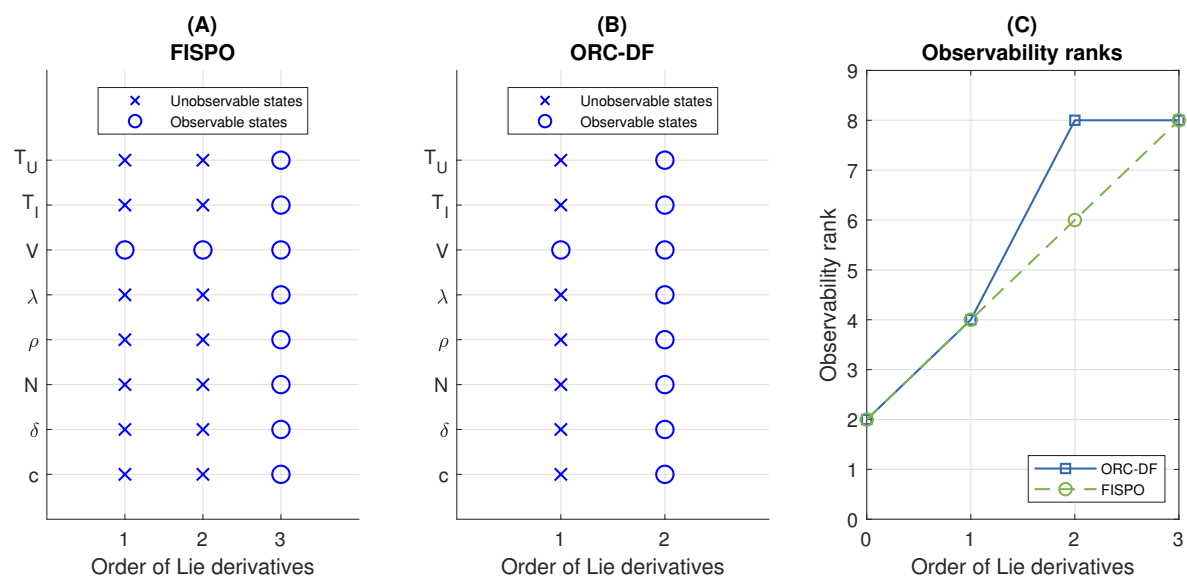


Figure 5. Analysis of the HIV model with the FISPO and ORC-DF algorithms, with the input $\eta(t)$ considered known. (A,B) Results of the FISPO and ORC-DF algorithms, respectively: the panel shows the states classified as observable or unobservable as a function of the number of Lie derivatives calculated by each algorithm. (C) Observability rank obtained by each algorithm as a function of the number of Lie derivatives. The full rank is equal to the number of states, i.e., eight.

5.5. A Genetic Toggle Switch with Two Inputs: "TS"

Let us now consider the following model of a genetic toggle switch [23]:

$$\begin{cases} \dot{x}_1 = k_{01} + \frac{k_1}{1 + (x_2 / (1 + (aTc / \theta_{aTc})^{\eta_{aTc}}))^{\eta_{TetR}}} - x_1, \\ \dot{x}_2 = k_{02} + \frac{k_2}{1 + (x_1 / (1 + (IPTG / \theta_{IPTG})^{\eta_{IPTG}}))^{\eta_{LacI}}} - x_2, \\ y_1 = x_1, \\ y_2 = x_2, \end{cases}$$

where $x = (x_1, x_2)$ is the state vector and the inputs are $aTc(t)$ and $IPTG(t)$. The remaining variables are unknown parameters.

This model is an example that cannot be analysed by ORC-DF algorithm, since it is not affine in inputs. It was analysed with FISPO in [16], considering both measured and unmeasured inputs. If both inputs are known, FISPO classifies the model as structurally identifiable, as long as neither input is constant. If the inputs are unknown FISPO concludes that some parameters become unidentifiable. For more details we refer the reader to [16].

5.6. A Signalling Pathway with Five Known Inputs: "JAK-STAT"

To show the computational limitations of the two algorithms, we analyse here a model that pushes them to their limits. It is a classic model of the JAK-STAT signalling pathway presented by [24], which has 25 states, 26 unknown parameters, and 5 inputs. The output consists on 15 measured functions of the model variables that depend only on one of the external signals u_i , which is not involved in system dynamics, that is,

$$f_{u_i}(x, \theta) = 0_{25 \times 1} \quad (20)$$

$$h_{u_j}(x, \theta) = 0_{15 \times 1}, \quad 1 \leq j \leq 5, j \neq i \quad (21)$$

The model equations are provided in Appendix C.

This model was analysed with FISPO in [25], concluding that all its parameters are structurally identifiable but two of its 25 states are non-observable. The calculations are computationally expensive, requiring the use of procedures supported in STRIKE-GOLDD—such as model decomposition or successive executions after removing parameters previously classified as identifiable—in order to reach the conclusion. Thus, the model was first analysed after setting the maximum computation time of each Lie derivative to 100 seconds, which allowed FISPO to calculate 5 Lie derivatives and to classify 17 parameters and 4 states as observable. Next, the 17 parameters were specified as previously classified in the FISPO options, thereby removing them from further consideration and decreasing the size of the problem, and the model was decomposed. The post-decomposition analysis classified five additional parameters as identifiable. After removing them, it was possible to analyse the remainder of the model and reach the aforementioned conclusion.

