Differential elimination for dynamical models via projections with applications to structural identifiability

Ruiwen Dong,* Christian Goodbrake,† Heather A Harrington,‡ and Gleb Pogudin§

Abstract. Elimination of unknowns in a system of differential equations is often required when analysing (possibly nonlinear) dynamical systems models, where only a subset of variables are observable. One such analysis, identifiability, often relies on computing input-output relations via differential algebraic elimination. Determining identifiability, a natural prerequisite for meaningful parameter estimation, is often prohibitively expensive for medium to large systems due to the computationally expensive task of elimination. We propose an algorithm that computes a description of the set of differential-algebraic relations between the input and output variables of a dynamical system model. The resulting algorithm outperforms general-purpose software for differential elimination on a set of benchmark models from literature. We use the designed elimination algorithm to build a new randomized algorithm for assessing structural identifiability of a parameter in a parametric model. A parameter is said to be identifiable if its value can be uniquely determined from input-output data assuming the absence of noise and sufficiently exciting inputs. Our new algorithm allows the identification of models that could not be tackled before. Our implementation is publicly available as a Julia package at https://github.com/SciML/StructuralIdentifiability.jl.

Key words. differential elimination, parameter identifiability, dynamical systems

AMS subject classifications. 12H05, 13P25, 93B30, 93B25, 34A55

1. Introduction.

1.1. Overview.
Models defined by systems of differential equations are widely used in engineering and sciences. One of the fundamental challenges in designing, testing, calibrating, and using such models comes from the fact that, in practice, often only few of the variables are observed/measured. Despite a deluge of data and advances in technology, the appropriate data may be inaccessible (e.g., prohibitively expensive to attach sensors to all parts of a mechanism, infeasible to observe protein complexes in distinct states, or impossible to measure all the proteins in a model in a single experimental set up). Therefore, the relations involving only observable variables play an important role in the systems and control theory (referred to as *input-output relations*, see, e.g., the textbooks [14, 56] and [63]). For example, computing these input-output relations explicitly is a key step in a so-called differential algebra approach to assessing structural identifiability of a dynamical model [45, 4, 40, 43]. Other applications include linearization [26], model selection [29], parameter estimation [61, 20], fault diagnosis [36, 57], and control [37].

*École Polytechnique, Route de Saclay, 91120 Palaiseau, France; Department of Computer Science, University of Oxford, Oxford, OX1 3QD, United Kingdom

†Oden Institute for Computational Engineering and Sciences, University of Texas, Austin, 201 E 24th St, Austin, TX 78712; Mathematical Institute, University of Oxford, Oxford, OX2 6GG, United Kingdom

‡Mathematical Institute, University of Oxford, Oxford, OX2 6GG, United Kingdom; Wellcome Centre for Human Genetics, University of Oxford, Oxford, OX3 7BN, United Kingdom

§LIX, CNRS, École Polytechnique, Institute Polytechnique de Paris, 1 rue Honoré d'Estienne d'Orves, 91120, Palaiseau, France, email: gleb.pogudin@polytechnique.edu

The problem of computing the input-output relations can be viewed as the *differential elimination problem*: given a system of differential equations

$$f_1(\mathbf{x}, \mathbf{y}) = \ldots = f_n(\mathbf{x}, \mathbf{y}) = 0$$

in two groups of unknowns $\mathbf{x} = (x_1, \ldots, x_s)$ and $\mathbf{y} = (y_1, \ldots, y_\ell)$, describe all equations $g(\mathbf{y}) = 0$ in $\mathbf{y}$ only, which hold for every solution of the system. This problem has been one of the central problems in the algebraic theory of differential equations. Its study has been initiated by Ritt, the founder of differential algebra, in the 1930s [52]. He developed the foundations of the characteristic set approach, which has been made fully constructive by Seidenberg [55]. The algorithmic aspect of this research culminated in the Rosenfeld-Gröbner algorithm [7, 34] implemented in the BLAD library [6] (available through MAPLE). See [1, 62] for related software. These algorithms and packages are very versatile: they can be applied to arbitrary systems of polynomials PDEs. There is a price to pay for such versatility: many interesting examples coming from applications cannot be tackled in a reasonable time. On the other hand, since differential equations in sciences and engineering are typically used to describe how the system of interest will evolve from a given state, many dynamical models in the literature are described by systems in *the state-space form*:

(1.1) $$\mathbf{x}' = \mathbf{f}(\mathbf{x}, \mathbf{u}), \quad \mathbf{y} = \mathbf{g}(\mathbf{x}, \mathbf{u})$$

where $\mathbf{f}$ and $\mathbf{g}$ are tuples of rational functions, $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{u}$ are tuples of differential unknowns (the state, output, and input variables, respectively). For such a system, one typically wants to eliminate the x-variables, that is, compute the *input-output relations*, the relations between the y-variables and u-variables. For example, such a computation is sufficient for all the applications mentioned in the beginning.

The contribution of the present paper is two-fold:

1. **Elimination.** We propose *a new way to describe the input-output relations* of (1.1) and design *a computationally tractable algorithm* for computing this description. We demonstrate that this algorithm outperforms the existing general-purpose elimination software (e.g., computations that took hours or didn't finish are computed in minutes, see Table 5).

2. **Identifiability.** We build *a new randomized algorithm for assessing structural identifiability* of parametric dynamical models on top of the elimination algorithm. The algorithm can handle problems that could not be solved by any existing identifiability software (see comparison to the state-of-the-art methods in Table 4). Our software is available as a Julia package at https://github.com/SciML/StructuralIdentifiability.jl.

The next two subsections describe the contributions in more detail.

1.2. Elimination. We propose to use a projection-based description of the input-output relations of (1.1) which can be viewed as a generalization of the state-space form (that is, the form (1.1)) itself. In order to motivate and introduce this description, we adopt the following algebro-geometric viewpoint on (1.1): we consider all the derivatives of the equations in (1.1) as an infinite system of equations describing a variety in an infinite-dimensional space with the coordinates $\mathbf{x}, \mathbf{y}, \mathbf{u}, \mathbf{x}', \mathbf{y}', \mathbf{u}', \ldots$. The points of this variety over $\mathbb{C}$ will be in bijective

correspondence (via the Taylor series [50, Lemma 3.5]) with the formal power series solutions of (1.1). We observe that the variety is rationally parametrized by $\mathbf{x}, \mathbf{u}, \mathbf{u}', \ldots$ (we will refer to these variables as *the base variables*). Moreover, each equation in (1.1) relates these base variables and one non-base variable of the lowest order, that is, one of $\{\mathbf{x}', \mathbf{y}\}$.

We can now generalize such a description to the notion of *a projection-based representation* of the ideal $\mathcal{I}$ generated by the derivatives of (1.1) by allowing

- any set of base variables such that they form a transcendence basis modulo $\mathcal{I}$ and, if $w_j^{(i)}$ (the i-th derivative of w_j) with $w \in \{x, y, u\}$ is a base variable, then $w_j^{(0)}, \ldots, w_j^{(i-1)}$ are base variables as well;
- the relations between the base variables and one of the lowest order non-base variables (*projections*) be nonlinear in the latter (i.e., parametrization of the variety may no longer be rational).

Example 1. *A toy example of changing the base variables (base variables are underlined):*

$$(1.2) \quad \begin{cases} x' = \underline{x}, \\ y = \underline{x}^2 + \underline{u} \end{cases} \implies (using\ y' = 2xx' + u' = 2x^2 + u') \implies \begin{cases} x^2 = \underline{y} - \underline{u}, \\ y' = 2\underline{y} - 2\underline{u} + \underline{u}'. \end{cases}$$

Note that such a change of the set of base variables may add extraneous prime components to the variety, this subtlety and the way we deal with it are discussed in Remark 3.

Example 1 also suggests how projection-based representations could be used for differential elimination: observe that the last equation in (1.2), $y' = 2y - 2u + u'$, is an input-output equation of minimal order for the original model. Indeed, we will show that in general, if one considers a set of base variables containing as many derivatives of y-variables as possible, a subset of projections will form a projection-based representation of the ideal of input-output relations (see Lemma 2.11). The main idea is to replace the x-variables in the set of base variables one by one with y-variables and their derivatives. We visualize such a computation for a simple artificial example on Figure 1.

$$\begin{cases} x_1' = x_2, \\ x_2' = x_3, \\ x_3' = x_1, \\ y_1 = x_1 + x_2, \\ y_2 = x_3 \end{cases} \implies \begin{cases} x_1' = x_2, \\ x_2' = y_2, \\ x_3 = y_2, \\ y_1 = x_1 + x_2, \\ y_2' = x_1 \end{cases} \implies \begin{cases} x_1' = y_1 - x_1, \\ x_2 = y_1 - x_1, \\ x_3 = y_2, \\ y_1' = y_1 - x_1 + y_2, \\ y_2' = x_1 \end{cases} \implies \begin{cases} x_1 = y_2', \\ x_2 = y_1 - y_2', \\ x_3 = y_2, \\ y_1' = y_1 - y_2' + y_2, \\ y_2'' = y_1 - y_2' \end{cases}$$

Figure 1: Elimination via a chain of projection-based representations. The diagrams in the second row describe the set of base variables at each step.

The key features of the projection-based representation that allow us to translate such an approach into a practically efficient algorithm are the following:

- The projection based representation is given by equations of the projections of the original variety defined by $\mathcal{I}$; hence, *geometric* in nature. In contrast to syntactic representations, such as characteristic sets or Gröbner bases, the projection based representation enables us to use tools from constructive algebraic geometry.
- The elimination process visualized in Figure 1 is *adaptive* in the sense that, after each step, the algorithm often has a choice which of the x's to replace with one of the y's (e.g., at the first step in Figure 1, we could eliminate any of x_1, x_2, x_3 from the base variables). By employing simple degree-based heuristics for this choice, we are able to speed up the computation substantially. While the described algorithm may be reminiscent of the Gröbner walk algorithm [13] and its relatives [16, 27], especially [8], the ranking of y's is not predetermined. The distinctive feature of the proposed algorithm is that the final ranking of y's can be constructed *adaptively and dynamically*. Note that the choice of ranking affects only the runtime of the algorithm but not the correctness of the result.

