



HAL
open science

Nonlinear unknown input observability and unknown input reconstruction: The general analytical solution

Agostino Martinelli

► **To cite this version:**

Agostino Martinelli. Nonlinear unknown input observability and unknown input reconstruction: The general analytical solution. *Information Fusion*, 2022, 85, pp.23-51. 10.1016/j.inffus.2022.03.004 . hal-03920679

HAL Id: hal-03920679

<https://inria.hal.science/hal-03920679>

Submitted on 26 Jan 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial - NoDerivatives 4.0 International License

Nonlinear Unknown Input Observability and Unknown Input Reconstruction: The General Analytical Solution

Agostino Martinelli

^aUniv. Grenoble Alpes, INRIA, Avenue de l'Europe, Grenoble, F-38000, FRANCE

Abstract

Observability is a fundamental structural property of any dynamic system and describes the possibility of reconstructing the state that characterizes the system from fusing the observations of its inputs and outputs. Despite the effort made to study this property and to introduce analytical criteria capable of verifying whether or not a dynamic system satisfies this property, there is no general analytical criterion to obtain the observability of the state when the dynamics are also driven by unknown inputs. Here, we introduce the general analytical solution of this fundamental open problem, often called the unknown input observability problem. We provide the systematic procedure, based on automatic calculation (differentiation and determination of the matrix rank), which allows us to check the observability of the state even in the presence of unknown inputs. One of the fundamental ingredients to obtain this solution is the characterization of the group of invariance of observability. We have very recently introduced this group, together with a new set of tensor fields with respect to this group of transformations [31]. The analytical solution of the unknown input observability problem is expressed in terms of these tensor fields. In [31] we provided the solution by restricting our investigation to systems that satisfy a special assumption that is called *canonicity with respect to the unknown inputs*. Here, after an exhaustive characterization of the concept of canonicity, we also account for the case when this assumption is not satisfied and we provide the general solution. This solution is also provided in the form of a new algorithm. In addition, even in the canonic case dealt with in [31], here we provide a new fundamental result that regards the convergence properties of the solution. Finally, as a consequence of the results obtained here, we also provide the condition to reconstruct the unknown inputs, and, when this condition is not met, what can be reconstructed on the unknown inputs. We illustrate the implementation of the new algorithm by studying the observability properties of a nonlinear system in the framework of visual-inertial sensor fusion, whose dynamics are driven by two unknown inputs and one known input. In particular, for this system, we follow step by step the algorithm introduced by this paper, which solves the unknown input observability problem in the most general case.

Keywords: Nonlinear observability; Unknown Input Observability; Observability Rank Condition; Unknown Input Reconstruction; Visual-Inertial Sensor Fusion

1. Introduction

Observability refers to the state that characterizes a dynamic system (e.g., if the system is an aerial drone, its state, under suitable conditions, can be its position and its orientation). A system is also characterized by one or more inputs, which drive its dynamics and one or more outputs (e.g., for the drone, the inputs could be the speeds of its rotators and the outputs the ones provided by an on-board monocular camera and/or a GPS). A state is observable if the fusion of knowledge of system inputs and outputs, during a given time interval, allows us its determination. In other words, observability is a fundamental prerequisite that a sensor fusion problem must satisfy.

The concept of observability was first introduced for linear systems [1, 2] and the analytic condition to check if a linear system satisfies this property has also been obtained. The nonlinear case is much more complex. First, this concept becomes local, i.e., the state can be observable in a given region of the space of the states and unobservable in other regions. Second, in general, in the nonlinear case we cannot distinguish states

that are not close¹. This is the reason why, for nonlinear systems, the concept of *weak local observability* has been introduced [4]. In this paper, with the word observability we actually mean the concept of weak local observability, as defined in [4, 5] (definitions 8, 9, 10, 11, in [5]). Third, unlike the linear case, observability depends on the system inputs.

The analytical condition to check if a nonlinear system satisfies this property has also been introduced [4, 5, 6, 7]. It is known as the *observability rank condition*. The observability rank condition is a fundamental result that has extensively been used in many application domains, ranging from computer vision (e.g., [8]), robotics (e.g., [9, 10, 11]), calibration (e.g., [12]), mechanical engineering (e.g., [13]) up to biology (e.g., [14, 15, 16, 17, 18]) and chemistry (e.g., [19]). It has also been used in the context of estimation theory [20, 21, 22]. It is a

¹This difference between the linear and the nonlinear case should not surprise. The observability concept is strongly related to the inverse function problem [3]. A nonlinear function, even when its Jacobian is nonsingular, can take the same value at two or more separated points and, consequently, cannot be inverted. Invertibility becomes a local property.

simple systematic procedure that allows us to give an answer to the previous fundamental question, i.e., whether the state is observable or not. It is based on very simple systematic computation (differentiation and matrix rank determination) on the functions that describe the system. The observability rank condition can deal with any system, independently of its complexity and type of nonlinearity. It works automatically, i.e., it is a systematic procedure that can be performed without supervision (it only requires to perform differentiation and to compute matrix ranks).

On the other hand, the observability rank condition presents an important limitation: it does not account for the presence of unknown inputs. This is a severe limitation. The dynamics of most real systems are driven by inputs that are usually unknown. This holds in robotics, in biology, in chemistry, in physics, in economics, etc. For instance, in the case of our drone, its dynamics could be driven also by the wind, which is in general unknown. The wind is a disturbance and acts on the dynamics as an (unknown) input. The problem of unknown input observability (i.e., the problem of obtaining the analytical and automatic criterion that extends the observability rank condition to the case when some of the inputs are unknown) was defined long time ago [23, 24] and remained unsolved for half a century. The control theory community has spent a huge effort to design observers for both linear and nonlinear systems in the presence of unknown inputs, in many cases in the context of fault diagnosis. In this framework, interesting conditions for the existence of an unknown input observer were introduced. On the other hand, these conditions refer to a restricted class of systems. In particular, the considered systems are often characterized by linearity (or some specific type of nonlinearity) with respect to the state in some of the functions that characterize the dynamics and/or the system outputs². No condition refers to any type of nonlinearity in the aforementioned functions. This is the case of a recent work [25], where the observability analysis adopts a linear model (see Equation (3) in [25]). Note that the observability rank condition in [4, 7] can deal with any nonlinear model. However, as we mentioned, this condition cannot be used in the presence of unknown inputs. The solution of the unknown input observability problem is the extension of the observability rank condition to the unknown input case, i.e., the analytic condition able to provide the state observability in the presence of unknown inputs for any nonlinear model.

Recently, this problem has been approached by introducing an extended state that includes the original state together with the unknown inputs and their time derivatives up to a given order [26, 27, 28, 29]. All these works proposed automatic iterative algorithms able to study the observability properties of nonlinear systems driven by also unknown inputs. On the other hand, all these algorithms suffer from the following fundamental limitations:

- They do not converge, automatically. In particular, at each iterative step, the state is extended by including new time

²These are the functions that appear in (1). Specifically, for the dynamics, these functions are $g^0(x), g^1(x), \dots, g^{m_w}(x), f^1(x), \dots, f^{m_u}(x)$. For the outputs, they are $h_1(x), \dots, h_p(x)$.

derivatives of the unknown inputs. Consequently, if the extended state is observable at a given step, convergence is achieved. However, if this were not the case, we can never exclude that, at a later step, the extended state becomes observable. Therefore, in the presence of unobservability, all these algorithms remain inconclusive.

- Due to the previous state augmentation, the computational burden becomes prohibitive after a few steps.

Both of these drawbacks have recently been overcome in [30] and in [31]. The solutions proposed in [30] and in [31] do not need to extend the state and, as a result, do not encounter the above limitations. On the other hand, these solutions are not general. In particular, the solution introduced in [30] has a very limited range of applicability: it can only be applied to systems without drift and in the presence of a single unknown input. In other words, this solution applies only to systems characterized by (1) with a null g^0 and with $m_w = 1$. The solution introduced in [31] applies only to systems that satisfy a fundamental assumption, called *canonicity with respect to the unknown inputs*. In addition, both the solutions introduced in [30] and [31] present a further important limitation. These solutions were provided in the form of algorithms (Algorithm 3 in [30] and Algorithm 8.3 in [31], respectively). The criteria that were provided to check the convergence of these algorithms do not hold always. In particular, the validity of the stopping criterion provided in [30] is limited to the systems that satisfy the condition given by Equation (31) in [30]. The problem is that, not only there are systems that do not meet this condition but, more importantly, checking if a system satisfies this condition can be very difficult (in [32], some examples of systems that violate Equation (31) in [30] are shown, together with a general convergence criterion that holds for driftless systems and with a single unknown input). The same holds for Algorithm 8.3 in [31]³.

This paper provides the general analytical solution of the unknown input observability problem. In addition, it also provides the condition to reconstruct the unknown inputs, and, when this condition is not satisfied, what can be reconstructed on the unknown inputs. These answers are automatically obtained by the usage of an algorithm that converges in a finite number of steps.

One of the ingredients to achieve this solution is the introduction of a group of transformations that must preserve observability. The concept of observability is invariant under several transformations, in addition to any coordinates change that we are allowed to operate to characterize the state⁴. We have very

³In this case, the analogous of Equation (31) in [30] is not provided. The condition that ensures the validity of the convergence criterion proposed in [31] coincides with the condition that ensures the possibility of inverting Equation (8.67) in [31]. Note that the case treated by Lemma 8.11 is not the only exception in which the invertibility of Equation (8.67) does not hold. There are other exceptions that are not taken into account in [31].

⁴Clearly, we are free to characterize the state of a system by choosing the coordinates (e.g., Cartesian, spheric, etc). The result of the observability analysis must be independent of this choice. This is automatically ensured by the fact that the observability rank condition is based on Lie derivatives which are scalar functions with respect to any coordinate change.

recently introduced this group, together with a new set of tensor fields with respect to this group of transformations [31]. The analytical solution of the unknown input observability problem is expressed in terms of these tensor fields.

This paper is organized as follows. In section 2 we provide the general characterization of the systems here investigated, together with some reminders on basic algebraic operations. Section 3 provides a new set of definitions to characterize the complex concept of canonicity with respect to the unknown inputs. This new characterization allows us to achieve the solution of the unknown input observability problem even for systems not canonic and not even canonizable. Specifically, we provide the new definitions of *unknown input reconstructability matrix from a set of scalar functions* and *unknown input degree of reconstructability*. Based on them, we introduce the definitions of *canonic system with respect to the unknown inputs* and *canonical form with respect to the unknown inputs* (Definition 3 and Definition 4, respectively).

Section 4 provides the solution for systems that are in canonical form with respect to their unknown inputs. The solution is given in the form of an algorithm (Algorithm 1). This solution differs from the one introduced in [31], which is reported here (Algorithm 4). The advantage of the new algorithm consists in a simpler criterion for verifying its convergence and in the fact that this criterion always holds. Section 4, together with the results derived in Appendix A, and Appendix B, proves the validity of the new algorithm (Theorem 1). In Sections 5, 6, and 7, we remove the assumption that the system is in canonical form with respect to its unknown inputs. We provide the automatic procedure that, in a finite number of steps, provides all the observability properties in any case. This automatic procedure is Algorithm 9. It uses iteratively Algorithm 8, which is a special implementation of Algorithm 1. This new algorithm is a fundamental new result that provides the complete solution to the problem⁵. Sections 5, 6, and 7, together with the results derived in Appendix C, Appendix D, and Appendix E, provide all the theoretical foundation for the analytical derivation of Algorithm 9 and all the steps to prove its validity and generality. Section 8 provides a possible use of Algorithm 9 to solve a problem strongly interconnected with the problem of state observability in the presence of unknown inputs. This is the problem of unknown input reconstruction. Section 9 illustrates the implementation of Algorithm 9 by studying the observability properties of a nonlinear system in the framework of visual-inertial sensor fusion. The dynamics of this system are driven by two unknown inputs and one known input and they are also characterized by a nonlinear drift. The system is not in canonical form with respect to its unknown inputs. However, by following the steps of Algorithm 9, it can be set in canonical form. Section 10 provides our conclusion.

⁵To this regard, note that the *Canonization* procedure introduced in Appendix C of [31] can fail. It is incomplete and there are systems that are not canonizable. Algorithm 9 is the complete and thorough solution of the unknown input observability problem in the most general case.

2. System characterization and basic concepts

2.1. Basic equations

A general characterization of a dynamic system, which includes the presence of known and unknown inputs, is given by the following equations:

$$\begin{cases} \dot{x} &= g^0(x, t) + \sum_{k=1}^{m_u} f^k(x, t)u_k + \sum_{j=1}^{m_w} g^j(x, t)w_j \\ y &= [h_1(x, t), \dots, h_p(x, t)], \end{cases} \quad (1)$$

where:

- $x \in \mathcal{M}$ is the state and \mathcal{M} is a differential manifold of dimension n .
- t is the time.
- u_1, \dots, u_{m_u} are the known inputs. They are m_u independent functions of time that can be assigned. In control theory they are called *controls* or *control inputs*.
- w_1, \dots, w_{m_w} are the unknown inputs or *disturbances*. They are m_w independent functions of time. In particular, in this paper we assume that they are analytic functions of time.
- $f^1, \dots, f^{m_u}, g^0, g^1, \dots, g^{m_w}$ are $m_u + m_w + 1$ vector fields, which are assumed to be smooth functions of x and t .
- y is the output/s and it is characterized by p scalar functions, h_1, \dots, h_p , which are assumed to be smooth functions of x and t .

The vector field g^0 is often called the *drift* since, when all the inputs vanish, the state evolution is still non-vanishing in the presence of g^0 .

In many cases the system is time-invariant (from now on TI), namely it has not an explicit time-dependence and all the functions that appear in (1) do not depend explicitly on time. Nevertheless, we also account for an explicit time dependence to be as general as possible. From now on, we use the acronym TV to indicate a system with an explicit time-dependence (TV stands for Time-Variant). We also use the acronym UI to mean unknown input and UIO to mean unknown input observability.

Finally, we show that the characterization given in (1) is very general. In particular, we show that, apparently more general characterizations, can be easily transformed into the above characterization.

A first category of more general systems would also include the presence of the inputs in the output functions, i.e.,

$$\begin{cases} \dot{x} &= g^0(x, t) + \sum_{k=1}^{m_u} f^k(x, t)u_k + \sum_{j=1}^{m_w} g^j(x, t)w_j \\ y &= [h_1(x, t, u, w), \dots, h_p(x, t, u, w)], \end{cases}$$

where $u = [u_1, \dots, u_{m_u}]^T$ and $w = [w_1, \dots, w_{m_w}]^T$. However, this case can be easily converted to (1) by including the inputs in the state and by considering the system driven by the new inputs $\dot{u}_1, \dots, \dot{u}_{m_u}$ and $\dot{w}_1, \dots, \dot{w}_{m_w}$.

A second category of more general systems would take into account for a general nonlinear dependence of the dynamics

with respect to the inputs (both known and unknown). In accordance with Equation (1), this dependence is affine. This second category is characterized by the new dynamics:

$$\dot{x} = f(x, t, u, w)$$

Again, also this case can be easily converted to (1) by including the inputs in the state and by considering the system driven by the new inputs $\dot{u}_1, \dots, \dot{u}_{m_u}$ and $\dot{w}_1, \dots, \dot{w}_{m_w}$.

2.2. Reminders on basic algebraic operations

In this section, we provide some basic algebraic operations that will be widely used in the rest of this paper.

We remind the reader of the Lie derivative operation and of some related fundamental properties. The Lie derivative evaluates the change of a tensor field⁶ along a given vector field. In this paper we only use this operation when the tensor field is a scalar (i.e., a tensor of rank 0), when it is a vector (i.e., a tensor of rank 1 and type (0, 1)) and when it is a row vector, or covector (i.e., a tensor of rank 1 and type (1, 0)). We use the following notation:

- We denote by ∇ the differential operator. If $h(x)$ is a scalar field defined on the manifold \mathcal{M} , $\nabla h = \frac{\partial}{\partial x} h$. This is evidently a row vector of dimension n . Note that, in some cases we work in extended spaces that include the space of the states (x) and some of the UIs together with their time derivatives up to a given order. In this paper, with the symbol ∇ we always mean the differential with respect to all the coordinates of the considered space. When we wish to consider the differential with respect to a given subset of coordinates, we adopt the symbol ∂_* , where $*$ stands for the considered subset of coordinates. For instance, if we work in the extended space that includes the original state x together with the unknown inputs w_1, \dots, w_{m_w} , the differential with respect to the original state alone (i.e., x) will be indicated by the symbol $\partial_x = [\partial_{x^1}, \dots, \partial_{x^n}]$ and, in this case, $\nabla = [\partial_x, \partial_{w_1}, \dots, \partial_{w_{m_w}}]$.
- Given a vector field f (defined on \mathcal{M}), \mathcal{L}_f denotes the Lie derivative along f . When applied to the scalar field $h(x)$ we obtain the following new scalar field

$$\mathcal{L}_f h \triangleq \nabla h \cdot f$$

(the product of a row vector times a column vector is a scalar). When applied to the covector field ω , we obtain the covector field

$$\mathcal{L}_f \omega \triangleq f^T \left(\frac{\partial \omega^T}{\partial x} \right)^T + \omega \frac{\partial f}{\partial x},$$

⁶Note that, here, tensor is with respect to a coordinate change in our manifold \mathcal{M} . Later, the same concept of tensor will be referred to another group of transformations, which is the group of invariance of observability.

where f^T is the transpose of f (i.e., a row vector) and $\left(\frac{\partial \omega^T}{\partial x} \right)^T$ is the transpose of the Jacobian of ω^T , and it is an $n \times n$ matrix. An important special case occurs when ω is the differential of a scalar field, i.e., $\omega = \nabla h$. In this case, the above equation simplifies as follows:

$$\mathcal{L}_f \nabla h = \nabla \mathcal{L}_f h. \quad (2)$$

Finally, when applied to a vector field, g , we obtain the following new vector field:

$$\mathcal{L}_f g \triangleq \frac{\partial g}{\partial x} f - \frac{\partial f}{\partial x} g = [f, g],$$

where the parenthesis $[\cdot, \cdot]$ are called *Lie brackets*.

- In accordance with the control theory literature, we use the term *distribution* to denote the span⁷ of a set of $d \leq n$ vector fields, $\Delta = \text{span}\{f^1, f^2, \dots, f^d\}$. It is basically a vector space that depends on $x \in \mathcal{M}$.
- Similarly, we also use the term *codistribution* to denote the span of a set of $s \leq n$ covector fields, $\Omega = \text{span}\{\omega_1, \omega_2, \dots, \omega_s\}$. Again, it is a vector space that depends on $x \in \mathcal{M}$.
- Given the codistribution Ω and a vector field f , we set $\mathcal{L}_f \Omega$ the span of all the covectors $\mathcal{L}_f \omega$ for any $\omega \in \Omega$.
- Given two vector spaces V_1 and V_2 , $V_1 + V_2$ is their sum, i.e., the span of all the generators of V_1 and V_2 . When we have $k (> 2)$ vector spaces V_1, \dots, V_k , we denote its sum in the compact notation $\sum_{i=1}^k V_i$.

2.3. A simple illustrative example

We provide a simple example to illustrate the above concepts and to highlight the goal of this paper. We consider a wheeled robot that moves on a plane. By introducing on this plane a global frame, we can characterize the position and orientation of the robot by the three parameters x_R, y_R, θ_R (see figure 1).

Under the assumption of the unicycle constraint, these parameters satisfy the following dynamics equations (unicycle dynamics)

$$\begin{cases} \dot{x}_R &= v \cos \theta_R, \\ \dot{y}_R &= v \sin \theta_R, \\ \dot{\theta}_R &= \omega, \end{cases} \quad (3)$$

where v and ω are the linear and the rotational robot speed, respectively. We assume that a landmark is placed at the origin of the global frame. In addition, our robot is equipped with a sensor that perceives the landmark and provides its bearing angle in the local frame (i.e., the angle β in figure 1). This angle β

⁷Note that, here, span is over the ring of the scalar functions which are smooth in \mathcal{M} (or in a given open set of \mathcal{M} , when it is specified). In other words, any element of Δ is a vector field, f , that can be expressed as follows: $f = \sum_{i=1}^d c_i(x) f^i$, with $c_1(x), \dots, c_d(x)$ scalar and smooth functions in \mathcal{M} . When we work in a given extended space, the ring is extended accordingly.

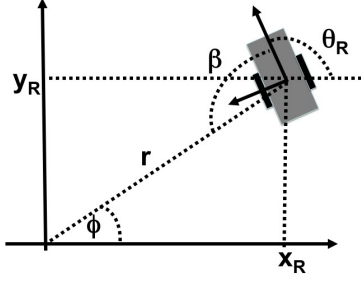


Figure 1: Wheeled robot moving on a plane.

can be expressed in terms of the robot position and orientation. We have

$$\beta = \pi - \theta_R + \text{atan2}(y_R, x_R). \quad (4)$$

This system is a special case of the systems characterized by (1). In particular:

- $x = [x_R, y_R, \theta_R]^T$ is the state, it has dimension $n = 3$ and it belongs to the manifold $\mathbb{R}^2 \times \mathcal{S}^1$.
- The vector g^0 is the zero 3-column vector.
- $m_u = 2$ and $u_1 = v, u_2 = \omega, f^1 = [\cos \theta_R, \sin \theta_R, 0]^T$ and $f^2 = [0, 0, 1]^T$.
- $m_w = 0$.
- y is the output and has a single component ($p = 1$), i.e., $y = h_1(x) = \pi - \theta_R + \text{atan2}(y_R, x_R)$.

We study the observability properties of this system by following an intuitive procedure. To check if the robot configuration $[x_R, y_R, \theta_R]^T$ is observable, we have to prove that it is possible to uniquely reconstruct the initial robot configuration by fusing the knowledge of inputs and outputs over a period of time. When, at the initial time, the bearing angle β of the origin is available, the robot can be everywhere in the plane but, for each position, only one orientation provides the right bearing β . In fig. 2a all the three positions A, B and C are compatible with the observation β , provided that the robot orientation satisfies (4). In particular, the orientation is the same for A and B but not for C.

Let us suppose that the robot moves according to the inputs $v(t)$ and $\omega(t)$. With the exception of the special motion consisting of a line passing by the origin, by only performing a further bearing observation it is possible to distinguish all the points belonging to the same line passing by the origin. In fig. 2b the two initial positions in A and B do not reproduce the same observations after the movement, i.e., $\alpha \neq \gamma$ (note that the segments AA' and BB' have the same length, which is known thanks to the knowledge of the system inputs). On the other hand, all the initial positions whose distance from the origin is the same, cannot be distinguished independently of the chosen trajectory. In fig. 2c, the two indicated trajectories provide the same bearing observations, at any time. Therefore, the dimension of the

unobservable region is 1. In particular, we introduce the following transformation

$$\begin{aligned} x_R &\rightarrow x'_R = \cos \delta x_R - \sin \delta y_R, \\ y_R &\rightarrow y'_R = \sin \delta x_R + \cos \delta y_R, \\ \theta_R &\rightarrow \theta'_R = \theta_R + \delta, \end{aligned} \quad (5)$$

where $\delta \in [-\pi, \pi)$ is the parameter that defines the transformation. The system inputs and output at any time are compatible with all the trajectories that differ because the initial state was transformed as above.

We wonder what is possible to reconstruct. The answer is immediate. All the physical quantities that are invariant with respect to the transform given in (5). In other words, all the functions of the following two quantities:

$$\begin{aligned} r &= r(x_R, y_R, \theta_R) = \sqrt{x_R^2 + y_R^2}, \\ \theta &= \theta(x_R, y_R, \theta_R) = \theta_R - \arctan 2(y_R, x_R). \end{aligned} \quad (6)$$

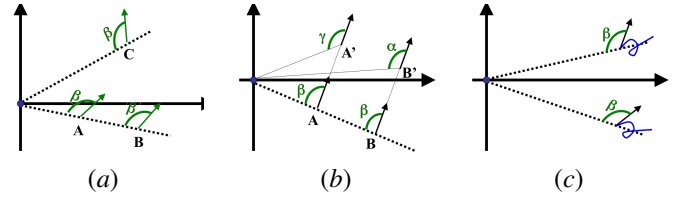


Figure 2: In (a), the three initial robot configurations are compatible with the same initial observation (β). In (b), the two initial positions (A and B) do not reproduce the same observations ($\alpha \neq \gamma$). In (c) the two indicated trajectories provide the same bearing observations at every time.

Now, we assume that the first input, v , is unknown. In other words, we have $m_u = m_w = 1$ and g^1 coincides with f^1 in the previous case and, the new f^1 is the old f^2 . We immediately remark that, in addition to the invariance given by (5), we have a new invariance, which is the scale. Indeed, both the known input (ω) and the output (β) are angular measurements. The system has no source of metric information. Hence, the unobservable region is characterized by the further transformation

$$\begin{aligned} x_R &\rightarrow x'_R = \lambda x_R, \\ y_R &\rightarrow y'_R = \lambda y_R, \\ \theta_R &\rightarrow \theta'_R = \theta_R, \end{aligned} \quad (7)$$

where $\lambda \in \mathbb{R}^+$ is the parameter that defines the transformation. In this case, we cannot reconstruct the distance r and we can only reconstruct any function of the only angle θ in (6) since θ is invariant both with respect to (5) and with respect to (7) ($\arctan 2(\lambda y_R, \lambda x_R) = \arctan 2(y_R, x_R)$, for any $\lambda \in \mathbb{R}^+$).

The above analysis was possible because the system is trivial: the state has dimension 3 and, both the dynamics and the output have a very simple expression. We wish to provide the analytic tool able to automatically perform such analysis, i.e., by following a systematic analytical procedure. Note that, in the absence of unknown inputs, this procedure is the observability rank condition introduced in [4]. In the presence of a single unknown input and for driftless systems, the solution has

very recently been obtained in [30]. Finally, for all the systems that satisfy the fundamental condition introduced in the next section (i.e., for systems in canonical form with respect to their unknown inputs) the solution is given in [31]. In this paper we provide the general solution that can be also applied to non canonic systems and even not canonizable (the concept of canonizable system will be given in Section 7). All these analytical systematic procedures use the algebraic operations summarized in section 2.2.

2.4. Observable Codistribution

When the state is not observable, there are functions of the state that can be observable, i.e., functions for which we have the possibility of reconstructing the value that they take at the initial state. In the simple example discussed in section 2.3, when all the inputs are known, we found that we cannot reconstruct the initial state. However, we can reconstruct the initial value of all the functions that depend on the state only via the two quantities r and θ in (6). In other words, for this system all the observable functions are all the functions that depend on the state only via $r = \sqrt{x_R^2 + y_R^2}$ and $\theta = \theta_R - \arctan 2(y_R, x_R)$ (e.g., a function of x_R and/or of θ_R is not an observable function). When the input v is unknown, we can reconstruct the initial value of the functions that only depend on θ . In this case, all the observable functions are all the functions that depend only on θ ⁸. The *observable codistribution*, or *observability codistribution*, is the span of the differentials of all the observable functions. By construction, the dimension of this codistribution cannot exceed the dimension of the state (n). It is immediate to remark that, if the dimension of this codistribution is n , every component of the state x is observable.

The observability rank condition is precisely the analytical procedure that allows us to compute the observable codistribution in the absence of unknown inputs. Our previous contributions in [30] and [31] provide the automatic procedures that compute the observable codistribution in the presence of unknown inputs in some special cases, as we mentioned at the end of Section 2.3. The procedure provided in [31] is here summarized in Section 4 and it is Algorithm 4. Section 4 also provides Algorithm 1, which improves Algorithm 4 by exhaustively accounting for the convergence properties. The automatic procedure to compute the observable codistribution in the general case is Algorithm 9. It uses iteratively Algorithm 8, which is a special implementation of Algorithm 1. It will be given in Section 7.

3. Canonic systems and canonical form with respect to the unknown inputs

We now introduce the new concepts of unknown input degree of reconstructability, canonic system with respect to its unknown inputs and canonical form with respect to the unknown

inputs. This is obtained by introducing an important matrix that will be called the *Unknown Input Reconstructability Matrix* from a finite set of scalar functions.

Definition 1 (Unknown input reconstructability matrix). *Given the k scalar functions of the state, $\lambda_1(x), \dots, \lambda_k(x)$, the unknown input reconstructability matrix from $\lambda_1, \dots, \lambda_k$ is defined as follows:*

$$\mathcal{RM}(\lambda_1, \dots, \lambda_k) \triangleq \begin{bmatrix} \mathcal{L}_{g^1} \lambda_1 & \mathcal{L}_{g^2} \lambda_1 & \dots & \mathcal{L}_{g^{m_w}} \lambda_1 \\ \mathcal{L}_{g^1} \lambda_2 & \mathcal{L}_{g^2} \lambda_2 & \dots & \mathcal{L}_{g^{m_w}} \lambda_2 \\ \dots & \dots & \dots & \dots \\ \mathcal{L}_{g^1} \lambda_k & \mathcal{L}_{g^2} \lambda_k & \dots & \mathcal{L}_{g^{m_w}} \lambda_k \end{bmatrix} \quad (8)$$

This matrix depends on x and, for TV systems, on t . Based on this matrix, we introduce the following definition:

Definition 2 (Unknown input degree of reconstructability). *Given the state that satisfies the dynamics in (1), and the functions $\lambda_1, \dots, \lambda_k$, the unknown input degree of reconstructability from $\lambda_1, \dots, \lambda_k$ is the rank of $\mathcal{RM}(\lambda_1, \dots, \lambda_k)$.*

By construction, the unknown input degree of reconstructability from any set of functions cannot exceed m_w . In addition, it depends in general on x and, for TV systems, also on t .

Given the system characterized by (1), we consider two special cases, depending on the set of functions $\lambda_1, \dots, \lambda_k$:

- This set of functions consists of the output functions, i.e., h_1, \dots, h_p .
- This set of functions consists of all the independent observable functions⁹.