With ORC-DF we did not manage to analyse the model due to computational limitations (specifically, insufficient memory). The different computational requirements of ORC-DF and FISPO are shown in Table 2.

Table 2. Results and computation times of ORC-DF and FISPO for the JAK-STAT model.

	ORC-DF					FISPO				
Iteration	1	2	3	4	5	1	2	3	4	5
Number of rows	180	930	4680	23430	117180	30	45	60	75	90
Rank	24	35	40	42	44	20	28	34	39	43
Rank computation time [s]	0.54	2.18	15.36	170.95	3600.54	0.40	0.76	3.46	59.55	362.78
Observable variables	14	21	29	29	29	8	12	15	15	21

Table 2 shows that the number of rows of the matrix built by ORC-DF grows rapidly at each iteration (even though the implementation removes null rows arising from dependencies in (20) and (21), which is why the number of rows of the ORC-DF matrix does not match the number given in Section 3.3). Although this matrix leads to higher ranks than the one built by FISPO, especially at the beginning of the execution (i.e., with few Lie derivatives), the difference decreases soon and the ranks of the two matrices are similar despite the big difference in the number of rows. With five Lie derivatives, ORC-DF labels 27 model variables as observable (9 states and 18 parameters), including those classified as observable by FISPO (21: 4 states and 17 parameters). However, the computation time of ORC-DF at that point is roughly ten times higher than FISPO, and memory requirements impede further progress with this algorithm.

6. Conclusions

In this paper we have analysed two recent algorithms for observability analysis of nonlinear systems with known and/or unknown inputs, which we refer to as ORC-DF and FISPO. Our analyses have revealed the key similarities and differences between them. The main conclusions can be summarized as follows.

First, we have proven theoretically that for models *without known inputs* both algorithms are basically equivalent, since they calculate the rank of the same observability matrix. In contrast, for models with known inputs—e.g., for controlled systems—the two algorithms differ, since they build different observability matrices. Specifically, the number of rows of the matrix built by ORC-DF increases more at each iteration than the one built by FISPO. We have shown that this increased growth is often an advantage of ORC-DF, since it makes it possible to reach full rank—and thus, to conclude that a model is observable—with less Lie derivatives, and hence less computational cost; an example was shown in Section 5.1. However, said increased growth is not always advantageous: as we have noted in Section 3.3, it can also be detrimental to the efficiency of ORC-DF. The latter situation may happen when the structure of the model equations is such that the increase in problem dimension outweighs the increase in information resulting from the inclusion of a new Lie derivative; we provided an example in Section 5.6.

When applying FISPO to a model with unknown input(s) it is usually necessary to assume that their derivatives, $w^{(s)}$, are zero for orders higher than a finite s , which amounts to assuming that the unknown inputs are polynomial functions. This is not a theoretical requirement—and in fact, a counter-example that did not require this assumption was shown in [16]—but it is often necessary in practice in order to reach a conclusion in finite time. This was indeed the case for the models with unknown inputs that we analysed in this paper. It should be noted that the results obtained under this assumption are not necessarily valid if the unknown input has nonzero derivatives of order higher than s .

We remark that this assumption of polynomial inputs does not apply to known inputs; FISPO does not require this assumption for their analysis, and it is not used by ORC-DF either. That being said, it must be taken into account that in certain applications there are experimental limitations that do not allow one to use sophisticated input signals for system identification. This situation is common in biological modelling, where often times only constant or ramp inputs can be applied. The implications of such limitations in observability can be taken into account by specifying that the derivatives of the known inputs, $u^{(s)}$, are zero for orders higher than a finite s . This option is available in STRIKE-GOLDD.

Another difference between both algorithms lies in the types of models and known inputs that they can analyse. In regard to *model types*, FISPO is applicable to a general class of nonlinear ODE models, while ORC-DF is applicable to a subclass of those models: the ones that are affine in the inputs. There is an additional, albeit subtle, difference between both algorithms in regard to the *known inputs*: FISPO considers infinitely differentiable ("smooth") functions, while ORC-DF considers piecewise constant inputs. It should be noted however that an affine system that is observable for

piecewise constant inputs is also observable for smooth inputs; therefore, ORC-DF can also establish the observability of (affine) models with continuous inputs.

In a new release of the STRIKE-GOLDD toolbox (v2.2) we have included an implementation of the ORC-DF algorithm, thus allowing the user to apply different algorithms with the same tool and model definition. It should be noted that the implementations of the FISPO and ORC-DF algorithms included in the STRIKE-GOLDD toolbox have a number of additional features that increase the efficiency of the core algorithms analysed here, as noted in Section 4. We used the new implementations in STRIKE-GOLDD 2.2 to benchmark the algorithms with several models taken from the literature. Our selection of case studies included both simple models, used for illustrating the inner workings of the algorithms in detail, and more complex models whose analysis is computationally challenging, which we used for pushing the algorithms to their limits. We also provided an example of a model that cannot be analysed with ORC-DF due to being not affine in the inputs.