Since each step of the algorithm boils down to elimination of a single variable (one of the x's), we use resultants as the main algebraic elimination tool. While resultants are known to produce extraneous factors, we use two strategies to address this issue:

1. Thanks to efficient algorithms [5, 59, 60] for computing power series solutions of the original system (1.1), we have an efficient randomized membership test for ideal $\mathcal{I}$. This membership test allows us to remove extraneous factors after each resultant computation and evade accumulation of these factors during repeated resultant computation.
2. Our algorithm may perform a change of variables in the original system. Therefore, some of these extraneous factors are extracted *before* computing the resultant (see Section 4.5), hence speeding up the computation substantially (Table 1).

1.3. Identifiability. In a parametric ODE model (that is, (1.1) in which coefficients involve unknown scalar parameters), a parameter or a function of parameters is called *structurally globally identifiable* (in what follows, just "identifiable") if its value can be uniquely determined from the input-output data, assuming the absence of noise and sufficiently exciting inputs. This identifiability property is a natural prerequisite for practical parameter estimation, and, therefore, it is an important step in the experimental design process. The problem of assessing identifiability has been studied since the 1970-s [3]. Since then, a number of different approaches have been proposed (see [11, 32] and references therein) and several software packages and webapps have been developed [4, 44, 39, 35]. However, many models of practical interest remain out of reach with the existing tools.

One popular approach to assess identifiability is the *input-output relations* approach proposed in [45] (recently used, e.g., in [51, 65, 58]) and relies on theory of differential algebra. It has been implemented in DAISY [4], COMBOS [44], and Structural Identifiability Toolbox [35]. First, the software computes a set of generators of the field of definition F of the ideal of the input-output relations, that is, the minimal subfield of the field of rational functions in parameters sufficient to write down the generators of this ideal [49, Definition 2.5]. These

generators are typically taken to be the coefficients of a finite set of input-output relations, which may be computed using a characteristic set (as in DAISY [4]), a Gröbner basis (as in COMBOS [44]), or elimination by hand (e.g., [51, 58]). Next, the identifiability of a parameter is assessed by testing whether it belongs to F. There are two important subtleties with existing algorithms following this approach:

1. **Single vs. multiple experiments.** In general, F is equal to the field of functions identifiable from *several* experiments [47, Theorem 19]. There is a sufficient condition ensuring that all the elements of F can be identified from a single experiment which can be verified by computing the Wronskian of the monomials of some input-output relations [49, Lemma 1]. The condition is not checked by DAISY or COMBOS. However, it has been checked manually in several case studies [21, 64, 61] with about a dozen of monomials and can be checked automatically by [35] if the number of monomials does not exceed a hundred.

2. **Probability of correctness.** The membership test of a parameter of interest in F is typically framed as an injectivity test for a so-called coefficient map. While this can be done deterministically using Gröbner bases with rational function coefficients, such a computation would be very costly. Because of this, both DAISY and COMBOS take a random point in the image of the map and check whether the preimage of the point is of cardinality one. Such an approach may yield an incorrect result if the chosen point was not generic, and the bounds on the probability of error are not provided by DAISY or COMBOS.

We address these issues and design a new algorithm for assessing structural parameter identifiability. More precisely, we:

1. show how to compute the field of definition F from a projection-based representation of the ideal of input-output relations (Algorithm 4.5).
2. design and implement a practical algorithm for checking the necessary condition for the elements of F to be identifiable from a single experiment, which terminates even when the input-output relations have a couple of thousands of monomials (Algorithm 5.2).
3. give a probability bound for testing field membership via randomization (Theorem 3.3).

The last two items above can be used with any other elimination algorithms (e.g., used in [4, 44, 35]).

The resulting algorithm is implemented in Julia in the STRUCTURALIDENTIFIABILITY package[1] which is a part of the SciML (scientific machine learning) ecosystem. Our implementation addresses the two issues outlined above and can solve problems that could not be computed before (see Table 4).

1.4. Structure of the paper. The rest of the paper is organized as follows. Section 2 contains preliminaries on differential algebra, structural identifiability, and a precise description of the projection-based representation. Section 3 describes our main theoretical results. In Section 4 we describe and justify our algorithm for differential elimination via the projection-based representation. In Section 5, we propose an algorithm for assessing structural identifiability based on our elimination algorithm. Section 6 describes our implementation and its performance. The benchmark models are listed in the Appendix.

---

2. Preliminaries.

2.1. Differential algebra. Throughout the paper, the ideal generated by $f_1, \ldots, f_n$ will be denoted by $\langle f_1, \ldots, f_n \rangle$.

Definition 2.1 (Differential rings and fields). *A differential ring $(R, {}')$ is a commutative ring with a derivation ${}' \colon R \to R$, that is, a map such that, for all $a, b \in R$, $(a + b)' = a' + b'$ and $(ab)' = a'b + ab'$. A differential field is a differential ring that is a field. For $i > 0$, $a^{(i)}$ denotes the i-th order derivative of $a \in R$.*

Notation 1. *Let x be an element of a differential ring and $h \in \mathbb{Z}_{\geqslant 0}$. We introduce*

$$x^{(<h)} := (x, x', \ldots, x^{(h-1)}),$$
$$x^{(\infty)} := (x, x', x'', \ldots).$$

$x^{(\leqslant h)}$ *is defined analogously. If $\mathbf{x} = (x_1, \ldots, x_n)$ is a tuple of elements of a differential ring and $\mathbf{h} = (h_1, \ldots, h_n) \in (\mathbb{Z}_{\geqslant 0} \cup \{\infty\})^n$, then*

$$\mathbf{x}^{(<h)} := (x_1^{(<h)}, \ldots, x_n^{(<h)}),$$
$$\mathbf{x}^{(<\mathbf{h})} := (x_1^{(<h_1)}, \ldots, x_n^{(<h_n)}),$$
$$\mathbf{x}^{(\infty)} := (x_1^{(\infty)}, \ldots, x_n^{(\infty)}).$$

Definition 2.2 (Differential polynomials). *Let R be a differential ring. Consider a ring of polynomials in infinitely many variables*

$$R[x^{(\infty)}] := R[x, x', x'', x^{(3)}, \ldots]$$

and extend the derivation from R to this ring by $(x^{(j)})' := x^{(j+1)}$. The resulting differential ring is called the ring of differential polynomials in x over R.

The ring of differential polynomials in several variables is defined by iterating this construction.

Definition 2.3 (Differential ideals). *Let $S := R[x_1^{(\infty)}, \ldots, x_n^{(\infty)}]$ be a ring of differential polynomials over a differential ring R. An ideal $I \subset S$ is called* a differential ideal *if $a' \in I$ for every $a \in I$.*

One can verify that, for every $f_1, \ldots, f_s \in S$, the ideal

$$\langle f_1^{(\infty)}, \ldots, f_s^{(\infty)} \rangle$$

is a differential ideal. Moreover, this is the minimal differential ideal containing $f_1, \ldots, f_s$, and we will denote it by $\langle f_1, \ldots, f_s \rangle^{(\infty)}$.

Notation 2 (Saturation). *Let R be a ring, $I \subset R$ be an ideal, and $a \in R$. We introduce*

$$I : a^\infty := \{b \in R \mid \exists N \colon a^N b \in I\},$$

which is also an ideal in R.

6

2.2. Structural identifiability. Consider an ODE system of the form

$$(2.1) \qquad \Sigma = \begin{cases} \mathbf{x}' = \mathbf{f}(\mathbf{x}, \boldsymbol{\mu}, \mathbf{u}), \\ \mathbf{y} = \mathbf{g}(\mathbf{x}, \boldsymbol{\mu}, \mathbf{u}), \end{cases}$$

where
- $\mathbf{x} = (x_1, \ldots, x_n)$, $\mathbf{y} = (y_1, \ldots, y_m)$, and $\mathbf{u} = (u_1, \ldots, u_s)$ are the vectors of state, output, and input variables, respectively;
- $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_\ell)$ is a vector of constant parameters;
- $\mathbf{f} = (f_1, \ldots, f_n)$ and $\mathbf{g} = (g_1, \ldots, g_m)$ are vectors of elements of $\mathbb{C}(\mathbf{x}, \boldsymbol{\mu}, \mathbf{u})$

By reducing $\mathbf{f}$ and $\mathbf{g}$ to the common denominator, we write $\mathbf{f} = \mathbf{F}/Q$ and $\mathbf{g} = \mathbf{G}/Q$, for $F_1, \ldots, F_n, G_1, \ldots, G_m, Q \in \mathbb{C}[\mathbf{x}, \boldsymbol{\mu}, \mathbf{u}]$. Consider the (prime, see [32, Lemma 3.2]) differential ideal

$$(2.2) \qquad I_\Sigma := \langle Qx_i' - F_i, Qy_j - G_j, 1 \leqslant i \leqslant n, 1 \leqslant j \leqslant m \rangle^{(\infty)} \colon Q^\infty \subset \mathbb{C}(\boldsymbol{\mu})[\mathbf{x}, \mathbf{y}, \mathbf{u}].$$

Note that every element of I_Σ vanishes on every analytic solution of (2.1).

We will use the following algebraic definition of identifiability. Its equivalence with the general analytic definition [32, Definition 2.5] (see also [48, Definition 2.3]) has been established [32, Proposition 3.4].

Definition 2.4 (Single-experiment identifiability). *A rational function $h \in \mathbb{C}(\boldsymbol{\mu})$ will be called globally (single-experiment, or SE-) identifiable for (2.1) if, there exist $P_0, P_1 \in \mathbb{C}[\mathbf{y}^{(\infty)}, \mathbf{u}^{(\infty)}]$ such that $P_1 \notin I_\Sigma$ and $P_1 h - P_0 \in I_\Sigma$.*

We also say that h is locally SE-identifiable *if there exist a positive integer s and $P_0, \ldots, P_s \in \mathbb{C}[\mathbf{y}^{(\infty)}, \mathbf{u}^{(\infty)}]$ such that $P_s \notin I_\Sigma$ and $P_s h^s + P_{s-1} h^{s-1} + \ldots + P_0 \in I_\Sigma$.*

Remark 1. *Informally, the definition can be stated as follows: h is globally SE-identifiable if and only if it can be expressed via inputs, outputs, and their derivatives via a formula $h = \frac{P_0(\mathbf{y}, \mathbf{u})}{P_1(\mathbf{y}, \mathbf{u})}$. While the existence of a formula may look like a special case of identifiability, [32, Proposition 3.4] shows that the possibility of a unique identification of h (formulated in analytic terms) is equivalent to the existence of such a formula.*

Example 2 (SE-identifiability). *Consider a harmonic oscillator:*

$$\Sigma \colon x_1' = -\mu x_2, \ x_2' = \mu x_1, \ y = x_1.$$

Observe that

$$(y - x_1)'' + (x_1' + \mu x_2)' - \mu(x_2' - \mu x_1) + \mu^2 (y - x_1) = y'' + \mu^2 y \in I_\Sigma.$$

From $y'' + \mu^2 y \in I_\Sigma$ we can conclude that $h = \mu^2$ is globally identifiable (with $P_1 = y, P_0 = y''$) and μ is locally identifiable (with $s = 2, P_2 = y, P_1 = 0, P_0 = y''$).