In the first case, we refer to the *unknown input degree of reconstructability from the outputs*. In the second case, we omit to specify the functions and we refer to the *unknown input degree of reconstructability*. We introduce the following definition:

Definition 3 (Canonic system wrt the UIs). *The system in (1) is canonic with respect to the unknown inputs if its unknown input degree of reconstructability is m_w .*

Let us consider the systems for which the unknown input degree of reconstructability from the outputs is m_w . Since the output functions are observable functions, these systems are canonic with respect to their unknown inputs. We say that these systems are in canonical form. In particular, we introduce the following definition:

Definition 4 (System in canonical form wrt the UIs). *The system in (1) is in canonical form with respect to the unknown inputs if its unknown input degree of reconstructability from the outputs is m_w .*

⁸The general definition of observable function is provided in [30], where Definition 2 defines the observation space in the presence of unknown inputs, starting from the concept of indistinguishability. An observable function is precisely an element of this function space.

⁹Here, with *independent* we mean that their differentials with respect to the state are independent. Hence, we can have at most n independent observable functions (when there are n independent observable functions the entire state is evidently observable).

Let us consider a system that satisfies (1). Its observability properties are the same of the new system defined as follows:

- It is characterized by the same state.
- The time evolution of the state is the same (i.e., it satisfies the first equation of (1)).
- The outputs are h_1, \dots, h_p, h_{p+1} , where h_{p+1} is an observable function for the first system.

Indeed, if the function h_{p+1} is observable, its value can be reconstructed and, as a result, for the observability analysis, it can be set as an output of the system. For the simple example in Section 2.3, it is easy to realize that, when both the inputs are known, we can add to the single output $h_1 = \beta = \pi - \theta_R + \phi$, any function h_2 that depends on r and/or $\theta_R - \phi$. The observable codistribution of the resulting system remains the same (e.g., we can add $h_2 = r$). Hence, for any system that is canonic with respect to its unknown inputs there exists a system that shares the same observability properties and that is in canonical form with respect to its unknown inputs. However, for a canonic system that is not in canonical form, a set of observable functions that makes the reconstructability matrix full rank, is not necessarily available. One of the outcomes of Algorithm 9 is to automatically determine this set of functions.

Note that, as the rank of the matrix in (8) depends in general on x and, for TV systems, on t , all the above definitions are meant in a given open set where this rank takes the same value.

4. Systems in Canonical Form: the Analytical Solution

In this section we provide the analytical and automatic procedure that builds, in a finite number of steps, the observable codistribution for systems characterized by (1) and that are in canonical form with respect to their unknown inputs. This is the case investigated in [31]. On the other hand, the solution here derived (Algorithm 1) differs from the one introduced in [31] (which is here Algorithm 4). In particular, the new solution overcomes the limits of the solution presented in [31] and that regard the criterion of convergence of Algorithm 4, as we mentioned in Section 1. The criterion to check the convergence of Algorithm 1 is very simple and holds for any system. We start by providing Algorithm 1. Then, we prove its validity (Theorem 1). The general solution, i.e., for systems neither in canonical form nor canonic and even not canonizable, is Algorithm 9 and will be given in Section 7. It uses, iteratively, a special implementation of Algorithm 1, which is Algorithm 8.

4.1. The solution

In accordance with Definition 1 and Definition 4, we know that we can extract from the output m_w functions, which we denote by $\tilde{h}_1, \dots, \tilde{h}_{m_w}$, such that the reconstructability matrix is full rank. We set:

$$\mu_j^i = \mathcal{L}_{g^i} \tilde{h}_j, \quad i, j = 1, \dots, m_w \quad (9)$$

Note that, the above object (μ) is characterized by two indices. On the other hand, these indices are unrelated to the dimension (n) of our manifold (\mathcal{M}) and, consequently, they cannot be tensorial indices with respect to a coordinates' change in \mathcal{M} . In addition, with respect to a coordinates' change, μ_j^i behaves as a scalar field, for any i, j . On the other hand, and this is the fundamental key to achieving a profound understanding, μ is a two index tensor of type (1, 1) with respect to a new group of transformations. This is the group of invariance of observability, introduced in [31] and denoted by *SUIO*. Actually, the indices of the tensors associated to *SUIO* take $m_w + 1$ values. The extra value is set to 0. The tensor μ must be completed as follows:

$$\mu_0^0 = 1, \mu_0^i = 0, \mu_i^0 = \frac{\partial \tilde{h}_i}{\partial t} + \mathcal{L}_{g^0} \tilde{h}_i, \quad i = 1, \dots, m_w. \quad (10)$$

We denote by ν the inverse of μ . In other words, we have:

$$\sum_{\gamma=0}^{m_w} \mu_\gamma^\alpha \nu_\beta^\gamma = \delta_\beta^\alpha, \quad \alpha, \beta = 0, 1, \dots, m_w, \quad (11)$$

where δ_β^α is the delta Kronecker¹⁰. We have:

$$\nu_0^0 = 1, \nu_0^i = 0, \nu_i^0 = - \sum_{k=1}^{m_w} \mu_k^0 \nu_i^k, \quad (12)$$

$$\sum_{k=1}^{m_w} \nu_k^i \mu_j^k = \sum_{k=1}^{m_w} \mu_k^i \nu_j^k = \delta_j^i.$$

As ν is the inverse of a tensor field of type (1, 1) associated to *SUIO*, it is a tensor of type (1, 1).

Note that, in our theory, we deal with two types of tensors, simultaneously. This because observability is invariant both with respect to a coordinates' change in \mathcal{M} and with respect to *SUIO*¹¹. To avoid confusion, we use a matrix format for the tensors associated to a coordinates' change in \mathcal{M} and we use indices only for *SUIO*. In this notation, g^α , for a given $\alpha = 0, 1, \dots, m_w$, is a vector (i.e., tensor of type (0, 1)) with respect to a coordinates' change and the component α of a tensor of type (0, 1) with respect to *SUIO*.

We set

$$\tilde{h}_0 = t. \quad (13)$$

\tilde{h}_α , for a given $\alpha = 0, 1, \dots, m_w$, is a scalar (i.e., tensor of type (0, 0)) with respect to a coordinates' change and the component α of a tensor of type (1, 0) (i.e., a covector) with respect to *SUIO*.

¹⁰According to Einstein's notation, Greek indices take the values 0, 1, 2, ... while Latin indices take the values 1, 2, ...

¹¹This is similar to what happens in the Standard Model of particle physics. Also in that case, two types of tensors coexist, simultaneously: tensors with respect to the global Poincaré symmetry and tensors with respect to the local $SU(3) \times SU(2) \times U(1)$ gauge symmetry.

We denote by \widetilde{g}^α the following vector fields in \mathcal{M} :

$$\widetilde{g}^\alpha = \sum_{\beta=0}^{m_w} v_\beta^\alpha g^\beta, \quad \alpha = 0, 1, \dots, m_w. \quad (14)$$

Since they are obtained by an index contraction of two tensor fields of \mathcal{SUIO} , \widetilde{g}^α is the component α of a tensor of type $(0, 1)$ with respect to \mathcal{SUIO} .

The analytical and automatic procedure that builds, in a finite number of steps, the observable codistribution is Algorithm 1.

Algorithm 1: The algorithm that builds the observable codistribution for systems in canonical form.

$$\begin{aligned} O_0 &= \widetilde{O} + \text{span}\{\nabla h_1, \dots, \nabla h_p\} \\ O_{k+1} &= O_k + \sum_{i=1}^{m_u} \mathcal{L}_{f^i} O_k + \sum_{\beta=0}^{m_w} \mathcal{L}_{\widetilde{g}^\beta} O_k \end{aligned}$$

where \widetilde{O} is an integrable codistribution and is provided in (20) in Section 4.2. Algorithm 1 can be easily extended to cope with TV systems. Specifically, we simply need to introduce the following substitution at the first term in the sum at the recursive step (the term with $\beta = 0$):

$$\mathcal{L}_{\widetilde{g}^0} \rightarrow \dot{\mathcal{L}}_{\widetilde{g}^0} \triangleq \mathcal{L}_{\widetilde{g}^0} + \frac{\partial}{\partial t} \quad (15)$$

Algorithm 1 converges at the smallest k for which

$$O_k = O_{k-1} \quad (16)$$

and this occurs in at most $n - m_w$ steps (note that the dimension of O_0 is at least m_w). Note, however, that the computation of \widetilde{O} requires the calculation of the Lie brackets of the vector fields f^1, \dots, f^{m_u} along g^0, g^1, \dots, g^{m_w} , up to the $(2n - m_w)^{th}$ order (see Section 4.2).

The limit codistribution of the series O_k computed by Algorithm 1 will be denoted by O .

4.2. Ingredients of the Solution

The initialization step of Algorithm 1 (i.e., the computation of O_0) requires the computation of the codistribution \widetilde{O} . In this section, we define this codistribution and we provide its expression (i.e., Equation (20)). It is based on the knowledge of two integers, denoted by s and r . They are defined as follows:

- s is the smallest number of steps requested for the convergence of Algorithm 2, which computes the codistribution Ω^s .
- $r + 1$ is the smallest number of steps requested for the convergence of Algorithm 3, which computes the distribution Δ .

Algorithm 2: The algorithm that builds the codistribution Ω^s .

$$\begin{aligned} \Omega_0^s &= \sum_{j=1}^{m_w} \text{span}\{\widetilde{\nabla} h_j\} \\ \Omega_k^s &= \Omega_{k-1}^s + \sum_{\beta=0}^{m_w} \mathcal{L}_{g^\beta} \Omega_{k-1}^s \end{aligned}$$

4.2.1. The codistribution Ω^s and the integer s

Ω^s is the smallest codistribution that includes $\widetilde{\nabla} h_1, \dots, \widetilde{\nabla} h_{m_w}$ and is invariant with respect to \mathcal{L}_{g^β} ($\beta = 0, 1, \dots, m_w$). It is automatically constructed by Algorithm 2, which converges at the smallest integer s such that $\Omega_s^s = \Omega_{s-1}^s (= \Omega^s)$. As the dimension of Ω^s cannot exceed n , we have:

$$s \leq n - m_w + 1 \quad (17)$$

In the TV case, we simply need to replace: $\mathcal{L}_{g^0} \rightarrow \dot{\mathcal{L}}_{g^0} \triangleq \mathcal{L}_{g^0} + \frac{\partial}{\partial t}$.

The computation of \widetilde{O} needs to know the value of s . This is automatically obtained by running Algorithm 2.

4.2.2. The distribution Δ and the integer r

We introduce the following abstract operation¹²:

Definition 5 (Autobacket). Given the quantity $\phi^{\alpha_1, \dots, \alpha_k}$, which is a vector field with respect to a change of coordinates in \mathcal{M} and one component of a tensor of type $(0, k)$ with respect to \mathcal{SUIO} , we define its autobacket with respect to the system characterized by equation (1) the following object:

$$\begin{aligned} [\phi^{\alpha_1, \dots, \alpha_k}]^{\alpha_{k+1}} &= \sum_{\beta=0}^{m_w} v_\beta^{\alpha_{k+1}} [g^\beta, \phi^{\alpha_1, \dots, \alpha_k}] + v_0^{\alpha_{k+1}} \frac{\partial \phi^{\alpha_1, \dots, \alpha_k}}{\partial t} = \\ &= \sum_{\beta=0}^{m_w} v_\beta^{\alpha_{k+1}} [g^\beta, \phi^{\alpha_1, \dots, \alpha_k}] + \delta_0^{\alpha_{k+1}} \frac{\partial \phi^{\alpha_1, \dots, \alpha_k}}{\partial t}, \end{aligned} \quad (18)$$

where, the square brackets on the right hand side are the Lie brackets.

The output of this operation is still a vector field with respect to a coordinates' change and one component of a tensor of type $(0, k + 1)$ with respect to \mathcal{SUIO} . The autobacket increases by one the tensor rank with respect to \mathcal{SUIO} . We can apply the autobacket repetitively. We denote by

$$[\phi]^{(\alpha_1, \dots, \alpha_m)}$$

the vector field obtained by applying the autobacket repetitively m consecutive times, first along α_1 and last along α_m (these operations are not commutative).

We introduce the distribution Δ . It is the smallest distribution that includes f^1, \dots, f^{m_u} and is invariant under the autobacket operation.

¹²The definition given here (Equation (18)) differs from the one given in [31] (Equation (8.20)) by a minus sign. This simplifies the proof of Theorem 1, given in Appendix A.

Algorithm 3: The algorithm that builds the distribution Δ .

$$\begin{aligned}\Delta_0 &= \text{span} \{f^1, \dots, f^{m_u}\} \\ \Delta_k &= \Delta_{k-1} + \sum_{\beta} [\Delta_{k-1}]^\beta\end{aligned}$$

with:

$$[\Delta]^\beta \triangleq \sum_{v \in \Delta} \text{span} \{[v]^\beta\},$$

for any $\beta = 0, 1, \dots, m_w$. It is immediate to remark that, if v_1, \dots, v_l is a basis of Δ_{k-1} , then $v_1, \dots, v_l, [v_1]^\beta, \dots, [v_l]^\beta$ is a set of generators of $\Delta_{k-1} + [\Delta_{k-1}]^\beta$.

Algorithm 3 converges at the smallest integer k such that $\Delta_k = \Delta_{k-1} (= \Delta)$. We set $k = r + 1$. As the dimension of Δ cannot exceed n , we have:

$$r \leq n - 1 \quad (19)$$

The computation of $\tilde{\mathcal{O}}$ needs to know the value of r . This is automatically obtained by running Algorithm 3.

4.2.3. The codistribution $\tilde{\mathcal{O}}$

We are ready to provide $\tilde{\mathcal{O}}$. It is:

$$\tilde{\mathcal{O}} \triangleq \sum_{q=1}^{m_w} \sum_{j=0}^{s+r} \sum_{\alpha_1, \dots, \alpha_j} \sum_{i=1}^{m_u} \text{span} \{ \nabla \mathcal{L}_{[f^i]^{(\alpha_1, \dots, \alpha_j)}} \tilde{h}_q \} \quad (20)$$

where the sum $\sum_{\alpha_1, \dots, \alpha_j}$ stands for $\sum_{\alpha_1=0}^{m_w} \dots \sum_{\alpha_j=0}^{m_w}$. The computation of $\tilde{\mathcal{O}}$ requires the computation of the Autobracket of the vector fields f^1, \dots, f^{m_u} up to the $(s+r)^{\text{th}}$ order. From (17) and (19) we obtain that this order cannot exceed $2n - m_w$.

4.3. Validity of Algorithm 1

In this section we prove that Algorithm 1, with $\tilde{\mathcal{O}}$ defined in (20), provides the observable codistribution for systems characterized by (1) and in canonical form with respect to the unknown inputs.

We start from the solution given in [31]. In [31] we proved that the entire observable codistribution is computed by Algorithm 8.3, that is here Algorithm 4.

Algorithm 4: The algorithm introduced in [31] to build the observable codistribution, for systems in canonical form.

$$\begin{aligned}\Omega_0 &= \text{span} \{ \nabla h_1, \dots, \nabla h_p \} \\ \Omega_{k+1} &= \Omega_k + \sum_{i=1}^{m_u} \mathcal{L}_{f^i} \Omega_k + \sum_{\beta=0}^{m_w} \mathcal{L}_{\tilde{g}^\beta} \Omega_k + \\ &\sum_{q=1}^{m_w} \sum_{i=1}^{m_u} \sum_{\alpha_1=0}^{m_w} \dots \sum_{\alpha_k=0}^{m_w} \text{span} \{ \mathcal{L}_{[f^i]^{(\alpha_1, \dots, \alpha_k)}} \tilde{\nabla} h_q \}\end{aligned}$$

The following new fundamental result, establishes that we can use Algorithm 1 to build the entire observable codistribution (we remind the reader that \mathcal{O} is the limit codistribution of the series of codistributions built by Algorithm 1).

Theorem 1. *There exists \widehat{k} such that, for any $k \geq \widehat{k}$, $\mathcal{O} \subseteq \Omega_k$. Conversely, for any k , $\Omega_k \subseteq \mathcal{O}$.*

Proof The proof is given in Appendix A and Appendix B. ◀

We wish to emphasize that the convergence criterion to check the convergence of Algorithm 1 is trivially the condition in (16), which holds always, and it ensures that, the limit codistribution (\mathcal{O}) is obtained in less than $n - m_w$ steps.

We conclude this section by remarking that to run the algorithm that provides the observable codistribution (Algorithm 1) the system must be in canonical form with respect to its unknown inputs. In other words, in accordance with Definition 4, the output must include m_w functions, $\tilde{h}_1, \dots, \tilde{h}_{m_w}$, such that the reconstructability matrix $\mathcal{RM}(\tilde{h}_1, \dots, \tilde{h}_{m_w})$ is non singular (or, equivalently, its rank is equal to m_w). But this is a special case. We want to deal with the case where this is not possible (e.g., when $m_w > p$). In addition, we also want to deal with systems that are not canonic with respect to their unknown inputs. As we will see, in some cases, we can transform a non canonic system into a canonic system by including in the state some of the unknown inputs together with their time derivatives up to a given order (Equation (21)). We will call these systems *canonizable*. We also want to deal with the case where the system is not canonic and not even canonizable. In few words, we want to deal with any system¹³.

In the next sections we introduce the automatic procedure that, in a finite number of steps, provides the observable codistribution in any case. This automatic procedure is Algorithm 9. It uses iteratively Algorithm 8, which is a special implementation of Algorithm 1. Before providing these new algorithms, we need to introduce several new concepts.

5. Unknown Input Extension (UIE)

The procedure introduced in this section generalizes the state augmentation adopted in many previous works (e.g., [26, 27, 28, 29]), where the state is extended by including all the unknown inputs and their time derivatives up to a given order.

Here the inclusion of the unknown inputs is targeted, in the sense that we include only those necessary to make the system canonic, when this is possible, or to achieve the *unknown input highest degree of reconstructability* (see later Definition 6). As we will see, Algorithm 9 automatically selects which inputs must be included.

Let us start by characterizing a generic inclusion, i.e., where not necessarily all the inputs are included. Given the system in (1), we can characterize the same system by introducing an extended state that includes part of (or all) the unknown inputs, together with their time derivatives up to a given order. Without loss of generality, i.e., by reordering the unknown inputs, we assume that we include the last $d \leq m_w$ unknown inputs. In particular, we include w_{m_w-d+1} with its time derivatives up to

¹³Note that, in [31] we introduced some heuristic procedures to apply Algorithm 4 to systems that are not in canonical form. These procedures are not general and, moreover, the system must be imperatively canonic.

the order k_1 , w_{m_w-d+2} with its time derivatives up to the order k_2 and so on and so forth up to w_{m_w} and its time derivatives up to the order k_d . k_j is the highest order time derivative of w_{m_w-d+j} included in the state. By using the usual notation:

$$w_j^{(k)} \triangleq \frac{d^k w_j}{dt^k}$$

and

$$w_j^{(0)} \triangleq w_j$$

the extended state is:

$$x_d = \left[x^T, w_{m_w-d+1}^{(0)}, \dots, w_{m_w-d+1}^{(k_1)}, w_{m_w-d+2}^{(0)}, \dots, w_{m_w-d+2}^{(k_2)}, \dots, w_{m_w}^{(0)}, \dots, w_{m_w}^{(k_d)} \right]^T \quad (21)$$

The extended state is fully characterized by the $d (\leq m_w)$ integers k_1, \dots, k_d . Each of them can take any non negative value. If a given k_j is equal to zero it means that the state only includes the unknown input w_{m_w-d+j} and no time derivatives of it. We call the system characterized by the above extended state an *unknown input extension* of the system in (1). Specifically, it is the (k_1, k_2, \dots, k_d) unknown input extension of the system in (1).

We will need the following two types of unknown input extensions:

1. **Finite** unknown input extension, when all the integers (k_1, k_2, \dots, k_d) take a finite value. In this case the unknown input extension will be denoted by

$$\mathcal{UIE}(k_1, k_2, \dots, k_d).$$

2. **Infinite** unknown input extension, when all the integers (k_1, k_2, \dots, k_d) take an infinite value. In this case the unknown input extension will be denoted by

$$\mathcal{UIE}_{d\infty}.$$

The first case will be fundamental to introduce the concepts of *unknown input highest degree of reconstructability* and *canonical system with respect to its unknown inputs*. The second case will be fundamental to obtain observable functions when the system is not in canonical form (or the system is even not canonic).

5.1. Finite Unknown Input Extension

Let us consider the system characterized by (1) and its (k_1, k_2, \dots, k_d) unknown input extension. We assume that all the integers (k_1, k_2, \dots, k_d) are finite numbers. The equations that characterize the new system, i.e., $\mathcal{UIE}(k_1, k_2, \dots, k_d)$, read as follows:

$$\begin{cases} \dot{x}_d &= g_d^0 + \sum_{i=1}^{m_u} f_d^i u_i + \sum_{j=1}^{m_w-d} g_d^j w_j + \sum_{j=1}^d e_d^j w_{m_w-d+j}^{(k_j+1)} \\ y &= [h_1, \dots, h_p], \end{cases} \quad (22)$$

where:

$$g_d^0 \triangleq \begin{bmatrix} g^0 + \sum_{j=m_w-d+1}^{m_w} g^j w_j \\ w_{m_w-d+1}^{(1)} \\ \dots \\ w_{m_w-d+1}^{(k_1)} \\ 0 \\ w_{m_w-d+2}^{(1)} \\ \dots \\ w_{m_w-d+2}^{(k_2)} \\ 0 \\ \dots \\ w_{m_w}^{(1)} \\ \dots \\ w_{m_w}^{(k_d)} \\ 0 \end{bmatrix}, \quad (23)$$

$$f_d^i \triangleq \begin{bmatrix} f^i \\ 0_{k_1+1} \\ 0_{k_2+1} \\ \dots \\ 0_{k_d+1} \end{bmatrix}, \quad i = 1, \dots, m_u$$

$$g_d^j \triangleq \begin{bmatrix} g^j \\ 0_{k_1+1} \\ 0_{k_2+1} \\ \dots \\ 0_{k_d+1} \end{bmatrix}, \quad j = 1, \dots, m_w - d$$

$$e_d^1 \triangleq \begin{bmatrix} 0_n \\ 0_{k_1} \\ 1 \\ 0_{k_2} \\ 0 \\ \dots \\ 0_{k_d} \\ 0 \end{bmatrix}, \quad e_d^2 \triangleq \begin{bmatrix} 0_n \\ 0_{k_1} \\ 0 \\ 0_{k_2} \\ 1 \\ \dots \\ 0_{k_d} \\ 0 \end{bmatrix}, \quad \dots, \quad e_d^d \triangleq \begin{bmatrix} 0_n \\ 0_{k_1} \\ 0 \\ 0_{k_2} \\ 0 \\ \dots \\ 0_{k_d} \\ 1 \end{bmatrix},$$

and we denoted by 0_K the K -dimensional zero column vector.

We remark that $\mathcal{UIE}(k_1, k_2, \dots, k_d)$ has still m_u known inputs and m_w unknown inputs. However, while the m_u known inputs coincide with the original ones, only the first $m_w - d$ unknown inputs coincide with the original ones. The last d unknown inputs of $\mathcal{UIE}(k_1, k_2, \dots, k_d)$ are the time derivatives of a given order of the original unknown inputs. Specifically, the new j^{th} unknown input is $w_{m_w-d+j}^{(k_j+1)}$ ($j = 1, \dots, d$). The state evolution depends on the known inputs via the vector fields f_d^i , ($i = 1, \dots, m_u$), on the first $m_w - d$ unknown inputs via the vector fields g_d^j , ($j = 1, \dots, m_w - d$), and on the last d unknown inputs via the unit vectors e_d^j , ($j = 1, \dots, d$). Finally, we remark that only the vector field g_d^0 depends on the new state elements.

We introduce the following definition, which extends Definition 2:

Definition 6 (UI Highest Degree of Reconstructability).

Given the system in (1), the unknown input highest degree

of reconstructability is the largest unknown input degree of reconstructability of all its finite unknown input extensions.

By construction, the unknown input highest degree of reconstructability cannot exceed m_w . We introduce the following last definition, which extends Definition 3:

Definition 7 (Canonizable System with respect to its UIs).

The system in (1) is canonizable with respect to the unknown inputs if its unknown input highest degree of reconstructability is equal to m_w .

In other words, the system is canonizable with respect to the unknown inputs if there exists a finite unknown input extension of it that is canonic with respect to the unknown inputs.

5.2. Infinite Unknown Input Extension

An infinite unknown input extension of the system in (1) is obtained by including in the state $d \leq m_w$ unknown inputs together with *all* their time derivatives. The state will be denoted by $x_{d\infty}$, which is defined as follows:

$$x_{d\infty} \triangleq [x^T, w_{d\infty}] \quad (24)$$

with

$$w_{d\infty} \triangleq [w_{m_w-d+1}^{(0)}, w_{m_w-d+1}^{(1)}, w_{m_w-d+1}^{(2)}, \dots, w_{m_w-d+2}^{(0)}, w_{m_w-d+2}^{(1)}, w_{m_w-d+2}^{(2)}, \dots, \dots, w_{m_w}^{(0)}, w_{m_w}^{(1)}, w_{m_w}^{(2)}, \dots] \quad (25)$$

The system is characterized by the following equations:

$$\begin{cases} \dot{x}_{d\infty} &= g_{d\infty}^0 + \sum_{i=1}^{m_u} f_{d\infty}^i u_i + \sum_{j=1}^{m_w-d} g_{d\infty}^j w_j \\ y &= [h_1, \dots, h_p], \end{cases} \quad (26)$$

where:

$$g_{d\infty}^0 \triangleq \begin{bmatrix} g^0 + \sum_{j=m_w-d+1}^{m_w} g^j w_j \\ w_{m_w-d+1}^{(1)} \\ w_{m_w-d+1}^{(2)} \\ w_{m_w-d+1}^{(3)} \\ \dots \\ w_{m_w-d+2}^{(1)} \\ w_{m_w-d+2}^{(2)} \\ w_{m_w-d+2}^{(3)} \\ \dots \\ \dots \\ w_{m_w}^{(1)} \\ w_{m_w}^{(2)} \\ w_{m_w}^{(3)} \\ \dots \end{bmatrix}, \quad (27)$$

$$f_{d\infty}^i \triangleq \begin{bmatrix} f^i \\ 0 \\ 0 \\ 0 \\ \dots \\ 0 \\ 0 \\ 0 \\ \dots \\ \dots \\ 0 \\ 0 \\ 0 \\ \dots \end{bmatrix}, \quad g_{d\infty}^j \triangleq \begin{bmatrix} g^j \\ 0 \\ 0 \\ 0 \\ \dots \\ 0 \\ 0 \\ 0 \\ \dots \\ \dots \\ 0 \\ 0 \\ 0 \\ \dots \end{bmatrix},$$

with $i = 1, \dots, m_u$, and $j = 1, \dots, m_w - d$.

6. Implementations of Algorithm 1 and 4 on $\mathcal{UIE}_{d\infty}$

Let us consider the system defined by (1) and let us suppose that its unknown input degree of reconstructability from all the available observable functions is $m < m_w$. In addition, we denote by $\tilde{h}_1, \dots, \tilde{h}_m$ a set of m available observable functions such that the reconstructability matrix from them has rank equal to m . Without loss of generality, we assume that the rank of the matrix that consists of the first m columns of the reconstructability matrix is m (this is general as it is always obtained by re-ordering the unknown inputs). We consider the infinite unknown input extension defined in Section 5.2 and denoted by $\mathcal{UIE}_{d\infty}$, where we include in the state the remaining $d = m_w - m$ unknown inputs together with their time derivatives up to any order.

We introduce the two series of codistributions Ω_k^∞ , and \mathcal{O}_k^∞ , ($k = 0, 1, \dots$), by running two new algorithms, Algorithm 5 and Algorithm 8, respectively. These algorithms are obtained by proceeding as follow. Basically, we introduce some modifications in order to run Algorithm 4 and Algorithm 1 on $\mathcal{UIE}_{d\infty}$. As $\mathcal{UIE}_{d\infty}$ is not in canonical form, the implementation of Algorithm 4 and Algorithm 1 is not possible. In particular, we cannot introduce the tensor μ in (9), and (10), its inverse ν in (11), and (12), the vector fields \tilde{g} in (14), and the autobracket operation in (18).

Starting from the functions $\tilde{h}_1, \dots, \tilde{h}_m$ and the $m + 1$ vector fields $g_{d\infty}^0, g_{d\infty}^1, \dots, g_{d\infty}^m$ in (27) we build the analogous of the tensor μ defined by (9) and (10). We denote it by ${}^m\mu$. We set:

$${}^m\mu_j^i = \mathcal{L}_{g_{d\infty}^i} \tilde{h}_j, \quad i, j = 1, \dots, m \quad (28)$$

$${}^m\mu_0^0 = 1, \quad {}^m\mu_0^i = 0, \quad {}^m\mu_i^0 = \frac{\partial \tilde{h}_i}{\partial t} + \mathcal{L}_{g_{d\infty}^0} \tilde{h}_i, \quad i = 1, \dots, m. \quad (29)$$

We denote by ${}^m\nu$ the inverse of ${}^m\mu$, namely: $\sum_{\gamma=0}^m {}^m\mu_\gamma^\alpha {}^m\nu_\beta^\gamma = \delta_\beta^\alpha$ for any $\alpha, \beta = 0, 1, \dots, m$.

We denote by ${}^m\tilde{g}_{d\infty}^\alpha$ the following vector fields:

$${}^m\tilde{g}_{d\infty}^\alpha = \sum_{\beta=0}^m {}^m\nu_\beta^\alpha g_{d\infty}^\beta, \quad \alpha = 0, 1, \dots, m. \quad (30)$$

We define the analogous of the Autobracket. In particular, we directly define the analogous of the vector fields $[f^i]^{(\alpha_1, \dots, \alpha_k)}$, which appear at the recursive step of Algorithm 4. They are recursively defined as follows:

$${}^m [f_{d\infty}^i]^{(\alpha_1, \dots, \alpha_{k-1}, \alpha_k)} = \sum_{\beta=0}^m {}^m \nu_{\beta}^{\alpha_k} \left[g_{d\infty}^{\beta}, {}^m [f_{d\infty}^i]^{(\alpha_1, \dots, \alpha_{k-1})} \right] + \delta_0^{\alpha_k} \frac{\partial^m [f_{d\infty}^i]^{(\alpha_1, \dots, \alpha_{k-1})}}{\partial t}, \quad (31)$$

and all the indices $\alpha_1, \dots, \alpha_k$ take the values among 0 and m .