In conclusion, the theoretical and computational analyses presented here have informed us about the differences between the ORC-DF and FISPO algorithms, showing that they represent complementary techniques for solving an often challenging problem, and clarifying when one may be preferred over the other. The release of a new version of the MATLAB toolbox STRIKE-GOLDD that includes implementations of both algorithms provides the convenience of performing different analyses with minimal intervention from the user.

Author Contributions: A.F.V. designed and supervised the research; N.M. implemented the software and performed the computational experiments; N.M. and A.F.V. analysed the algorithms and the results; N.M. and A.F.V. wrote the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Spanish Ministry of Science, Innovation and Universities through the project SYN BIOCONTROL (reference DPI2017-82896-C2-2-R). We acknowledge support of the publication fee by the CSIC Open Access Publication Support Initiative through its Unit of Information Resources for Research (URICI) and by the Xunta de Galicia through grant ref. IN607B 2020-03.

Conflicts of Interest: The authors declare no competing interests.

Data and Materials Availability: The methods and models used in this paper are available in the GitHub repository as part of release v2.2 of the STRIKE-GOLDD toolbox: <https://github.com/afvillaverde/strike-goldd>.

Appendix A. Analysis of the Lie Derivatives of the C2M Case Study

Let us first consider the FISPO algorithm and the model with a non-constant input. As shown in Figure 1C, in this case there are six independent extended Lie derivatives of the output, which are given by:

$$L_f^0 h(x) = x_1, \quad (\text{A1})$$

$$L_f h(x) = -(k_{1e} + k_{12})x_1 + k_{21}x_2 + bu, \quad (\text{A2})$$

$$L_f^2 h(x) = \left((k_{1e} + k_{12})^2 + k_{21}k_{12} \right) x_1 - (k_{1e} + k_{12} + k_{21})k_{21}x_2 - (k_{1e} + k_{12})bu + b\ddot{u}, \quad (\text{A3})$$

$$L_f^3 h(x) = - \left((k_{1e} + k_{12})^3 + k_{12}k_{21}(k_{1e} + k_{12}) + k_{12}k_{21}(k_{1e} + k_{12} + k_{21}) \right) x_1 + (k_{1e} + k_{12} + k_{21})k_{21}^2x_2 + \left((k_{1e} + k_{12})^2 + k_{12}k_{21} \right) k_{21}x_2 + \left((k_{1e} + k_{12})^2 + k_{21}k_{12} \right) bu - (k_{1e} + k_{12})b\ddot{u} + b\ddot{u}, \quad (\text{A4})$$

$$L_f^4 h(x) = \left((k_{1e} + k_{12})^4 + 2k_{12}k_{21}(k_{1e} + k_{12})^2 + k_{12}k_{21}(k_{1e} + k_{12} + k_{21})^2 + k_{12}^2k_{21}^2 \right) x_1 - \left((k_{1e} + k_{12})^3 + 2k_{12}k_{21}(k_{1e} + k_{12}) + 2k_{12}k_{21}^2 + k_{21}(k_{1e} + k_{12})^2 + k_{21}^2(k_{1e} + k_{12} + k_{21}) \right) k_{21}x_2 - \left((k_{1e} + k_{12})^3 + 2k_{12}k_{21}(k_{1e} + k_{12}) + k_{12}k_{21}^2 \right) bu + \left((k_{1e} + k_{12})^2 + k_{21}k_{12} \right) b\ddot{u} - b(k_{1e} + k_{12})\ddot{u} + bu^3, \quad (\text{A5})$$

$$L_f^5 h(x) = - \left((k_{1e} + k_{12})^5 + 3k_{12}k_{21}(k_{1e} + k_{12})^3 + 3k_{12}^2k_{21}^2(k_{1e} + k_{12}) + k_{12}k_{21}^2(k_{1e} + k_{12})^2 + 2k_{12}^2k_{21}^2 \right) x_1 + \left(k_{12}k_{21}(k_{1e} + k_{12})(k_{1e} + k_{12} + k_{21})^2 + k_{12}k_{21}^3(k_{1e} + k_{12} + k_{21}) \right) x_1 + (k_{1e} + k_{12})^4 k_{21}x_2 + \left(k_{21}(k_{1e} + k_{12})^2(2k_{12} + k_{21}) + k_{12}k_{21}(k_{1e} + k_{12} + k_{21})^2 + k_{21}^3(k_{1e} + k_{12} + k_{21}) \right) k_{21}x_2 + \left(k_{21}(k_{1e} + k_{12})^3 + 2k_{12}k_{21}^2(k_{1e} + k_{12}) + 2k_{12}k_{21}^3 + k_{12}^2k_{21}^2 + 3k_{12}^2k_{21}^2(k_{1e} + k_{12}) \right) k_{21}x_2 + \left((k_{1e} + k_{12})^4 + 2k_{12}k_{23}(k_{1e} + k_{12})^2 + k_{12}^2k_{21}^2 + k_{12}k_{21}(k_{1e} + k_{12} + k_{21})^2 \right) bu + (k_{1e} + k_{12})^3 b\ddot{u} + \left(2k_{12}k_{21}(k_{1e} + k_{12}) + k_{12}k_{21}^2 \right) b\ddot{u} + \left((k_{1e} + k_{12})^2 + k_{21}k_{12} \right) b\ddot{u} - b(k_{1e} + k_{12})u^3 + bu^4. \quad (\text{A6})$$