Definition 2.5 (Multi-experiment identifiability).

- *For a model Σ and a positive integer r, we define* the r-fold replica *of Σ as*

$$\Sigma_r := \begin{cases} \mathbf{x}'_i = \mathbf{f}(\mathbf{x}_i, \boldsymbol{\mu}, \mathbf{u}_i), & i = 1, \dots, r, \\ \mathbf{y}_i = \mathbf{g}(\mathbf{x}_i, \boldsymbol{\mu}, \mathbf{u}_i), & i = 1, \dots, r, \end{cases}$$

  where $\mathbf{x}_1, \dots, \mathbf{x}_r, \mathbf{y}_1, \dots, \mathbf{y}_r, \mathbf{u}_1, \dots, \mathbf{u}_r$ are new tuples of indeterminates (note that the vector of parameters is not being replicated).
- *For a model Σ, a rational function $h \in \mathbb{C}(\boldsymbol{\mu})$ is* globally multi-experimental identifiable *(ME-identifiable) if there exists a positive integer r such that $h(\boldsymbol{\mu})$ is globally SE-identifiable in Σ_r.* Local ME-identifiability *is defined analogously.*

Definition 2.6 (Field of definition). *Let $J \subset K[\mathbf{x}^{(\infty)}]$ (with $\mathbf{x} = (x_1, \dots, x_n)$) be a differential ideal over a differential field K. Then the smallest differential subfield $L \subset K$ such that J is generated by $J \cap L[\mathbf{x}^{(\infty)}]$ is called the* field of definition *of J.*

Theorem 2.7 ([47, Theorem 21]). *The field of multi-experimental identifiable functions is generated over $\mathbb{C}$ by the field of definition of*

$$I_\Sigma \cap \mathbb{C}(\boldsymbol{\mu})[\mathbf{y}^{(\infty)}, \mathbf{u}^{(\infty)}].$$

Therefore, the problem of computing the field of ME-identifiable functions reduces to the problem of computing the field of definition of a projection of an irreducible differential-algebraic variety. Furthermore, using a Wronskian-based criterion [49, Lemma 1], one can in many cases (see Table 3) establish that SE-identifiable and ME-identifiable functions coincide.

Example 3 (ME-identifiability). *In order to illustrate the difference between the SE- and ME-identifiability, we consider the following artificial example (a version of [32, Example 2.14]):*

$$\Sigma \colon x' = 0, \ y_1 = x, \ y_2 = \mu_1 x + \mu_2.$$

Since $y'_1 = y'_2 = 0$ and there is no algebraic relation between y_1 and y_2 modulo I_Σ, we have $I_\Sigma \cap \mathbb{C}[y_1^{(\infty)}, y_2^{(\infty)}] = \langle y_1, y_2 \rangle^{(\infty)}$. Therefore, no function of μ_1, μ_2 is SE-identifiable.
  On the other hand, modulo I_{Σ_2}, we will have

$$y_{1,2} = \mu_1 y_{1,1} + \mu_2 \quad and \quad y_{2,2} = \mu_1 y_{1,2} + \mu_2.$$

These equations yield a linear system for μ_1 and μ_2, in particular, by subtracting these equations, we obtain:

$$\mu_1(y_{1,1} - y_{2,1}) - (y_{1,2} - y_{2,2}) \in I_{\Sigma_2},$$

so μ_1 is globally identifiable in Σ_2 and, thus, globally ME-identifiable in Σ (same for μ_2).

2.3. Projections of an irreducible differential-algebraic variety.

Notation 3.
- $\overline{\mathbb{Z}}_{\geqslant 0} := \mathbb{Z}_{\geqslant 0} \cup \{\infty\}$.
- *For $n > 0$ and $1 \leqslant i \leqslant n$, $\mathbf{e}_{i,n}$ denotes the i-th basis vector in $\mathbb{Z}^n$. If n is clear from the context, we will write simply $\mathbf{e}_i$.*

Definition 2.8 (Parametric profile). *For a prime differential ideal $P \subset K[\mathbf{x}^{(\infty)}]$ with $\mathbf{x} = (x_1, \ldots, x_n)$, a tuple $\mathbf{h} \in \overline{\mathbb{Z}}_{\geqslant 0}^n$ is called a* parametric profile *for P if the images of $\mathbf{x}^{(<\mathbf{h})}$ form a transcendence basis of $K[\mathbf{x}^{(\infty)}]/P$. This notion may be viewed as a generalization of observability indices from control theory [14, §4.6].*

Lemma 2.9. *Let $P \subset K[\mathbf{x}^{(\infty)}]$ (with $\mathbf{x} = (x_1, \ldots, x_n)$) be a prime differential ideal and $\mathbf{h} \in \overline{\mathbb{Z}}_{\geqslant 0}^n$ be a parametric profile for P. Then the ideal*

$$P \cap K[\mathbf{x}^{(<\mathbf{h}+\mathbf{e}_i)}]$$

is principal for every $1 \leqslant i \leqslant n$ such that $h_i \neq \infty$.

Proof. Fix i such that $h_i \neq \infty$. Let $f \in P \cap K[\mathbf{x}^{(<\mathbf{h}+\mathbf{e}_i)}]$ be an irreducible polynomial of minimal degree in $x_i^{(h_i)}$. Since the leading coefficient of f does not belong to P, for every other polynomial $g \in P \cap K[\mathbf{x}^{(<\mathbf{h}+\mathbf{e}_i)}]$, the result of pseudo-division of g by f with respect to $x_i^{(h_i)}$ belongs to P. The minimality of $\deg_{x_i^{(h_i)}} f$ implies that this result is zero, so g is pseudo-divisible by f with respect to $x_i^{(h_i)}$. Then the Gauss lemma implies that g is divisible by f. ∎

Definition 2.10. *For a prime differential ideal $P \subset K[\mathbf{x}^{(\infty)}]$ (where $\mathbf{x} = (x_1, \ldots, x_n)$) and its parametric profile $\mathbf{h}$, a tuple of differential polynomials $(f_1, \ldots, f_n)$ is called* the projections *corresponding to $\mathbf{h}$ if, for every $1 \leqslant i \leqslant n$,*
1. *if $h_i = \infty$, $f_i = 0$;*
2. *if $h_i < \infty$, f_i is the generator of $P \cap K[\mathbf{x}^{(<\mathbf{h}+\mathbf{e}_i)}]$.*

The following lemma describes how one can use the projections for differential elimination.

Lemma 2.11 (Projections under projection). *Let $(f_1, \ldots, f_n)$ be the projections of a prime differential ideal $P \subset K[\mathbf{x}^{(\infty)}]$ with respect to its profile $\mathbf{h}$. Assume that, for some $m \leqslant n$, differential polynomials $f_1, \ldots, f_m$ do not involve $x_{m+1}, \ldots, x_n$ and their derivatives.*

Then $(h_1, \ldots, h_m)$ is a parametric profile of $P \cap K[x_1^{(\infty)}, \ldots, x_m^{(\infty)}]$, and the corresponding projections are $(f_1, \ldots, f_m)$.

Proof. Since $x_1^{(<h_1)}, \ldots, x_m^{(<h_m)}$ are algebraically independent modulo P, they are algebraically independent modulo $P \cap K[x_1^{(\infty)}, \ldots, x_m^{(\infty)}]$ as well. Moreover, they form a transcendence basis of $K[x_1^{(<h_1)}, \ldots, x_m^{(<h_m)}]/P \cap K[x_1^{(\infty)}, \ldots, x_m^{(\infty)}]$ because, for every $1 \leqslant i \leqslant m$, f_i involves only $x_1^{(<h_1)}, \ldots, x_m^{(<h_m)}, x_i^{(h_i)}$ and thus provides an algebraic dependence between their images. Therefore, $(h_1, \ldots, h_m)$ is indeed a parametric profile for $P \cap K[x_1^{(\infty)}, \ldots, x_m^{(\infty)}]$. Then one can see that $(f_1, \ldots, f_m)$ are the corresponding projections. ∎

Remark 2 (Projections for ODE models). *For an ideal $I_\Sigma \subset \mathbb{C}(\boldsymbol{\mu})[\mathbf{x}^{(\infty)}, \mathbf{y}^{(\infty)}, \mathbf{u}^{(\infty)}]$, corresponding to a system Σ of the form (2.1), we have the following parametric profile (see [32, Lemma 3.1]):*

$$(2.3) \qquad (\underbrace{1, 1, \ldots, 1}_{n \; times}, \underbrace{0, 0, \ldots, 0}_{m \; times}, \underbrace{\infty, \ldots, \infty}_{s \; times}).$$

Furthermore, the corresponding projections are

$$(Qx_1' - F_1, \ldots, Qx_n' - F_n, Qy_1 - G_1, \ldots, Qy_m - G_m, \underbrace{0, \ldots, 0}_{s \text{ times}}).$$

By Lemma 2.11, one can compute the projections of the ideal of input-output relations, that is $I_\Sigma \cap \mathbb{C}(\boldsymbol{\mu})[\mathbf{y}^{(\infty)}, \mathbf{u}^{(\infty)}]$, by "moving" as many of the first n ones in (2.3) as possible to the y-variables (see Figure 1 and Example 4).