6.1. The codistributions Ω_k^∞ and their properties

The codistributions $\Omega_k^\infty, k = 0, 1, \dots$, are obtained by running Algorithm 5.

Algorithm 5: The algorithm that builds the codistributions Ω_k^∞ .

$$\begin{aligned} \Omega_0^\infty &= \text{span} \{ \nabla h_1, \dots, \nabla h_p \} \\ \Omega_{k+1}^\infty &= \Omega_k^\infty + \sum_{i=1}^{m_u} \mathcal{L}_{f_{d\infty}^i} \Omega_k^\infty + \sum_{\beta=0}^m \mathcal{L}_{m \tilde{g}_{d\infty}^\beta} \Omega_k^\infty + \\ &\sum_{q=1}^m \sum_{i=1}^{m_u} \sum_{\alpha_1=0}^m \dots \sum_{\alpha_k=0}^m \text{span} \left\{ \mathcal{L}_{m [f_{d\infty}^i]^{(\alpha_1, \dots, \alpha_k)}} \nabla \tilde{h}_q \right\} \end{aligned}$$

In this algorithm, all the operations that appear at the recursive step (the autobracket ${}^m [f_{d\infty}^i]^{(\alpha_1, \dots, \alpha_k)}$, the Lie derivatives along ${}^m \tilde{g}_{d\infty}^\beta$) are based on the new definitions given by Equations (28-31). As a result, these operations can be executed when the system is not in canonical form. In particular, their execution, requires that m observable functions are available such that the reconstructability matrix from them has rank m (and this is precisely the case we are dealing with).

It holds the following result:

Theorem 2. *Let us consider a scalar function $\theta(x)$ (i.e., of the only original state). If there exists an integer k such that $\nabla \theta \in \Omega_k^\infty$ at $x_0 \in \mathcal{M}$, then $\theta(x)$ is observable at x_0 . Conversely, if $\theta(x)$ is observable on a given open set $\mathcal{A} \subseteq \mathcal{M}$, then there exists an integer k such that $\nabla \theta \in \Omega_k^\infty$ on a dense set of \mathcal{A} .*

Proof The proof is given in Appendix C. ◀

This result allows us to use Algorithm 5 to build the observable codistribution of a system which is not in canonical form. We remark that the state that characterizes the system where we apply Algorithm 5 has infinite dimensions. As a result, this algorithm does not converge, in general. Note, however, that the computation of the codistribution at a given step k (i.e., Ω_k^∞) can be done by only including in the state the time derivatives of the last d unknown inputs up to the order $k - 1$. This is due to the special structure of the vector fields in (27), which makes the functions computed by Algorithm 5 at the step k independent of the time derivatives of the unknown input of order larger than $k - 1$. In other words, from a practical point of view, we could obtain Ω_k^∞ by working with a finite unknown input extension. In our final procedure, however, we do not compute Ω_k^∞ but O_k^∞ , defined in Section 6.2. Theorem 2 will be only used as an intermediate theoretical result to prove the validity of Theorem 3 in Section 6.2.

6.2. The codistributions O_k^∞ and their properties

As in the case of the codistributions $\Omega_k^\infty, k = 0, 1, \dots$, we would like to generate the codistributions $O_k^\infty, k = 0, 1, \dots$, by re-adapting Algorithm 1 based on the new definitions given by Equations (28-31). In addition, we wish to obtain for these codistributions the same result stated by Theorem 2. On the other hand, we immediately encounter a fundamental problem. As $\mathcal{UIE}_{d\infty}$ is characterized by an infinite dimension state, the initialization step, which includes the computation of \tilde{O} , can require infinite operations. In accordance with Equation (20), we need to compute all the terms that correspond to the sum on the index j , which runs from 0 up to $s + r$. The problem which arises stems from the fact that these integers can diverge (both s and r are only bounded by the state dimension, which is infinite in the specific case of $\mathcal{UIE}_{d\infty}$).

In Section 6.2.1, we show that, thanks to the special structure of the vector fields $f_{d\infty}^1, \dots, f_{d\infty}^{m_u}, g_{d\infty}^0, g_{d\infty}^1, \dots, g_{d\infty}^m$ in (27), the integer r is actually bounded by n , which is the dimension of x , i.e., the dimension of the original state. In particular, $r \leq n - 1$. In Section 6.2.2, we provide a new definition for the integer s and we denote it by s_x . In accordance with this new definition, also s_x is bounded by the dimension of the original state. Specifically, we have $s_x \leq n - m + 1$.

In Section 6.2.3, we provide the iterative algorithm that builds the codistributions O_k^∞ . It is Algorithm 8. Its initialization step computes \tilde{O}^∞ by using s_x instead of s (see (33)). Finally, in Section 6.2.4, we provide the properties of the codistributions built by Algorithm 8. We show that, by using all the above definitions, we have the same results stated by Theorem 2 (i.e., Theorem 3).

6.2.1. Upper bound for the integer r

In the case of the system $\mathcal{UIE}_{d\infty}$, Algorithm 3 becomes Algorithm 6.

Algorithm 6: The algorithm that builds the distribution Δ for $\mathcal{UIE}_{d\infty}$.

$$\begin{aligned} \Delta_0 &= \text{span} \{ f_{d\infty}^1, \dots, f_{d\infty}^{m_u} \} \\ \Delta_k &= \Delta_{k-1} + \sum_{\beta} {}^m [\Delta_{k-1}]^\beta \end{aligned}$$

The autobracket at the recursive step is obtained by using the new definition in (31). We have the following result:

Proposition 1. *Let us denote by $\phi_{d\infty}$ any vector field computed by Algorithm 6. The entries of $\phi_{d\infty}$, with index larger than n , are null.*

Proof See Appendix D. ◀

From the above result, it is immediate to realize that the dimension of Δ_k cannot exceed n . As a result, the condition that characterizes the convergence of Algorithm 6 (i.e., $\Delta_k = \Delta_{k-1}$) occurs in at most $n - 1$ steps. In other words, $k \leq n$. As $r = k - 1$ by definition, we obtain $r \leq n - 1$.

6.2.2. The new integer s_x and its upper bound

Let us apply Algorithm 2 to $\mathcal{UIE}_{d\infty}$ and let us denote by ${}^x\Omega_k^g$ the span of all the differentials with respect to the original state x of all the functions computed by Algorithm 2, up to its k^{th} step. We obtain Algorithm 7.

Algorithm 7: The algorithm that builds the codistributions Ω_k^g and ${}^x\Omega_k^g$

$$\begin{aligned}\Omega_0^g &= \sum_{j=1}^m \text{span} \{ \nabla \tilde{h}_j \} \\ \Omega_k^g &= \Omega_{k-1}^g + \sum_{\beta=0}^m \mathcal{L}_{g_{d\infty}^\beta} \Omega_{k-1}^g \\ \text{Set } \nabla \theta_k^1, \dots, \nabla \theta_k^{D_k} &\text{ a basis of } \Omega_k^g \\ {}^x\Omega_k^g &= \text{span} \{ \partial_x \theta_k^1, \dots, \partial_x \theta_k^{D_k} \}\end{aligned}$$

The integer D_k is the dimension of the codistribution Ω_k^g . We define the integer s_x as the smallest integer such that

$${}^x\Omega_{s_x}^g = {}^x\Omega_{s_x-1}^g \quad (32)$$

As the dimension of ${}^x\Omega_k^g$ cannot exceed n , and the dimension of ${}^x\Omega_0^g$ is m , we have $s_x \leq n - m + 1$. Note that, we cannot conclude that the condition in (32) ensures the convergence of the series of codistributions ${}^x\Omega_k^g$, $k = 0, 1, \dots$. On the other hand, for the special system $\mathcal{UIE}_{d\infty}$, due to the special structure of the vector fields in (27) that characterize the dynamics, this is the case and will be proved in Appendix E.2 (Proposition 11).

6.2.3. The algorithm that builds the codistributions O_k^∞

The codistributions O_k^∞ , $k = 0, 1, \dots$, are obtained by running Algorithm 8.

Algorithm 8: The algorithm that builds the codistributions O_k^∞ .

$$\begin{aligned}O_0^\infty &= \text{span} \{ \nabla h_1, \dots, \nabla h_p \} + \tilde{O}^\infty \\ O_{k+1}^\infty &= O_k^\infty + \sum_{i=1}^{m_u} \mathcal{L}_{f_{d\infty}^i} O_k^\infty + \sum_{\beta=0}^m \mathcal{L}_{g_{d\infty}^\beta} O_k^\infty\end{aligned}$$

The vector fields $m\tilde{g}_{d\infty}^\beta$ that appear at the recursive step, are obtained from Equations (28-30) and the codistribution \tilde{O}^∞ is obtained by using the integer r and the new integer s_x , i.e.:

$$\tilde{O}^\infty \triangleq \sum_{q=1}^m \sum_{j=0}^{s_x+r} \sum_{\alpha_1, \dots, \alpha_j} \sum_{i=1}^{m_u} \text{span} \left\{ \nabla \mathcal{L}_{m[f_{d\infty}^i]^{\alpha_1, \dots, \alpha_j}} \tilde{h}_q \right\} \quad (33)$$

where the autobracket $m[f_{d\infty}^i]^{\alpha_1, \dots, \alpha_j}$ is computed by using Equation (31). In (33) we must compute the autobracket of the vectors $f_{d\infty}^1, \dots, f_{d\infty}^{m_u}$ up to the $r + s_x$ order. As $r \leq n - 1$ and $s_x \leq n - m + 1$, this order cannot exceed $2n - m$.

6.2.4. Properties of the codistributions O_k^∞

We have exactly the same result that holds for the codistributions Ω_k^∞ . We have:

Theorem 3. Let us consider a scalar function $\theta(x)$ (i.e., of the only original state). If there exists an integer k such that $\nabla \theta \in O_k^\infty$ at $x_0 \in \mathcal{M}$, then $\theta(x)$ is observable at x_0 . Conversely, if $\theta(x)$ is observable on a given open set $\mathcal{A} \subseteq \mathcal{M}$, then there exists an integer k such that $\nabla \theta \in O_k^\infty$ on a dense set of \mathcal{A} .

Proof The proof is given in Appendix E. ◀

This result is fundamental and will be adopted in our final procedure (Algorithm 9). If our system is not in canonical form (or even worse is neither canonic nor canonizable), we cannot run Algorithm 1. Let us suppose that its unknown input degree of reconstructability from all the available observable functions (at the beginning these are the output functions) is equal to $m < m_w$. We select the m observable functions that make the above degree equal to m and we denote them by $\tilde{h}_1, \dots, \tilde{h}_m$. From them, we can build the quantities defined in (28-31) and then we can run Algorithm 8. Theorem 3 ensures that this algorithm builds the entire observable codistribution. On the other hand, as the state that characterizes $\mathcal{UIE}_{d\infty}$ has infinite dimensions, this algorithm does not converge, in general. In the next section, we provide a final fundamental result that allows us to conclude that either this algorithm converges in at most $n - m$ steps (with n the dimension of the original state) or, in at most $n - m$ steps, it provides a new observable function able to increase the unknown input degree of reconstructability. In this latter case, if the new degree is equal to m_w it means that the new system is now in canonical form and we can apply Algorithm 1. Otherwise, we repeat the above procedure, starting from a higher degree of reconstructability.

7. The complete solution

This section provides the general solution that allows us to obtain, in a finite number of steps, the entire observable codistribution for any system that satisfies (1). No further assumption is required. The system does not necessarily have to be in canonical form with respect to its unknown inputs and it can also be neither canonical nor canonizable.

The automatic procedure that builds the observable codistribution is Algorithm 9. Before providing and describing the procedure, we need a final theoretical result.

7.1. System extensibility

As we mentioned at the end of Section 6.2.4, when a system is not in canonical form, we cannot run Algorithm 1. We denote by $\tilde{h}_1, \dots, \tilde{h}_m$ ($m \leq m_w$) a set of available¹⁴ observable functions from which the unknown input degree of reconstructability is m . They are such that the following condition holds:

$$\text{rank} \left(\mathcal{RM}(\tilde{h}_1, \dots, \tilde{h}_m) \right) = m. \quad (34)$$

Without loss of generality (i.e., by reordering the UIs) we assume that the submatrix obtained by considering the first m

¹⁴At the beginning, this means that $p \geq m$ and $\tilde{h}_1, \dots, \tilde{h}_m$ are m functions among the output functions, h_1, \dots, h_p .

columns of $\mathcal{RM}(\tilde{h}_1, \dots, \tilde{h}_m)$ has non vanishing determinant. We can run Algorithm 8. The condition that ensures its convergence is trivially:

$$O_k^\infty = O_{k-1}^\infty$$

Unfortunately, as the state has infinite dimensions, the smallest integer at which the above condition occurs is unbounded. We denote by ∇h_k^∞ a generic generator of O_k^∞ . Note that Algorithm 8 automatically computes all these generators. We also denote by ∂_w the operator that includes all the derivatives with respect to the components of the state extension $w_{d\infty}$ in (25). We introduce the following definition:

Definition 8. *The system in (1) with reconstructability matrix of rank $m < m_w$ is said non-extensible if, for any integer k , $\partial_w h_k^\infty$ vanishes for all the generators of O_k^∞ . In addition, it is said extensible if it is not non-extensible.*

We have the following fundamental result.

Proposition 2. *If the system is non-extensible, then Algorithm 8 converges in at most $n - m$ steps. Conversely, if the system is extensible, then there exists a generator h_k^∞ , which depends on a given component of $w_{d\infty}$, with $k \leq n - m + 1$.*

Proof Let us prove the first statement. The convergence of Algorithm 8 is characterized by the condition $O_k^\infty = O_{k-1}^\infty$. As no generator of O_k^∞ depends on $w_{d\infty}$ in (25), the dimension of O_k^∞ cannot exceed n . As a result, the condition that characterizes the convergence is attained in at most $n - m$ steps (note that the dimension of O_0^∞ is at least m). This proves the first statement.

Let us prove the second statement. We proceed by contradiction. Let us suppose that all the generators of O_j^∞ are independent of $w_{d\infty}$ when $j \leq n - m + 1$. By proceeding as in the proof of the first statement, we attain the condition that characterizes the convergence of Algorithm 8. We have $O_j^\infty = O_{j-1}^\infty$, for some $j \leq n - m + 1$. As a result, $O_k^\infty = O_{j-1}^\infty$ for any $k \geq j - 1$. But this means that the generators of O_k^∞ are independent of $w_{d\infty}$, namely the system is non-extensible, in contrast with the assumption of the second statement. ◀

The above result ensures that, when we run Algorithm 8, we have one of the following two results, in at most $n - m$ steps:

1. **(Non-extensible)** we attain the convergence, and, consequently, the observable codistribution.
2. **(Extensible)** we find a generator of O_k^∞ (h_k^∞) that depends on at least one component of $w_{d\infty}$.

In the first case, we have achieved our final goal. In the second case, we build a new unknown input extension of the original system (denoted by Σ) by proceeding as follows. We know that h_k^∞ depends on $w_{d\infty}$. Let us characterize a general dependency of h_k^∞ on $w_{d\infty}$. We denote by $z \leq d$ the number of the original UIs that appear in h_k^∞ . We can assume that h_k^∞ depends on $w_{m_w-z+1}, w_{m_w-z+2}, \dots, w_{m_w}$ (we re-order the last d original UIs). For each of them, h_k^∞ can depend directly on it and/or on a given time derivative of it. We denote by k_1, k_2, \dots, k_z

the highest orders time derivative of the corresponding original unknown inputs that appear in h_k^∞ . We introduce the following unknown input extension:

Definition 9. *Given h_k^∞ , we set $\Sigma \triangleq \mathcal{UIE}(k_1, k_2, \dots, k_z)$.*

We have the following result:

Lemma 1. *For the system Σ , we have:*

$$\text{rank}(\mathcal{RM}(\tilde{h}_1, \dots, \tilde{h}_m, h_k^\infty)) = m + 1$$

Proof The proof is obtained by a direct computation. For $\underline{\Sigma} = \mathcal{UIE}(k_1, k_2, \dots, k_z)$, the reconstructability matrix from $\tilde{h}_1, \dots, \tilde{h}_m, h_k^\infty$ is:

$$\mathcal{RM}(\tilde{h}_1, \dots, \tilde{h}_m, h_k^\infty) = \begin{bmatrix} \mathcal{L}_{g_z^1} \tilde{h}_1 & \dots & \mathcal{L}_{g_z^m} \tilde{h}_1 & \dots & \mathcal{L}_{g_z^{m_w-z}} \tilde{h}_1 & \mathcal{L}_{e_z^1} \tilde{h}_1 & \dots & \mathcal{L}_{e_z^z} \tilde{h}_1 \\ \mathcal{L}_{g_z^1} \tilde{h}_2 & \dots & \mathcal{L}_{g_z^m} \tilde{h}_2 & \dots & \mathcal{L}_{g_z^{m_w-z}} \tilde{h}_2 & \mathcal{L}_{e_z^1} \tilde{h}_2 & \dots & \mathcal{L}_{e_z^z} \tilde{h}_2 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \mathcal{L}_{g_z^1} \tilde{h}_m & \dots & \mathcal{L}_{g_z^m} \tilde{h}_m & \dots & \mathcal{L}_{g_z^{m_w-z}} \tilde{h}_m & \mathcal{L}_{e_z^1} \tilde{h}_m & \dots & \mathcal{L}_{e_z^z} \tilde{h}_m \\ \mathcal{L}_{g_z^1} h_k^\infty & \dots & \mathcal{L}_{g_z^m} h_k^\infty & \dots & \mathcal{L}_{g_z^{m_w-z}} h_k^\infty & \mathcal{L}_{e_z^1} h_k^\infty & \dots & \mathcal{L}_{e_z^z} h_k^\infty \end{bmatrix} = \begin{bmatrix} \mathcal{L}_{g_z^1} \tilde{h}_1 & \dots & \mathcal{L}_{g_z^m} \tilde{h}_1 & \dots & \mathcal{L}_{g_z^{m_w-z}} \tilde{h}_1 & 0 & \dots & 0 \\ \mathcal{L}_{g_z^1} \tilde{h}_2 & \dots & \mathcal{L}_{g_z^m} \tilde{h}_2 & \dots & \mathcal{L}_{g_z^{m_w-z}} \tilde{h}_2 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \mathcal{L}_{g_z^1} \tilde{h}_m & \dots & \mathcal{L}_{g_z^m} \tilde{h}_m & \dots & \mathcal{L}_{g_z^{m_w-z}} \tilde{h}_m & 0 & \dots & 0 \\ \mathcal{L}_{g_z^1} h_k^\infty & \dots & \mathcal{L}_{g_z^m} h_k^\infty & \dots & \mathcal{L}_{g_z^{m_w-z}} h_k^\infty & \mathcal{L}_{e_z^1} h_k^\infty & \dots & \mathcal{L}_{e_z^z} h_k^\infty \end{bmatrix},$$

From (34) we know that the rank of the submatrix that consists of the first m rows of the above matrix is equal to m . Let us consider the last row of the above matrix. By assumption, h_k^∞ depends on the time derivatives of order k_1, k_2, \dots, k_z of the last z unknown inputs. As a result, the last z entries of this row do not vanish. Therefore, the rank of the above matrix is $m + 1$. ◀

7.2. Description of Algorithm 9

We provide Algorithm 9, which builds, in a finite number of steps, the entire observable codistribution for any system that satisfies (1). These are its main features/outcomes:

- If the system is canonic, it finds a set of m_w observable functions, $\tilde{h}_1, \dots, \tilde{h}_{m_w}$, such that

$$\text{rank}(\mathcal{RM}(\tilde{h}_1, \dots, \tilde{h}_{m_w})) = m_w.$$

In this case, the above functions can be used to define the tensor μ in (9) and, consequently, they allow us to run Algorithm 1.

- If the system is non canonic but it is canonizable, it returns a finite unknown input extension which is canonic.
- If the system is neither canonic nor canonizable, it returns an unknown input extension of the original system with the highest unknown input degree of reconstructability. In addition, it also provides the observable codistribution.

The procedure is iterative. At each step, it computes new scalar functions that are observable. At the beginning these functions are the output functions (i.e, h_1, \dots, h_p).

The procedure consists of a main loop that contains a nested loop. The main loop terminates when one of the following two conditions is achieved:

1. m_w observable functions, $\tilde{h}_1, \dots, \tilde{h}_{m_w}$, such that $\text{rank}(\mathcal{RM}(\tilde{h}_1, \dots, \tilde{h}_{m_w})) = m_w$, are found.
2. The entire observable codistribution is obtained

Let us denote by m the rank of the unknown input reconstructability matrix, at a given step of the main loop. The procedure obtains new observable functions, when they exist. This is obtained by introducing the new system, denoted by $\mathcal{UIE}_{d\infty}$. Hence, the procedure can obtain new observable functions by running Algorithm 8. This is carried out by the nested loop. Because of the results stated by Proposition 2 and Lemma 1, the nested loop terminates, in at most $n - m$ steps, with one of the following outcomes:

1. Algorithm 8 converges (i.e., $O_k^\infty = O_{k-1}^\infty$), and the entire observable codistribution is obtained.
2. Algorithm 8 determines a new observable function such that the unknown input reconstructability matrix increases its rank by one.

In the first case, also the main loop terminates and we have achieved our goal because we have obtained the observable codistribution. In the second case, there are two possibilities depending on the new value of m , that is the rank of the new unknown input reconstructability matrix:

1. $m = m_w$.
2. $m < m_w$.

In the first case, the main loop terminates and we can run Algorithm 1 to obtain the observable codistribution. In the second case, we skip to the next iterative step of the main loop. Since the rank of the unknown input reconstructability matrix cannot exceed m_w , the procedure converges in at most $m_w - 1$ steps of the main loop.

We conclude with the following remark. Lemma 1 ensures that, the dependence of h_k^∞ on the extension $w_{d\infty}$ in (25), allows us to introduce a finite unknown input extension ($\Sigma = \mathcal{UIE}(k_1, k_2, \dots, k_z)$ in Definition 9) for which the rank of the reconstructability matrix from $\tilde{h}_1, \dots, \tilde{h}_m$ and h_k^∞ is $m + 1$. On the other hand, this is a sufficient condition that guarantees the possibility of increasing such a rank. We cannot exclude the possibility that, there exists a generator h_k^∞ , which is independent of $w_{d\infty}$, but still increases the rank of the reconstructability matrix. Considering this possibility is convenient, as, if it occurs, we do not need to extend the state and, consequently, we can reduce the computational load. In Algorithm 9, this possibility is accounted by the second *if* in the nested loop.

Algorithm 9: The algorithm that builds the observable codistribution for any nonlinear system characterized by (1).

```

Set  $m = \text{rank}(\mathcal{RM}(h_1, \dots, h_p))$ .
Select  $\tilde{h}_1, \dots, \tilde{h}_m$  from  $h_1, \dots, h_p$  s.t.
 $\text{rank}(\mathcal{RM}(\tilde{h}_1, \dots, \tilde{h}_m)) = m$ .
Set  $run\_find\_Extension = true$ 
while  $m < m_w$  and  $run\_find\_Extension$  do
  Set  $run\_find\_O = true$ 
  Set  $d = m_w - m$  and build  $\mathcal{UIE}_{d\infty}$ 
  while  $run\_find\_O$  do
    This is the nested loop and consists of the
    implementation of Algorithm 8.
    Its  $k^{th}$  iteration returns the generators of  $O_k^\infty$ 
     $\nabla h_k^\infty$  is a generic generator of  $O_k^\infty$ 
    if  $h_k^\infty$  contains component/s of  $w_{d\infty}$  then
      Set  $\tilde{h}_{m+1} \triangleq h_k^\infty$ 
      Reset the system as  $\Sigma$  in Definition 9
      Set  $m = m + 1$ 
      Set  $run\_find\_O = false$ 
    else
      if  $\text{rank}(\mathcal{RM}(\tilde{h}_1, \dots, \tilde{h}_m, h_k^\infty)) = m + 1$  then
        Set  $\tilde{h}_{m+1} \triangleq h_k^\infty$ 
        Set  $m = m + 1$ 
        Set  $run\_find\_O = false$ 
      end if
      if  $O_k^\infty = O_{k-1}^\infty$  then
        Set  $run\_find\_O = false$ 
        Set  $run\_find\_Extension = false$ 
      end if
    end if
  end while
end while
if  $run\_find\_Extension$  then
  The system under consideration is set in canonical form.
  This can be the original system or an unknown input
  extension.
  Compute  $O$  by applying Algorithm 1.
else
  The system is neither canonic nor canonizable. Algorithm
  8 has converged.
  Set  $O = O_k^\infty$ 
end if

```

8. Unknown input reconstruction

In this section, we investigate the problem of reconstructing the unknown inputs. We start by providing a property (Proposition 3) that establishes the equivalence between non-extensible systems (Definition 8) and systems with unknown input highest degree of reconstructability (Definition 6). Then, Section 8.2 provides a new result that characterizes the observability of the unknown inputs (Theorem 4). Finally, based on the strategy adopted to prove this result, we discuss how to set up a possible framework to estimate a set of unknown inputs. This framework is suitable both in case Theorem 4 tells us that these unknown inputs are observable and in the case they are not.

8.1. Non-extensible systems and unknown input highest degree of reconstructability

Let us refer to the system in (1) and let us suppose that we have $m < m_w$ observable functions, $\tilde{h}_1, \dots, \tilde{h}_m$, such that $\text{rank}(\mathcal{RM}(\tilde{h}_1, \dots, \tilde{h}_m)) = m$.

We have the following result:

Proposition 3. *The integer m is the unknown input highest degree of reconstructability if and only if the system is non-extensible.*

Proof (\Rightarrow) We proceed by contradiction. If the system is extensible we find a function h_k^∞ that depends on at least one component of w_{d_∞} . Lemma 1 proves that there exists a finite unknown input extension ($\mathcal{UIE}(k_1, k_2, \dots, k_z)$) with degree of reconstructability equal to $m + 1$. But this would mean that m is not the unknown input highest degree of reconstructability.

(\Leftarrow) We know that, for any observable function, there exists an integer k such that its differential belongs to \mathcal{O}_k^∞ (Theorem 3). Hence, the fact that the system is non-extensible means that all the observable functions are independent of all the components of w_{d_∞} . As a result, for any unknown input extension, the reconstructability matrix from any set of observable functions has a rank that cannot exceed m . Therefore, m is the unknown input highest degree of reconstructability. \blacktriangleleft

8.2. Observability of the unknown inputs

Based on the results obtained for the state observability it is immediate to prove the following fundamental result:

Theorem 4. *If a system is canonic with respect to its unknown inputs and its state is observable, then all the unknown inputs can be reconstructed.*

Proof Let us consider the system characterized by (1) and let us suppose that the state x is observable and that the system is canonic with respect to its unknown inputs. From the canonicity assumption, there exist m_w observable functions, $\tilde{h}_1, \dots, \tilde{h}_{m_w}$, such that the reconstructability matrix $\mathcal{RM}(\tilde{h}_1, \dots, \tilde{h}_{m_w})$ is non singular or, in other words, the tensor μ_j^i defined in (9) is non singular. We consider the unknown input extension obtained by including in the state all the unknown inputs (without any time derivative). In other words, we

are considering the system $\mathcal{UIE}(k_1, k_2, \dots, k_{m_w})$ with $k_1 = k_2 = \dots = k_{m_w} = 0$. We use the following result from the state of the art. Given an observable function of the state x , all the first order Lie derivatives of this function are observable functions for the above unknown input extension (see the first statement of Theorem 5 in Appendix C.1, or Theorem 6.11 in [31] or Proposition 3 in [27]¹⁵). Hence, we know that the m_w scalar functions obtained by computing the first order Lie derivative of $\tilde{h}_1, \dots, \tilde{h}_{m_w}$ along $g_{m_w}^0$ (which is the vector field in (23) for this specific unknown input extension) are observable functions or, in other words, their initial value can be reconstructed. We have:

$$\mathcal{L}_{g_{m_w}^0} \tilde{h}_j = \mathcal{L}_{g^0} \tilde{h}_j + \sum_{i=1}^{m_w} \mathcal{L}_{g^i} \tilde{h}_j w_i = \mu_j^0 + \sum_{i=1}^{m_w} \mu_j^i w_i \quad (35)$$

for $j = 1, \dots, m_w$. This is a system of m_w linear equations in the m_w unknowns: w_1, \dots, w_{m_w} . As μ is non singular, we invert and we obtain:

$$w_k = \sum_{l=1}^{m_w} \nu_k^l (\mathcal{L}_{g_{m_w}^0} \tilde{h}_l - \mu_l^0) \quad (36)$$

Now we use the assumption that the state is observable. This means that we know all the components of the state at the initial time and, consequently, we know all the components of ν and μ at the initial time. Therefore, from the above equation we can also reconstruct the unknown inputs \blacktriangleleft

8.3. Possible framework for reconstructing the unknown inputs

Theorem 4 provides a sufficient condition to check the observability of the unknown inputs. On the other hand, it could be also important from a practical point of view, namely to set up a possible framework that allows us to estimate a set of unknown physical quantities. In the remainder of this section, we only provide a preliminary discussion about this idea.

Let us suppose that we want to estimate one or more time-varying quantities and that we are able to build a dynamic system in which these quantities drive its dynamics by acting as unknown inputs. Thanks to Algorithm 9, we can check whether the state that characterizes this system is observable or not. In the second case, as Algorithm 9 builds the observable codistribution \mathcal{O} , we can try to characterize our system by an observable state. Note that this step is non trivial as it cannot be accomplished by a systematic/automatic procedure. Algorithm 9 provides an observable state automatically. However, for a complete description of our system, we need to express its dynamics only in terms of the components of the observable state and the system inputs. This step, not only cannot be accomplished automatically, but may also require a redefinition of the unknown inputs.

Let us suppose that we are able to obtain a description of our system in terms of an observable state. We have now the following two possibilities:

¹⁵Note that this result is widely used to introduce the strategies proposed in [28, 29].