Assuming (for simplicity) that $u^{(k)} = 0$ for $k > 1$, and writing $L_f^i h = y^{(i)}$ ($1 \leq i \leq 5$), from (A1)–(A6) we obtain:

$$x_1 = y, \quad (\text{A7})$$

$$bu = y' + (k_{1e} + k_{12})y - k_{21}x_2, \quad (\text{A8})$$

$$k_{21}^2x_2 = k_{12}k_{21}y - (k_{1e} + k_{12})y' - y'' + b\ddot{u}, \quad (\text{A9})$$

$$k_{21}b\ddot{u} = y''' + k_{1e}k_{21}y' + (k_{1e} + k_{12} + k_{21})y'', \quad (\text{A10})$$

$$k_{1e}k_{21}y'' = -y^{(4)} - (k_{1e} + k_{12} + k_{21})y''', \quad (\text{A11})$$

$$(k_{1e} + k_{12} + k_{21})(y''' - y^{(4)}) = y^{(5)}y'' - y^{(4)}y'''. \quad (\text{A12})$$

Therefore, from the above system we can extract directly the following input–output expressions:

$$x_1 = \phi_0(y, y', \dots, y^{(5)}, u, \dot{u}), \quad (\text{A13})$$

$$k_{1e} + k_{12} + k_{21} = \phi_1(y, y', \dots, y^{(5)}, u, \dot{u}), \quad (\text{A14})$$

$$k_{1e}k_{21} = \phi_2(y, y', \dots, y^{(5)}, u, \dot{u}), \quad (\text{A15})$$

$$k_{21}b = \phi_3(y, y', \dots, y^{(5)}, u, \dot{u}), \quad (\text{A16})$$

where ϕ_i ($0 \leq i \leq 3$) are functions that depend only on the output, the input, and their time derivatives. It is possible to obtain similar input–output expressions for any variable involved in the model by determining the unique solution of the system (A7)–(A12), which consists of six independent equations and six unknowns. Note that the assumption $u^{(k)} = 0$ for $k > 1$ does not imply loss of generality, because the presence of higher order derivatives does not affect the unicity of the solution of (A1)–(A6) (if anything, they would contribute to the independence of the equations).

By inspecting Equations (A7)–(A12) it is possible to explain the classification obtained by FISPO and shown in Figure 1A (blue line), since it is not possible to obtain such an input–output expression

for any unmeasured variable until calculating the fifth Lie derivative (A6), when the ambiguities in the equations are finally resolved. (Note that this result requires excluding those states and parameters in the phase space for which the denominators in the input–output expressions vanish. However, since the system is analytical these states form a zero measurement subset.)

Let us consider now the FISPO algorithm in the constant input case. From Figure 1C it follows that the equations obtained (A1)–(A6) are dependent. The following system of equations is extracted from (A1)–(A5):

$$x_1 = y, \quad (\text{A17})$$

$$k_{21}x_2 = y' + (k_{1e} + k_{12})y - bu, \quad (\text{A18})$$

$$k_{21}bu = y'' + k_{1e}k_{21}y + (k_{1e} + k_{12} + k_{21})y', \quad (\text{A19})$$

$$k_{1e}k_{21}y' = -y''' - (k_{1e} + k_{12} + k_{21})y'', \quad (\text{A20})$$

$$(k_{1e} + k_{12} + k_{21})(y''^2 - y'''y') = y'y^{(4)} - y''y'''. \quad (\text{A21})$$

Using Equations (A17) and (A19–A21), it is possible to write the combinations $x_1, k_{21}b, k_{1e} + k_{12} + k_{21}$ and $k_{1e}k_{21}$ exclusively in terms of the input and output of the model. Thus, after replacing in (A18), the system to be solved is:

$$x_1 = \phi_0(y, y', \dots, y^{(4)}, u), \quad (\text{A22})$$

$$k_{1e} + k_{12} + k_{21} = \phi_1(y, y', \dots, y^{(4)}, u), \quad (\text{A23})$$

$$k_{1e}k_{21} = \phi_2(y, y', \dots, y^{(4)}, u), \quad (\text{A24})$$

$$k_{21}b = \phi_3(y, y', \dots, y^{(4)}, u), \quad (\text{A25})$$

$$k_{21}x_2 = y' + (\phi_1(y, y', \dots, y^{(4)}, u) - k_{21})y - \phi_3(y, y', \dots, y^{(4)}, u)u/k_{21}, \quad (\text{A26})$$

which has six unknowns and five independent equations (infinite solutions), so it is not possible to write (at least, locally) any of the parameters or the unknown state x_2 as a function of the input and output. This scenario is shown in Figure 1A (red line).