Example 4. Consider system Σ in variables x_1, x_2, y which is a harmonic oscillator with one of the coordinates being observed:

$$(2.4) \qquad\qquad x_1' = -x_2, \quad x_2' = x_1, \quad y = x_1.$$

As explained in Remark 2, (2.4) are the projections corresponding to profile $(1, 1, 0)$. Since a general solution of (2.4) is of the form $y = c_1 e^{it} + c_2 e^{-it}$ for arbitrary $c_1, c_2 \in \mathbb{C}$ and every element of I_Σ must vanish on every solution of (2.4), there is no nonzero element in I_Σ depending only on y and y', so they are algebraically independent. Since, modulo I_Σ, any polynomial can be reduced to a polynomial in x_1 and x_2, the transcendence degree modulo I_Σ is 2, so y and y' form a transcendence basis modulo I_Σ. This yields a different profile $(0, 0, 2)$ for the same ideal. Now we compute the corresponding projections:

- *the projection relating x_1, y, y' is just one of (2.4): $x_1 - y = 0$;*
- *the projection relating x_2, y, y' can be obtained from the previous one by differentiation: $x_2 - y' = 0$;*
- *the projection relating y, y, y'' is the classical equation for the harmonic oscillator: $y'' + y = 0$. By Lemma 2.11, $y'' + y = 0$ is the projection of the ideal of input-output relations.*

Remark 3 (Extra components in the projection-based representation). Let $(f_1, \ldots, f_n)$ be the projections of a prime differential ideal $P \subset K[\mathbf{x}^{(\infty)}]$ corresponding to a profile $\mathbf{h}$. Similar to (2.2), one can define

$$(2.5) \qquad\qquad I_{\mathbf{h}} := \langle f_1, \ldots, f_n \rangle^{(\infty)} : (H \cdot S)^\infty,$$

where H and S are, respectively, the products, over all i's with $f_i \neq 0$, of the leading coefficients and the derivatives of f_i with respect to $x_i^{(h_i)}$. Since P is a prime containing $I_{\mathbf{h}}$ and has the same transcendence basis as all associated primes of $I_{\mathbf{h}}$ by [33, Theorem 4.4], P is a prime component of $I_{\mathbf{h}}$. However, in general, $I_{\mathbf{h}}$ may have additional prime components as the following example shows. Although the example may look involved, it is an ODE version of the geometric fact that an intersection of two circular cylinders of the same radius and with intersecting axes consists of two components (ellipses).

Consider an ODE system Σ of the form (2.1):

$$x_1' = \frac{1}{2}(1 + x_1^2), \quad x_2' = \frac{1 - x_1^2}{1 + x_1^2}, \quad y_1 = \frac{2x_1}{\mu(1 + x_1^2)}, \quad y_2 = x_2.$$

The corresponding ideal I_Σ is prime, and one can show (using Algorithm 4.3) that it has a parametric profile $\mathbf{h} := (0, 0, 1, 1)$, and the corresponding projections are:

$$\mu y_1(\underline{x_1}^2 + 1) - 2\underline{x_1} = 0, \quad \underline{x_2} - y_1 = 0, \quad \mu^2(\underline{y_1'})^2 + \mu^2 y_1^2 - 1 = 0, \quad (\underline{y_2'})^2 + \mu^2 y_1^2 - 1 = 0,$$

where the variables of the form $x_i^{(h_i)}$ are underlined. By computing the difference of the last two equations, we have

(2.6) $(\mu^2(y_1')^2 + \mu^2 y_1^2 - 1) - ((y_2')^2 + \mu^2 y_1^2 - 1) = \mu^2(y_1')^2 - (y_2')^2 = (\mu y_1' - y_2')(\mu y_1' + y_2') \in I_{\mathbf{h}}.$

One can check that neither of $\mu y_1' - y_2'$ and $\mu y_1' + y_2'$ belongs to $I_{\mathbf{h}}$, so it is not prime. Because of this phenomenon, the projections will be typically used in combination with a weak membership test for the ideal (see Section 4.1 for details) to distinguish between the factors as in (2.6).

Furthermore, the coefficients of the projections $\mu^2(y_1')^2 + \mu^2 y_1^2 - 1$ and $(y_2')^2 + \mu^2 y_1^2 - 1$ generate $\mathbb{C}(\mu^2)$ while the field of definition of the ideal of input-output relations (and, thus, by [47, Theorem 21], the field of ME-identifiable functions) contains μ since $\mu y_1' - y_2' \in I_\Sigma$. We overcome this difficulty by designing Algorithm 4.5 for computing the field of definition for a given set of projections together with a weak membership test.

Remark 4 (Projections vs. characteristic sets). *Perhaps the most popular way of representing differential ideals is via characteristic sets [34, Section 3.3]. One can check that, by [34, Definition 3.17], the set of nonzero projections $\{f_i \mid i = 1, \ldots, n, f_i \neq 0\}$ for a profile $\mathbf{h}$ of a prime differential ideal P form a characteristic set of $I_{\mathbf{h}}$ (2.5) with respect to the following ranking:*

(2.7) $\qquad x_i^{(a)} > x_j^{(b)} \iff (a - h_i > b - h_j) \text{ or } (a - h_i = b - h_j \text{ and } i > j).$

However, as we have shown in Remark 3, $I_{\mathbf{h}}$ may be not equal to P. Nevertheless, if the algebraic ideal

$$\langle f_1, \ldots, f_n \rangle \colon H^\infty$$

has a unique component of the top dimension, then it is prime by [33, Theorem 4.4], so $I_{\mathbf{h}}$ is prime by [34, Theorem 4.13] implying that $I_{\mathbf{h}} = P$. In this case $\{f_i \mid i = 1, \ldots, n, f_i \neq 0\}$ is characteristic set for P with respect to the ranking (2.7). We give and implement an algorithm for checking this (Algorithm 4.4) and it turns out that this property holds for the ideals of input-output relations for the majority of systems we have considered, see Table 3 (column "Char. set?"). The prevalence of input-output projections which are characteristic sets makes the comparison of our algorithm with the methods based on characteristic sets in Table 5 even more meaningful.

Remark 5 (Complexity). *The geometric definition via the projections allows one to expect to obtain useful geometry-based bounds for the projection-based representation and the algorithms presented in this paper. This interesting problem is beyond the scope of the present paper.*

3. Main theoretical results. In this section, we collect the main theoretical results of the paper. For implementation and performance of the corresponding algorithms, see Section 6.

3.1. Computing projections of ODE systems.

Theorem 3.1 (Model $\implies$ Projections). *Consider an ODE system Σ as in (2.1) and the corresponding differential ideal I_Σ (see (2.2)). Then Algorithm 4.2 computes a parametric profile and the corresponding projections for the ideal of input-output relations, that is, $I_\Sigma \cap \mathbb{C}(\boldsymbol{\mu})[\mathbf{y}^{(\infty)}, \mathbf{u}^{(\infty)}]$.*

As we have described in Section 2.2, the field of ME-identifiable functions is equal to the field of definition of $I_\Sigma \cap \mathbb{C}(\boldsymbol{\mu})[\mathbf{y}^{(\infty)}, \mathbf{u}^{(\infty)}]$. The next theorem shows that one can compute generators of this field from the input-output projections.

Theorem 3.2 (Projections $\implies$ Field of definition). *Consider an ODE system Σ as in (2.1) and the corresponding differential ideal I_Σ (see (2.2)). Let $\mathbf{h}$ be a parametric profile for $I_\Sigma \cap \mathbb{C}(\boldsymbol{\mu})[\mathbf{y}^{(\infty)}, \mathbf{u}^{(\infty)}]$ with projections $(f_1, \ldots, f_n)$ such that each nonzero f_i has at least one coefficient that is one.*

Then Algorithm 4.5 computes a differential polynomial $f \in I_\Sigma$ with at least one coefficient (as a polynomial in $\{\mathbf{y}^{(\infty)}, \mathbf{u}^{(\infty)}\}$) being equal to one, so that the field of ME-identifiable functions for Σ is generated by the coefficients of $f, f_1, \ldots, f_n$.

3.2. Assessing identifiability. Theorems 3.1 and 3.2 provide a way to compute a set of generators of the field of multi-experiment identifiable functions for a model Σ. However, these generators may be complicated and hard to interpret, which is what typically happens in practice. For any given function of parameters, the following theorem allows to find with high probability of correctness whether the function belongs to the field of multi-experiment identifiable functions.

Theorem 3.3 (Field of definition $\implies$ ME-identifiability). *Let $f, f_1, \ldots, f_N, g \in \mathbb{C}[\mathbf{X}]$, where $\mathbf{X} = (X_1, \ldots, X_n)$. Let $0 < p < 1$ be a real number. We define*

$$d := \max(\deg g + 1, \deg f, \deg f_1, \ldots, \deg f_N) \quad and \quad M := \frac{6d^{n+3}}{1-p}$$

Let $\mathbf{a} = (a_1, \ldots, a_n)$ be integers sampled uniformly and independently from $[0, M]$. Consider the field $F := \mathbb{C}\left(\frac{f_1(\mathbf{X})}{g(\mathbf{X})}, \ldots, \frac{f_N(\mathbf{X})}{g(\mathbf{X})}\right)$ and ideal

$$I := \langle f_1(\mathbf{X})g(\mathbf{a}) - f_1(\mathbf{a})g(\mathbf{X}), \ldots, f_N(\mathbf{X})g(\mathbf{a}) - f_N(\mathbf{a})g(\mathbf{X})\rangle \colon g(\mathbf{X})^\infty \subset \mathbb{C}[\mathbf{X}]$$

Then
1. *if $f(\mathbf{X})g(\mathbf{a}) - f(\mathbf{a})g(\mathbf{X}) \in I$, then $\frac{f(\mathbf{X})}{g(\mathbf{X})} \in F$ with probability at least p;*
2. *if $f(\mathbf{X})g(\mathbf{a}) - f(\mathbf{a})g(\mathbf{X}) \notin I$, then $\frac{f(\mathbf{X})}{g(\mathbf{X})} \notin F$ with probability at least p.*

Remark 6.
- *We will apply Theorem 3.3 to normalized coefficients of input-output equations (see Algorithm 5.3). Since there will be usually only a few resulting input-output equations, there will be typically few distinct denominators. Thus, the common denominator g will not have excessively high degree.*
- *In the identifiability context, N is usually a large number, so it is important that it does not appear in the formula for M, the size of the sampling range.*

Remark 7. *Note that it may be beneficial, instead of running the test from Theorem 3.3 once, run it $\log_2 \frac{1}{1-\varepsilon}$ times for probability $p = \frac{1}{2}$ to achieve overall probability of success ε.*

Finally, by combining Theorems 3.1, 3.2, and 3.3 with Algorithm 5.2 for checking the Wronskian condition of the equivalence of SE- and ME-identifiability, we obtain:

Theorem 3.4 (Projection-based identifiability testing). *For any given ODE system Σ as in (2.1), function of parameters $h(\boldsymbol{\mu}) \in \mathbb{C}(\boldsymbol{\mu})$, and a real number $0 < p < 1$, Algorithm 5.3 returns the following information:*

- *whether $h(\boldsymbol{\mu})$ is globally, locally, or not ME-identifiable (correct with probability at least p);*
- *whether the algorithm was able to conclude that SE- and ME-identifiability coincide for Σ.*

4. Projection-based (PB) elimination. In this section, we will describe the algorithms from Theorem 3.2 and 3.1 and prove the theorems. The results and algorithms will be formulated in a more general context of prime differential ideals with membership oracles.