1. The system is canonic with respect to its unknown inputs or canonizable.
2. The system is neither canonic with respect to its unknown inputs nor canonizable.

In the first case, if the system is canonizable but not canonic, we need to find a system characterization that is canonic. This is automatically built by Algorithm 9. Hence, in the first case, we achieve automatically a proper system characterization that allows us to estimate the unknown inputs. The strategy to perform this estimation depends on the specific case. We only mention that, on the basis of the proof of Theorem 4 (and in particular equation (36)), in order to design a suitable strategy, it could be very useful to have a set of observable functions, $\tilde{h}_1, \dots, \tilde{h}_{m_w}$, that makes the system canonic. These are automatically generated by Algorithm 9.

In the second case, we cannot obtain the unknown inputs because we cannot invert equation (35), for any choice of observable functions. On the other hand, Algorithm 9 provides, in a finite number of steps, the unknown input extension with the highest degree of reconstructability. Let us denote it by m . In addition, Algorithm 9 automatically builds a set of m observable functions, $\tilde{h}_1, \dots, \tilde{h}_m$, that makes the reconstructability matrix of rank m . Equation (35) becomes:

$$\mathcal{L}_{g_{m_w}^0} \tilde{h}_j = \mathcal{L}_{g^0} \tilde{h}_j + \sum_{i=1}^{m_w} \mathcal{L}_{g^i} \tilde{h}_j w_i \quad (37)$$

for $j = 1, \dots, m$. This is a system of $m < m_w$ linear equations in the m_w unknown inputs. We cannot obtain the unknown inputs as in the proof of Theorem 4. On the other hand, by using this linear system, we can easily obtain the linear combinations of w_1, \dots, w_{m_w} that can be reconstructed.

9. Applications

This section illustrates the implementation of Algorithm 9. We refer to the visual inertial sensor fusion problem. For the sake of clarity, we restrict our analysis to a 2D environment. The 3D case differs only in a more laborious calculation. We consider three variants of this sensor fusion problem depending on the available inertial signals. The description of these variants is given in Section 9.1. In Section 9.2, we use Algorithm 9 to obtain the observability properties. In particular, for one of the considered variants, we provide all the details of the computation, following, step by step, the implementation of Algorithm 9 (Section 9.2.2). Finally, in Section 9.3 we provide a concise study of the same observability problem by using the approach introduced in [27].

9.1. The system

We consider a rigid body (\mathcal{B}) equipped with a visual sensor (\mathcal{V}) and an inertial measurement unit (\mathcal{I}). The body moves on a plane. A complete \mathcal{I} measures the body acceleration and the angular speed. In 2D, the acceleration is a two dimensional vector and the angular speed is a scalar. The visual sensor provides

the bearing angle of the features in its own local frame. We assume that the \mathcal{V} frame coincides with the \mathcal{I} frame. We call this frame, the body frame. In addition, the inertial measurements are unbiased. Figure 3 depicts our system.

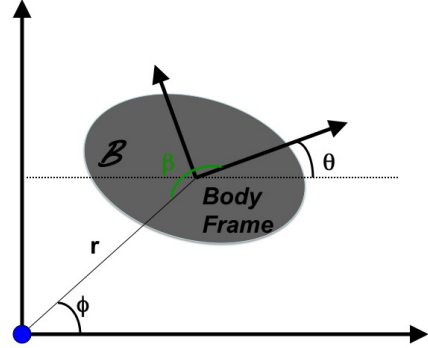


Figure 3: The global frame, the body frame and the observation provided by the \mathcal{V} sensor (β).

It is very convenient to work in polar coordinates. Hence, we define the state:

$$x = [r, \phi, v, \alpha, \theta]^T \quad (38)$$

where r and ϕ characterize the body position, θ its orientation (see Fig. 3 for an illustration), and v and α the body speed in polar coordinates. In particular, $v = \sqrt{v_x^2 + v_y^2}$ and $\alpha = \arctan\left(\frac{v_y}{v_x}\right)$, where $[v_x, v_y]^T$ is the body speed in Cartesian coordinates. The dynamics are:

$$\begin{cases} \dot{r} &= v \cos(\alpha - \phi) \\ \dot{\phi} &= \frac{v}{r} \sin(\alpha - \phi) \\ \dot{v} &= A_x \cos(\alpha - \theta) + A_y \sin(\alpha - \theta) \\ \dot{\alpha} &= -\frac{A_x}{v} \sin(\alpha - \theta) + \frac{A_y}{v} \cos(\alpha - \theta) \\ \dot{\theta} &= \omega \end{cases} \quad (39)$$

where $[A_x, A_y]^T$ is the body acceleration in the body frame and ω the angular speed. Without loss of generality, we assume that the feature is positioned at the origin of the global frame. The \mathcal{V} sensor provides the angle $\beta = \pi - \theta + \phi$. Hence, we can perform the observability analysis by using the output (we ignore π):

$$y = h(x) = \phi - \theta \quad (40)$$

We consider the following three variants, which differ in a different setting of the \mathcal{I} sensor:

1. Variant 1: \mathcal{I} provides the body acceleration ($[A_x, A_y]^T$) and the angular speed (ω).
2. Variant 2: \mathcal{I} consists of a single gyroscope that only provides the angular speed (ω).
3. Variant 3: \mathcal{I} consists of a single accelerometer that only provides the component of the acceleration along the x -axis of the body frame (A_x).

9.2. State observability

We use Algorithm 9 to obtain the observability properties of our system. For the brevity sake, for Variant 3, we do not provide the details of the computation (which are similar to the ones for Variant 2) but only the result.

9.2.1. State observability for Variant 1

When A_x , A_y , and ω are known, by comparing (39) and (40) with (1) we have: $n = 5$, $m_u = 3$, $m_w = 0$, $p = 1$, $u_1 = \omega$, $u_2 = A_x$, $u_3 = A_y$,

$$g^0 = \begin{bmatrix} v \cos(\alpha - \phi) \\ \frac{v}{r} \sin(\alpha - \phi) \\ 0 \\ 0 \\ 0 \end{bmatrix}, f^1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix},$$

$$f^2 = \begin{bmatrix} 0 \\ 0 \\ \cos(\alpha - \theta) \\ -\frac{1}{v} \sin(\alpha - \theta) \\ 0 \end{bmatrix}, f^3 = \begin{bmatrix} 0 \\ 0 \\ \sin(\alpha - \theta) \\ \frac{1}{v} \cos(\alpha - \theta) \\ 0 \end{bmatrix}$$

As all the inputs are known, we can compute the observable codistribution by using the standard observability rank condition, which is a special case of Algorithm 1 when $m_w = 0$.

We obtain, at the initialization, $\mathcal{O}_0 = \text{span}\{\nabla h\} = \text{span}\{[0 \ 1 \ 0 \ 0 \ -1]\}$ (note that, when $m_w = 0$, the codistribution $\bar{\mathcal{O}}$ vanishes). Its dimension is equal to 1. At the first iterative step we obtain $\mathcal{O}_1 = \text{span}\{\nabla h, \nabla \mathcal{L}_{g^0} h\}$ and its dimension is $2 > 1$. At the second iterative step we obtain $\mathcal{O}_2 = \text{span}\{\nabla h, \nabla \mathcal{L}_{g^0} h, \nabla \mathcal{L}_{g^0} \mathcal{L}_{g^0} h, \nabla \mathcal{L}_{f^2} \mathcal{L}_{g^0} h\}$ and its dimension is $4 > 2$. At the next step \mathcal{O}_3 remains the same, meaning that the algorithm has converged. Therefore:

$$\mathcal{O} = \mathcal{O}_2 = \text{span}\{\nabla h, \nabla \mathcal{L}_{g^0} h, \nabla \mathcal{L}_{g^0} \mathcal{L}_{g^0} h, \nabla \mathcal{L}_{f^2} \mathcal{L}_{g^0} h\}$$

We compute the orthogonal distribution. By a direct computation, we obtain:

$$(\mathcal{O})^\perp = \text{span}\{[0 \ 1 \ 0 \ 1 \ 1]^T\}.$$

The generator of $(\mathcal{O})^\perp$ expresses the following invariance (ϵ is an infinitesimal parameter):

$$\begin{bmatrix} r \\ \phi \\ v \\ \alpha \\ \theta \end{bmatrix} \rightarrow \begin{bmatrix} r' \\ \phi' \\ v' \\ \alpha' \\ \theta' \end{bmatrix} = \begin{bmatrix} r \\ \phi \\ v \\ \alpha \\ \theta \end{bmatrix} + \epsilon \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 1 \end{bmatrix}, \quad (41)$$

which is an infinitesimal rotation about the vertical axis.

In conclusion, for this system (Variant 1), only the first and the third component of the state are observable (i.e., r and v), while the three remaining components, ϕ , α , and θ , are not. However, the difference between any two of these angles (e.g., $\phi - \theta$, or $\alpha - \theta$) is observable.

9.2.2. State observability for Variant 2

When A_x , A_y are unknown and ω is known, by comparing (39) and (40) with (1) we have: $n = 5$, $m_u = 1$, $m_w = 2$, $p = 1$, $u_1 = \omega$, $w_1 = A_x$, $w_2 = A_y$,

$$g^0 = \begin{bmatrix} v \cos(\alpha - \phi) \\ \frac{v}{r} \sin(\alpha - \phi) \\ 0 \\ 0 \\ 0 \end{bmatrix}, f^1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix},$$

$$g^1 = \begin{bmatrix} 0 \\ 0 \\ \cos(\alpha - \theta) \\ -\frac{1}{v} \sin(\alpha - \theta) \\ 0 \end{bmatrix}, g^2 = \begin{bmatrix} 0 \\ 0 \\ \sin(\alpha - \theta) \\ \frac{1}{v} \cos(\alpha - \theta) \\ 0 \end{bmatrix}$$

As $p = 1 < 2 = m_w$ the system is certainly not in canonical form with respect to w_1 and w_2 . We even do not know if it is canonic with respect to them. We apply Algorithm 9.

We obtain $\mathcal{L}_{g^1} h = \mathcal{L}_{g^2} h = 0$. As a result, $\text{rank}(\mathcal{RM}(h)) = 0$ and

$$m = 0, \quad d = m_w - 0 = 2$$

Hence, we build the system $\mathcal{UIE}_{2\infty}$ and we apply Algorithm 8.

$\mathcal{UIE}_{2\infty}$

We follow the steps in section 5.2 to build this system. It is characterized by the following state:

$$x_{2\infty} = [x^T, A_x, A_x^{(1)}, A_x^{(2)}, A_x^{(3)}, \dots, A_y, A_y^{(1)}, A_y^{(2)}, A_y^{(3)}, \dots]^T$$

and the equations in (26) with:

$$g_{2\infty}^0 = \begin{bmatrix} g^0 + g^1 A_x + g^2 A_y \\ A_x^{(1)} \\ A_x^{(2)} \\ A_x^{(3)} \\ \dots \\ A_y^{(1)} \\ A_y^{(2)} \\ A_y^{(3)} \\ \dots \end{bmatrix}, f_{2\infty}^1 = \begin{bmatrix} f^1 \\ 0 \\ 0 \\ 0 \\ \dots \\ 0 \\ 0 \\ 0 \\ \dots \end{bmatrix}.$$

Application of Algorithm 8 to $\mathcal{UIE}_{2\infty}$

In this special case ($m = 0$), the tensor ${}^m \mu = {}^0 \mu$ in (29) has the single component ${}^0 \mu_0^0 = 1$, and, consequently, the same holds for its inverse ${}^0 \nu$: we have ${}^0 \nu_0^0 = 1$. In addition, we only have the vector field ${}^0 \hat{g}_{2\infty}^0$ in (30) and we have ${}^0 \hat{g}_{2\infty}^0 = g_{2\infty}^0$. As a result, the steps of Algorithm 8 trivially consist of the computation of the Lie derivatives along the drift and the vector field that corresponds to the known input, i.e., the above $g_{2\infty}^0$ and $f_{2\infty}^1$. In addition, the initialization step is trivial because $\bar{\mathcal{O}}$ vanishes. As a result, $\mathcal{O}_0^\infty = \text{span}\{\nabla h\}$.

At the first step we obtain:

$$\begin{aligned} \mathcal{O}_1^\infty &= \text{span} \left\{ \nabla h, \nabla \left(\mathcal{L}_{g_{2\infty}^0} h \right), \nabla \left(\mathcal{L}_{f_{2\infty}^1} h \right) \right\} = \\ &\text{span} \left\{ \nabla h, \nabla \left(\frac{v}{r} \sin(\alpha - \phi) \right) \right\} \end{aligned}$$

Both the generators are independent of $w_{2\infty}$. On the other hand, for the latter we have:

$$\text{rank} \left(\mathcal{RM} \left(\frac{v}{r} \sin(\alpha - \phi) \right) \right) = 1 > m = 0$$

In other words, we meet the condition stated by the second *if* in the nested loop. Hence, we set:

$$\tilde{h}_{m+1} = \tilde{h}_1 \triangleq \frac{v}{r} \sin(\alpha - \phi),$$

$$m = m + 1 = 1,$$

and we stop Algorithm 8 (*run_find_O = false*).

As $m = 1 < m_w = 2$ we repeat the iterative step in the main loop of Algorithm 9. We have now $d = m_w - m = 2 - 1 = 1$. We build $\mathcal{UIE}_{1\infty}$.

$\mathcal{UIE}_{1\infty}$

We follow the steps in section 5.2 to build this system. We obtain the state by including one of the two unknown inputs, together with all its time derivatives. We can choose any of them, as both the entries of $\mathcal{RM} \left(\frac{v}{r} \sin(\alpha - \phi) \right)$ do not vanish. We choose A_x and we have:

$$x_{1\infty} = [x^T, A_x, A_x^{(1)}, A_x^{(2)}, A_x^{(3)}, \dots]^T$$

$\mathcal{UIE}_{1\infty}$ is characterized by this state and the equations in (26) with:

$$g_{1\infty}^0 = \begin{bmatrix} g^0 + g^1 A_x \\ A_x^{(1)} \\ A_x^{(2)} \\ A_x^{(3)} \\ \dots \end{bmatrix}, f_{1\infty}^1 = \begin{bmatrix} f^1 \\ 0 \\ 0 \\ 0 \\ \dots \end{bmatrix}, g_{1\infty}^1 = \begin{bmatrix} g^2 \\ 0 \\ 0 \\ 0 \\ \dots \end{bmatrix}$$

Application of Algorithm 8 to $\mathcal{UIE}_{1\infty}$

We apply Algorithm 8 to $\mathcal{UIE}_{1\infty}$ (nested loop in Algorithm 9). We have:

$$\mathcal{O}_0^\infty = \text{span} \left\{ \nabla h, \nabla \tilde{h}_1 \right\} + \tilde{\mathcal{O}}$$

To proceed, we need to compute $\tilde{\mathcal{O}}$. This requires the computation of the integers r and s_x and the autobrackets of $f_{1\infty}^1$ up to the $(r + s_x)^{\text{th}}$ order. We need, first of all, to compute the tensor ${}^m \mu = {}^1 \mu$ in (28) and (29). We obtain: ${}^1 \mu_0^0 = 1, {}^1 \mu_0^1 = 0,$

$${}^1 \mu_1^0 = \mathcal{L}_{g_{1\infty}^0} \tilde{h}_1 = -\frac{\sin(2\alpha - 2\phi)v^2 + A_x r \sin(\phi - \theta)}{r^2},$$

$${}^1 \mu_1^1 = \mathcal{L}_{g_{1\infty}^1} \tilde{h}_1 = \frac{\cos(\phi - \theta)}{r}$$

By computing its inverse we obtain: ${}^1 v_0^0 = 1, {}^1 v_0^1 = 0,$

$${}^1 v_1^0 = \frac{\sin(2\alpha - 2\phi)v^2 + A_x r \sin(\phi - \theta)}{r \cos(\phi - \theta)}, \quad {}^1 v_1^1 = \frac{r}{\cos(\phi - \theta)}$$

We start by computing the first order autobracket of $f_{1\infty}^1$. We obtain, for the former: ${}^1 [f_{1\infty}^1]^0 =$

$$\begin{bmatrix} 0 \\ 0 \\ \frac{\cos(\alpha - \theta)(\sin(2\alpha - 2\phi)v^2 + r A_x \sin(\phi - \theta))}{r \cos(\phi - \theta)} - A_x \sin(\alpha - \theta) \\ -\frac{A_x \cos(\alpha - \theta)}{v} - \frac{\sin(\alpha - \theta)(\sin(2\alpha - 2\phi)v^2 + r A_x \sin(\phi - \theta))}{r v \cos(\phi - \theta)} \\ 0 \\ 0 \end{bmatrix}$$

and $\mathcal{L}_{[f_{1\infty}^1]^0} \tilde{h}_1 =$

$$-\frac{2r A_x - v^2 \cos(\phi - 2\alpha + \theta) + v^2 \cos(2\alpha - 3\phi + \theta)}{2r^2 \cos(\phi - \theta)}$$

The differential of the above function is one of the generator of $\tilde{\mathcal{O}}$. As it depends on A_x , we meet the condition stated by the first *if* in the nested loop. Hence, we set: $\tilde{h}_{m+1} =$

$$\tilde{h}_2 \triangleq -\frac{2r A_x - v^2 \cos(\phi - 2\alpha + \theta) + v^2 \cos(2\alpha - 3\phi + \theta)}{2r^2 \cos(\phi - \theta)},$$

$$m = m + 1 = 2 = m_w.$$

Additionally, we reset the system as Σ in Definition 9, and we stop Algorithm 8 (*run_find_O = false*).

The new system Σ

The new system (Σ in Definition 9) is obtained starting from the dependency of the new function \tilde{h}_2 on $w_{d\infty}$. In this specific case, we have $z = 1$ and $k_z = 0$ (the function only depends on A_x and it is independent of any order time derivative of A_x). The new state is:

$$x = [r, \phi, v, \alpha, \theta, A_x]^T$$

Σ is characterized by a single known input (as the starting system), which is $u_1 = \omega$. It is also characterized by two unknown inputs. However, the second unknown input is now the first order time derivative of the first unknown input of the original system. In other words, $w_2 = A_x^{(1)} = \dot{A}_x$. The first unknown input coincides with the second unknown input of the original system, i.e., $w_1 = A_y$. The vector fields that characterize the state dynamics are:

$$g^0 = \begin{bmatrix} v \cos(\alpha - \phi) \\ \frac{v}{r} \sin(\alpha - \phi) \\ A_x \cos(\alpha - \theta) \\ -\frac{A_x}{v} \sin(\alpha - \theta) \\ 0 \\ 0 \end{bmatrix}, f^1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix},$$

$$g^1 = \begin{bmatrix} 0 \\ 0 \\ \sin(\alpha - \theta) \\ \frac{1}{v} \cos(\alpha - \theta) \\ 0 \\ 0 \end{bmatrix}, g^2 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}.$$

Finally, the nested loop returned $m = 2$, which is equal to m_w . Therefore, also the main loop is stopped and we can run Algorithm 1 on the above system because it is in canonical form. In particular, we set its outputs as the original output and the two functions \tilde{h}_1 and \tilde{h}_2 , namely:

$$h_1 = h = \phi - \theta, h_2 = \tilde{h}_1 = \frac{v}{r} \sin(\alpha - \phi),$$

and

$$h_3 = \tilde{h}_2 = -\frac{2rA_x - v^2 \cos(\phi - 2\alpha + \theta) + v^2 \cos(2\alpha - 3\phi + \theta)}{2r^2 \cos(\phi - \theta)}.$$

Application of Algorithm 1 to the system in canonical form

The initialization consists of the computation of the following codistribution:

$$\mathcal{O}_0 = \text{span}\{\nabla h_1, \nabla h_2, \nabla h_3\} + \tilde{\mathcal{O}}$$

To proceed, we need to compute $\tilde{\mathcal{O}}$. This requires the computation of the integers r and s and the autobrackets of f^1 up to the $(r + s)^{\text{th}}$ order. We need, first of all, to compute the tensor μ . We obtain: $\mu_0^0 = 1, \mu_0^1 = 0, \mu_0^2 = 0, \mu_1^0 = -\frac{\sin(2\alpha - 2\phi)v^2 + rA_x \sin(\phi - \theta)}{r^2}, \mu_1^1 = \frac{\cos(\phi - \theta)}{r}, \mu_1^2 = 0, \mu_2^0 = \frac{1}{r^3(\cos(2\phi - 2\theta) + 1)}[v^3 \cos(3\alpha - 3\phi) - v^3 \cos(\phi - 3\alpha + 2\theta) - v^3 \cos(\alpha - \phi) + v^3 \cos(3\alpha - 5\phi + 2\theta) - rvA_x \cos(\alpha - 4\phi + 3\theta) + 3rvA_x \cos(\alpha - \theta)], \mu_2^1 = \frac{v \sin(\alpha - 3\phi + 2\theta) - v \sin(\alpha - \phi)}{r^2 \cos(\phi - \theta)}, \mu_2^2 = -\frac{1}{r \cos(\phi - \theta)}$. By computing its inverse, we obtain the tensor ν (we do not provide its analytical expression, for the brevity sake).

We compute the integers s and r by running Algorithm 2 and Algorithm 3, respectively. We obtain $s = r = 2$. Finally, we use (20) to compute $\tilde{\mathcal{O}}$. We have:

$$\mathcal{O}_0 = \text{span}\{\nabla h_1, \nabla h_2, \nabla h_3, \nabla h_4\}$$

The new generator, is $h_4 = \frac{1}{2r^2 \cos(\phi - \theta)^2}[\sin(\phi - \theta)(2rA_x - v^2 \cos(\phi - 2\alpha + \theta) + v^2 \cos(2\alpha - 3\phi + \theta))] - \cos(\phi - \theta)[v^2 \sin(\phi - 2\alpha + \theta) - v^2 \sin(2\alpha - 3\phi + \theta)]$.

This concludes the initialization step. By running the recursive step of Algorithm 1 we obtain that

$$\mathcal{O}_1 = \mathcal{O}_0$$

meaning that Algorithm 1 has converged and, the observable codistribution is:

$$\mathcal{O} = \text{span}\{\nabla h_1, \nabla h_2, \nabla h_3, \nabla h_4\}.$$

We compute the orthogonal distribution. By a direct computation, we obtain:

$$(\mathcal{O})^\perp = \text{span}\{[0 \ 1 \ 0 \ 1 \ 1 \ 0]^T, [r \ 0 \ v \ 0 \ 0 \ A_x]^T\}.$$

The first generator of $(\mathcal{O})^\perp$ expresses the same invariance found in the case discussed in section 9.2.1. This is the rotation around the vertical axis (see equation (41)). This fact is not surprising. By removing the accelerometer we lose information and, consequently, all the degrees of unobservability remain.

The second generator expresses the following new invariance (ϵ is an infinitesimal parameter):

$$\begin{bmatrix} r \\ \phi \\ v \\ \alpha \\ \theta \\ A_x \end{bmatrix} \rightarrow \begin{bmatrix} r' \\ \phi' \\ v' \\ \alpha' \\ \theta' \\ A'_x \end{bmatrix} = \begin{bmatrix} r \\ \phi \\ v \\ \alpha \\ \theta \\ A_x \end{bmatrix} + \epsilon \begin{bmatrix} r \\ 0 \\ v \\ 0 \\ 0 \\ A_x \end{bmatrix}, \quad (42)$$

which is an infinitesimal scale transform. The presence of this new degree of unobservability is also not surprising. All the information is provided by the measurements that now only consist of angular measurements (the sensor \mathcal{V} only provides the angle β in Fig. 3 and the sensor \mathcal{I} only provides the angular speed). As a result, the system does not have any source of metric information.

In conclusion, for this system (Variant 2), no component of the state is observable. However, as for Variant 1, the difference between any two of the three angles ϕ, α , and θ , (e.g., $\phi - \theta$, or $\alpha - \theta$) is observable. In addition, regarding the metric quantities in the new state (i.e., r, v , and A_x), the ratio between any two of them (e.g., $\frac{v}{r}$, or $\frac{A_x}{r}$) is observable.

Note that, even if the above computation could seem laborious, it is automatic. In particular, it is carried out by a simple code that uses symbolic computation.

9.2.3. State observability for Variant 3

By running Algorithm 9, we obtain that the observable codistribution coincides with the one obtained in Section 9.2.1, i.e., in the case of a complete \mathcal{I} (Variant 1). In other words, in this case, the system has a single degree of unobservability that is the system invariance against a rotation around the vertical axis.

In general, it is trivial to realize that, by adding new sensors, the observability of the system cannot deteriorate. It is less trivial to realize in which cases, by adding a new sensor, the observability of the state improves and in which it remains the same. In our examples, if we add a new single-axis accelerometer to the system in Variant 2, the observability improves. Conversely, if we add a gyroscope and/or a new single-axis accelerometer to the system in Variant 3, the observability remains the same. An explanation for the first case was done at the end of Section 9.2.2: the new sensor (the accelerometer) provides a new type of information that is the absolute scale. This information is capable of breaking the system symmetry in (42). The second case can be intuitively explained when the vision sensor observes at least 5 points, simultaneously. Indeed, with at least 5 points, a monocular vision sensor alone provides the angular speed ω [33]. It is less intuitive to realize the redundancy of the gyroscope for Variant 3 in the presence of a single point (in all our systems the vision sensor observes only one point).

k	$\dim(x_{km_w})$	$\dim(\overline{\Omega}_k)$		# states		# obs	
		V2	V3	V2	V3	V2	V3
0	5	1	1	0	0	1	1
1	7	2	2	0	0	1	1
2	9	3	4	0	1	1	2
3	11	5	7	0	2	3	4
4	13	8	9	0	2	3	4

Table 1: Results from Algorithm 10, for Variant 2 (V2) and Variant 3 (V3). The step (k) of the algorithm (1st column), the dimension of the extended state (2nd column), the dimension of $\overline{\Omega}_k$ (3^d column), the number of the components of the original state with differential in $\overline{\Omega}_k$ (4th column), the number of all the independent observable functions with differential in $\overline{\Omega}_k$ (5th column).

9.3. Results obtained with the approach introduced in [27]

We study the same observability problem by using the approach proposed in [27]. It consists in the implementation of Algorithm 10 in Appendix C.1. We emphasize that this algorithm does not converge alone. To understand what is the step after which the algorithm does not provide further information on the original state, we actually exploited our knowledge of all the observable functions. Specifically, we know that a choice of all the independent observable functions is $\frac{v}{r}$, $\phi - \theta$, $\alpha - \theta$, for Variant 2, and v , r , $\phi - \theta$, $\alpha - \theta$, for Variant 3 (see sections 9.2.2 and 9.2.3).

Table 1 provides the results obtained by running Algorithm 10. The last column provides how many independent observable functions have the differential in $\overline{\Omega}_k$. We can see that, for both Variant 2 and 3, we achieve the totality (3 and 4) at the third step. We emphasize that, in order to obtain the values of this last column, we exploit some extra knowledge that is not obtained by using Algorithm 10. Therefore, to get a complete answer with this approach, we are actually using the solution of the problem itself (which is the determination of all the observable functions). We also emphasize that the computation becomes prohibitive starting from the 8th step (the computation of $\overline{\Omega}_8$ takes almost 10 days by using the symbolic tool of MATLAB with the processor 3.1 GHz Intel Core i7). Finally, we highlight that the results obtained with this algorithm agree with the results obtained with Algorithm 9.

10. Conclusion

This paper provided the general analytical solution of a fundamental open problem introduced long time ago (in the middle of the 1960's). The problem is to find an analytical and automatic test able to check the observability of the state that characterizes a system whose dynamics are also driven by inputs that are unknown. This is the extension of the well known *Observability Rank Condition* introduced in the 1970's and that does not account for the presence of unknown inputs. This paper provided this extension. This problem arises in a large class of domains, ranging from mechanical engineering, robotics, computer vision, up to biology, chemistry and economics. Observability is a fundamental structural property of any dynamic

system and describes the possibility of reconstructing the state that characterizes the system from observing its inputs and outputs. The dynamics of most real systems are driven by inputs that are usually unknown. Very surprisingly, the complexity of the solution here introduced is comparable to the complexity of the observability rank condition. Given any nonlinear system characterized by any type of nonlinearity, driven by both known and unknown inputs, the state observability is obtained automatically, i.e., by the usage of a very simple code that uses symbolic computation. This is a fundamental practical (and unexpected) advantage. To obtain this solution, the paper used several concepts recently introduced [31]. Specifically, it used the *Group of invariance of observability*, and the solution was expressed in terms of a new set of tensor fields with respect to the above group of invariance. In addition, the paper introduced the new concepts of *canonic system with respect to the unknown inputs* (Definition 3), *canonical form with respect to the unknown inputs* (Definition 4), *canonizable system* (Definition 7), and *unknown input highest degree of reconstructability* (Definition 6). These new definitions allowed us to achieve the solution of the unknown input observability problem even for systems not canonic and not even canonizable. Note that, the solution provided in [31] is based on the assumption that the system is in canonical form with respect to its unknown inputs. The paper reported the solution in [31] (Algorithm 4) by also providing an improved solution (Algorithm 1), which exhibits an easier and unrestricted convergence criterion. Then, the paper provided the automatic procedure that, in a finite number of steps, provides all the observability properties for systems neither in canonical form, nor canonizable. This automatic procedure is Algorithm 9. It uses iteratively Algorithm 8, which is a special implementation of Algorithm 1.

Finally, as a simple consequence of the results here obtained, the paper provided the answer to the problem of unknown input reconstruction, which is intimately related to the problem of state observability.

The solution was illustrated by a simple application. We studied the observability properties of a nonlinear system in the framework of visual-inertial sensor fusion. The dynamics of this system are driven by two unknown inputs and one known input and they are also characterized by a nonlinear drift. The system is not in canonical form with respect to its unknown inputs. However, by following the steps of Algorithm 9, it was set in canonical form. We emphasize that we already dealt with the same sensor fusion problem in 3D, in the presence of biased measurements and when the visual sensor is not extrinsically calibrated. For this system the state dimension is 18 and the overall time requested for the computation of the entire observable codistribution was about 1 hour by using the symbolic tool of MATLAB with the processor 3.1 GHz Intel Core i7. On the other hand, we also highlight that the computation is symbolic and it is not clear how the time requested for the computation scales for more complex systems.