We consider now the ORC-DF algorithm. Instead of (A1)–(A6), this method computes the following Lie derivatives (independent of the control values, since they are assumed to be piecewise constant):

$$L_0 = h(x) = x_1, \quad (\text{A27})$$

$$L_1 = L_{f_{xw}}h(x) = -(k_{1e} + k_{12})x_1 + k_{21}x_2, \quad (\text{A28})$$

$$L_2 = L_{f_u}h(x) = b, \quad (\text{A29})$$

$$L_3 = L_{f_{xw}}^2h(x) = ((k_{1e} + k_{12})^2 + k_{21}k_{12})x_1 - (k_{1e} + k_{12} + k_{21})k_{21}x_2, \quad (\text{A30})$$

$$L_4 = L_{f_u}L_{f_{xw}}h(x) = -b(k_{1e} + k_{12}), \quad (\text{A31})$$

$$L_5 = L_{f_{xw}}^3h(x) = -((k_{1e} + k_{12})^3 + 2k_{21}k_{12}(k_{1e} + k_{12}) + k_{21}^2k_{12})x_1 + ((k_{1e} + k_{12})^2 + k_{21}^2 + k_{21}(k_{1e} + 2k_{12}))k_{21}x_2, \quad (\text{A32})$$

$$L_6 = L_{f_u}L_{f_{xw}}^2h(x) = b((k_{1e} + k_{12})^2 + k_{21}k_{12}). \quad (\text{A33})$$

The system (A27)–(A33) has a unique solution, in which one of the equations depends on the others. The solution, obtained from (A27)–(A31) and (A33), is given as a function of Lie derivatives L_i , $i \in \{0, 1, 2, 3, 4, 6\}$, as follows:

$$\begin{aligned}x_1 &= L_0, \\x_2 &= (L_1 L_2 - L_0 L_4)^2 / (L_2 (L_0 L_6 - L_2 L_3) + L_4 (L_1 L_2 - L_0 L_4)), \\k_{1e} &= L_2 (L_3 L_4 - L_1 L_6) / (L_2 (L_0 L_6 - L_2 L_3) + L_4 (L_1 L_2 - L_0 L_4)), \\k_{12} &= (L_1 L_2 - L_0 L_4) (L_2 L_6 - L_4^2) / (L_2 (L_0 L_6 - L_2 L_3) + L_4 (L_1 L_2 - L_0 L_4)), \\k_{21} &= (L_2 (L_0 L_6 - L_2 L_3) + L_4 (L_1 L_2 - L_0 L_4)) / (L_1 L_2^2 - L_0 L_2 L_4), \\b &= L_2.\end{aligned}$$

Therefore, the ORC-DF algorithm classifies the C2M model as observable and identifiable after the third iteration. We note that Equation (A29) implies that parameter b can be calculated directly from the output.

Appendix B. Analysis of the Lie Derivatives of the Bolie Case Study

Assuming non-constant input, FISPO calculates six independent Lie derivatives of the output, as shown in Figure 2C:

$$L_f^0 h(x, \theta, u) = q_1 / V_p, \quad (\text{A34})$$

$$L_f^1 h(x, \theta, u) = (p_1 q_1 - p_2 q_2 + u) / V_p, \quad (\text{A35})$$

$$L_f^2 h(x, \theta, u) = ((p_1^2 - p_2 p_4) q_1 - (p_1 + p_3) p_2 q_2 + p_1 u + \dot{u}) / V_p, \quad (\text{A36})$$

$$\begin{aligned}L_f^3 h(x, \theta, u) &= ((p_1(p_1^2 - p_2 p_4) - p_2 p_4(p_1 + p_3)) q_1 - (p_1^2 - p_2 p_4 + p_3(p_1 + p_3)) p_2 q_2) / V_p + \\& \quad ((p_1^2 - p_2 p_4) u + p_1 \dot{u} + \ddot{u}) / V_p,\end{aligned} \quad (\text{A37})$$

$$\begin{aligned}L_f^4 h(x, \theta, u) &= (((p_1^2 - p_2 p_4)^2 - p_2 p_4(p_1 + p_3)^2) q_1 + (p_1 + p_3)(p_1^2 - 2p_2 p_4 + p_3^2) p_2 q_2) / V_p + \\& \quad ((p_1^3 - 2p_1 p_2 p_4 - p_2 p_3 p_4) u - ((p_1^2 - p_2 p_4) \dot{u} + p_1 \ddot{u} + \ddot{u})) / V_p,\end{aligned} \quad (\text{A38})$$

$$\begin{aligned}L_f^5 h(x, \theta, u) &= ((p_1^2(p_1 - p_2 p_4)^2 - p_2 p_4(p_1 + p_3)(2p_1^2 + p_3^2 - p_2 p_4)) q_1 - (p_1 - p_2 p_4)^2 p_2 q_2) / V_p - \\& \quad ((p_3(p_1 + p_3)(p_1^2 - 2p_2 p_4 + p_3^2) - p_2 p_4(p_1 + p_3)^2) p_2 q_2 + (p_1^2 - p_2 p_4)^2 u) / V_p + \\& \quad (-p_2 p_4(p_1 + p_3)^2 u + (p_1^3 - 2p_1 p_2 p_4 - p_2 p_3 p_4) \dot{u} + (p_1^2 - p_2 p_4) \ddot{u} + p_1 u^{(3)} + u^{(4)}) / V_p.\end{aligned} \quad (\text{A39})$$

System (A34)–(A39) has infinite solutions, since it is composed by six equations and seven unknowns, but similarly to the C2M case study, it is possible to perform a series of algebraic manipulations on the above equations to obtain an input–output expression for the variables q_1 , p_1 , p_3 and V_p , as it shown in Figure 2.