4.1. Membership oracle for an ODE model.

Definition 4.1 (Weak membership oracle). *Let $P \subset K[\mathbf{x}^{(\infty)}]$ be a prime differential ideal over a differential field K in variables $\mathbf{x} = (x_1, \ldots, x_n)$. Let $\mathcal{O}$ be an algorithm taking as input a list of differential polynomials from $K[\mathbf{x}^{(\infty)}]$ and returning, if it terminates, an element of $K[\mathbf{x}^{(\infty)}]$.*

Then $\mathcal{O}$ is called a weak membership oracle for P if, for every $f_1, \ldots, f_s \in K[\mathbf{x}^{(\infty)}]$ such that exactly one of $f_1, \ldots, f_s$ belongs to P (denote it by f), we have $\mathcal{O}(f_1, \ldots, f_s) = f$.

Let Σ be an ODE model as in (2.1). In this section, we describe an efficient sampling-based weak membership oracle for the corresponding ideal I_Σ (see (2.2)). Note that the representation (2.2) for I_Σ is already a characteristic set of this ideal [32, Lemma 3.2], so one can test membership in I_Σ by reducing with respect to this characteristic set. However, in practice, this may be very inefficient.

Algorithm 4.1 Weak membership oracle for I_Σ

Input A system Σ as in (2.1) and polynomials $f_1, \ldots, f_s \in \mathbb{C}(\boldsymbol{\mu})[\mathbf{y}^{(\infty)}, \mathbf{u}^{(\infty)}]$ such that exactly one of $f_1, \ldots, f_s$ belongs to I_Σ.

Output Index i such that $f_i \in I_\Sigma$.

(Step 1) Set $D := 1$, let h be the maximum of the orders of $f_1, \ldots, f_s$;

(Step 2) Repeat

(a)   Choose random values for parameters $\boldsymbol{\mu}$, initial conditions $\mathbf{x}$, and $\mathbf{u}^{(\leqslant h)}$ uniformly from $[1, D]$

(b)   If Q, the common denominator of $f_1, \ldots, f_s$, vanishes at the chosen values, go to the next iteration of the loop;

(c)   Compute the truncated power series solution of Σ up to order h using the algorithm from [5] for the values chosen in **(Step 2)**a;

(d)   For each $1 \leqslant i \leqslant s$, plug the computed solution to f_i and evaluate at $t = 0$;

(e)   If exactly one of the f_i's vanished after evaluation, return its index;

(f)   Set $D := 2D$.

Lemma 4.2. *Algorithm 4.1 is correct and terminates with probability one.*

Proof. Assume that Algorithm 4.1 terminates and returns index i_0. This implies, that the algorithm has computed a truncated power series solution of Σ on which all the f_j's with $j \neq i_0$ do not vanish. Therefore, $f_j \notin I_\Sigma$ for $j \neq i_0$, so $f_{i_0} \in I_\Sigma$.

To analyze the probability of termination, let i_0 be the correct output. Let $F = \prod_{j \neq i_0} f_j$. As has been shown in [32, Proof of Lemma 3.2] there exists a positive integer N and nonzero $\widetilde{F} \in \mathbb{C}[\boldsymbol{\mu}, \mathbf{x}, \mathbf{u}^{(\leqslant h)}]$ such that $Q^N F - \widetilde{F} \in I_\Sigma$. We denote the total degree of $\widetilde{F}Q$ by d.

Let $\boldsymbol{\mu}, \mathbf{x}^*, (\mathbf{u}^{(\leqslant h)})^*$ be the values chosen at **(Step 2)**a and $(\mathbf{x}(t), \mathbf{y}(t), \mathbf{u}(t))$ be the truncated power series solution computed at **(Step 2)**c. Then, if $\widetilde{F}(\boldsymbol{\mu}^*, \mathbf{x}(t), \mathbf{y}(t), \mathbf{u}(t))$ does not vanish at $t = 0$, then the same is true for $F(\boldsymbol{\mu}^*, \mathbf{x}(t), \mathbf{y}(t), \mathbf{u}(t))$. Since $\widetilde{F} \in \mathbb{C}[\boldsymbol{\mu}, \mathbf{x}, \mathbf{u}^{(\leqslant h)}]$, the value of $\widetilde{F}(\boldsymbol{\mu}^*, \mathbf{x}(t), \mathbf{y}(t), \mathbf{u}(t))$ at $t = 0$ is equal to $\widetilde{F}(\boldsymbol{\mu}^*, \mathbf{x}^*, (\mathbf{u}^{(\leqslant h)})^*)$. The probability of this value or Q being zero does not exceed $\frac{d}{D}$ due to the Demillo-Lipton-Schwartz-Zippel lemma [66, Proposition 98]. Therefore, for each iteration of the main loop, the algorithm will not terminate on this iteration with the probability at most $\frac{d}{D}$. Therefore, the probability of termination is at least

$$1 - \frac{d}{1} \cdot \frac{d}{2} \cdot \frac{d}{4} \cdot \frac{d}{8} \cdot \ldots = 1. \qquad \blacksquare$$

4.2. "Socoban" algorithm: changing the profile.

Lemma 2.11 suggests that computing a parametric profile and the corresponding projections for the ideal of input-output relations of system Σ as in (2.1) can be carried out by changing the original profile (2.3) to a profile of the form

$$(\underbrace{\leqslant 1, \leqslant 1, \ldots, \leqslant 1}_{n \text{ times}}, \underbrace{h_1, h_2, \ldots, h_m}_{m \text{ times}}, \underbrace{\infty, \ldots, \infty}_{s \text{ times}}),$$

where none of the projections corresponding to the outputs contains a state variable. This is done by Algorithm 4.2 below which in turn uses Algorithm 4.3 for performing an elementary "carrying" step.

Algorithm 4.2 Projecting an ODE system

Input   A system Σ as in (2.1).
Output   A parametric profile and the corresponding projections for $I_\Sigma \cap \mathbb{C}(\boldsymbol{\mu})[\mathbf{y}^{(\infty)}, \mathbf{u}^{(\infty)}]$.
(Step 1) Let $\mathcal{O}$ be the weak membership oracle for I_Σ given by Algorithm 4.1;
(Step 2) Consider the following profile $\mathbf{h}$ and the corresponding projections $\mathbf{f}$ for I_Σ (see Remark 2)

$$\mathbf{h} := (\underbrace{1, 1, \ldots, 1}_{n \text{ times}}, \underbrace{0, 0, \ldots, 0}_{m \text{ times}}, \underbrace{\infty, \ldots, \infty}_{s \text{ times}}),$$
$$\mathbf{p} := (Qx_1' - F_1, \ldots, Qx_n' - F_n, Qy_1 - G_1, \ldots, Qy_m - G_m, 0, \ldots, 0).$$

(Step 3) While there exist $n + 1 \leqslant i \leqslant n + m$ and $1 \leqslant j \leqslant n$ such that x_j appears in p_i:
(a)   Apply Algorithm 4.3 with $P = I_\Sigma$, $\mathbf{h} = \mathbf{h}$, $\mathbf{f} = \mathbf{p}$, $\mathcal{O} = \mathcal{O}$, $i = i$, and $j = j$.
(b)   Set $\mathbf{h} := \mathbf{h} - \mathbf{e_j} + \mathbf{e_i}$ and $\mathbf{p}$ to be the projections returned by Algorithm 4.3.
(Step 4) Return $(h_{n+1}, \ldots, h_{n+m+s})$ and $(p_{n+1}, \ldots, p_{n+n+s})$.

Remark 8. *It often happens that there are several possible pairs (i, j) at **(Step 3)** of the algorithm. The choice of the one to process may have significant impact on the performance of the algorithm. The choice made in our implementation is to have $\deg_{x_j} p_i$ as small as*

possible, similar quantities (e.g., the degree of the prolongation of p_i) are used to break the ties. Systematic study of choice heuristics is an interesting question for future research.

Proposition 4.3 (Proof of Theorem 3.1). *Algorithm 4.2 is correct and terminates with probability one.*

Proof. Correctness follows from Lemma 2.11. Now we will prove termination. After each iteration of the while loop at **(Step 3)**, the sum $h_{n+1} + \ldots + h_{n+m}$ increases by one. Since it is bounded by n, there will be at most n iterations. Moreover, each iteration will terminate with probability one due to Proposition 4.5. ∎

Algorithm 4.3 "Socoban" algorithm

Input
- A prime differential ideal $P \subset K[\mathbf{x}^{(\infty)}]$ defined by
    - a parametric profile $\mathbf{h}$ and corresponding projections $\mathbf{f} = (f_1, \ldots, f_n)$;
    - and a weak membership oracle $\mathcal{O}$.
- $1 \leqslant i \neq j \leqslant n$ such that f_i involves x_j and $h_j = 1$.

Output the projections of P corresponding to $\mathbf{h} - \mathbf{e_j} + \mathbf{e_i}$.

(Step 1) Set $f := f_i'$.

(Step 2) For $\ell \in \{1, \ldots, n\} \setminus \{i\}$ such that $h_\ell < \infty$ do

(a) Compute $g_1, \ldots, g_r$, the squarefree factorization over K of $\mathrm{Res}_{x_\ell^{(h_\ell)}}(f, f_\ell)$;

(b) Set $f := \mathcal{O}(g_1, \ldots, g_r)$.

(Step 3) Set $\tilde{f}_j := f_i$ and $f_i := f$.

(Step 4) For $\ell \in \{1, \ldots, n\} \setminus \{j\}$ such that $h_\ell < \infty$ do

(a) Compute $g_1, \ldots, g_r$, the squarefree factorization over K of $\mathrm{Res}_{x_j}(\tilde{f}_j, f_\ell)$;

(b) Set $\tilde{f}_\ell := \mathcal{O}(g_1, \ldots, g_r)$.

(Step 5) Return $(\tilde{f}_1, \ldots, \tilde{f}_n)$.