Future research will be devoted, from one side, to develop an open source toolbox to study very complex observability problems (this will make use of a combination of symbolic and random numerical computation). From another side, it will be very

interesting to extend these results (or at least part of them), to the problem of controllability in the presence of unknown inputs and to the problem of detectability. Both these concepts are strongly related to the concept of observability.

Appendix A. Proof of Theorem 1

The proof of Theorem 1 requires the introduction of two fundamental new ingredients:

- The set of generators ψ .
- The quantities $C_{\mathbf{kz}}^\alpha$.

Additionally, this proof is based on the validity of Equation (B.12), which is the result stated by Proposition 4. This is a technical result and it will be given, separately, in Appendix B.

The set of generators ψ

We denote by ψ the vectors computed by Algorithm 3 before its convergence. Note that these vectors generate the entire distribution Δ and they will be not necessarily independent. They will be listed by using a double index $\mathbf{i} \triangleq (i, j_i)$. The first index is the index of the step of Algorithm 3. It takes the values $i = 0, 1, \dots, r$. The second index lists all the vectors which are computed by Algorithm 3, when computing Δ_i . For this proof, we do not need to specify how we list these vectors. We only emphasize that at the initialization (computation of Δ_0) we have m_u vectors: $\psi_{(0,1)} = f^1, \psi_{(0,2)} = f^2, \dots, \psi_{(0,m_u)} = f^{m_u}$. In other words, the second index j_0 take the values $1, 2, \dots, m_u$. Then, when computing Δ_i , we have, for every $\psi_{i-1,j_{i-1}}$, $m_w + 1$ vectors with the first index equal to i (each of them is obtained by computing the autobracket $[\psi_{i-1,j_{i-1}}]^\alpha$, for $\alpha = 0, 1, \dots, m_w$). Therefore, the index j_i , take the values: $1, 2, \dots, (m_w + 1)^i m_u$.

The vectors $\psi_{\mathbf{k}} = \psi_{(k,j_k)}$ for $k = 0, 1, \dots, r$, generate the entire distribution Δ . However, they are not independent, in general.

The quantities $C_{\mathbf{kz}}^\alpha$

We introduce the following quantities C . They are characterized by three indices:

$$C_{\mathbf{kz}}^\alpha$$

The lower indices \mathbf{k} and \mathbf{z} are double indices, as explained above ($\mathbf{k} = (k, j_k)$ and $\mathbf{z} = (z, j_z)$, with $k, z = 0, 1, \dots, r$). The upper index α takes the $m_w + 1$ values: $0, 1, \dots, m_w$. These quantities are implicitly defined by the following equation:

$$[\psi_{\mathbf{k}}]^\alpha = \sum_{\mathbf{z}} C_{\mathbf{kz}}^\alpha \psi_{\mathbf{z}} \quad (\text{A.1})$$

where $\sum_{\mathbf{z}} = \sum_{(z,j_z)}$ stands for $\sum_{z=0}^r \sum_{j_z=1}^{(m_w+1)^z m_u}$. Indeed, the term at the left hand side of the above equation belongs to Δ and can be expressed in terms of the set of vectors ψ (we remind the reader that Δ is invariant under the autobracket operation, which is the operation $[\cdot]^\alpha$ at the left side of (A.1)). As the vectors ψ are not necessarily independent, the choice of C is not unique. When the index $\mathbf{k} = (k, j_k)$ is such that $k < r$ we trivially choose C such that it selects the vector ψ with the first index incremented by one. For $\mathbf{k} = (r, j_r)$ the expression of $C_{\mathbf{kz}}^\alpha$ is not trivial and provides $[\psi_{\mathbf{k}}]^\alpha$ in terms of the vectors $\psi_{\mathbf{z}}$ with $\mathbf{z} = (z, j_z)$ and $z = 0, 1, \dots, r$. The choice is not unique, as the vectors are not independent. The result is independent of the choice (in other words, any choice can be done).

Now, we have all the ingredients to prove Theorem 1.

Proof Let us prove the first statement. From the $(k + 1)^{\text{th}}$ step of Algorithm 4, we obtain that $\sum_{q=1}^{m_w} \sum_{i=1}^{m_u} \sum_{\alpha_1=0}^{m_w} \dots \sum_{\alpha_k=0}^{m_w} \text{span} \left\{ \mathcal{L}_{[f^i]^{(\alpha_1, \dots, \alpha_k)}} \nabla \tilde{h}_q \right\} \in \Omega_{k+1}$. From (20), we immediately obtain that

$$\bar{\mathcal{O}} \subseteq \Omega_{r+s+1}.$$

By comparing the recursive step of Algorithm 1 with the one of Algorithm 4 it is immediate to conclude that, for any integer $v \geq 0$:

$$\mathcal{O}_v \subseteq \Omega_{r+s+1+v}$$

and, as Algorithm 1 converges in at most $n - m_w$ steps to \mathcal{O} , we conclude that:

$$\mathcal{O} \subseteq \Omega_{r+s+1+n-m_w}$$

and this proves the first statement with $\widehat{k} = r + s + 1 + n - m_w \leq 3n - 2m_w + 1$.

Let us prove the second statement. We must prove that, for any k , $\Omega_k \subseteq \mathcal{O}$. We proceed by induction. It is true at $k = 0$ as $\Omega_0 \subseteq \mathcal{O}_0 \subseteq \mathcal{O}$.

Let us assume that

$$\Omega_p \subseteq \mathcal{O}$$

We must prove that

$$\Omega_{p+1} \subseteq \mathcal{O}$$

As \mathcal{O} is invariant under \mathcal{L}_{g^β} and \mathcal{L}_{f^i} , we only need to prove that

$$\nabla \mathcal{L}_{[f^i]^{(\alpha'_1, \dots, \alpha'_p)}} \tilde{h}_q \in \mathcal{O}$$

for any choice of $\alpha'_1, \dots, \alpha'_p = 0, 1, \dots, m_w$, and every $q = 1, \dots, m_w$. We distinguish the following two cases:

1. $p \leq r + s$.
2. $p \geq r + s + 1$.

The first case is trivial as $\nabla \mathcal{L}_{[f^i]^{(\alpha'_1, \dots, \alpha'_p)}} \tilde{h}_q \in \bar{\mathcal{O}} \subseteq \mathcal{O}$.

Let us consider the second case. We start by remarking that, when $p \geq r + s + 1$ we have:

$$[f^i]^{(\alpha'_1, \dots, \alpha'_p)} = [\psi_{(r,j_r)}]^{(\alpha'_{r+1}, \dots, \alpha'_p)}$$

where j_r is such that $[f^i]^{(\alpha'_1, \dots, \alpha'_p)} = \psi_{(r,j_r)}$.

We set $p = r + s + l'$, with $l' \geq 1$, and we use Equation (B.12), with $l = s + l'$, and $\alpha_1, \dots, \alpha_l = \alpha'_{r+1}, \dots, \alpha'_p$. We have:

$$\nabla \mathcal{L}_{[f^i]^{(\alpha'_1, \dots, \alpha'_p)}} \tilde{h}_q = \nabla \mathcal{L}_{[\psi_{(r,j_r)}]^{(\alpha'_{r+1}, \dots, \alpha'_p)}} \tilde{h}_q = \quad (\text{A.2})$$

$$\sum_{\beta^{(l-1)}} M_{\beta_1, \dots, \beta_{l-1}}^{\alpha'_{r+2}, \dots, \alpha'_p} \sum_{\mathbf{z}} \left(\nabla \mathcal{C}_{(r,j_r)}^{\alpha'_{r+1}} \right) \mathcal{L}_{\psi_{\mathbf{z}}} \mathcal{L}_{g^{\beta_{p-r}}} \dots \mathcal{L}_{g^{\beta_1}} \tilde{h}_q \quad \text{mod } \Omega_p + \mathcal{L}_{g^\beta} \Omega_p$$

where:

- The sum $\sum_{\beta^{(l-1)}}$ is defined as follows:

$$\sum_{\beta^{(l-1)}} \triangleq \sum_{\beta_1=0}^{m_w} \sum_{\beta_2=0}^{m_w} \dots \sum_{\beta_{l-1}=0}^{m_w},$$

- The sum $\sum_{\mathbf{z}}$ is defined as follows:

$$\sum_{\mathbf{z}} = \sum_{(z,j_z)} \triangleq \sum_{z=0}^r \sum_{j_z=1}^{(m_w+1)^z m_u}$$

- M is a suitable multi-index object, which is non singular (i.e., it can be inverted with respect to all its indices).
- $\mathcal{L}_{\tilde{g}} \Omega_p$ stands for $\sum_{\gamma=0}^{m_w} \mathcal{L}_{\tilde{g}^\gamma} \Omega_p$
- the mod at the end of Equation (A.2) means that the equality holds when the codistribution $\Omega_p + \mathcal{L}_{\tilde{g}} \Omega_p$ is added on both sides.

The validity of the above result is proved in Appendix B (see Proposition 4).

Let us consider now the function $\mathcal{L}_{g^{\beta_{p-r-1}}} \dots \mathcal{L}_{g^{\beta_1}} \tilde{h}_q$. Its differential belongs to Ω^g .

We denote by t the dimension of Ω^g and by h_1^g, \dots, h_t^g a basis of Ω^g , i.e.:

$$\Omega^g = \text{span} \{ \nabla h_1^g, \dots, \nabla h_t^g \}$$

In addition, we choose these generators (h_1^g, \dots, h_t^g) directly among the functions built by Algorithm 2. As a result,

$$\nabla \mathcal{L}_{g^{\beta_{p-r-1}}} \dots \mathcal{L}_{g^{\beta_1}} \tilde{h}_q = \sum_{j=1}^t c_j^p \nabla h_j^g \quad (\text{A.3})$$

with c_1^p, \dots, c_t^p suitable coefficients (scalar functions in the manifold \mathcal{M}). We have:

$$\mathcal{L}_{\psi_z} \mathcal{L}_{g^{\beta_{p-r-1}}} \dots \mathcal{L}_{g^{\beta_1}} \tilde{h}_q = \quad (\text{A.4})$$

$$\nabla \mathcal{L}_{g^{\beta_{p-r-1}}} \dots \mathcal{L}_{g^{\beta_1}} \tilde{h}_q \cdot \psi_z = \left(\sum_{j=1}^t c_j^p \nabla h_j^g \right) \cdot \psi_z = \sum_{j=1}^t c_j^p \mathcal{L}_{\psi_z} h_j^g$$

By replacing this expression in (A.2) we obtain:

$$\nabla \mathcal{L}_{[f_i]^{(\alpha'_1, \dots, \alpha'_p)}} \tilde{h}_q = \quad (\text{A.5})$$

$$\sum_{j=1}^t c_j^p \sum_{\beta(l-1)} M_{\beta_1, \dots, \beta_{l-1}}^{\alpha'_1, \dots, \alpha'_p} \sum_{\mathbf{z}} \left(\nabla C_{(r,j_r)\mathbf{z}}^{\alpha'_1} \right) \mathcal{L}_{\psi_z} h_j^g \text{ mod } \Omega_p + \mathcal{L}_{\tilde{g}} \Omega_p$$

Now we use again Equation (B.12) with $l = s$. We have:

$$\begin{aligned} \nabla \mathcal{L}_{[\psi_r]^{(\alpha_1, \dots, \alpha_s)}} \tilde{h}_q &= \\ \sum_{\beta(s-1)} M_{\beta_1, \dots, \beta_{s-1}}^{\alpha_2, \dots, \alpha_s} \sum_{\mathbf{z}} \left(\nabla C_{\mathbf{r}\mathbf{z}}^{\alpha_1} \right) \mathcal{L}_{\psi_z} \mathcal{L}_{g^{\beta_{s-1}}} \dots \mathcal{L}_{g^{\beta_1}} \tilde{h}_q \\ \text{mod } \Omega_{r+s} + \mathcal{L}_{\tilde{g}} \Omega_{r+s} \end{aligned}$$

We remark that

$$\nabla \mathcal{L}_{[\psi_r]^{(\alpha_1, \dots, \alpha_s)}} \tilde{h}_q \in \Omega_{r+s+1}$$

for any choice of $\alpha_1, \dots, \alpha_s$ and q . In addition, because of the non singularity of M , we obtain that

$$\sum_{\mathbf{z}} \left(\nabla C_{\mathbf{r}\mathbf{z}}^{\alpha_1} \right) \mathcal{L}_{\psi_z} \mathcal{L}_{g^{\beta_{s-1}}} \dots \mathcal{L}_{g^{\beta_1}} \tilde{h}_q \in \Omega_{r+s+1}$$

for any choice of $\beta_1, \dots, \beta_{s-1}$, q and α_1 . On the other hand, the generators of Ω^g (∇h_j^g) are built by Algorithm 2, and they are the differentials of the functions that belong to the function space that includes the

outputs (\tilde{h}_q) together with their Lie derivatives along g^β of order that does not exceed $s - 1$. Therefore, by setting $\alpha_1 = \alpha'_{r+1}$, we obtain:

$$\sum_{\mathbf{z}} \left(\nabla C_{\mathbf{r}\mathbf{z}}^{\alpha'_{r+1}} \right) \mathcal{L}_{\psi_z} h_j^g \in \Omega_{r+s+1}, \quad j = 1, \dots, t.$$

By using this in (A.5) and by knowing that $p \geq r + s + 1$, we immediately obtain that

$$\nabla \mathcal{L}_{[f_i]^{(\alpha'_1, \dots, \alpha'_p)}} \tilde{h}_q \in \Omega_p + \mathcal{L}_{\tilde{g}} \Omega_p$$

From the inductive assumption, we know that $\Omega_p \subseteq \mathcal{O}$. In addition, as \mathcal{O} is invariant under $\mathcal{L}_{\tilde{g}^0}, \mathcal{L}_{\tilde{g}^1}, \dots, \mathcal{L}_{\tilde{g}^{m_w}}$, we have $\Omega_p + \mathcal{L}_{\tilde{g}} \Omega_p \subseteq \mathcal{O}$ and, consequently, $\nabla \mathcal{L}_{[f_i]^{(\alpha'_1, \dots, \alpha'_p)}} \tilde{h}_q \in \mathcal{O}$. ◀

Appendix B. Proof of the validity of Equation (B.12)

The goal of this section is to prove the validity of Equation (B.12), which has been used twice in the proof of Theorem 1. To achieve this goal, we basically need to proceed along two directions:

- Introduce the quantities $C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_l}$, according to Equation (B.1), and obtaining the recursive law in (B.2) and (B.3).
- First proving the validity of Equation (B.6), which is more general than Equation (B.12). In particular, this latter is obtained from the former in a special setting.

We introduce the quantities $C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_l}$ by extending the quantities $C_{\mathbf{kz}}^\alpha$ in (A.1) to the case when the autobracket is applied repetitively multiple times. $C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_l}$ are implicitly defined by the following equation:

$$[\psi_{\mathbf{k}}]^{(\alpha_1, \dots, \alpha_l)} = \sum_{\mathbf{z}} C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_l} \psi_{\mathbf{z}} \quad (\text{B.1})$$

where $\sum_{\mathbf{z}} = \sum_{(z,j_z)}$ stands for $\sum_{z=0}^r \sum_{j_z=1}^{(m_w+1)^z m_u}$, and the operation at the left side, $[\cdot]^{(\alpha_1, \dots, \alpha_l)}$, is the multiple autobracket defined in Section 4. As in the case of (A.1), the term at the left hand side of the above equation belongs to Δ and can be expressed in terms of the set of vectors ψ (again, we remind the reader that Δ is invariant under the autobracket operation and the vectors ψ generate the entire distribution). Equation (B.1) does not provide uniquely the quantities $C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_l}$. This because the vectors ψ are not necessarily independent. On the other hand, it is possible to set them in such a way that they satisfy the following recursive law. The law is stated by the following Lemma:

Lemma 2. *It is possible to set the quantities $C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_l}$ such that they satisfy the following recursive law:*

$$C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_l, \alpha_{l+1}} = \mathcal{L}_{\tilde{g}^{\alpha_{l+1}}} C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_l} + \sum_{\mathbf{z}'} C_{\mathbf{kz}'}^{\alpha_1, \dots, \alpha_l} C_{\mathbf{z}'\mathbf{z}}^{\alpha_{l+1}} \quad (\text{B.2})$$

and its differential form:

$$\nabla C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_l, \alpha_{l+1}} = \quad (\text{B.3})$$

$$\mathcal{L}_{\tilde{g}^{\alpha_{l+1}}} \nabla C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_l} + \sum_{\mathbf{z}'} \nabla C_{\mathbf{kz}'}^{\alpha_1, \dots, \alpha_l} C_{\mathbf{z}'\mathbf{z}}^{\alpha_{l+1}} + \sum_{\mathbf{z}'} C_{\mathbf{kz}'}^{\alpha_1, \dots, \alpha_l} \nabla C_{\mathbf{z}'\mathbf{z}}^{\alpha_{l+1}}$$

where $\sum_{\mathbf{z}'} = \sum_{(z',j_{z'})}$ stands for $\sum_{z'=0}^r \sum_{j_{z'}=1}^{(m_w+1)^{z'} m_u}$.

Proof This proof needs the following equality, which is a property of the autobracket. Given a vector field ϕ and a scalar field a , we have:

$$[a\phi]^\alpha = (\mathcal{L}_{\tilde{g}^\alpha} a) \phi + a[\phi]^\alpha \quad (\text{B.4})$$

for any $\alpha = 0, 1, \dots, m_w$. This is obtained by a direct computation. We have:

$$[a\phi]^\alpha = \sum_{\beta=0}^{m_w} v_\beta^\alpha [g^\beta, a\phi] = \sum_{\beta=0}^{m_w} v_\beta^\alpha a [g^\beta, \phi] + \sum_{\beta=0}^{m_w} (v_\beta^\alpha \mathcal{L}_{g^\beta} a) \phi =$$

$$a[\phi]^\alpha + (\mathcal{L}_{\tilde{g}^\alpha} a) \phi.$$

Now, let us prove the validity of (B.2). By definition we have:

$$[\psi_{\mathbf{k}}]^{(\alpha_1, \dots, \alpha_l, \alpha_{l+1})} = \sum_{\mathbf{z}} C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_l, \alpha_{l+1}} \psi_{\mathbf{z}} \quad (\text{B.5})$$

On the other hand,

$$[\psi_{\mathbf{k}}]^{(\alpha_1, \dots, \alpha_l, \alpha_{l+1})} = \left[[\psi_{\mathbf{k}}]^{(\alpha_1, \dots, \alpha_l)} \right]^{\alpha_{l+1}} = \sum_{\mathbf{z}} \left[C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_l} \psi_{\mathbf{z}} \right]^{\alpha_{l+1}}$$

and, by using (B.4), we obtain

$$\begin{aligned} &= \sum_{\mathbf{z}} (\mathcal{L}_{\tilde{g}^{\alpha_{l+1}}} C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_l}) \psi_{\mathbf{z}} + \sum_{\mathbf{z}} C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_l} [\psi_{\mathbf{z}}]^{\alpha_{l+1}} = \\ &\sum_{\mathbf{z}} (\mathcal{L}_{\tilde{g}^{\alpha_{l+1}}} C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_l}) \psi_{\mathbf{z}} + \sum_{\mathbf{z}'} C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_l} C_{\mathbf{z}'\mathbf{z}}^{\alpha_{l+1}} \psi_{\mathbf{z}'} = \\ &\sum_{\mathbf{z}} \left(\mathcal{L}_{\tilde{g}^{\alpha_{l+1}}} C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_l} + \sum_{\mathbf{z}'} C_{\mathbf{kz}'}^{\alpha_1, \dots, \alpha_l} C_{\mathbf{z}'\mathbf{z}}^{\alpha_{l+1}} \right) \psi_{\mathbf{z}} \end{aligned}$$

By comparing with (B.5) we obtain (B.2) and, by differentiating (B.3). \blacktriangleleft

We have the following fundamental result:

Lemma 3. Given $\psi_{\mathbf{k}}$ ($\mathbf{k} = (k, j_k)$), for any integer $l \geq 1$, and any set of integers $\alpha_1, \dots, \alpha_l$ that take the values $0, 1, \dots, m_w$, for any integer $0 \leq j \leq l-1$, and any $q = 1, \dots, m_w$, we have:

$$\nabla \mathcal{L}_{[\psi_{\mathbf{k}}]^{(\alpha_1, \dots, \alpha_l)}} \tilde{h}_q = \quad (\text{B.6})$$

$$\sum_{\beta_1=0}^{m_w} \sum_{\beta_2=0}^{m_w} \dots \sum_{\beta_j=0}^{m_w} M_{\beta_1, \dots, \beta_j}^{\alpha_1, \dots, \alpha_l} \sum_{\mathbf{z}} (\nabla C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_l, j}) \mathcal{L}_{\psi_{\mathbf{z}}} \mathcal{L}_{g^{\beta_1}} \dots \mathcal{L}_{g^{\beta_j}} \tilde{h}_q$$

$$\text{mod } \Omega_{r+l} + \mathcal{L}_{\tilde{g}} \Omega_{r+l}$$

where M is a suitable multi-index object, which is non singular (i.e., it can be inverted with respect to all its indices).

Proof For notation simplicity we omit the index q , i.e., we denote by h the function \tilde{h}_q . In addition, we use the notation:

$$\sum_{\beta(j)} := \sum_{\beta_1=0}^{m_w} \sum_{\beta_2=0}^{m_w} \dots \sum_{\beta_j=0}^{m_w}.$$

We proceed by induction on l . Let us set $l = 1$. We only have $j = 0$. We have:

$$\nabla \mathcal{L}_{[\psi_{\mathbf{k}}]^{j=1}} h = \sum_{\mathbf{z}} \nabla \mathcal{L}_{C_{\mathbf{kz}}^{\alpha_1}} h = \sum_{\mathbf{z}} \nabla (C_{\mathbf{kz}}^{\alpha_1} \mathcal{L}_{\psi_{\mathbf{z}}} h) =$$

$$\sum_{\mathbf{z}} C_{\mathbf{kz}}^{\alpha_1} \nabla \mathcal{L}_{\psi_{\mathbf{z}}} h + \sum_{\mathbf{z}} (\nabla C_{\mathbf{kz}}^{\alpha_1}) \mathcal{L}_{\psi_{\mathbf{z}}} h$$

On the other hand,

$$\nabla \mathcal{L}_{\psi_{\mathbf{z}}} h \in \Omega_{r+1},$$

Hence,

$$\nabla \mathcal{L}_{[\psi_{\mathbf{k}}]^{j=1}} h = \sum_{\mathbf{z}} (\nabla C_{\mathbf{kz}}^{\alpha_1}) \mathcal{L}_{\psi_{\mathbf{z}}} h, \quad \text{mod } \Omega_{r+1},$$

which proves (B.6) when $l = 1$, and $j = 0$.

Let us consider the recursive step. We assume that (B.6) holds at a given $l = l^* > 1$. We have:

$$\nabla \mathcal{L}_{[\psi_{\mathbf{k}}]^{(\alpha_1, \dots, \alpha_{l^*})}} h = \quad (\text{B.7})$$

$$\sum_{\beta(j)} M_{\beta_1, \dots, \beta_j}^{\alpha_1, \dots, \alpha_{l^*}} \sum_{\mathbf{z}} (\nabla C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_{l^*}-j}) \mathcal{L}_{\psi_{\mathbf{z}}} \mathcal{L}_{g^{\beta_1}} \dots \mathcal{L}_{g^{\beta_j}} h$$

$$\text{mod } \Omega_{r+l^*} + \mathcal{L}_{\tilde{g}} \Omega_{r+l^*}$$

for any $j = 0, 1, \dots, l^* - 1$ and with M non singular. In addition, (B.6) holds at any $l \leq l^*$ for any $j \leq l-1$. In particular, it holds for any $l \leq l^*$ and $j = l-1$. Hence we also have:

$$\nabla \mathcal{L}_{[\psi_{\mathbf{k}}]^{(\alpha_1, \dots, \alpha_{l^*})}} h = \quad (\text{B.8})$$

$$\sum_{\beta(l^*-1)} M_{\beta_1, \dots, \beta_{l^*-1}}^{\alpha_1, \dots, \alpha_{l^*}} \sum_{\mathbf{z}} (\nabla C_{\mathbf{kz}}^{\alpha_1}) \mathcal{L}_{\psi_{\mathbf{z}}} \mathcal{L}_{g^{\beta_1}} \dots \mathcal{L}_{g^{\beta_{l^*-1}}} h$$

$$\text{mod } \Omega_{r+l^*} + \mathcal{L}_{\tilde{g}} \Omega_{r+l^*}$$

for any $l \leq l^*$ with M non singular.

We must prove the validity of (B.6) at $l^* + 1$ and for any $j = 0, \dots, l^*$, i.e.:

$$\nabla \mathcal{L}_{[\psi_{\mathbf{k}}]^{(\alpha_1, \dots, \alpha_{l^*}, \alpha_{l^*+1})}} h = \quad (\text{B.9})$$

$$\sum_{\beta(j)} M_{\beta_1, \dots, \beta_j}^{\alpha_1, \dots, \alpha_{l^*+1}} \sum_{\mathbf{z}} (\nabla C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_{l^*+1}-j}) \mathcal{L}_{\psi_{\mathbf{z}}} \mathcal{L}_{g^{\beta_1}} \dots \mathcal{L}_{g^{\beta_j}} h$$

$$\text{mod } \Omega_{r+l^*+1} + \mathcal{L}_{\tilde{g}} \Omega_{r+l^*+1}$$

for a suitable non singular M .

We proceed by induction on j . Let us consider $j = 0$. We have:

$$\nabla \mathcal{L}_{[\psi_{\mathbf{k}}]^{(\alpha_1, \dots, \alpha_{l^*}, \alpha_{l^*+1})}} h = \nabla \mathcal{L}_{\sum_{\mathbf{z}} C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_{l^*+1}} \psi_{\mathbf{z}}} h =$$

$$\sum_{\mathbf{z}} \nabla (C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_{l^*+1}} \mathcal{L}_{\psi_{\mathbf{z}}} h) =$$

$$\sum_{\mathbf{z}} C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_{l^*+1}} \nabla \mathcal{L}_{\psi_{\mathbf{z}}} h + \sum_{\mathbf{z}} (\nabla C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_{l^*+1}}) \mathcal{L}_{\psi_{\mathbf{z}}} h$$

On the other hand,

$$\nabla \mathcal{L}_{\psi_{\mathbf{z}}} h \in \Omega_{r+1} \subseteq \Omega_{r+l^*+1} + \mathcal{L}_{\tilde{g}} \Omega_{r+l^*+1},$$

Hence,

$$\nabla \mathcal{L}_{[\psi_{\mathbf{k}}]^{(\alpha_1, \dots, \alpha_{l^*}, \alpha_{l^*+1})}} h =$$

$$\sum_{\mathbf{z}} (\nabla C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_{l^*+1}}) \mathcal{L}_{\psi_{\mathbf{z}}} h \quad \text{mod } \Omega_{r+l^*+1} + \mathcal{L}_{\tilde{g}} \Omega_{r+l^*+1}$$

which proves (B.6) when $l = l^* + 1$, and $j = 0$.