In the case of constant input, Equations (A34)–(A39) become redundant, so there are only five independent Lie derivatives, as is shown in Figure 2. The combinations $p_1 + p_3$, $p_1 p_3 + p_2 p_4$ are the only observable functions of the variables that can be extracted from (A34)–(A38). Since the rational combinations of these functions are insufficient to determine any of the parameters or states, all variables are non-observable, as shown in Figure 2A (red line).

On the other hand, ORC-DF calculates the following Lie derivatives:

$$h(x, \theta) = q_1 / V_p, \quad (\text{A40})$$

$$L_{f_{xw}} h(x, \theta) = (p_1 q_1 - p_2 q_2) / V_p, \quad (\text{A41})$$

$$L_{f_u} h(x, \theta) = 1 / V_p, \quad (\text{A42})$$

$$L_{f_{xw}}^2 h(x, \theta) = \left((p_1^2 - p_2 p_4) q_1 - (p_1 + p_3) p_2 q_2 \right) / V_p, \quad (\text{A43})$$

$$L_{f_u} L_{f_{xw}} h(x, \theta) = p_1 / V_p, \quad (\text{A44})$$

$$L_{f_{xw}}^3 h(x, \theta) = \left((p_1(p_1^2 - p_2 p_4) - (p_1 + p_3) p_2 p_4) q_1 - ((p_1^2 - p_2 p_4) + p_3(p_1 + p_3)) p_2 q_2 \right) / V_p, \quad (\text{A45})$$

$$L_{f_u} L_{f_{xw}}^2 h(x, \theta) = (p_1^2 - p_2 p_4) / V_p. \quad (\text{A46})$$

From Equations (A40), (A42), and (A44) it is easy to obtain the state q_1 and the parameters V_p and p_1 uniquely from the measurements, using Lie derivatives up to order two. This property is not fulfilled by the system formed by (A34)–(A39); Figure 2A shows that the aforementioned variables are classified as observable by FISPO only after considering fifth order derivatives. Using the input–output expressions of q_1 , V_p , and p_1 extracted from (A40), (A42) and (A44), in conjunction with Equations (A41), (A43) and (A46), it is also possible to determine parameter p_3 as a function of the Lie derivatives of the output. However, Figure 2C shows that system (A40)–(A46) contains one redundant equation, so it is not possible to determine uniquely an input–output expression of the remaining unmeasured states. It can also be noted that any rational combination of the observable variables with the functions $p_2 q_2$ and $p_2 p_4$ is also observable.

Appendix C. Equations of the JAK-STAT Model

The dynamics of the JAK-STAT model analysed in Section 5.6 is given by:

$$\begin{aligned} \dot{x}_1 &= x_{2345} x_8 \theta_{11} / \theta_{26} - k_5 x_1 \theta_{10} / M_1, \\ \dot{x}_2 &= k_5 x_1 \theta_{10} / M_1 - x_2 \theta_7 / M_1 - x_2 x_8 \theta_{11} / \theta_{26} - 3 x_2 \theta_7 / ((\theta_8 x_6 + 1) M_1), \\ \dot{x}_3 &= \theta_7 x_2 / M_1 - \theta_{11} x_8 x_3 / \theta_{26} - 3 \theta_7 x_3 / ((\theta_8 x_6 + 1) M_1), \\ \dot{x}_4 &= 3 x_2 \theta_7 / ((\theta_8 x_6 + 1) M_1) - \theta_7 x_4 / M_1 - \theta_{11} x_8 x_4 / \theta_{26}, \\ \dot{x}_5 &= \theta_7 x_4 / M_1 - \theta_{11} x_8 x_5 / \theta_{26} + 3 \theta_7 x_3 / ((\theta_8 x_6 + 1) M_1), \\ \dot{x}_6 &= -x_6 (\theta_9 / \theta_{25}) (x_5 + x_3), \\ \dot{x}_7 &= \theta_{13} x_8 - x_7 (\theta_{12} / \theta_{25}) x_{2345}, \\ \dot{x}_8 &= x_7 (\theta_{12} / \theta_{25}) x_{2345} - \theta_{13} x_8, \\ \dot{x}_9 &= k_6 \theta_{23} x_{11} / k_7 - x_9 (\theta_{22} / \theta_{25}) x_{2345} / M_1 - x_9 \theta_{21} (x_5 + x_3)^2 / ((x_{18} \theta_3 / \theta_1 + 1) M_1 \theta_{25}^2), \\ \dot{x}_{10} &= x_9 \theta_{22} x_{2345} M_1 / \theta_{25} - \theta_{24} x_{10} + x_9 \theta_{21} (x_5 + x_3)^2 / (\theta_{25}^2 (x_{18} \theta_3 / \theta_1 + 1) M_1), \\ \dot{x}_{11} &= k_7 \theta_{24} x_{10} / k_6 - \theta_{23} x_{11}, \\ \dot{x}_{12} &= -x_{12} \theta_4 - \theta_5 x_{11} (k_1 - 1) / \theta_{27}, \\ \dot{x}_{13} &= x_{12} \theta_4 - x_{13} \theta_4, \\ \dot{x}_{14} &= x_{13} \theta_4 - x_{14} \theta_4, \\ \dot{x}_{15} &= x_{14} \theta_4 - x_{15} \theta_4, \\ \dot{x}_{16} &= x_{15} \theta_4 - x_{16} \theta_4, \\ \dot{x}_{17} &= x_{16} \theta_4 k_6 / k_7 - x_{17} \theta_5, \\ \dot{x}_{18} &= x_{17} \theta_1 \theta_6 - x_{18} \theta_6 + k_2 \theta_6 \theta_2 \theta_1, \\ \dot{x}_{19} &= -x_{19} \theta_{18} - \theta_{19} x_{11} (k_1 - 1) / \theta_{27}, \\ \dot{x}_{20} &= x_{19} \theta_{18} - x_{20} \theta_{18}, \\ \dot{x}_{21} &= x_{20} \theta_{18} - x_{21} \theta_{18}, \\ \dot{x}_{22} &= x_{21} \theta_{18} - x_{22} \theta_{18}, \end{aligned}$$