Lemma 4.4. *Let $\mathbb{A}$ be an arbitrary integral domain,*

$$r \in \mathbb{A}[z] \setminus \mathbb{A}, R \in \mathbb{A}[w_1, w_2, \ldots, w_t, z, v],$$

$$s_1 \in \mathbb{A}[w_1] \setminus \mathbb{A}, s_2 \in \mathbb{A}[w_2] \setminus \mathbb{A}, \ldots, s_t \in \mathbb{A}[w_t] \setminus \mathbb{A}$$

be non-zero polynomials such that r is irreducible. $r \nmid \mathrm{lc}_v(R)$ (where $\mathrm{lc}_v(R)$ denotes the leading coefficient of R with respect to the variable v), and that $\mathrm{lc}_v(R)$ does not involve $w_1, \ldots, w_t$.

Then $\mathrm{Res}_z(r, \mathrm{Res}_{w_t}(s_t, \ldots \mathrm{Res}_{w_2}(s_2, \mathrm{Res}_{w_1}(R, s_1)) \ldots)) \neq 0$.

Proof. By replacing $\mathbb{A}$ with its algebraic closure, we will further assume that $\mathbb{A}$ is algebraically closed. Then each of $r, s_1, \ldots, s_t$ factors into linear factors over $\mathbb{A}$. We will denote the roots of r and s_i for $1 \leqslant i \leqslant t$ in $\mathbb{A}$ by $\alpha_1, \ldots, \alpha_{\deg r}$ and $\beta_{i,1}, \ldots, \beta_{i,\deg s_i}$, respectively. Applying iteratively the formula for the resultant in terms of roots [15, Chapter 3, §1, Ex. 10], we show that the resultant of interest is equal to

$$C \prod R(\beta_{1,i_1}, \beta_{2,i_2}, \ldots, \beta_{t,i_j}, \alpha_j, v),$$

where C is a power product of the leading coefficients of $r, s_1, \ldots, s_t$. Since none of $w_1, \ldots, w_t$ appears in $\mathrm{lc}_v(R)$ and $\mathrm{lc}_v(R)$ does not vanish at any of $\alpha_1, \ldots, \alpha_{\deg r}$, each $R(\beta_{1,i_i}, \beta_{2,i_2}, \ldots, \beta_{t,i_j}, \alpha_j, v)$ is a nonzero polynomial in v. Thus, the resultant is nonzero. ∎

Proposition 4.5. *Algorithm 4.3 terminates with probability one and is correct.*

Proof. By Lemma 4.2, each of **(Step 2)**b and **(Step 4)**b terminates with probability 1. Therefore algorithm 4.3 terminates with probability one.

Each of the computed polynomials $\tilde{f}_1, \ldots, \tilde{f}_n$ is obtained as an irreducible factor of a polynomial, so is irreducible. Moreover, $\tilde{f}_1, \ldots, \tilde{f}_n$ are obtained from elements of P by a chain of resultant computations, so they also belong to P. Therefore, in order to prove the correctness, it remains to prove that the resultants computed on steps **(Step 2)** and **(Step 4)** are non-zero. We apply Lemma 4.4 with

$$(w_1, \ldots, w_t) = (x_\ell^{(h_\ell)} : \ell \in \{1, \ldots, n\} \setminus \{i\}, \ h_\ell < \infty), \qquad z = x_j, \qquad v = x_i^{(h_i+1)},$$
$$(s_1, \ldots, s_t) = (f_\ell : \ell \in \{1, \ldots, n\} \setminus \{i\}, \ h_\ell < \infty), \quad r = f_i, \quad R = f_i', \quad \mathbb{A} = K[\mathbf{x}^{(<\mathbf{h}-\mathbf{e_j}+\mathbf{e_i})}],$$

where all values are taken at the beginning of the algorithm.

Let us verify that the conditions of Lemma 4.4 are indeed satisfied. Since $\mathbf{h}$ is a parametric profile, $\deg_{x_\ell^{(h_\ell)}} f_\ell > 0$, so $s_k \in \mathbb{A}[w_k] \setminus \mathbb{A}, k = 1, \ldots, t$. Since f_i involves x_j, $r \in \mathbb{A}[z] \setminus \mathbb{A}$. $r = f_i$ is irreducible by the primality of P. $\deg_{x_i^{(h_i)}} f_i > \deg_{x_i^{(h_i)}} \mathrm{lc}_{x_i^{(h_i+1)}}(f_i')$, so $r \nmid \mathrm{lc}_v(R)$. f_i does not involve $x_\ell^{(h_\ell)} : \ell \in \{1, \ldots, n\} \setminus \{i\}$, thus by properties of the Lie derivative, $\mathrm{lc}_v(R) = \mathrm{lc}_{x_i^{(h_i+1)}}(f_i')$ does not involve $(w_1, \ldots, w_t) = (x_\ell^{(h_\ell)} : \ell \in \{1, \ldots, n\} \setminus \{i\})$.

Then the result of the computation performed by **(Step 2)** and the iteration of **(Step 4)** with $\ell = i$ will divide the resultant from Lemma 4.4 and, thus, will be nonzero.

For the remaining iterations (i.e. $\ell \neq i$) of the loop in **(Step 4)**, $\mathrm{Res}_{x_j}(f_i, f_\ell)$ will be nonzero since both f_i and f_ℓ are irreducible and $f_i \neq f_\ell$ (they have different degree in $x_i^{(h_i)}$). ∎

4.3. Checking the uniqueness of the top-dimensional component. Algorithm 4.4 below will be one of the key ingredients for Section 4.4.

Proposition 4.6.
1. *If Algorithm 4.4 returns **True**, then there is a unique prime component $\mathfrak{p} \subset I$ with $\mathfrak{p} \cap \mathbb{Q}[\mathbf{y}] = 0$.*
2. *There exist positive constants C_0, C_1 depending only on $q_1, \ldots, q_m$ and the degree of q such that, if there is a unique prime component $\mathfrak{p} \subset I$ with $\mathfrak{p} \cap \mathbb{Q}[\mathbf{y}] = 0$ and $N > C_0$, then Algorithm 4.4 returns **True** with probability at least $1 - \frac{C_1}{\sqrt[3]{N}}$.*

In the proof of the proposition, we will use the following quantitative version of the Hilbert irreducibility theorem due to Cohen [12] (see also [10]).

Lemma 4.7 (Follows from [12, Theorem 2.1]). *For every polynomial $p \in \mathbb{Q}[x_1, \ldots, x_n, t]$, there exist positive constants C_0, C_1 such that: for every positive integer $N > C_0$:*

$$P(\text{for every } \mathbb{Q}\text{-irreducible factor } q \text{ of } p: q(a_1, \ldots, a_n, t) \text{ is irreducible in } \mathbb{Q}[t]) > 1 - C_1/\sqrt[3]{N},$$

where integers $a_1, \ldots, a_n$ are sampled independently uniformly at random from $[-N, N]$

16

Algorithm 4.4 Checking the uniqueness of the top-dimensional component

---

Input a positive integer N and squarefree polynomials $q_1, \ldots, q_m, q$ such that
  1. $q \in \mathbb{Q}[\mathbf{x}, \mathbf{y}]$, where $\mathbf{x} = (x_1, \ldots, x_n)$ and $\mathbf{y} = (y_1, \ldots, y_m)$;
  2. $q_i \in \mathbb{Q}[\mathbf{x}, y_i]$ and $\deg_{y_i} q_i > 0$ for every $1 \leqslant i \leqslant m$;
  3. $I \cap \mathbb{Q}[\mathbf{x}] = 0$, where $I := \langle q_1, \ldots, q_m, q \rangle$.

Output **True** or **False**, for precise interpretation (parametrized by N), see Proposition 4.6

Part 1: Specializing

(Step 1) For each $1 \leqslant i \leqslant m$, compute $r_i \in \mathbb{Q}[\mathbf{x}]$, the discriminant of q_i w.r.t. y_i. Define $R = \prod_{i=1}^{m} r_i$.

(Step 2) Sample integers $\mathbf{a} = (a_1, \ldots, a_n)$ independently uniformly at random from $[-N, N]$.

(Step 3) If $R(\mathbf{a}) = 0$, return **False**.

(Step 4) Set $\tilde{q}_i(y_i) := q_i(a_1, \ldots, a_n, y_i)$ for every $1 \leqslant i \leqslant m$ and $\tilde{q}(\mathbf{y}) := q(a_1, \ldots, a_n, \mathbf{y})$.

Part 2: Checking the primality of the specialization

(Step 5) Sample integers $\mathbf{b} = (b_1, \ldots, b_m)$ independently uniformly at random from $[-N, N]$.

(Step 6) Use Gröbner bases to compute the minimal polynomial $p(z)$ of the linear form $b_1 y_1 + \ldots + b_m y_m$ modulo zero-dimensional ideal $\widetilde{I} := \langle \tilde{q}_1, \ldots, \tilde{q}_m, \tilde{q} \rangle \subset \mathbb{Q}[\mathbf{y}]$.

(Step 7) If p is irreducible over $\mathbb{Q}$ and $\deg p = \dim_{\mathbb{Q}} \mathbb{Q}[\mathbf{y}]/\widetilde{I}$, return **True**. Otherwise, **False**.

Proof of Proposition 4.6. Consider zero-dimensional ideals $J := \langle q_1, \ldots, q_m, q \rangle \subset \mathbb{Q}(\mathbf{x})[\mathbf{y}]$ and $J_0 := \langle q_1, \ldots, q_m \rangle \subset \mathbb{Q}(\mathbf{x})[\mathbf{y}]$. The prime components of J are in bijection with the prime components $\mathfrak{p} \subset I$ with $\mathfrak{p} \cap k[\mathbf{x}] = 0$, so there is a unique such component iff J is prime.

Consider a linear form $\ell = \lambda_1 y_1 + \ldots + \lambda_m y_m$ with $\boldsymbol{\lambda} := (\lambda_1, \ldots, \lambda_m)$ being new indeterminates. Then ℓ separates the roots of J_0 in $\overline{k(\mathbf{x})}$. Let $P, P_0 \in \mathbb{Q}(\boldsymbol{\lambda}, \mathbf{x})[t]$ be the monic minimal polynomials for ℓ modulo J and J_0, respectively. We have $P \mid P_0$. Since $q_1, \ldots, q_m$ form a Gröbner basis, P_0 can be represented as a characteristic polynomial of a matrix with the entries in $\mathbb{Q}[\boldsymbol{\lambda}, \mathbf{x}, 1/R(\mathbf{x})]$, so its coefficients belong to this ring. Then the same holds for P by the Gauss lemma. [18, Theorem 3] implies that J is prime iff P is irreducible.