Let us assume that (B.9) holds at a given $j = j^* \leq l^* - 1$.

$$\begin{aligned} & \nabla \mathcal{L}_{\{\psi_{\mathbf{k}}\}^{(\alpha_1, \dots, \alpha_{l^*}, \alpha_{l^*+1})}} h = \\ & \sum_{\beta(j^*)} M_{\beta_1, \dots, \beta_{j^*}}^{\alpha_{l^*} - j^* + 2, \dots, \alpha_{l^*} + 1} \sum_{\mathbf{z}} (\nabla C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_{l^*} + 1 - j^*}) \mathcal{L}_{\psi_{\mathbf{z}}} \mathcal{L}_g^{\beta_{j^*}} \dots \mathcal{L}_g^{\beta_1} h \\ & \text{mod } \Omega_{r+l^*+1} + \mathcal{L}_{\bar{g}} \Omega_{r+l^*+1} \end{aligned}$$

We must prove:

$$\begin{aligned} & \nabla \mathcal{L}_{\{\psi_{\mathbf{k}}\}^{(\alpha_1, \dots, \alpha_{l^*}, \alpha_{l^*+1})}} h = \\ & \sum_{\beta(j^*+1)} M_{\beta_1, \dots, \beta_{j^*}, \beta_{j^*+1}}^{\alpha_{l^*} - j^* + 1, \dots, \alpha_{l^*} + 1} \sum_{\mathbf{z}} (\nabla C_{\mathbf{kz}}^{\alpha_1, \dots, \alpha_{l^*} - j^*}) \mathcal{L}_{\psi_{\mathbf{z}}} \mathcal{L}_g^{\beta_{j^*+1}} \mathcal{L}_g^{\beta_{j^*}} \dots \mathcal{L}_g^{\beta_1} h \\ & \text{mod } \Omega_{r+l^*+1} + \mathcal{L}_{\bar{g}} \Omega_{r+l^*+1} \end{aligned}$$

We adopt the notation:

$$\begin{aligned} \lambda & \triangleq \mathcal{L}_g^{\beta_{j^*}} \dots \mathcal{L}_g^{\beta_1} h \\ C^- & = C^{\alpha_1, \dots, \alpha_{l^*} - j^*} \\ C^{+\alpha_{l^*} - j^* + 1} & = C^{\alpha_1, \dots, \alpha_{l^*} - j^*, \alpha_{l^*} - j^* + 1} \end{aligned}$$

We have:

$$\begin{aligned} & \nabla \mathcal{L}_{\{\psi_{\mathbf{k}}\}^{(\alpha_1, \dots, \alpha_{l^*}, \alpha_{l^*+1})}} h = \tag{B.10} \\ & \sum_{\beta(j^*)} M_{\beta_1, \dots, \beta_{j^*}}^{\alpha_{l^*} - j^* + 2, \dots, \alpha_{l^*} + 1} \sum_{\mathbf{z}} (\nabla C_{\mathbf{kz}}^{+\alpha_{l^*} - j^* + 1}) \mathcal{L}_{\psi_{\mathbf{z}}} \lambda \\ & \text{mod } \Omega_{r+l^*+1} + \mathcal{L}_{\bar{g}} \Omega_{r+l^*+1} \end{aligned}$$

We must prove:

$$\begin{aligned} & \nabla \mathcal{L}_{\{\psi_{\mathbf{k}}\}^{(\alpha_1, \dots, \alpha_{l^*}, \alpha_{l^*+1})}} h = \tag{B.11} \\ & \sum_{\beta(j^*+1)} M_{\beta_1, \dots, \beta_{j^*}, \beta_{j^*+1}}^{\alpha_{l^*} - j^* + 1, \dots, \alpha_{l^*} + 1} \sum_{\mathbf{z}} (\nabla C_{\mathbf{kz}}^-) \mathcal{L}_{\psi_{\mathbf{z}}} \mathcal{L}_g^{\beta_{j^*+1}} \lambda \\ & \text{mod } \Omega_{r+l^*+1} + \mathcal{L}_{\bar{g}} \Omega_{r+l^*+1} \end{aligned}$$

From (B.3) we have:

$$\begin{aligned} \nabla C_{\mathbf{kz}}^{+\alpha_{l^*} - j^* + 1} & = \mathcal{L}_g^{\alpha_{l^*} - j^* + 1} \nabla C_{\mathbf{kz}}^- + \sum_{\mathbf{z}'} \nabla C_{\mathbf{kz}'}^- C_{\mathbf{z}'\mathbf{z}}^{\alpha_{l^*} - j^* + 1} + \\ & \sum_{\mathbf{z}'} C_{\mathbf{kz}'}^- \nabla C_{\mathbf{z}'\mathbf{z}}^{\alpha_{l^*} - j^* + 1} \end{aligned}$$

We substitute in (B.10) and we obtain:

$$\begin{aligned} & \nabla \mathcal{L}_{\{\psi_{\mathbf{k}}\}^{(\alpha_1, \dots, \alpha_{l^*}, \alpha_{l^*+1})}} h = \\ & \sum_{\beta(j^*)} M_{\beta_1, \dots, \beta_{j^*}}^{\alpha_{l^*} - j^* + 2, \dots, \alpha_{l^*} + 1} \sum_{\mathbf{z}} \left(\mathcal{L}_g^{\alpha_{l^*} - j^* + 1} \nabla C_{\mathbf{kz}}^- + \sum_{\mathbf{z}'} \nabla C_{\mathbf{kz}'}^- C_{\mathbf{z}'\mathbf{z}}^{\alpha_{l^*} - j^* + 1} + \right. \\ & \left. \sum_{\mathbf{z}'} C_{\mathbf{kz}'}^- \nabla C_{\mathbf{z}'\mathbf{z}}^{\alpha_{l^*} - j^* + 1} \right) \mathcal{L}_{\psi_{\mathbf{z}}} \lambda \text{ mod } \Omega_{r+l^*+1} + \mathcal{L}_{\bar{g}} \Omega_{r+l^*+1} \end{aligned}$$

Let us consider the first term on the right hand side.

$$\sum_{\beta(j^*)} M_{\beta_1, \dots, \beta_{j^*}}^{\alpha_{l^*} - j^* + 2, \dots, \alpha_{l^*} + 1} \sum_{\mathbf{z}} (\mathcal{L}_g^{\alpha_{l^*} - j^* + 1} \nabla C_{\mathbf{kz}}^-) \mathcal{L}_{\psi_{\mathbf{z}}} \lambda =$$

$$\begin{aligned} & \sum_{\beta(j^*)} M_{\beta_1, \dots, \beta_{j^*}}^{\alpha_{l^*} - j^* + 2, \dots, \alpha_{l^*} + 1} \left[\sum_{\mathbf{z}} \mathcal{L}_g^{\alpha_{l^*} - j^* + 1} ((\nabla C_{\mathbf{kz}}^-) \mathcal{L}_{\psi_{\mathbf{z}}} \lambda) - \right. \\ & \left. \sum_{\mathbf{z}} (\nabla C_{\mathbf{kz}}^-) \mathcal{L}_g^{\alpha_{l^*} - j^* + 1} \mathcal{L}_{\psi_{\mathbf{z}}} \lambda \right] \end{aligned}$$

Note that $j^* \leq l^* - 1$. Hence, we are allowed to use (B.7) at $j = j^*$. In the new notation, it tells us that

$$\begin{aligned} & \nabla \mathcal{L}_{\{\psi_{\mathbf{k}}\}^{(\alpha_1, \dots, \alpha_{l^*})}} h = \sum_{\beta(j^*)} M_{\beta_1, \dots, \beta_{j^*}}^{\alpha_{l^*} - j^* + 1, \dots, \alpha_{l^*}} \sum_{\mathbf{z}} (\nabla C_{\mathbf{kz}}^-) \mathcal{L}_{\psi_{\mathbf{z}}} \lambda \\ & \text{mod } \Omega_{r+l^*} + \mathcal{L}_{\bar{g}} \Omega_{r+l^*} \end{aligned}$$

On the other hand,

$$\nabla \mathcal{L}_{\{\psi_{\mathbf{k}}\}^{(\alpha_1, \dots, \alpha_{l^*})}} h \in \Omega_{k+l^*+1} \subseteq \Omega_{r+l^*+1}$$

and, from the invertibility of M , it follows that

$$\sum_{\mathbf{z}} (\nabla C_{\mathbf{kz}}^-) \mathcal{L}_{\psi_{\mathbf{z}}} \lambda \in \Omega_{r+l^*+1}$$

As a result,

$$\sum_{\beta(j^*)} M_{\beta_1, \dots, \beta_{j^*}}^{\alpha_{l^*} - j^* + 2, \dots, \alpha_{l^*} + 1} \sum_{\mathbf{z}} \mathcal{L}_g^{\alpha_{l^*} - j^* + 1} ((\nabla C_{\mathbf{kz}}^-) \mathcal{L}_{\psi_{\mathbf{z}}} \lambda) \in \mathcal{L}_{\bar{g}} \Omega_{r+l^*+1}$$

and we have:

$$\begin{aligned} & \sum_{\beta(j^*)} M_{\beta_1, \dots, \beta_{j^*}}^{\alpha_{l^*} - j^* + 2, \dots, \alpha_{l^*} + 1} \sum_{\mathbf{z}} \left(\mathcal{L}_g^{\alpha_{l^*} - j^* + 1} \nabla C_{\mathbf{kz}}^- + \sum_{\mathbf{z}'} \nabla C_{\mathbf{kz}'}^- C_{\mathbf{z}'\mathbf{z}}^{\alpha_{l^*} - j^* + 1} + \right. \\ & \left. \sum_{\mathbf{z}'} C_{\mathbf{kz}'}^- \nabla C_{\mathbf{z}'\mathbf{z}}^{\alpha_{l^*} - j^* + 1} \right) \mathcal{L}_{\psi_{\mathbf{z}}} \lambda = \\ & \sum_{\beta(j^*)} M_{\beta_1, \dots, \beta_{j^*}}^{\alpha_{l^*} - j^* + 2, \dots, \alpha_{l^*} + 1} \sum_{\mathbf{z}} \left(-\nabla C_{\mathbf{kz}}^- \mathcal{L}_g^{\alpha_{l^*} - j^* + 1} \mathcal{L}_{\psi_{\mathbf{z}}} \lambda + \right. \\ & \left. \sum_{\mathbf{z}'} \nabla C_{\mathbf{kz}'}^- C_{\mathbf{z}'\mathbf{z}}^{\alpha_{l^*} - j^* + 1} \mathcal{L}_{\psi_{\mathbf{z}}} \lambda + \sum_{\mathbf{z}'} C_{\mathbf{kz}'}^- \nabla C_{\mathbf{z}'\mathbf{z}}^{\alpha_{l^*} - j^* + 1} \mathcal{L}_{\psi_{\mathbf{z}}} \lambda \right) \end{aligned}$$

Let us consider the last term of the above. We consider (B.8). It holds for any $l \leq l^*$. As $j^* \leq l^* - 1$, we are allowed to set $l = j^* + 1$. We have:

$$\begin{aligned} & \nabla \mathcal{L}_{\{\psi_{\mathbf{k}}\}^{(\alpha_1, \dots, \alpha_{j^*+1})}} h = \sum_{\beta(j^*)} M_{\beta_1, \dots, \beta_{j^*}}^{\alpha_2, \dots, \alpha_{j^*+1}} \sum_{\mathbf{z}} (\nabla C_{\mathbf{kz}}^{\alpha_1}) \mathcal{L}_{\psi_{\mathbf{z}}} \lambda \\ & \text{mod } \Omega_{r+j^*+1} + \mathcal{L}_{\bar{g}} \Omega_{r+j^*+1} \end{aligned}$$

Hence, from the invertibility of M , it follows that

$$\sum_{\mathbf{z}} (\nabla C_{\mathbf{kz}}^{\alpha_1}) \mathcal{L}_{\psi_{\mathbf{z}}} \lambda \in \Omega_{r+j^*+2}$$

As $j^* \leq l^* - 1$, this proves that:

$$\sum_{\mathbf{z}} (\nabla C_{\mathbf{kz}}^{\alpha_1}) \mathcal{L}_{\psi_{\mathbf{z}}} \lambda \in \Omega_{r+l^*+1} \quad \forall \alpha_1$$

We have:

$$\sum_{\beta(j^*)} M_{\beta_1, \dots, \beta_{j^*}}^{\alpha_{l^*} - j^* + 2, \dots, \alpha_{l^*} + 1} \sum_{\mathbf{z}} \left(-\nabla C_{\mathbf{kz}}^- \mathcal{L}_g^{\alpha_{l^*} - j^* + 1} \mathcal{L}_{\psi_{\mathbf{z}}} \lambda \right)$$

$$\begin{aligned}
& + \sum_{z'} \nabla C_{\mathbf{k}z'}^- C_{z'z}^{\alpha_{l^s}-j^s+1} \mathcal{L}_{\psi_z} \lambda + \sum_{z'} C_{\mathbf{k}z'}^- \nabla C_{z'z}^{\alpha_{l^s}-j^s+1} \mathcal{L}_{\psi_z} \lambda \Big) = \\
& \sum_{\beta(j^s)} M_{\beta_1, \dots, \beta_{j^s}}^{\alpha_{l^s}-j^s+2, \dots, \alpha_{l^s}+1} \sum_z \left(-\nabla C_{\mathbf{k}z}^- \mathcal{L}_g^{\alpha_{l^s}-j^s+1} \mathcal{L}_{\psi_z} \lambda \right. \\
& \left. + \sum_{z'} \nabla C_{\mathbf{k}z'}^- C_{z'z}^{\alpha_{l^s}-j^s+1} \mathcal{L}_{\psi_z} \lambda \right) \text{ mod } \Omega_{r+l^s+1}
\end{aligned}$$

We have:

$$\begin{aligned}
& - \sum_z \nabla C_{\mathbf{k}z}^- \mathcal{L}_g^{\alpha_{l^s}-j^s+1} \mathcal{L}_{\psi_z} \lambda + \sum_{z'} \nabla C_{\mathbf{k}z'}^- C_{z'z}^{\alpha_{l^s}-j^s+1} \mathcal{L}_{\psi_z} \lambda = \\
& \sum_z \nabla C_{\mathbf{k}z}^- \left(-\mathcal{L}_g^{\alpha_{l^s}-j^s+1} \mathcal{L}_{\psi_z} \lambda + \sum_{z'} C_{z'z}^{\alpha_{l^s}-j^s+1} \mathcal{L}_{\psi_z} \lambda \right) = \\
& \sum_z \sum_{\gamma=0}^{m_w} \nabla C_{\mathbf{k}z}^- \mathcal{V}_\gamma^{\alpha_{l^s}-j^s+1} \left(-\mathcal{L}_{g^\gamma} \mathcal{L}_{\psi_z} \lambda + \mathcal{L}_{[g^\gamma, \psi_z]} \lambda \right) = \\
& \sum_{\gamma=0}^{m_w} \sum_z \nabla C_{\mathbf{k}z}^- \mathcal{V}_\gamma^{\alpha_{l^s}-j^s+1} \left(-\mathcal{L}_{\psi_z} \mathcal{L}_{g^\gamma} \lambda \right)
\end{aligned}$$

Hence, (B.10) becomes:

$$\begin{aligned}
& \nabla \mathcal{L}_{[\psi_{\mathbf{k}}]^{(\alpha_1, \dots, \alpha_{l^s}, \alpha_{l^s+1})}} h = \\
& \sum_{\beta(j^s)} M_{\beta_1, \dots, \beta_{j^s}}^{\alpha_{l^s}-j^s+2, \dots, \alpha_{l^s}+1} \sum_{\gamma=0}^{m_w} \sum_z \left(-\mathcal{V}_\gamma^{\alpha_{l^s}-j^s+1} \right) \nabla C_{\mathbf{k}z}^- \left(\mathcal{L}_{\psi_z} \mathcal{L}_{g^\gamma} \lambda \right) = \\
& \sum_{\beta(j^s)} \sum_{\gamma=0}^{m_w} M_{\beta_1, \dots, \beta_{j^s}}^{\alpha_{l^s}-j^s+2, \dots, \alpha_{l^s}+1} \left(-\mathcal{V}_\gamma^{\alpha_{l^s}-j^s+1} \right) \sum_z \nabla C_{\mathbf{k}z}^- \left(\mathcal{L}_{\psi_z} \mathcal{L}_{g^\gamma} \lambda \right) \\
& \text{mod } \Omega_{r+j^s+1} + \mathcal{L}_g \Omega_{r+j^s+1}
\end{aligned}$$

which coincides with the right left side of (B.11) with the dummy index γ equal to the dummy index β_{j^s+1} in (B.11) and

$$M_{\beta_1, \dots, \beta_{j^s}, \beta_{j^s+1}}^{\alpha_{l^s}-j^s+1, \dots, \alpha_{l^s}+1} = M_{\beta_1, \dots, \beta_{j^s}}^{\alpha_{l^s}-j^s+2, \dots, \alpha_{l^s}+1} \left(-\mathcal{V}_{\beta_{j^s+1}}^{\alpha_{l^s}-j^s+1} \right),$$

which remains non singular as ν is non singular. \blacktriangleleft

We have the following fundamental result:

Proposition 4. Given ψ_r ($\mathbf{r} = (r, j_r)$), for any integer $l \geq 1$, and any set of integers $\alpha_1, \dots, \alpha_l$ that take the values $0, 1, \dots, m_w$, and any $q = 1, \dots, m_w$, we have:

$$\begin{aligned}
& \nabla \mathcal{L}_{[\psi_r]^{(\alpha_1, \dots, \alpha_l)}} \tilde{h}_q = \tag{B.12} \\
& \sum_{\beta(l-1)} M_{\beta_1, \dots, \beta_{l-1}}^{\alpha_2, \dots, \alpha_l} \sum_z \left(\nabla C_{\mathbf{r}z}^{\alpha_1} \right) \mathcal{L}_{\psi_z} \mathcal{L}_{g^{\beta_{l-1}}} \dots \mathcal{L}_{g^{\beta_1}} \tilde{h}_q \\
& \text{mod } \Omega_{r+l} + \mathcal{L}_g \Omega_{r+l}
\end{aligned}$$

with:

- M a suitable multi-index object, which is non singular (i.e., it can be inverted with respect to all its indices).
- $\sum_{\beta(l-1)} \triangleq \sum_{\beta_1=0}^{m_w} \sum_{\beta_2=0}^{m_w} \dots \sum_{\beta_{l-1}=0}^{m_w}$.
- $\sum_z = \sum_{(z, j_z)} \triangleq \sum_{z=0}^r \sum_{j_z=1}^{(m_w+1)^2 m_u}$.

Proof The above equality is obtained from Lemma 3, with $j = l - 1$ and $\mathbf{k} = \mathbf{r}$. \blacktriangleleft

Appendix C. Proof of Theorem 2

The proof of this Theorem is complex. It will be divided into the following three steps:

1. Appendix C.1, where we remind the reader of four results in the state of the art, which hold for systems in canonical form with respect to their unknown inputs.
2. Appendix C.2, where we extend the above results to systems that are not in canonical form (and that could even be neither canonic nor canonizable).
3. Appendix C.3, where, based on the results obtained in Appendix C.2, we provide the final proof.

Appendix C.1. Basic results in the state of the art

The four results in the state of the art are Theorem 5, Lemma 4, Proposition 5, and Proposition 6 in this appendix¹⁶.

We consider a system that satisfies (1) and that is in canonical form with respect to its unknown inputs. We denote by $\tilde{h}_1, \dots, \tilde{h}_{m_w}$ a set of observable functions, which are available, and such that the unknown input degree of reconstructability from them is m_w . We set up a special unknown input extension, where *all* the unknown inputs are included in the state, up to the same order. We denote this order by $k-1$. In other words, we are considering the system: $\mathcal{UIE}(k_1, k_2, \dots, k_{m_w})$ with $k_1 = k_2 = \dots = k_{m_w} = k - 1$. The dimension of the state that characterizes this system (i.e., the state in (21)) is $n + km_w$. For simplicity we set: $G \triangleq g_{m_w}^0$, $F^i \triangleq f_{m_w}^i$ ($i = 1, \dots, m_w$), where $g_{m_w}^0$ and $f_{m_w}^i$ are defined in (23) with $d = m_w$. Finally, we denote this unknown input extension by:

$$\mathcal{UIE}_{m_w k}$$

Based on the results derived in [27, 28, 29], in [31] we introduced Algorithm 6.1, that is here Algorithm 10.

Algorithm 10: The algorithm that builds the observable codistribution for the state and all the unknown inputs.

```

Set  $k = 0$ 
Set  $\bar{\Omega}_k = \text{span}\{\nabla h_1, \dots, \nabla h_p\}$ 
loop
  Set  $\bar{\Omega}_k = [\bar{\Omega}_k, 0_{m_w}]$ 
  Set  $k = k + 1$  and the vectors  $G$  and  $F^i$  according to the new  $k$ 
  Set  $\bar{\Omega}_k = \bar{\Omega}_{k-1} + \mathcal{L}_G \bar{\Omega}_{k-1} + \sum_{i=1}^{m_u} \mathcal{L}_{F^i} \bar{\Omega}_{k-1}$ 
end loop

```

The first line in the loop, i.e., $\bar{\Omega}_k = [\bar{\Omega}_k, 0_{m_w}]$, consists in embedding the codistribution $\bar{\Omega}_k$ in the new extended space (the one that also includes the new m_w axes, $w_1^{(k)}, \dots, w_{m_w}^{(k)}$). In practice, each covector $\omega \in \bar{\Omega}_k$ that is represented by a row-vector of dimension $n + km_w$ is extended as follows:

$$\omega \rightarrow [\omega, \underbrace{0, \dots, 0}_{m_w}]$$

¹⁶Note that these results are used in [31] to prove the validity of Algorithm 4, i.e., to prove that this algorithm computes the observable codistribution of systems in canonical form. We show this at end of Appendix C.1 (Remark 1).

For a given k , the codistribution $\widetilde{\Omega}_k$ contains mixed information on both the original initial state x_0 and the values that the unknown inputs (together with their time derivatives up to the $(k-1)^{\text{th}}$ order) take at the initial time.

Based on the derivations in [27, 28, 29], we have the following fundamental result:

Theorem 5. *All the functions of the extended state computed by Algorithm 10 are observable (i.e., their value at the initial time can be reconstructed). In addition, let us consider a scalar function $\theta(x)$ (i.e., of the only original state). If there exists an integer k such that $\nabla\theta = [\partial_x\theta, 0_{km_w}] \in \widetilde{\Omega}_k$ at $x_0 \in \mathcal{M}$, then $\theta(x)$ is observable at x_0 . Conversely, if $\theta(x)$ is observable on a given open set $\mathcal{A} \subseteq \mathcal{M}$, then there exists an integer k such that $\nabla\theta \in \widetilde{\Omega}_k$ on a dense set of \mathcal{A} .*

Proof A proof of these statements can be found in Chapter 6 of [31]. The proof of their validity can also be found in the previous works [27, 28, 29]. ◀

The result stated by Theorem 5 can be used to obtain the observability properties in the presence of unknown inputs (and this is exactly what was done in [27, 28, 29]). However, Algorithm 10 has two strong limitations: (i) It does not converge, automatically, and (ii) its implementation is computationally very expensive (because of the state augmentation).

Let us consider the codistribution Ω_k generated by Algorithm 4. The codistribution $[\Omega_k, 0_{km_w}]$ consists of the covectors whose first n components are covectors in Ω_k and the last components are all zero. Additionally, $L_{m_w}^k$ is the codistribution that is the span of the Lie derivatives of $\widetilde{h}_1, \dots, \widetilde{h}_{m_w}$ up to the order k along the vector G , which is the drift of the considered unknown input extension ($\mathcal{UIE}_{m_w k}$). We have:

$$L_{m_w}^k = \sum_{i=1}^{m_w} \sum_{j=1}^k \text{span} \{ \mathcal{L}_G^j \widetilde{h}_i \} \quad (\text{C.1})$$

Finally, we introduce the following codistribution:

$$\widetilde{\Omega}_k \triangleq [\Omega_k, 0_{km_w}] + L_{m_w}^k$$

In [31], we proved an important property (Lemma 8.3) that regards the above codistribution and that extends the result stated by Lemma 1 in [30], which holds in the driftless case with a single unknown input. Here, we provide a simple modification of Lemma 8.3 in [31]. We have:

Lemma 4. *Given a scalar function θ of the original state x , $\nabla\theta \in \Omega_k$ if and only if $\nabla\theta \in \widetilde{\Omega}_k$.*

Proof This result immediately follows from Lemma 8.3 in [31]. ◀
In addition, it holds the following result:

Proposition 5. *For any integer k we have: $\widetilde{\Omega}_k \subseteq \widetilde{\Omega}_{k+1}$.*

Proof The proof of this statement is available in [31] (Proposition 8.7). ◀

Note that, for driftless system with a single unknown input ($m_w = 1$) the two codistributions in the statement of Proposition 5 coincide, i.e.: $\widetilde{\Omega}_k = \widetilde{\Omega}_{k+1}, \forall k$. See Theorem 1 in [30]. In general, this is not the case and the inclusion holds (\subseteq).

We have the following result:

Proposition 6. *For any integer k , the codistribution $\widetilde{\Omega}_k$ is observable.*

Proof The proof of this statement is available in [31] (Proposition 8.9). ◀

Remark 1. *The results stated by Theorem 5, Lemma 4, Proposition 5, and Proposition 6 allow us to build the entire observable codistribution of a system that is in canonical form, by using Algorithm 4.*

Proof Proposition 6 ensures that for any k , Ω_k is observable, because it is included in $\widetilde{\Omega}_k$, which is observable. On the other hand, it also holds the viceversa. If a function is observable on a given open set \mathcal{A} , Theorem 5 ensures that its differential belongs to $\widetilde{\Omega}_k$ for a given k and on a dense set of \mathcal{A} . As a result, because of Proposition 5, this differential certainly belongs to $\widetilde{\Omega}_k$. Finally, because of Lemma 4, this differential belongs to Ω_k . ◀

Appendix C.2. Extension to the non canonical case

Let us consider the codistributions $\Omega_k^\infty, k = 0, 1, \dots$, obtained by running Algorithm 5. Note that these codistributions, are integrable codistributions generated by the differentials of scalar functions of the original state x and the last $d = m_w - m$ inputs together with their time derivatives up to the order $(k-1)^{\text{th}}$. In particular, it is important to remark that these functions are independent of the first m unknown inputs.

Now, starting from \mathcal{UIE}_{d^∞} , we set up a further unknown input extension. It consists of the extension of the system \mathcal{UIE}_{d^∞} that includes in the state also the first m unknown inputs, together with their time derivatives up to the $(k-1)^{\text{th}}$ order (the same for all these first m unknown inputs). In practice, our final extended system includes the first m unknown inputs with their time derivatives up to the $(k-1)^{\text{th}}$ order and the last $d = m_w - m$ inputs with their time derivatives up to any order. We denote this new unknown input extension by $\mathcal{UIE}_{m_k, d^\infty}$. In addition, we denote by \mathcal{G} the drift of $\mathcal{UIE}_{m_k, d^\infty}$ and by $\mathcal{F}^1, \dots, \mathcal{F}^{m_u}$ the vector fields that correspond to the known inputs. By definition, we have:

$$\mathcal{G} \triangleq \begin{bmatrix} g_{d^\infty}^0 + \sum_{j=1}^m g_{d^\infty}^j w_j \\ w_1^{(1)} \\ \dots \\ w_1^{(k-1)} \\ 0 \\ w_2^{(1)} \\ \dots \\ w_2^{(k-1)} \\ 0 \\ \dots \\ w_m^{(1)} \\ \dots \\ w_m^{(k-1)} \\ 0 \end{bmatrix}, \quad \mathcal{F}^i \triangleq \begin{bmatrix} f_{d^\infty}^i \\ 0_{km} \end{bmatrix}, i = 1, \dots, m_u.$$

In addition, we denote by \overline{w} the first m unknown inputs and by \underline{w} the last $d = m_w - m$ unknown inputs (i.e., the ones included in the state of \mathcal{UIE}_{d^∞}). In other words:

$$\overline{w} = [w_1, \dots, w_m], \quad \underline{w} = [w_{m+1}, \dots, w_{m_w}]$$

Finally, we use the capital W to include their time derivatives up to the $(k-1)^{\text{th}}$ order:

$$\overline{W} = [\overline{w}, \overline{w}^{(1)}, \dots, \overline{w}^{(k-1)}], \quad \underline{W} = [\underline{w}, \underline{w}^{(1)}, \dots, \underline{w}^{(k-1)}]$$

We define the analogous of the codistribution $L_{m_w}^k$ defined in (C.1). In particular, for any integer $z \leq k$, we define:

$$L_m^{z^\infty} \triangleq \sum_{i=1}^m \sum_{j=1}^z \text{span} \{ \mathcal{L}_{\mathcal{G}}^j \widetilde{h}_i \} \quad (\text{C.2})$$

Finally, we introduce the following codistribution:

$$\widetilde{\Omega}_k^\infty \triangleq [\Omega_k^\infty, 0_{km}] + L_m^{\text{ko}}$$

The codistribution in (C.2) is observable because it belongs to $\widetilde{\Omega}_k$, once embedded in the extended space of $\mathcal{UIE}_{mk,d\infty}$.

We start by proving the following result, which extends Lemma 4 to the non canonical case.

Lemma 5. *Given a scalar function θ of x and \underline{W} , $\nabla\theta \in \Omega_k^\infty$ if and only if $\nabla\theta \in \widetilde{\Omega}_k^\infty$.*

Proof As $[\Omega_k^\infty, 0_{km}] \subseteq \widetilde{\Omega}_k^\infty$, we only need to prove that, if $\nabla\theta \in \widetilde{\Omega}_k^\infty$, then $\nabla\theta \in \Omega_k^\infty$. Let us suppose that $\nabla\theta \in \widetilde{\Omega}_k^\infty$. We have:

$$\nabla\theta = \omega + \sum_{i=1}^N c_i \omega_i \quad (\text{C.3})$$

where $\omega \in [\Omega_k^\infty, 0_{km}]$, $\omega_1, \dots, \omega_N$ are N covectors among the generators of L_m^{ko} in (C.2), and c_1, \dots, c_N are all non vanishing coefficients. We must prove that $N = 0$. We proceed by contradiction. Let us suppose that $N \geq 1$. We remark that the covector ω has the last mk entries equal to zero. The same holds for $\nabla\theta$. The covectors ω_i consist of the Lie derivatives of $\widetilde{\nabla}h_1, \dots, \widetilde{\nabla}h_m$ along \mathcal{G} up to the k order. Let us select the highest order Lie derivatives along \mathcal{G} of $\widetilde{\nabla}h_1, \dots, \widetilde{\nabla}h_m$ that appear among $\omega_1, \dots, \omega_N$. Let us denote by j'_1, \dots, j'_m these highest orders. For a given $\widetilde{\nabla}h_i$, if no Lie derivative along \mathcal{G} of it appears among $\omega_1, \dots, \omega_N$, we set $j'_i = 0$. Let us denote by j' the largest value among j'_1, \dots, j'_m (note that $j' > 0$ because $j' = 0$ means that $N = 0$). We denote by $j'_{m_1}, \dots, j'_{m_t}$ all the $t(\geq 1)$ highest orders that are equal to j' . It is immediate to realize that $\mathcal{L}_{\mathcal{G}}^{j'} \widetilde{\nabla}h_{m_1}, \dots, \mathcal{L}_{\mathcal{G}}^{j'} \widetilde{\nabla}h_{m_t}$ are the only generators among the N generators of above that depend on the components of \underline{W} with the highest order time derivative. In other words, they are the only generators that depend on $w_1^{(j'-1)}, \dots, w_m^{(j'-1)}$. By a direct computation, we can derive this dependency. We easily obtain that $\mathcal{L}_{\mathcal{G}}^{j'} \widetilde{h}_{m_j}$ ($j = 1, \dots, t$) depends on them, by the following linear expression: $\sum_{l=1}^m \mu_{m_j}^l w_l^{(j'-1)}$. Let us back to (C.3). In the sum at the right side, the elements that contribute to the entries that correspond to $w_1^{(j'-1)}, \dots, w_m^{(j'-1)}$ can be regrouped in the following sum:

$$\sum_{j=1}^t c_j^* \mathcal{L}_{\mathcal{G}}^{j'} \widetilde{\nabla}h_{m_j}$$

where c_1^*, \dots, c_t^* are among c_1, \dots, c_N and, consequently, they are all different from 0. We remark that the function $c_1^* \mathcal{L}_{\mathcal{G}}^{j'} \widetilde{h}_{m_1} + \dots + c_t^* \mathcal{L}_{\mathcal{G}}^{j'} \widetilde{h}_{m_t}$ depends on $w_1^{(j'-1)}, \dots, w_m^{(j'-1)}$ as follows: $\sum_{l=1}^m \sum_{j=1}^t c_j^* \mu_{m_j}^l w_l^{(j'-1)}$. We also remark that it must exist at least one value of l such that $\sum_{j=1}^t c_j^* \mu_{m_j}^l \neq 0$ (if this is not true it means that the tensor $m\mu$ is singular). Hence, we obtain that $\nabla\theta$ has at least one entry, among the last mk entries, different from zero and this is not possible. \blacktriangleleft

The codistribution $\widetilde{\Omega}_k$ is an integrable codistribution that is generated by the differentials of scalar functions of x , \underline{W} , and \overline{W} . We need to embed this codistribution in the extended space that also includes the axes that correspond to the time derivatives of w_{m+1}, \dots, w_{m_w} , from the order k^{th} , up to any order. We denote this codistribution by $\widetilde{\Omega}_k^\infty$.