$$\begin{aligned}
 \dot{x}_{23} &= x_{22}\theta_{18} - x_{23}\theta_{18}, \\
 \dot{x}_{24} &= k_6 x_{23}\theta_{18}/k_7 - x_{24}\theta_{19}, \\
 \dot{x}_{25} &= x_{24}\theta_{15}\theta_{20} - x_{25}\theta_{20} + k_3\theta_{20}\theta_{16}\theta_{15}
 \end{aligned}$$

where the auxiliary variables $x_{2345} = x_2 + x_3 + x_4 + x_5$ and $M_1 = x_{25}\theta_{17}/\theta_{15} + 1$ have been used.

The 25 states x_1, x_2, \dots, x_{25} are, respectively, the following species: EpoRJAK2, EpoRpJAK2, p1EpoRpJAK2, p2EpoRpJAK2, p12EpoRpJAK2, EpoRJAK2_CIS, SHP1, SHP1Act, STAT5, pSTAT5, npSTAT5, CISnRNA1, CISnRNA2, CISnRNA3, CISnRNA4, CISnRNA5, CISRNA, CIS, SOCS3nRNA1, SOCS3nRNA2, SOCS3nRNA3, SOCS3nRNA4, SOCS3nRNA5, SOCS3RNA, and SOCS3.

The 27 unknown parameters, θ_i , were written in the original publication [24] as: CISEqc, CISEqcOE, CISInh, CISRNADelay, CISRNATurn, CISTurn, EpoRActJAK2, EpoRCISInh, EpoRCISRemove, JAK2ActEpo, JAK2EpoRDeaSHP1, SHP1ActEpoR, SHP1Dea, SHP1ProOE, SOCS3Eqc, SOCS3EqcOE, SOCS3Inh, SOCS3RNADelay, SOCS3RNATurn, SOCS3Turn, STAT5ActEpoR, STAT5ActJAK2, STAT5Exp, STAT5Imp, init_EpoRJAK2, init_SHP1, and init_STAT5.

The model has seven known constants (k_1 – k_7), five of which correspond to experimental conditions that can be considered as constant inputs (k_1 – k_5), including the external signal ($k_5 \equiv \text{Epo}$).

The output equations are:

$$\begin{aligned}
 y_1 &= 2(x_2 + x_3 + x_4 + x_5)\theta_{25}, \\
 y_2 &= 16(x_3 + x_4 + x_5)\theta_{25}, \\
 y_3 &= x_{18}\theta_1, \\
 y_4 &= x_{25}/\theta_{14}, \\
 y_5 &= (x_9 + x_{10})/\theta_{27}, \\
 y_6 &= x_{10}\theta_{27}, \\
 y_7 &= x_9, \\
 y_8 &= x_7 + x_8, \\
 y_9 &= x_{18}, \\
 y_{10} &= x_{25}, \\
 y_{11} &= 100x_{10}/(x_{10} + x_9), \\
 y_{12} &= x_{24}, \\
 y_{13} &= x_{17}, \\
 y_{14} &= (x_7 + x_8)(1 + (k_4\theta_{27}))/\theta_{26},
 \end{aligned}$$

References

1. Chatzis, M.N.; Chatzi, E.N.; Smyth, A.W. On the observability and identifiability of nonlinear structural and mechanical systems. *Struct. Control Health Monit.* **2015**, *22*, 574–593. [\[CrossRef\]](#)
2. Villaverde, A.F. Observability and Structural Identifiability of Nonlinear Biological Systems. *Complexity* **2019**, *2019*, 8497093. [\[CrossRef\]](#)
3. Tuza, Z.A.; Ács, B.; Szederkényi, G.; Allgöwer, F. Efficient computation of all distinct realization structures of kinetic systems. *IFAC-PapersOnLine* **2016**, *49*, 194–200. [\[CrossRef\]](#)
4. Hermann, R.; Krener, A.J. Nonlinear controllability and observability. *IEEE Trans. Autom. Control* **1977**, *22*, 728–740. [\[CrossRef\]](#)
5. Bellman, R.; Åström, K.J. On structural identifiability. *Math. Biosci.* **1970**, *7*, 329–339. [\[CrossRef\]](#)
6. Bellu, G.; Saccomani, M.P.; 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*, 52–61. [\[CrossRef\]](#) [\[PubMed\]](#)
7. Meshkat, N.; Eisenberg, M.; DiStefano, J.J. An algorithm for finding globally identifiable parameter combinations of nonlinear ODE models using Gröbner Bases. *Math. Biosci.* **2009**, *222*, 61–72. [\[CrossRef\]](#) [\[PubMed\]](#)