Let $\mathbf{a} = (a_1, \ldots, a_n)$ be the integers sampled at **(Step 2)**. $R(\mathbf{a}) \neq 0$ implies that the total multiplicity of solutions of $\widetilde{I}_0 := \langle \tilde{q}_1, \ldots, \tilde{q}_m \rangle$ in $\overline{\mathbb{Q}}$ is the same as the number of solutions of J_0. Thus, the minimal polynomial of ℓ modulo $\widetilde{I}_0$ is equal to $P_0|_{\mathbf{x}=\mathbf{a}}$. Then the minimal polynomial of ℓ modulo $\widetilde{I}$ is divisible by $P|_{\mathbf{x}=\mathbf{a}}$.

Assume that the algorithm returned **True**. By [18, Theorem 3], we have that $\widetilde{I}$ is prime. Then $P|_{\mathbf{x}=\mathbf{a}}$ is irreducible, so P is irreducible as well, and thus J is prime. This proves the first part of the proposition.

Now we will prove the second part. Assume that J is prime. Let $\mathbf{b} = (b_1, \ldots, b_m)$ be the integers sampled at **(Step 5)**. There is a polynomial, say $T_1(\boldsymbol{\lambda})$, such that $T_1(\mathbf{b}) \neq 0$ implies that $z = b_1 y_1 + \ldots + b_m y_m$ separates the roots of J_0. We will now assume that $T_1(\mathbf{b}) \neq 0$. The multiplication map by q in the quotient ring $\mathbb{Q}(\mathbf{x})[\mathbf{y}]/J_0$ can be written as a matrix with the entries in $\mathbb{Q}[\mathbf{x}, 1/R(\mathbf{x})]$. The dimension of its kernel is equal to the number of roots of J. Let $T_2(\mathbf{x})$ be the numerator of any maximal minor of this matrix. If $T_2(\mathbf{a}) \neq 0$, the number of roots of $\widetilde{I}$ is the same as of J. Then the minimal polynomial of ℓ modulo $\widetilde{I}$ is or the same degree as $P_{\mathbf{x}=\mathbf{a}}$, so it is equal to $P_{\mathbf{x}=\mathbf{a}}$. We will now assume that $T_2(\mathbf{a}) \neq 0$. Then the polynomial $p(z)$ computed at the **(Step 6)** is equal to $P|_{\mathbf{x}=\mathbf{a}, \boldsymbol{\lambda}=\mathbf{b}}(z)$. Therefore, if $R(\mathbf{a})T_1(\mathbf{b})T_2(\mathbf{a}) \neq 0$

and the substitution $\mathbf{x} = \mathbf{a}, \boldsymbol{\lambda} = \mathbf{b}$ preserves the irreducibility of P, the algorithm will return **True**. Note that R and T_1, and P_0 depend only on $q_1, \ldots, q_m$, and not on q while the degree of T_2 can be bounded by a function of $q_1, \ldots, q_m$ and $\deg q$. We apply Lemma 4.7 to $P_0(z)$ (with $t = z$), let c_0 and c_1 be the corresponding constants. Then, by Demillo-Lipton-Schwartz-Zippel lemma [66, Proposition 98], the probability of $R(\mathbf{a})T_1(\mathbf{b})T_2(\mathbf{a}) \neq 0$ and $P|_{\mathbf{x}=\mathbf{a},\boldsymbol{\lambda}=\mathbf{b}}(z)$ being irreducible is at least

$$1 - \frac{\deg T_1 + \deg T_2 + \deg R}{N} - \frac{c_1}{\sqrt[3]{N}}, \quad \text{for } N > c_0.$$

Setting $C_0 = c_0$ and $C_1 = c_1 + \deg T_1 + \deg T_2 + \deg R$ finishes the proof. ∎

4.4. Computing the field of definition. Before presenting and justifying Algorithm 4.5 for computing generators of a field of definition, we prove several useful properties of the field of definition.

Lemma 4.8. *Let I be an ideal of the polynomial ring $K[\mathbf{x}]$ in variables $\mathbf{x} = (x_1, \ldots, x_n)$ over a field K of characteristic zero. Let F be the field of definition of I. Then*

1. *If I is a principal ideal, and its generator f has at least one coefficient equal to one, then F is generated by the coefficients of f.*
2. *For any $\mathbb{Q}$-linear forms $y_1, \ldots, y_r$ in $x_1, \ldots, x_n$, the field of definition of $I \cap K[y_1, \ldots, y_r]$ is a subfield of F.*
3. *Let I be radical and $1 \leqslant s \leqslant n$. Consider J, the union of prime components $\mathfrak{p}$ of I such that $\mathfrak{p} \cap \overline{K}[x_1, \ldots, x_s] = 0$. Then the field of definition of J is a subfield of F.*

Proof. Since I is generated by f, F is contained in the field generated by the coefficients of f. Conversely, $I \cap F[\mathbf{x}]$ must contain at least one polynomial of degree $\deg f$, and this polynomial will be proportional to f. Since normalizing the coefficients does not increase the field generated by the coefficients, the coefficients of f belong to F.

2. Since $I \cap F[y_1, \ldots, y_r] \subset I \cap F[\mathbf{x}]$ and $I \cap F[\mathbf{x}]$ spans I as a K-vector space, $I \cap F[y_1, \ldots, y_r]$ must span $I \cap K[y_1, \ldots, y_r]$ as a K-vector space. Therefore, F contains the field of definition of $I \cap K[y_1, \ldots, y_r]$.
3. It is sufficient to show that any automorphism $\alpha \colon \overline{K} \to \overline{K}$ over F fixes J set-wise. Since α permutes the primes of I and $\mathfrak{p} \cap \overline{K}[x_1, \ldots, x_s] \iff \alpha(\mathfrak{p}) \cap \overline{K}[x_1, \ldots, x_s]$, α fixes J as well. ∎

Lemma 4.9. *Let $P \subset K[\mathbf{x}^{(\infty)}]$ be a prime differential ideal with a parametric profile $\mathbf{h}$ and the corresponding projections $(f_1, \ldots, f_n)$. We define (cf. (4.1))*

$$h_i^\circ := \begin{cases} h_i, & \text{if } f_i \neq 0, \\ \max_{1 \leqslant j \leqslant n} \operatorname{ord}_{x_i} f_j, & \text{if } f_i = 0. \end{cases} \qquad and \qquad \mathbf{h}^\circ := (h_1^\circ, \ldots, h_n^\circ).$$

Then $P \cap K[\mathbf{x}^{(\leqslant \mathbf{h})}]$ is generated by $P \cap K[\mathbf{x}^{(\leqslant \mathbf{h}^\circ)}]$.

Proof. We consider the images of $\mathbf{x}^{(\infty)}$ modulo P and denote them by the same symbols. The lemma will follow from the fact that $\mathbf{x}^{(\leqslant \mathbf{h})} \setminus \mathbf{x}^{(\leqslant \mathbf{h}^\circ)}$ are algebraically independent over $\mathbf{x}^{(\leqslant \mathbf{h}^\circ)}$. To prove this, we assume the contrary. Let $\mathbf{h}^* := \mathbf{h}^\circ - \sum_{f_i \neq 0} \mathbf{e_i}$. Then $\mathbf{x}^{(\leqslant \mathbf{h}^*)}$ form a

18

transcendence basis of $\mathbf{x}^{(\leqslant \mathbf{h}^\circ)}$ over K modulo P. Therefore, there is an algebraic dependence between

$$\mathbf{x}^{(\leqslant \mathbf{h}^*)} \cup (\mathbf{x}^{(\leqslant \mathbf{h})} \setminus \mathbf{x}^{(\leqslant \mathbf{h}^\circ)}) = \mathbf{x}^{(<\mathbf{h})},$$

but this contradicts the fact that $\mathbf{h}$ is a parametric profile. ∎

Algorithm 4.5 Computing the differential field of definition

Input A prime differential ideal $P \subset \mathbb{Q}(\mathbf{p})[\mathbf{x}^{(\infty)}]$, where $\mathbf{p} = (p_1, \ldots, p_s)$ are transcendental constants, defined by
- a parametric profile $\mathbf{h}$ and the corresponding projections $(f_1, \ldots, f_n)$;
- and a weak membership oracle $\mathcal{O}$.

Output a differential polynomial $f \in P$ such that the coefficients of $f, f_1, \ldots, f_n$ (after normalizing so that every nonzero polynomial has at least one coefficient 1) generate the field of definition of P.

(Step 1) Introduce a new variable z, set $N := 1$.

(Step 2) Relabel $x_1, \ldots, x_n$ so that there exists $1 \leqslant n_0 \leqslant n$ such that $h_1, \ldots, h_{n_0} < \infty$ and $h_{n_0+1} = \ldots = h_n = \infty$.

(Step 3) Clear the denominators in $f_1, \ldots, f_{n_0}$ so that $f_1, \ldots, f_{n_0} \in \mathbb{Q}[\mathbf{p}, \mathbf{x}^{(\infty)}]$.

(Step 4) Let ℓ be the product of the leading terms of f_i w.r.t. $x_i^{(h_i)}$ for every $1 \leqslant i \leqslant n_0$.

(Step 5) For every $1 \leqslant i \leqslant n$, define

$$(4.1) \qquad h_i^\circ := \begin{cases} h_i, & \text{if } i \leqslant n_0, \\ \max_{1 \leqslant j \leqslant n_0} \operatorname{ord}_{x_i} f_j, & \text{if } i > n_0. \end{cases} \qquad \text{and} \qquad \mathbf{h}^\circ := (h_1^\circ, \ldots, h_n^\circ).$$

(Step 6) Repeat
- (a) Let $a_1, \ldots, a_{n_0}$ be integers sampled from $[1, N]$ independently, uniformly at random.
- (b) Set $f := z - a_1 x_1^{(h_1)} - \ldots - a_{n_0} x_n^{(h_{n_0})}$.
- (c) For each $j \in \{1, \ldots, n_0\}$ do $f := \operatorname{Res}_{x_\ell^{(h_j)}}(f, f_j)$.
- (d) Let $\tilde{f}$ be the result of applying $z \to a_1 x_1^{(h_1)} + \ldots + a_{n_0} x_n^{(h_{n_0})}$ to f.
- (e) Compute $g_1, \ldots, g_r$, the squarefree factorization of $\tilde{f}$ over $\mathbb{Q}$.
- (f) Set $\tilde{f} = \mathcal{O}(g_1, \ldots, g_r)$.
- (g) Apply Algorithm 4.4 to the ideal $\langle f_1, \ldots, f_{n_0}, \tilde{f} \rangle \subset \mathbb{Q}[\mathbf{p}, \mathbf{x}^{(\leqslant \mathbf{h}^\circ)}]$ with $N = N$ and $\mathbf{y} = (x_1^{(h_1)}, \ldots, x_{n_0}^{(h_{n_0})})$.
- (h) If the algorithm returns **True**, return $\tilde{f}$. Otherwise, set $N := 2N$.