The extension of Proposition 5 is Proposition 9. To obtain this extended result, we first need several properties. From now on, for the sake of simplicity, we consider the case of a single known input (i.e, $m_u = 1$). The extension to multiple known inputs is trivial. We denote the vector field that corresponds to the known input by $f = f^1$

for the original system, by $f_{d\infty} = f_{d\infty}^1$ for $\mathcal{UIE}_{d\infty}$, and by $\mathcal{F} = \mathcal{F}^1$ for $\mathcal{UIE}_{mk,d\infty}$.

Starting from \mathcal{F} , we build, for any integer $z \leq k$, the vector field $[\mathcal{F}]^{(z)}$, by computing, repetitively, the following Lie bracket:

$$[\mathcal{F}]^{(0)} = \mathcal{F}, \quad [\mathcal{F}]^{(z)} = [\mathcal{G}, [\mathcal{F}]^{(z-1)}].$$

We have the following property:

Lemma 6. *For any $i = 1, \dots, m$, and for any $z \leq k$, we have:*

$$\mathcal{L}_{\mathcal{G}} \widetilde{\Omega}_z^\infty + \text{span} \mathcal{L}_{[\mathcal{F}]^{(z)}} \widetilde{\nabla}h_i = \mathcal{L}_{\mathcal{G}} \widetilde{\Omega}_z^\infty + \text{span} \mathcal{L}_{\mathcal{F}} \mathcal{L}_{\mathcal{G}}^z \widetilde{\nabla}h_i$$

Proof The proof follows exactly the same steps of the proof of Lemma 7.3 in [31] \blacktriangleleft

The following property extends Lemma 8.4 in [31] to the non canonical case.

Lemma 7. *We have:*

$$[\mathcal{F}]^{(z)} = \sum_{j=0}^z \sum_{\alpha_1, \dots, \alpha_j=0}^m \begin{bmatrix} c_{\alpha_1, \dots, \alpha_j}^j [f_{d\infty}]^{(\alpha_1, \dots, \alpha_j)} \\ 0_{km} \end{bmatrix}, \quad (\text{C.4})$$

where the coefficients ($c_{\alpha_1, \dots, \alpha_j}^j$) depend on x , \underline{W} , \overline{W} only through the functions that generate the codistribution L_m^{ko} . The autobracket (${}^m[\]$) that appears at the right hand side of (C.4), is the one defined in (31).

Proof This can be proved by induction. For $z = 0$, we have: $[\mathcal{F}]^{(0)} = \mathcal{F} = \begin{bmatrix} f_{d\infty} \\ 0_{km} \end{bmatrix}$.

Inductive step: Let us assume that

$$[\mathcal{F}]^{(z-1)} = \sum_{j=0}^{z-1} \sum_{\alpha_1, \dots, \alpha_j=0}^m \begin{bmatrix} c_{\alpha_1, \dots, \alpha_j}^j [f_{d\infty}]^{(\alpha_1, \dots, \alpha_j)} \\ 0_{km} \end{bmatrix},$$

We have:

$$\begin{aligned} [\mathcal{F}]^{(z)} &= [\mathcal{G}, [\mathcal{F}]^{(z-1)}] = \\ &= \sum_{j=0}^{z-1} \sum_{\alpha_1, \dots, \alpha_j=0}^m \left[\mathcal{G}, c_{\alpha_1, \dots, \alpha_j}^j \begin{bmatrix} [f_{d\infty}]^{(\alpha_1, \dots, \alpha_j)} \\ 0_{km} \end{bmatrix} \right] = \\ &= \sum_{j=0}^{z-1} \sum_{\alpha_1, \dots, \alpha_j=0}^m c_{\alpha_1, \dots, \alpha_j}^j \left[\mathcal{G}, \begin{bmatrix} [f_{d\infty}]^{(\alpha_1, \dots, \alpha_j)} \\ 0_{km} \end{bmatrix} \right] + \\ &= \sum_{j=0}^{z-1} \sum_{\alpha_1, \dots, \alpha_j=0}^m \mathcal{L}_{\mathcal{G}} c_{\alpha_1, \dots, \alpha_j}^j \begin{bmatrix} [f_{d\infty}]^{(\alpha_1, \dots, \alpha_j)} \\ 0_{km} \end{bmatrix} \end{aligned} \quad (\text{C.5})$$

We directly compute the Lie bracket in the sum (note that ${}^m[f_{d\infty}]^{(\alpha_1, \dots, \alpha_j)}$ is independent of \overline{W}):

$$\begin{aligned} \left[\mathcal{G}, \begin{bmatrix} [f_{d\infty}]^{(\alpha_1, \dots, \alpha_j)} \\ 0_{km} \end{bmatrix} \right] &= \\ \left[\begin{bmatrix} g_{d\infty}^0, [f_{d\infty}]^{(\alpha_1, \dots, \alpha_j)} \end{bmatrix} + \sum_{j=1}^m \begin{bmatrix} g_{d\infty}^j, [f_{d\infty}]^{(\alpha_1, \dots, \alpha_j)} \end{bmatrix} w_j \right] & \end{aligned} \quad (\text{C.6})$$

On the other hand, we have:

$$\mathcal{L}_{\mathcal{G}} \widetilde{h}_l = \mathcal{L}_{g_{d\infty}^0} \widetilde{h}_l + \sum_{j=1}^m \mathcal{L}_{g_{d\infty}^j} \widetilde{h}_l w_j = {}^m\mu_l^0 + \sum_{j=1}^m {}^m\mu_l^j w_j$$

By multiplying both members by ${}^m v_i^l$ and summing up over $l = 1, \dots, m$ we obtain:

$$w_i = \sum_{l=1}^m \left\{ {}^m v_i^l \mathcal{L}_{\mathcal{G}} \widetilde{h}_l - {}^m v_i^l \mu_l^0 \right\} = \quad (\text{C.7})$$

$$\sum_{l=1}^m {}^m v_i^l \mathcal{L}_{\mathcal{G}} \widetilde{h}_l + {}^m v_i^0, \quad \forall i$$

By substituting (C.7) in (C.6), we obtain:

$$\begin{aligned} & \left[g_{d\infty}^0, {}^m [f_{d\infty}]^{(\alpha_1, \dots, \alpha_j)} \right] + \sum_{j=1}^m \left[g_{d\infty}^j, {}^m [f_{d\infty}]^{(\alpha_1, \dots, \alpha_j)} \right] w_j = \\ & {}^m [f_{d\infty}]^{(\alpha_1, \dots, \alpha_j, 0)} + \sum_{l=1}^m {}^m [f_{d\infty}]^{(\alpha_1, \dots, \alpha_j, l)} \mathcal{L}_{\mathcal{G}} \widetilde{h}_l. \end{aligned} \quad (\text{C.8})$$

Let us consider the second term in (C.5). From the inductive assumption, any coefficient $c_{\alpha_1, \dots, \alpha_j}^j$ depends on $x, \underline{W}, \overline{W}$ only through the functions that generate the codistribution $L_m^{(z-1)\infty}$. As a result, we have:

$$\mathcal{L}_{\mathcal{G}} c_{\alpha_1, \dots, \alpha_j}^j = \sum_{l=1}^{z-1} \sum_{i=1}^m \frac{\partial c_{\alpha_1, \dots, \alpha_j}^j}{\partial (\mathcal{L}_{\mathcal{G}}^l \widetilde{h}_i)} \mathcal{L}_{\mathcal{G}}^{l+1} \widetilde{h}_i.$$

By substituting this equality and the one in (C.8) in (C.5), we obtain that also the coefficients $c_{\alpha_1, \dots, \alpha_j}^j$ that appear in the expression of (C.4) depend on $x, \underline{W}, \overline{W}$ only through the functions that generate the codistribution L_m^{∞} . \blacktriangleleft

From the previous result, it immediately follows the same property stated by Proposition 8.5 in [31], for the canonical case. We have:

Proposition 7. *For any scalar function of the original state, $\theta(x)$, and any integer $z \leq k$, we have:*

$$\text{span} \left\{ \mathcal{L}_{[\mathcal{F}]^{(z)}} \nabla \theta \right\} + L_m^{z\infty} \subseteq$$

$$\left[\sum_{j=0}^z \sum_{\alpha_1, \dots, \alpha_j=0}^m \text{span} \left\{ \mathcal{L}_{m[f_{d\infty}]^{(\alpha_1, \dots, \alpha_j)}} \nabla \theta \right\}, 0_{km} \right] + L_m^{z\infty},$$

Proof This is immediately obtained by computing $\nabla \mathcal{L}_{[\mathcal{F}]^{(z)}} \theta$ and by using the expression of $[\mathcal{F}]^{(z)}$ in (C.4)¹⁷. \blacktriangleleft

The following property extends Lemma 8.6 in [31] to the non canonical case.

Proposition 8. *We have:*

$$L_m^{1\infty} + \mathcal{L}_{\mathcal{G}}[\Omega_{k-1}^{\infty}, 0_{km}] \subseteq L_m^{1\infty} + \sum_{\alpha=0}^m [\mathcal{L}_{m\overline{g}_{d\infty}^{\alpha}} \Omega_{k-1}^{\infty}, 0_{km}]$$

Proof Let us consider a scalar function $\theta(x, \underline{W})$ such that $\nabla \theta \in \Omega_{k-1}^{\infty}$. We have:

$$\mathcal{L}_{\mathcal{G}} \theta = \mathcal{L}_{g_{d\infty}^0} \theta + \sum_{i=1}^m \mathcal{L}_{g_{d\infty}^i} \theta w_i$$

and, by using (C.7), we have:

$$\mathcal{L}_{\mathcal{G}} \theta = \mathcal{L}_{g_{d\infty}^0} \theta + \sum_{i=1}^m \mathcal{L}_{g_{d\infty}^i} \theta \left(\sum_{l=1}^m {}^m v_i^l \mathcal{L}_{\mathcal{G}} \widetilde{h}_l + {}^m v_i^0 \right)$$

and, by using (30), we obtain:

$$\mathcal{L}_{\mathcal{G}} \theta = \mathcal{L}_{m\overline{g}_{d\infty}^0} \theta + \sum_{l=1}^m \mathcal{L}_{m\overline{g}_{d\infty}^l} \theta \mathcal{L}_{\mathcal{G}} \widetilde{h}_l \quad (\text{C.9})$$

from which the proof follows. \blacktriangleleft

We are finally ready to prove Proposition 9, which is the extension of Proposition 5 to the non canonical case.

Proposition 9. *For any integer k we have: $\overline{\Omega}_k^{\infty} \subseteq \overline{\Omega}_k^{\infty}$.*

Proof We proceed by induction.

By definition, $\overline{\Omega}_0^{\infty} = \overline{\Omega}_0^{\infty}$ since they are both the span of the differentials of the outputs.

Inductive step: Let us assume that $\overline{\Omega}_{k-1}^{\infty} \subseteq \overline{\Omega}_{k-1}^{\infty}$. We have: $\overline{\Omega}_k^{\infty} = \overline{\Omega}_{k-1}^{\infty} + \mathcal{L}_{\mathcal{F}} \overline{\Omega}_{k-1}^{\infty} + \mathcal{L}_{\mathcal{G}} \overline{\Omega}_{k-1}^{\infty} \subseteq \overline{\Omega}_{k-1}^{\infty} + \mathcal{L}_{\mathcal{F}} \overline{\Omega}_{k-1}^{\infty} + \mathcal{L}_{\mathcal{G}} \overline{\Omega}_{k-1}^{\infty} = \overline{\Omega}_{k-1}^{\infty} + [\mathcal{L}_{f_{d\infty}} \overline{\Omega}_{k-1}^{\infty}, 0_{km}] + \mathcal{L}_{\mathcal{F}} L_m^{(k-1)\infty} + \mathcal{L}_{\mathcal{G}} \overline{\Omega}_{k-1}^{\infty}$.

On the other hand,

$$\mathcal{L}_{\mathcal{F}} L_m^{(k-1)\infty} = \sum_{i=1}^m \sum_{j=1}^{k-1} \text{span} \left\{ \mathcal{L}_{\mathcal{F}} \mathcal{L}_{\mathcal{G}}^j \nabla \widetilde{h}_i \right\}$$

The only terms of the above codistribution that are not necessarily in $\overline{\Omega}_{k-1}^{\infty}$ are $\sum_{i=1}^m \text{span} \left\{ \mathcal{L}_{\mathcal{F}} \mathcal{L}_{\mathcal{G}}^{k-1} \nabla \widetilde{h}_i \right\}$.

Hence, we have:

$$\begin{aligned} \overline{\Omega}_k^{\infty} & \subseteq \overline{\Omega}_{k-1}^{\infty} + [\mathcal{L}_{f_{d\infty}} \overline{\Omega}_{k-1}^{\infty}, 0_{km}] + \sum_{i=1}^m \text{span} \left\{ \mathcal{L}_{\mathcal{F}} \mathcal{L}_{\mathcal{G}}^{k-1} \nabla \widetilde{h}_i \right\} + \\ & \mathcal{L}_{\mathcal{G}} \overline{\Omega}_{k-1}^{\infty} \end{aligned}$$

By using lemma 6 with $z = k - 1$, we obtain:

$$\begin{aligned} \overline{\Omega}_k^{\infty} & \subseteq \overline{\Omega}_{k-1}^{\infty} + [\mathcal{L}_{f_{d\infty}} \overline{\Omega}_{k-1}^{\infty}, 0_{km}] + \sum_{i=1}^m \text{span} \left\{ \mathcal{L}_{[\mathcal{F}]^{(k-1)}} \nabla \widetilde{h}_i \right\} + \\ & \mathcal{L}_{\mathcal{G}} \overline{\Omega}_{k-1}^{\infty} \end{aligned}$$

By using again the induction assumption we obtain:

$$\begin{aligned} \overline{\Omega}_k^{\infty} & \subseteq [\Omega_{k-1}^{\infty}, 0_{km}] + L_m^{(k-1)\infty} + [\mathcal{L}_{f_{d\infty}} \overline{\Omega}_{k-1}^{\infty}, 0_{km}] + \\ & \sum_{i=1}^m \text{span} \left\{ \mathcal{L}_{[\mathcal{F}]^{(k-1)}} \nabla \widetilde{h}_i \right\} + \mathcal{L}_{\mathcal{G}} [\Omega_{k-1}^{\infty}, 0_{km}] + \mathcal{L}_{\mathcal{G}} L_m^{(k-1)\infty} = \\ & [\Omega_{k-1}^{\infty}, 0_{km}] + L_m^{k\infty} + [\mathcal{L}_{f_{d\infty}} \overline{\Omega}_{k-1}^{\infty}, 0_{km}] + \\ & \sum_{i=1}^m \text{span} \left\{ \mathcal{L}_{[\mathcal{F}]^{(k-1)}} \nabla \widetilde{h}_i \right\} + \mathcal{L}_{\mathcal{G}} [\Omega_{k-1}^{\infty}, 0_{km}] \end{aligned}$$

By using Proposition 7 with $z = k - 1$, we obtain:

$$\begin{aligned} \overline{\Omega}_k^{\infty} & \subseteq [\Omega_{k-1}^{\infty}, 0_{km}] + L_m^{k\infty} + [\mathcal{L}_{f_{d\infty}} \overline{\Omega}_{k-1}^{\infty}, 0_{km}] + \\ & \left[\sum_{i=1}^m \sum_{j=0}^{k-1} \sum_{\alpha_1, \dots, \alpha_j=0}^m \text{span} \left\{ \mathcal{L}_{m[f_{d\infty}]^{(\alpha_1, \dots, \alpha_j)}} \nabla \widetilde{h}_i \right\}, 0_{km} \right] + \end{aligned}$$

¹⁷Note that, the above property actually holds for any function of x and \underline{W} (the proof remains the same). However, we only need to use this result when θ is one of the functions $\widetilde{h}_1, \dots, \widetilde{h}_m$, which are all functions of the only original state (x).

$$\begin{aligned} & \mathcal{L}_{\mathcal{G}}[\Omega_{k-1}^{\infty}, 0_{km}] = \\ & [\Omega_{k-1}^{\infty}, 0_{km}] + L_m^{k\infty} + [\mathcal{L}_{f_{d\infty}} \Omega_{k-1}^{\infty}, 0_{km}] + \\ & \left[\sum_{i=1}^m \sum_{\alpha_1, \dots, \alpha_{k-1}=0}^m \text{span} \left\{ \mathcal{L}_{m[f_{d\infty}]^{(\alpha_1, \dots, \alpha_{k-1})}} \nabla \tilde{h}_i \right\}, 0_{km} \right] + \mathcal{L}_{\mathcal{G}}[\Omega_{k-1}^{\infty}, 0_{km}], \end{aligned}$$

where, the last equality, follows from the fact that all the other terms of the sum on j (i.e., the terms up to $k-2$) are already included in Ω_{k-1}^{∞} .

Finally, by using Proposition 8, we obtain:

$$\begin{aligned} \widetilde{\Omega}_k^{\infty} & \subseteq [\Omega_{k-1}^{\infty}, 0_{km}] + L_m^{k\infty} + [\mathcal{L}_{f_{d\infty}} \Omega_{k-1}^{\infty}, 0_{km}] + \\ & \sum_{i=1}^m \sum_{\alpha_1, \dots, \alpha_{k-1}=0}^m \text{span} \left\{ \mathcal{L}_{m[f_{d\infty}]^{(\alpha_1, \dots, \alpha_{k-1})}} \nabla \tilde{h}_i \right\} + \\ & \sum_{\alpha=0}^m [\mathcal{L}_{m\tilde{g}_{d\infty}^{\alpha}} \Omega_{k-1}^{\infty}, 0_{km}], \end{aligned}$$

which is precisely $\widetilde{\Omega}_k^{\infty}$. \blacktriangleleft

The following result extends Lemma 8.8 in [31] to the non canonical case and will be used in the proof of Proposition 10.

Lemma 8. *Let us consider a scalar function $\lambda(x, \bar{w}, \underline{w})$ of the original state x and the unknown inputs w_1, \dots, w_m . Let us assume that it has the following expression:*

$$\lambda(x, \bar{w}, \underline{w}) = \theta(x, \underline{w}) + \sum_{i=1}^m \theta^i(x, \underline{w}) \mathcal{L}_{\mathcal{G}} \tilde{h}_i$$

with $\theta(x, \underline{w}), \theta^1(x, \underline{w}), \dots, \theta^m(x, \underline{w})$ scalar functions of the original state and the last d unknown inputs.

If $\lambda(x, \bar{w}, \underline{w})$ is observable, then, all the $m+1$ functions $\theta(x, \underline{w}), \theta^1(x, \underline{w}), \dots, \theta^m(x, \underline{w})$ are observable.

Proof This proof follows the same steps of the proof of Lemma 8.8 in [31] with the following differences:

- The tensors μ and ν must be replaced by the tensors ${}^m\mu$ and ${}^m\nu$.
- The indices that in the proof of Lemma 8.8 in [31] take values from 1 to m_w , here they take values from 1 to m .

\blacktriangleleft

Finally, the following property is the extension of Proposition 8.9 in [31] (here, in the statement, we directly consider Ω_k^{∞} instead of $\widetilde{\Omega}_k^{\infty}$).

Proposition 10. *For any integer k , the codistribution Ω_k^{∞} is observable.*

Proof We proceed by induction. By definition, Ω_0^{∞} is the span of the differentials of the outputs, which are observable functions.

Inductive step: Let us assume that the codistribution Ω_{k-1}^{∞} is observable. We want to prove that also Ω_k^{∞} is observable. From Algorithm 5, we need to prove that the following codistributions are observable:

$$\mathcal{L}_{f_{d\infty}} \Omega_{k-1}^{\infty}, \quad \mathcal{L}_{m\tilde{g}_{d\infty}^{\alpha}} \Omega_{k-1}^{\infty}, \quad \text{span} \left\{ \mathcal{L}_{m[f_{d\infty}]^{(\alpha_1, \dots, \alpha_{k-1})}} \nabla \tilde{h}_i \right\}$$

$\forall \alpha, \alpha_1, \dots, \alpha_{k-1}, i$.

Since Ω_{k-1}^{∞} is observable, also $\mathcal{L}_{\mathcal{F}} \widetilde{\Omega}_{k-1}^{\infty}$ it is. Hence, $\mathcal{L}_{\mathcal{F}}[\Omega_{k-1}^{\infty}, 0_{km}]$ is observable and, consequently, we obtain that $\mathcal{L}_{f_{d\infty}} \Omega_{k-1}^{\infty}$ is observable.

We also have that $\mathcal{L}_{\mathcal{G}} \widetilde{\Omega}_{k-1}^{\infty}$ is observable and consequently, also $\mathcal{L}_{\mathcal{G}}[\Omega_{k-1}^{\infty}, 0_{km}]$ it is. Let us consider a function $\theta(x, \underline{W})$ such that

$\nabla \theta \in \Omega_{k-1}^{\infty}$. We obtain that $\nabla \mathcal{L}_{\mathcal{G}} \theta$ is observable. By using (C.9) and the result stated by lemma 8 we immediately obtain that the codistribution $\mathcal{L}_{m\tilde{g}_{d\infty}^{\alpha}} \Omega_{k-1}^{\infty}$ is observable $\forall \alpha = 0, \dots, m$.

It remains to show that also the codistribution $\text{span} \left\{ \mathcal{L}_{m[f_{d\infty}]^{(\alpha_1, \dots, \alpha_{k-1})}} \nabla \tilde{h}_i \right\}$ is observable $\forall \alpha_1, \dots, \alpha_{k-1}, i$.

To prove this, we start by remarking that, by applying $\mathcal{L}_{\mathcal{F}}$ and $\mathcal{L}_{\mathcal{G}}$ repetitively on the observable codistribution, starting from $[\Omega_{k-1}^{\infty}, 0_{km}]$ and by proceeding as before, we finally obtain an observable codistribution in the space of the original state and \underline{W} , which is invariant under $\mathcal{L}_{m\tilde{g}_{d\infty}^{\alpha}}$ ($\forall \alpha$) and $\mathcal{L}_{f_{d\infty}}$. It is possible to show that this codistribution is also invariant under $\mathcal{L}_{m[f_{d\infty}]^{(\alpha_1, \dots, \alpha_{k-2})}}$ ($\forall \alpha_1, \dots, \alpha_{k-2}$). This means that the function

$$\mathcal{L} \left[\begin{array}{c} m[f_{d\infty}]^{(\alpha_1, \dots, \alpha_{k-2})} \\ 0_{km} \end{array} \right], \mathcal{G} \tilde{h}_i$$

is observable $\forall i$. Let us compute the above Lie bracket (for the brevity sake we set $\phi = m[f_{d\infty}]^{(\alpha_1, \dots, \alpha_{k-2})}$). We have:

$$\left[\begin{array}{c} \phi \\ 0_{km} \end{array} \right], \mathcal{G} = \left[\begin{array}{c} [\phi, g_{d\infty}^0] \\ 0_{km} \end{array} \right] + \sum_{i=1}^m \left[\begin{array}{c} [\phi, g_{d\infty}^i] w_i \\ 0_{km} \end{array} \right]$$

By using (C.7) we obtain:

$$= \left[\begin{array}{c} [\phi, g_{d\infty}^0] + \sum_{i=1}^m [\phi, g_{d\infty}^i] ({}^m v_i^j \mathcal{L}_{\mathcal{G}} \tilde{h}_j + {}^m v_i^0) \\ 0_{km} \end{array} \right] =$$

$$\left[\begin{array}{c} [\phi]^0 + \sum_{j=1}^m [\phi]^j \mathcal{L}_{\mathcal{G}} \tilde{h}_j \\ 0_{km} \end{array} \right]$$

Hence we have:

$$\mathcal{L} \left[\begin{array}{c} m[f_{d\infty}]^{(\alpha_1, \dots, \alpha_{k-2})} \\ 0_{km} \end{array} \right], \mathcal{G} \tilde{h}_i =$$

$$\mathcal{L}_{m[f_{d\infty}]^{(\alpha_1, \dots, \alpha_{k-2}, 0)}} \tilde{h}_i + \sum_{j=1}^m \mathcal{L}_{m[f_{d\infty}]^{(\alpha_1, \dots, \alpha_{k-2}, j)}} \tilde{h}_i \mathcal{L}_{\mathcal{G}} \tilde{h}_j$$

and the observability of $\mathcal{L}_{m[f_{d\infty}]^{(\alpha_1, \dots, \alpha_{k-2}, \alpha_{k-1})}} \nabla \tilde{h}_i$, $\forall \alpha_1, \dots, \alpha_{k-1}, i$, follows from lemma 8. \blacktriangleleft

Appendix C.3. The proof

We are now ready to prove the validity of Theorem 2. We actually prove a more general result, as we prove that the statement of the Theorem holds even when the function θ depends on \underline{W} .

Proof Proposition 10 ensures that, for any k , Ω_k^{∞} is observable. As a result, any function $\theta = \theta(x, \underline{W})$ such that its differential belongs to Ω_k^{∞} for a given k is observable. On the other hand, it also holds the viceversa. If a function is observable on a given open set \mathcal{A} , Theorem 5 ensures that its differential belongs to Ω_k for a given k and on a dense set of \mathcal{A} . As a result, because of Proposition 9, this differential certainly belongs to $\widetilde{\Omega}_k^{\infty}$. Finally, because of Lemma 5, this differential belongs to Ω_k^{∞} . \blacktriangleleft

Appendix D. Proof of Proposition 1

We proceed by induction. At the initialization step of Algorithm 3, we only have the vector fields $f_{d\infty}^i$, $i = 1, \dots, m_u$, in (27). All the components of these vectors, with index larger than n , vanish. Let us suppose that, at a given step k , all the generators of Δ_k satisfy this property. Let us denote a given generator by ϕ_k . We have:

$$\phi_k = \begin{bmatrix} \check{\phi}_k \\ 0 \\ 0 \\ \dots \end{bmatrix} \quad (\text{D.1})$$

where $\check{\phi}_k$ is a vector field with dimension n . We must prove that any generator of Δ_{k+1} has the same structure. In other words, we must prove that also the autobracket ${}^m[\phi_k]^\alpha$ has the same structure, for any $\alpha = 0, 1, \dots, m$. From the definition of the autobracket ${}^m[\cdot]$ given in (31), it suffices to prove that the vector fields,

$$\left[g_{d\infty}^0, \phi_k \right], \left[g_{d\infty}^j, \phi_k \right], \quad j = 1, \dots, m_w - d$$

with $g_{d\infty}^0$ and $g_{d\infty}^j$ in (27), satisfy the same property. Namely, the components of these vectors, with index larger than n , vanish, as for the vector in (D.1). This is obtained by an explicit computation of the Lie bracket, by using the structure of the vectors $g_{d\infty}^0$ and $g_{d\infty}^j$. In particular, regarding $\left[g_{d\infty}^j, \phi_k \right]$, for $j = 1, \dots, m_w - d$, the result is immediate as the entries that exceed n , of both $g_{d\infty}^j$ and ϕ_k , vanish, and consequently also their Jacobian. Regarding $\left[g_{d\infty}^0, \phi_k \right]$ we cannot say the same for $g_{d\infty}^0$. On the other hand, the entries that exceed n are independent of the original state and, consequently, the product of the Jacobian of these entries by a vector with the structure in (D.1) vanishes.

Appendix E. Proof of Theorem 3

The proof of this theorem is obtained by using the result stated by Theorem 2 and by proving the following two properties:

1. For any integer k , there exists an integer k_1 such that: $\mathcal{O}_k^\infty \subseteq \Omega_{k_1}^\infty$.
2. For any integer k , there exists an integer k_2 such that: $\Omega_k^\infty \subseteq \mathcal{O}_{k_2}^\infty$.

Once these statements are proved, the proof of Theorem 3 is immediate. If $\theta(x)$ is observable, because of Theorem 2, there exists an integer k such that $\nabla\theta \in \Omega_k^\infty$. Because of the second property, $\nabla\theta \in \mathcal{O}_{k_2}^\infty$. Conversely, if $\nabla\theta \in \mathcal{O}_k^\infty$, because of the first property, $\nabla\theta \in \Omega_{k_1}^\infty$ and, because of Theorem 2, θ is observable.

In the rest of this appendix, we prove the two above properties.

Appendix E.1. Proof of the first property

From the $(k+1)$ th step of Algorithm 5, we obtain that

$$\sum_{q=1}^m \sum_{i=1}^{m_u} \sum_{\alpha_1=0}^m \dots \sum_{\alpha_k=0}^m \text{span} \left\{ \mathcal{L}_{m[f_{d\infty}^i]^{(\alpha_1, \dots, \alpha_k)}} \nabla \tilde{h}_q \right\} \in \Omega_{k+1}^\infty.$$

From (33), we immediately obtain that

$$\tilde{\mathcal{O}}^\infty \subseteq \Omega_{r+s_x+1}^\infty.$$

By comparing the recursive step of Algorithm 8 with the one of Algorithm 5 it is immediate to conclude that, for any integer k :

$$\mathcal{O}_k^\infty \subseteq \Omega_{r+s_x+1+k}^\infty$$

and we obtain the validity of this property with $k_1 = r + s_x + 1 + k \leq k + 2n - m + 1$

Appendix E.2. Proof of the second property

We must prove that, for any k , there exists always an integer (k_2) such that $\Omega_k^\infty \subseteq \mathcal{O}_{k_2}^\infty$. We proceed by induction. It is true at $k = 0$ as $\Omega_0 \subseteq \mathcal{O}_0$. Hence, $k_2 = 0$.