8. Karlsson, J.; Anguelova, M.; Jirstrand, M. An Efficient Method for Structural Identifiability Analysis of Large Dynamic Systems. In Proceedings of the 16th IFAC Symposium on System Identification, Brussels, Belgium, 11–13 July 2012; Volume 16, pp. 941–946.
9. Villaverde, A.F.; Barreiro, A.; Papachristodoulou, A. Structural identifiability of dynamic systems biology models. *PLoS Comput. Biol.* **2016**, *12*, e1005153. [[CrossRef](#)] [[PubMed](#)]
10. Ligon, T.S.; Fröhlich, F.; Chiş, O.T.; Banga, J.R.; Balsa-Canto, E.; Hasenauer, J. GenSSI 2.0: Multi- experiment structural identifiability analysis of SBML models. *Bioinformatics* **2018**, *8*, 1421–1423. [[CrossRef](#)] [[PubMed](#)]
11. Hong, H.; Ovchinnikov, A.; Pogudin, G.; Yap, C. SIAN: A tool for assessing structural identifiability of parametric ODEs. *ACM Commun. Comput. Algebra* **2019**, *53*, 37–40. [[CrossRef](#)]
12. Evans, N.D.; Chapman, M.J.; Chappell, M.J.; Godfrey, K.R. Identifiability of uncontrolled nonlinear rational systems. *Automatica* **2002**, *38*, 1799–1805. [[CrossRef](#)]
13. Martinelli, A. Extension of the observability rank condition to nonlinear systems driven by unknown inputs. In Proceedings of the 2015 23rd Mediterranean Conference on Control and Automation (MED), Torremolinos, Spain, 16–19 June 2015; pp. 589–595.
14. Martinelli, A. Nonlinear Unknown Input Observability: Extension of the Observability Rank Condition. *IEEE Trans. Autom. Control* **2019**, *64*, 222–237. [[CrossRef](#)]
15. Maes, K.; Chatzis, M.; Lombaert, G. Observability of nonlinear systems with unmeasured inputs. *Mech. Syst. Signal Process.* **2019**, *130*, 378–394. [[CrossRef](#)]
16. Villaverde, A.F.; Tsiantis, N.; Banga, J.R. Full observability and estimation of unknown inputs, states, and parameters of nonlinear biological models. *J. R. Soc. Interface* **2019**, *16*, 20190043. [[CrossRef](#)] [[PubMed](#)]
17. Vidyasagar, M. *Nonlinear Systems Analysis*; Prentice Hall: Englewood Cliffs, NJ, USA, 1993.
18. Isidori, A. *Nonlinear Control Systems*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 1995.
19. Anguelova, M. Nonlinear observability and identifiability: General theory and a case study of a kinetic model for *S. cerevisiae*. Master's Thesis. Chalmers University of Technology and Göteborg University, Göteborg, Sweden, 2004.
20. Villaverde, A.F.; Evans, N.D.; Chappell, M.J.; Banga, J.R. Input-Dependent Structural Identifiability of Nonlinear Systems. *IEEE Control Syst. Lett.* **2019**, *3*, 272–277. [[CrossRef](#)]
21. Bolie, J. Coefficients of normal blood glucose regulation. *J. Appl. Physiol.* **1961**, *16*, 783–788. [[CrossRef](#)] [[PubMed](#)]
22. Miao, H.; Xia, X.; Perelson, A.S.; Wu, H. On identifiability of nonlinear ODE models and applications in viral dynamics. *SIAM Rev.* **2011**, *53*, 3–39. [[CrossRef](#)] [[PubMed](#)]
23. Lugagne, J.B.; Carrillo, S.S.; Kirch, M.; Köhler, A.; Batt, G.; Hersen, P. Balancing a genetic toggle switch by real-time feedback control and periodic forcing. *Nat. Commun.* **2017**, *8*, 1671. [[CrossRef](#)] [[PubMed](#)]
24. Bachmann, J.; Raue, A.; Schilling, M.; Böhm, M.E.; Kreutz, C.; Kaschek, D.; Busch, H.; Gretz, N.; Lehmann, W.D.; Timmer, J.; et al. Division of labor by dual feedback regulators controls JAK2/STAT5 signaling over broad ligand range. *Mol. Syst. Biol.* **2011**, *7*, 516. [[CrossRef](#)] [[PubMed](#)]
25. Villaverde, A.F.; Banga, J.R. Análisis de observabilidad e identificabilidad estructural de modelos no lineales: aplicación a la vía de señalización JAK/STAT. In *XL Jornadas de Automática. Universidade da Coruña*; Servizo de Publicacións—UDC: Coruña, Spain, 2019; pp. 631–638.