Let us assume that, for $k = p$, there exists $k_2 = k_p$ such that

$$\Omega_p^\infty \subseteq \mathcal{O}_{k_p}^\infty$$

We must prove that there exists an integer k_{p+1} such that:

$$\Omega_{p+1}^\infty \subseteq \mathcal{O}_{k_{p+1}}^\infty$$

From the inductive assumption we have

$$\mathcal{L}_{m_{\check{g}_{d\infty}^\beta}} \Omega_p^\infty \subseteq \mathcal{L}_{m_{\check{g}_{d\infty}^\beta}} \mathcal{O}_{k_p}^\infty, \quad \beta = 0, 1, \dots, m$$

and

$$\mathcal{L}_{f_{d\infty}^i} \Omega_p^\infty \subseteq \mathcal{L}_{f_{d\infty}^i} \mathcal{O}_{k_p}^\infty, \quad i = 1, \dots, m_u.$$

Hence, by setting $k_{p+1} \geq k_p + 1$, from the recursive step of Algorithm 8, we obtain that

$$\mathcal{L}_{m_{\check{g}_{d\infty}^\beta}} \Omega_p^\infty \subseteq \mathcal{O}_{k_{p+1}}^\infty, \quad \mathcal{L}_{f_{d\infty}^i} \Omega_p^\infty \subseteq \mathcal{O}_{k_{p+1}}^\infty,$$

for any $\beta = 0, \dots, m$ and any $i = 1, \dots, m_u$. As a result, from the recursive step of Algorithm 5, it only remains to prove that

$$\nabla \mathcal{L}_{m[f_{d\infty}^i]^{(\alpha'_1, \dots, \alpha'_p)}} \tilde{h}_q \in \mathcal{O}_{k_{p+1}}^\infty$$

for any choice of $\alpha'_1, \dots, \alpha'_p = 0, 1, \dots, m$. In the rest of this appendix, we will show that it suffices to set $k_{p+1} \geq k_p + 1$ in order to obtain the above condition.

We distinguish the following two cases:

1. $p \leq r + s_x$.
2. $p \geq r + s_x + 1$.

The first case is trivial as $\nabla \mathcal{L}_{m[f_{d\infty}^i]^{(\alpha'_1, \dots, \alpha'_p)}} \tilde{h}_q \in \tilde{\mathcal{O}}^\infty \subseteq \mathcal{O}_0$.

Let us consider the second case.

First of all, we introduce the same vector fields ψ and the quantities $C_{\mathbf{kz}}^\alpha$ introduced in Appendix A. In this case, they refer to the vectors generated by Algorithm 6 instead of Algorithm 3. We can obtain the same result derived in Appendix B, i.e., the equality given in (B.12). In this case, the equality becomes:

$$\nabla \mathcal{L}_{m[\psi_r]^{(\alpha_1, \dots, \alpha_r)}} \tilde{h}_q = \quad (\text{E.1})$$

$$\sum_{\beta(l-1)} M_{\beta_1, \dots, \beta_{l-1}}^{\alpha_2, \dots, \alpha_l} \sum_{\mathbf{z}} (\nabla C_{\mathbf{rz}}^{\alpha_l}) \mathcal{L}_{\psi_{\mathbf{z}}} \mathcal{L}_{g_{d\infty}^{\beta_{l-1}}} \dots \mathcal{L}_{g_{d\infty}^{\beta_1}} \tilde{h}_q$$

$$\text{mod } \Omega_{r+l}^\infty + \mathcal{L}_{m_{\check{g}_{d\infty}^\infty}} \Omega_{r+l}^\infty$$

where:

- The sum $\sum_{\beta(l-1)}$ is defined as follows:

$$\sum_{\beta(l-1)} \triangleq \sum_{\beta_1=0}^m \sum_{\beta_2=0}^m \dots \sum_{\beta_{l-1}=0}^m,$$

- The sum $\sum_{\mathbf{z}}$ is defined as follows:

$$\sum_{\mathbf{z}} = \sum_{(z, j_z)} \triangleq \sum_{z=0}^r \sum_{j_z=1}^{(m+1)^{m_u}}.$$

- M is a suitable multi-index object, which is non singular (i.e., it can be inverted with respect to all its indices).
- $\mathcal{L}^{m_{\tilde{g}_{d\infty}} \Omega_p^\infty}$ stands for $\sum_{\gamma=0}^m \mathcal{L}^{m_{\tilde{g}_{d\infty}}^\gamma \Omega_p^\infty}$
- the mod at the end of Equation (E.1) means that the equality holds when the codistribution $\Omega_p^\infty + \mathcal{L}^{m_{\tilde{g}_{d\infty}} \Omega_p^\infty}$ is added on both sides.

The above equality holds for any integer l , any set of integers $\alpha_1, \dots, \alpha_l$ that take values between 0 and m , and any $q = 1, \dots, m$.

We start by remarking that, when $p \geq r + s_x + 1$ we have:

$$m[f_{d\infty}^i]^{(\alpha'_1, \dots, \alpha'_p)} = m[\psi_{(r, j_r)}]^{(\alpha'_{r+1}, \dots, \alpha'_p)}$$

where j_r is such that $m[f_{d\infty}^i]^{(\alpha'_1, \dots, \alpha'_r)} = \psi_{(r, j_r)}$.

We set $p = r + s_x + l'$, with $l' \geq 1$, and we use Equation (E.1), with $l = s_x + l'$, and $\alpha_1, \dots, \alpha_l = \alpha'_{r+1}, \dots, \alpha'_p$. We have:

$$\nabla \mathcal{L}^{m[f_{d\infty}^i]^{(\alpha'_1, \dots, \alpha'_p)}} \tilde{h}_q = \nabla \mathcal{L}^{m[\psi_{(r, j_r)}]^{(\alpha'_{r+1}, \dots, \alpha'_p)}} \tilde{h}_q = \quad (E.2)$$

$$\sum_{\beta(l-1)} M_{\beta_1, \dots, \beta_{l-1}}^{\alpha'_{r+2}, \dots, \alpha'_p} \sum_z \left(\nabla G_{(r, j_r)z}^{\alpha'_{r+1}} \right) \mathcal{L}_{\psi_z} \mathcal{L}_{g_{d\infty}^{\beta_{p-r-1}}} \dots \mathcal{L}_{g_{d\infty}^{\beta_1}} \tilde{h}_q$$

$$\text{mod } \Omega_p^\infty + \mathcal{L}^{m_{\tilde{g}_{d\infty}} \Omega_p^\infty}$$

Let us consider now the term:

$$\mathcal{L}_{\psi_z} \mathcal{L}_{g_{d\infty}^{\beta_{p-r-1}}} \dots \mathcal{L}_{g_{d\infty}^{\beta_1}} \tilde{h}_q = \left(\nabla \mathcal{L}_{g_{d\infty}^{\beta_{p-r-1}}} \dots \mathcal{L}_{g_{d\infty}^{\beta_1}} \tilde{h}_q \right) \cdot \psi_z \quad (E.3)$$

The key point is that, in the above expression, thanks to the result stated by Proposition 1, the vector ψ_z (which has infinite dimensions) has only the first n entries which can be non vanishing. As a result, the scalar product in the above expression can be limited to the first n entries of the covector $\nabla \mathcal{L}_{g_{d\infty}^{\beta_{p-r-1}}} \dots \mathcal{L}_{g_{d\infty}^{\beta_1}} \tilde{h}_q$. The scalar product in the above expression becomes:

$$\left(\nabla \mathcal{L}_{g_{d\infty}^{\beta_{p-r-1}}} \dots \mathcal{L}_{g_{d\infty}^{\beta_1}} \tilde{h}_q \right) \cdot \psi_z = \left(\partial_x \mathcal{L}_{g_{d\infty}^{\beta_{p-r-1}}} \dots \mathcal{L}_{g_{d\infty}^{\beta_1}} \tilde{h}_q \right) \cdot \check{\psi}_z$$

with $\check{\psi}_z$ the n -dimensional vector field that includes the first n components of ψ_z .

Algorithm 7 computes scalar functions, starting from $\tilde{h}_1, \dots, \tilde{h}_m$, and by computing, at each step, the Lie derivatives of the functions computed at the previous step along $g_{d\infty}^1, \dots, g_{d\infty}^{m_w-d}$. At each step k , it computes the codistributions Ω_k^g and ${}^x\Omega_k^g$. We set $\nabla \theta_k^1, \dots, \nabla \theta_k^{D_k}$ a basis of Ω_k^g . Note that all the functions $\theta_k^1, \dots, \theta_k^{D_k}$ are automatically computed by the algorithm. Then, ${}^x\Omega_k^g$ is defined as the span of the differentials with respect to the only original state of these functions, i.e., ${}^x\Omega_k^g = \text{span} \left\{ \partial_x \theta_k^1, \dots, \partial_x \theta_k^{D_k} \right\}$. We extract from $\partial_x \theta_k^1, \dots, \partial_x \theta_k^{D_k}$ a basis of ${}^x\Omega_k^g$ and, without loss of generality (i.e., by reordering the above covectors), we assume that it consists of the first $D_k^x \leq D_k$ covectors. We have:

$${}^x\Omega_k^g = \text{span} \left\{ \partial_x \theta_k^1, \dots, \partial_x \theta_k^{D_k^x} \right\}$$

Note also that $D_k^x \leq n$.

The following result ensures that, by running Algorithm 7, we attain the limit codistribution ${}^x\Omega^g$, of the series ${}^x\Omega_k^g$, in a finite number of steps. In particular, at the smallest integer s_x that satisfies (32). As a result, $s_x \leq n$. This is a special property of the system $\mathcal{UIE}_{d\infty}$, due to the special structure of the vector fields $g_{d\infty}^\gamma$ in (27). In particular,

thanks to their structure, we have the following commutative property, $\forall h$:

$$\mathcal{L}_{g_{d\infty}^\gamma} \partial_x h = \partial_x \mathcal{L}_{g_{d\infty}^\gamma} h,$$

which does not hold, in general (the property that always holds is the above with ∇ instead of ∂_x). We have the following result:

Proposition 11. *Let us refer to the step s_x of Algorithm 7. For every scalar function θ , such that $\partial_x \theta \in {}^x\Omega_{s_x}^g$, we have $\partial_x \mathcal{L}_{g_{d\infty}^\gamma} \theta \in {}^x\Omega_{s_x}^g$, for any $\gamma = 0, 1, \dots, m_w - d$.*

Proof For simplicity, we denote by D^x the dimension of $\Omega_{s_x}^g$ (instead of $D_{s_x}^x$) and by $\theta^1, \dots, \theta^{D^x}$, a basis of ${}^x\Omega_{s_x}^g$ (instead of $\theta_{s_x}^1, \dots, \theta_{s_x}^{D^x}$). In other words:

$${}^x\Omega_{s_x}^g = \text{span} \left\{ \partial_x \theta^1, \dots, \partial_x \theta^{D^x} \right\} = {}^x\Omega_{s_x-1}^g$$

The last equality follows from the definition of s_x by (32). Because of this, we also have:

$$\partial_x \mathcal{L}_{g_{d\infty}^\gamma} \theta^i \in {}^x\Omega_{s_x}^g,$$

for any $i = 1, \dots, D^x$, and for any $\gamma = 0, 1, \dots, m_w - d$.

By definition, we have:

$$\nabla \mathcal{L}_{g_{d\infty}^\gamma} \theta = \left[\partial_x \mathcal{L}_{g_{d\infty}^\gamma} \theta, \partial_w \mathcal{L}_{g_{d\infty}^\gamma} \theta \right] \quad (E.4)$$

where ∂_w is the operator that includes all the partial derivatives with respect to the components of $w_{d\infty}$ in (25). On the other hand,

$$\nabla \mathcal{L}_{g_{d\infty}^\gamma} \theta = \mathcal{L}_{g_{d\infty}^\gamma} \nabla \theta = \mathcal{L}_{g_{d\infty}^\gamma} \left[\partial_x \theta, \partial_w \theta \right]$$

As by assumption $\partial_x \theta \in {}^x\Omega_{s_x}^g$, we have:

$$\partial_x \theta = \sum_{j=1}^{D^x} b_j \partial_x \theta^j$$

with b_1, \dots, b_{D^x} suitable coefficients (scalar functions of the extended state). Hence:

$$\nabla \mathcal{L}_{g_{d\infty}^\gamma} \theta = \mathcal{L}_{g_{d\infty}^\gamma} \left[\sum_{j=1}^{D^x} b_j \partial_x \theta^j, \partial_w \theta \right] =$$

$$\mathcal{L}_{g_{d\infty}^\gamma} \left(\sum_{j=1}^{D^x} b_j \left[\partial_x \theta^j, \partial_w \theta^j \right] - \sum_{j=1}^{D^x} b_j \left[0^n, \partial_w \theta^j \right] + \left[0^n, \partial_w \theta \right] \right)$$

where $0^n = [0, \dots, 0]$ and n is the dimension of the original state (x).

We obtain: $\nabla \mathcal{L}_{g_{d\infty}^\gamma} \theta =$

$$\mathcal{L}_{g_{d\infty}^\gamma} \left(\sum_{j=1}^{D^x} b_j \nabla \theta^j - \sum_{j=1}^{D^x} b_j \left[0^n, \partial_w \theta^j \right] + \left[0^n, \partial_w \theta \right] \right) =$$

$$\sum_{j=1}^{D^x} \left(\mathcal{L}_{g_{d\infty}^\gamma} b_j \right) \nabla \theta^j + \sum_{i=1}^{D^x} b_i \nabla \mathcal{L}_{g_{d\infty}^\gamma} \theta^i +$$

$$\mathcal{L}_{g_{d\infty}^\gamma} \left(- \sum_{j=1}^{D^x} b_j \left[0^n, \partial_w \theta^j \right] + \left[0^n, \partial_w \theta \right] \right) =$$

$$\sum_{j=1}^{D^x} (\mathcal{L}_{g_{d\infty}^\gamma} b_j) \nabla \theta^j + \sum_{i=1}^{D^x} b_i [\partial_x \mathcal{L}_{g_{d\infty}^\gamma} \theta^i, \partial_w \mathcal{L}_{g_{d\infty}^\gamma} \theta^i] + \mathcal{L}_{g_{d\infty}^\gamma} \left(- \sum_{j=1}^{D^x} b_j [0^n, \partial_w \theta^j] + [0^n, \partial_w \theta] \right)$$

On the other hand, by assumption, $\partial_x \mathcal{L}_{g_{d\infty}^\gamma} \theta^i \in {}^x \Omega_{s_x}^g$, i.e.:

$$\partial_x \mathcal{L}_{g_{d\infty}^\gamma} \theta^i = \sum_{j=1}^{D^x} a_j^{i\gamma} \partial_x \theta^j \quad (\text{E.5})$$

with $a_j^{i\gamma}$ ($i, j = 1, \dots, D^x$, and $\gamma = 0, \dots, m_w - d$) suitable coefficients (scalar functions of the extended state). Hence:

$$\begin{aligned} \nabla \mathcal{L}_{g_{d\infty}^\gamma} \theta &= \sum_{j=1}^{D^x} (\mathcal{L}_{g_{d\infty}^\gamma} b_j) \nabla \theta^j + \sum_{i=1}^{D^x} b_i \left[\sum_{j=1}^{D^x} a_j^{i\gamma} \partial_x \theta^j, \partial_w \mathcal{L}_{g_{d\infty}^\gamma} \theta^i \right] + \\ &\mathcal{L}_{g_{d\infty}^\gamma} \left(- \sum_{j=1}^{D^x} b_j [0^n, \partial_w \theta^j] + [0^n, \partial_w \theta] \right) = \\ &\left[\sum_{j=1}^{D^x} \left(\mathcal{L}_{g_{d\infty}^\gamma} b_j + \sum_{i=1}^{D^x} b_i a_j^{i\gamma} \right) \partial_x \theta^j, \sum_{j=1}^{D^x} \left(\mathcal{L}_{g_{d\infty}^\gamma} b_j \right) \partial_w \theta^j + b_j \partial_w \mathcal{L}_{g_{d\infty}^\gamma} \theta^j \right] \\ &+ \mathcal{L}_{g_{d\infty}^\gamma} \left(- \sum_{j=1}^{D^x} b_j [0^n, \partial_w \theta^j] + [0^n, \partial_w \theta] \right) \end{aligned}$$

Now, from the special structure of the vector fields $g_{d\infty}^\gamma$ given in (27), for any $\gamma = 0, 1, \dots, m$, we obtain that, for any covector in the extended space with the structure

$$[0^n, \omega]$$

the operator $\mathcal{L}_{g_{d\infty}^\gamma}$ does not change this structure. Namely:

$$\mathcal{L}_{g_{d\infty}^\gamma} [0^n, \omega] = [0^n, \omega^*]$$

for some ω^* .

Therefore: $\nabla \mathcal{L}_{g_{d\infty}^\gamma} \theta =$

$$\left[\sum_{j=1}^{D^x} \left(\mathcal{L}_{g_{d\infty}^\gamma} b_j + \sum_{i=1}^{D^x} b_i a_j^{i\gamma} \right) \partial_x \theta^j, \sum_{j=1}^{D^x} \left(\mathcal{L}_{g_{d\infty}^\gamma} b_j \right) \partial_w \theta^j + b_j \partial_w \mathcal{L}_{g_{d\infty}^\gamma} \theta^j \right] + [0^n, \omega^*]$$

for some ω^* .

By comparing this with (E.4) we obtain

$$\partial_x \mathcal{L}_{g_{d\infty}^\gamma} \theta = \sum_{j=1}^{D^x} c_j^\gamma \partial_x \theta^j$$

with

$$c_j^\gamma = \mathcal{L}_{g_{d\infty}^\gamma} b_j + \sum_{i=1}^{D^x} b_i a_j^{i\gamma}$$

In other words, $\partial_x \mathcal{L}_{g_{d\infty}^\gamma} \theta \in {}^x \Omega_{s_x}^g$, for any $\gamma = 0, 1, \dots, m_w - d$. ◀

Now, let us back to the term in (E.3). We have:

$$\begin{aligned} \mathcal{L}_{\psi_z} \mathcal{L}_{g_{d\infty}^{\beta_{p-r-1}}} \dots \mathcal{L}_{g_{d\infty}^{\beta_1}} \widetilde{h}_q &= \nabla \mathcal{L}_{g_{d\infty}^{\beta_{p-r-1}}} \dots \mathcal{L}_{g_{d\infty}^{\beta_1}} \widetilde{h}_q \cdot \psi_z = \\ \partial_x \mathcal{L}_{g_{d\infty}^{\beta_{p-r-1}}} \dots \mathcal{L}_{g_{d\infty}^{\beta_1}} \widetilde{h}_q \cdot \check{\psi}_z &= \left(\sum_{j=1}^{D^x} c_j^p \partial_x \theta^j \right) \cdot \check{\psi}_z = \end{aligned}$$

$$\left(\sum_{j=1}^{D^x} c_j^p \nabla \theta^j \right) \cdot \psi_z = \sum_{j=1}^{D^x} c_j^p \mathcal{L}_{\psi_z} \theta^j$$

with c_j^p ($j = 1, \dots, D^x$) suitable coefficients (scalar functions of the extended state).

By replacing this expression in (E.2) we obtain:

$$\begin{aligned} \nabla \mathcal{L}_{m_{[f_i]^{(\alpha'_1, \dots, \alpha'_p)}}} \widetilde{h}_q &= \quad (\text{E.6}) \\ \sum_{j=1}^{D^x} c_j^p \sum_{\beta(l-1)} M_{\beta_1, \dots, \beta_{l-1}}^{\alpha'_r+2, \dots, \alpha'_p} \sum_z \left(\nabla C_{(r,j)z}^{\alpha'_r+1} \right) \mathcal{L}_{\psi_z} \theta^j \text{ mod } \Omega_p^\infty &+ \mathcal{L}_{m_{\widehat{g}_{d\infty}}} \Omega_p^\infty \end{aligned}$$

Now we use again Equation (E.1) with $l = s_x$. We have:

$$\begin{aligned} \nabla \mathcal{L}_{m_{[\psi_r]^{(\alpha_1, \dots, \alpha_{s_x})}}} \widetilde{h}_q &= \\ \sum_{\beta(s_x-1)} M_{\beta_1, \dots, \beta_{s_x-1}}^{\alpha_2, \dots, \alpha_{s_x}} \sum_z \left(\nabla C_{\mathbf{rm}}^{\alpha_1} \right) \mathcal{L}_{\psi_z} \mathcal{L}_{g_{d\infty}^{\beta_{s_x-1}}} \dots \mathcal{L}_{g_{d\infty}^{\beta_1}} \widetilde{h}_q & \\ \text{mod } \Omega_{r+s_x}^\infty &+ \mathcal{L}_{m_{\widehat{g}_{d\infty}}} \Omega_{r+s_x}^\infty \end{aligned}$$

We remark that

$$\nabla \mathcal{L}_{m_{[\psi_r]^{(\alpha_1, \dots, \alpha_{s_x})}}} \widetilde{h}_q \in \Omega_{r+s_x+1}^\infty$$

for any choice of $\alpha_1, \dots, \alpha_{s_x}$ and q . In addition, because of the non singularity of M , we obtain that

$$\sum_z \left(\nabla C_{\mathbf{rm}}^{\alpha_1} \right) \mathcal{L}_{\psi_z} \mathcal{L}_{g_{d\infty}^{\beta_{s_x-1}}} \dots \mathcal{L}_{g_{d\infty}^{\beta_1}} \widetilde{h}_q \in \Omega_{r+s_x+1}^\infty$$

for any choice of $\beta_1, \dots, \beta_{s_x-1}$, q and α_1 .

On the other hand, the generators of ${}^x \Omega^g$ ($\nabla \theta^j$) are built by Algorithm 7, and they are the differentials of the functions that belong to the function space that includes the outputs (\widetilde{h}_q) together with their Lie derivatives along $g_{d\infty}^\beta$ of order that does not exceed $s_x - 1$. Therefore, by setting $\alpha_1 = \alpha'_{r+1}$, we obtain:

$$\sum_z \left(\nabla C_{\mathbf{rm}}^{\alpha'_{r+1}} \right) \mathcal{L}_{\psi_z} \theta^j \in \Omega_{r+s_x+1}^\infty, \quad j = 1, \dots, D^x.$$

By using this in (E.6) and by knowing that $p \geq r + s_x + 1$, we immediately obtain that

$$\nabla \mathcal{L}_{[f_i]^{(\alpha'_1, \dots, \alpha'_p)}} \widetilde{h}_q \in \Omega_p^\infty + \mathcal{L}_{m_{\widehat{g}_{d\infty}}} \Omega_p^\infty$$

From the inductive assumption, we know that $\Omega_p^\infty \subseteq \mathcal{O}_{k_p}^\infty$ and, as a result, $\Omega_p^\infty + \mathcal{L}_{m_{\widehat{g}_{d\infty}}} \Omega_p^\infty \subseteq \mathcal{O}_{k_p}^\infty + \mathcal{L}_{m_{\widehat{g}_{d\infty}}} \mathcal{O}_{k_p}^\infty$. On the other hand, from the recursive step of Algorithm 8 (where we include the Lie derivatives along all the vector fields $m_{\widehat{g}_{d\infty}}^0, m_{\widehat{g}_{d\infty}}^1, \dots, m_{\widehat{g}_{d\infty}}^m$), we have $\mathcal{O}_{k_p}^\infty + \mathcal{L}_{m_{\widehat{g}_{d\infty}}} \mathcal{O}_{k_p}^\infty \subseteq \mathcal{O}_{k_{p+1}}^\infty$ and, consequently, $\nabla \mathcal{L}_{[f_i]^{(\alpha'_1, \dots, \alpha'_p)}} \widetilde{h}_q \in \mathcal{O}_{k_{p+1}}^\infty$ as soon as we set $k_{p+1} \geq k_p + 1$.

References

- [1] Kalman R. E., "On the General Theory of Control Systems", Proc. 1st Int. Cong. of IFAC, Moscow 1960 1481, Butterworth, London 1961.
- [2] Kalman R. E., "Mathematical Description of Linear Dynamical Systems", SIAM J. Contr. 1963, 1, 152.
- [3] Kahl Dominik, Wendland Philipp, Neidhardt Matthias, Weber Andreas, Kschischo, Maik, Structural Invertibility and Optimal Sensor Node Placement for Error and Input Reconstruction in Dynamic Systems, Phys. Rev. X, doi 10.1103/PhysRevX.9.041046, December 2019
- [4] Hermann R. and Krener A.J., 1977, Nonlinear Controllability and Observability, Transaction On Automatic Control, AC-22(5): 728–740.
- [5] Casti J. L., Recent developments and future perspectives in nonlinear system theory, SIAM Review, vol. 24, No. 3, July 1982.
- [6] H.J. Sussman, Lie brackets, real analyticity, and geometric control In Differential Geometric Control Theory. Conference held at MTU (Houghton, MI, June 1982). R.W. Brockett, R.S. Millman, H.J. Sussman Eds. Progress in Mathematics 27. Birkhauser Boston/Basel/Stuttgart. 1- 116 (1983).
- [7] Isidori A., Nonlinear Control Systems, 3rd ed., London, Springer Verlag, 1995.
- [8] M. K. Paul and S. I. Roumeliotis, Alternating-stereo VINS: Observability analysis and performance evaluation, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pages 4729–4737
- [9] Joel A. Hesch, Dimitris G. Kottas, Sean L. Bowman, Stergios I. Roumeliotis, Camera-IMU-based Localization: Observability Analysis and Consistency Improvement, International Journal of Robotics Research, 2014, Vol 33(1) 182–201
- [10] Fabrizio Schiano, Roberto Tron, The dynamic bearing observability matrix nonlinear observability and estimation for multi-agent systems, 2018 IEEE International Conference on Robotics and Automation
- [11] Yulin Yang, Guoquan Huang, Observability Analysis of Aided INS With Heterogeneous Features of Points, Lines, and Planes, IEEE Transactions on Robotics, 2019 35(6), 1399–1418
- [12] Chao X Guo, Stergios I Roumeliotis, IMU-RGBD camera 3D pose estimation and extrinsic calibration: Observability analysis and consistency improvement, 2013 IEEE International Conference on Robotics and Automation, 2935–2942
- [13] Manolis N. Chatzis, Eleni N. Chatzi, Andrew W. Smyth, On the observability and identifiability of nonlinear structural and mechanical systems, Structural Control and Health Monitoring, Vol. 22, 574–593, 2015
- [14] Alejandro F. Villaverde, Observability and Structural Identifiability of Nonlinear Biological Systems, Complexity, 2019, Special Issue on Computational Methods for Identification and Modelling of Complex Biological Systems
- [15] J. III. DiStefano, Dynamic systems biology modeling and simulation, Academic Press, 2015.
- [16] A. F. Villaverde and A. Barreiro, Identifiability of large nonlinear biochemical networks, MATCH - Communications in Mathematical and in Computer Chemistry, vol. 76, no. 2, pp. 259–296, 2016.
- [17] O.-T. Chis, J. R. Banga, and E. Balsa-Canto, Structural identifiability of systems biology models: A critical comparison of methods, PLoS ONE, vol. 6, no. 11, 2011.
- [18] H.Miao, X. Xia, A. S. Perelson, and H.Wu, On identifiability of nonlinear ODE models and applications in viral dynamics, SIAM Review, vol. 53, no. 1, pp. 3–39, 2011.
- [19] Elias August 1, Antonis Papachristodoulou, A New Computational Tool for Establishing Model Parameter Identifiability, J Comput Biol. 2009, 16(6): 875–885.
- [20] G. P. Huang, A. I. Mourikis and S. I. Roumeliotis, Observability-based Rules for Designing Consistent EKF SLAM Estimators, The International Journal of Robotics Research, 2010, 29(5), 502–528.
- [21] G. P. Huang, A. I. Mourikis and S. I. Roumeliotis, A Quadratic-Complexity Observability-Constrained Unscented Kalman Filter for SLAM, in IEEE Transactions on Robotics, vol. 29, no. 5, pp. 1226-1243, Oct. 2013.
- [22] Alejandro F. Villaverde, Antonio Barreiro, and Antonis Papachristodoulou, Structural Identifiability of Dynamic Systems Biology Models, PLoS Comput Biol. 2016 Oct; 12(10).
- [23] G. Basile and G. Marro. On the observability of linear, time invariant systems with unknown inputs. J. Optimization Theory Appl., 3:410–415, 1969.
- [24] R. Guidorzi and G. Marro. On Wonham stabilizability condition in the synthesis of observers for unknown-input systems. Automatic Control, IEEE Transactions on, 16(5):499–500, oct 1971.
- [25] H. Geng, M. A. Haile, H. Fang, SSUE: Simultaneous state and uncertainty estimation for dynamical systems, International Journal of Robust and Nonlinear Control, Volume 31, Issue 4, March 2021
- [26] F. A. W. Belo, P. Salaris, and A. Bicchi, 3 Known Landmarks are Enough for Solving Planar Bearing SLAM and Fully Reconstruct Unknown Inputs, IROS 2010, Taipei, Taiwan
- [27] A. Martinelli. Extension of the Observability Rank Condition to Nonlinear Systems Driven by Unknown Inputs. 23th Mediterranean Conference on Control and Automation (MED), June, 2015. Torremolinos, Spain
- [28] Maes, K., Chatzis, M., Lombaert, G., 2019. Observability of nonlinear systems with unmeasured inputs. Mech. Syst. Signal Process. 130, 378–394.
- [29] 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 16, 2019
- [30] A. Martinelli, Nonlinear Unknown Input Observability: Extension of the Observability Rank Condition, IEEE Transactions on Automatic Control, Vol 64, No 1, pp 222–237
- [31] A. Martinelli, Observability: A New Theory Based on the Group of Invariance, published by SIAM in Advances in Design and Control, 2020, ISBN: 978-1-61197-624-3.
- [32] M. A. Sarafrazi, U. Kotta and Z. Bartosiewicz, "On the Stopping Criteria in Nonlinear Unknown Input Observability Condition," in IEEE Transactions on Automatic Control, doi: 10.1109/TAC.2022.3160133.
- [33] D. Nistér, An efficient solution to the five-point relative pose problem, IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 26(6):756-770, June 2004