Lemma 4.10. *Let $P \subset K[\mathbf{x}^{(\infty)}]$ be a prime differential ideal with a parameteric profile $\mathbf{h}$, and let $\mathbf{h}^\circ$ be defined in the same way as in Lemma 4.9. Then the differential field of definition of P is generated (as a differential field) by the field of definition of $P \cap K[\mathbf{x}^{(\leqslant \mathbf{h}^\circ)}]$.*

Proof. Let the differential field of definition of P be F and the differential field generated by the field of definition of $P \cap K[\mathbf{x}^{(\leqslant \mathbf{h})}]$ be F_0. Since $P \cap F[\mathbf{x}^{(\leqslant \mathbf{h})}]$ must generate $P \cap K[\mathbf{x}^{(<\mathbf{h})}]$, we have $F_0 \subset F$.

After reordering $x_1, \ldots, x_n$ if necessary, we will further assume that there exists $1 \leqslant n_0 \leqslant n$ such that $h_1, \ldots, h_{n_0} < \infty$ and $h_{n_0+1} = \ldots = h_n = \infty$. Let $f_1, \ldots, f_n$ be the projections corresponding to $\mathbf{h}$ such that each nonzero f_i is normalized to have at least one coefficient that is one. Then the first part of Lemma 4.8 implies that the coefficients of $f_1, \ldots, f_{n_0}$ belong to F_0. We introduce the following ranking [33, Definition 3.1] on $K[\mathbf{x}^{(<\infty)}]$ (as in Remark 4):

$$x_i^{(a)} \prec x_j^{(b)} \iff (a - h_i < b - h_j) \text{ or } (a - h_i = b - h_j \text{ and } i < j)$$

for every $i \neq j$. Then $f_1, \ldots, f_{n_0}$ is a weak differential triangular set [34, Defintion 3.7] w.r.t. this ranking. Let $g \in P$. By performing a partial reduction [34, Algorithm 3.12] of g with respect to $f_1, \ldots, f_{n_0}$, we obtain $\overline{g} \in P \cap K[\mathbf{x}^{(\leqslant \mathbf{h})}]$ and S, a product of powers of the separants of $f_1, \ldots, f_{n_0}$ such that

$$Sg - \sum c_{i,j} f_i^{(j)} = \overline{g}$$

for some $c_{i,j} \in K[\mathbf{x}^{(\infty)}]$. Let $\{e_\lambda\}_{\lambda \in \Lambda}$ be a F_0 basis of K, and, for every $\lambda \in \Lambda$, we denote by g_λ and $\overline{g}_\lambda$ the corresponding coordinates of g and $\overline{g}$, respectively. Since the coefficients of S and all the derivatives of $f_1, \ldots, f_n$ belong to F_0, for every $\lambda \in \Lambda$, we have $Sg_\lambda - \overline{g}_\lambda \in \langle f_1, \ldots, f_n \rangle^{(\infty)}$. Since $\overline{g} \in K[\mathbf{x}^{(\leqslant \infty)}]$, we have $\overline{g}_\lambda \in P$. Since $s \notin P$ and P is prime, we have $g_\lambda \in P$. Therefore, P is generated by $P \cap F_0[\mathbf{x}^{(\infty)}]$, so $F_0 = F$. $\blacksquare$

Lemma 4.11. *Consider an $n + m$-dimensional affine space $\mathbb{A}^{n+m}$ over an algebraically closed field k with coordinates $x_1, \ldots, x_n, y_1, \ldots, y_m$. Let $X \subset \mathbb{A}^{n+m}$ be an n-dimensional variety such that X projects dominantly onto $x_1, \ldots, x_n$. Then there is a nonzero polynomial $P \in K[z_1, \ldots, z_m]$ for some $K \supset k$ such that, for every $\mathbf{a} = (a_1, \ldots, a_m) \in \mathbb{Q}^m$ with $P(\mathbf{a}) \neq 0$, the images of the irreducible components of X with respect to the projection $\pi_{\mathbf{a}}(\mathbf{x}, \mathbf{y}) := (\mathbf{x}, a_1 y_1 + \ldots + a_m y_m)$ are distinct.*

Proof. The fact that the images of the components of X with respect to $\pi_{\mathbf{a}}$ are distinct is equivalent to the fact that the linear form $a_1 y_1 + \ldots + a_m y_m$ separates the solutions in $\overline{k(\mathbf{x})}$ of the zero-dimensional ideal J generated by I in $k(\mathbf{x})[\mathbf{y}]$. This condition can be written as a system of algebraic inequalities, so P can be taken to the be the product of these inequalities. $\blacksquare$

Proposition 4.12 (Proof of Theorem 3.2). *Algorithm 4.5 terminates with probability one and is correct.*

Proof. We will start with proving *the correctness* of the algorithm. Assume that the algorithm has terminated after sampling numbers $a_1^*, \ldots, a_{n_0}^*$ at **(Step 6)**a. Let $z^* := a_1^* x_1^{(h_1)} + \ldots + a_{n_0}^* x_n^{(h_{n_0})}$. Since $x_i^{(h_i)}$ is algebraic over $\mathbb{Q}[\mathbf{p}, \mathbf{x}^{(<\mathbf{h})}]$ modulo P for every $1 \leqslant i \leqslant n_0$, z^* is algebraic modulo P over this ring as well. Since $\mathbf{p}, \mathbf{x}^{(<\mathbf{h})}$ are $\mathbb{Q}$-algebraically independent modulo P, the ideal

$$P \cap \mathbb{Q}[\mathbf{p}, \mathbf{x}^{(<\mathbf{h})}, z^*]$$

is a prime principal ideal. Since $\tilde{f}$ computed at **(Step 6)**f belongs to this ideal and is irreducible, $\tilde{f}$ is the generator of the ideal. Therefore, it generates the prime principal ideal

$$P \cap \mathbb{Q}(\mathbf{p})[\mathbf{x}^{(<\mathbf{h})}, z^*].$$

Let F be the field of definition of P (since $\mathbb{Q}(\mathbf{p})$ is a constant field, this is the same as the differential field of definition) and F_0 be the field generated by the coefficients of $f_1, \ldots, f_{n_0}, \tilde{f}$ after normalizing at least one of the coefficients to be one. Lemma 4.10 implies that F also equals to the field of definition of $P \cap \mathbb{Q}(\mathbf{p})[\mathbf{x}^{(\leqslant \mathbf{h})}]$ which equals to the field of definition of $P_0 := P \cap \mathbb{Q}(\mathbf{p})[\mathbf{x}^{(\leqslant \mathbf{h}^\circ)}]$ due to Lemma 4.9. Combining parts 1 and 2 of Lemma 4.8, we deduce that $F_0 \subset F$. On the other hand, since the ideal $\langle f_1, \ldots, f_{n_0}, \tilde{f} \rangle \subset \mathbb{Q}[\mathbf{p}, \mathbf{x}^{(\leqslant \mathbf{h}^\circ)}]$ has a unique component of codimension n_0, this component is P_0, so $F \subset F_0$ due to the part 3 of Lemma 4.8. *Thus, the correctness of the algorithm is proved.*

Now we will prove that *the algorithm terminates with probability one.* The polynomial f computed at **(Step 6)**c will be always nonzero due to Lemma 4.4. Let d be the maximum of the degrees of $f_1, \ldots, f_{n_0}$. Then the degree of f will not exceed $2^{n_0} d^{n_0}$, so the same is true for $\tilde{f}$. Therefore, Proposition 4.6 implies that there exist constants C_0 and C_1 such that, if the ideal $\langle f_1, \ldots, f_{n_0}, \tilde{f} \rangle$ of $\mathbb{Q}[\mathbf{p}, \mathbf{x}^{(\leqslant \mathbf{h}^\circ)}]$ has a unique component projecting dominantly on $\{\mathbf{p}, \mathbf{x}^{(\leqslant \mathbf{h}^\circ)}\} \setminus \{x^{(h_1)}, \ldots, x^{(h_{n_0})}\}$ and $N > C_0$, then the probability that the algorithm will return the result at **(Step 6)**h is at least $1 - \frac{C_1}{\sqrt[3]{N}}$.

We apply Lemma 4.11 to the variety X consisting of all the components of $\langle f_1, \ldots, f_{n_0} \rangle \subset \mathbb{Q}[\mathbf{p}, \mathbf{x}^{(\leqslant \mathbf{h}^\circ)}]$ projecting dominantly on $\{\mathbf{p}, \mathbf{x}^{(\leqslant \mathbf{h}^\circ)}\} \setminus \{x^{(h_1)}, \ldots, x^{(h_{n_0})}\}$ with

$$\mathbf{x} = \{\mathbf{p}, \mathbf{x}^{(\leqslant \mathbf{h}^\circ)}\} \setminus \{x^{(h_1)}, \ldots, x^{(h_{n_0})}\} \quad \text{and} \quad \mathbf{y} = \{x^{(h_1)}, \ldots, x^{(h_{n_0})}\},$$

and denote the polynomial provided by the lemma by Q. We denote the degree of Q by d_0. Consider an iteration of the "repeat" loop such that $N > C_0$. We denote the integers sampled at **(Step 6)**a by $a_1^*, \ldots, a_{n_0}^*$. Assume that $Q(\mathbf{a}^*) \neq 0$. Since $\tilde{f}$ is the defining polynomial of the projection $\pi_{\mathbf{a}^*}$ (in the notation of Lemma 4.11) of the variety defined by $P \cap \mathbb{Q}[\mathbf{p}, \mathbf{x}^{(\leqslant \mathbf{h}^\circ)}]$, and this variety is one of the dominantly projected components of X, $Q(\mathbf{a}^*) \neq 0$ implies that $\tilde{f}$ does not vanish on any other dominantly projecting component of X. Therefore, the ideal $\langle f_1, \ldots, f_{n_0}, \tilde{f} \rangle$ has a unique such component, so the algorithm will return with the probability at least $1 - \frac{C_1}{\sqrt[3]{N}}$.

The Demillo-Lipton-Schwartz-Zippel lemma [66, Proposition 98] implies that $Q(\mathbf{a}^*) = 0$ with probability at most $\frac{d_0}{N}$. Therefore, for each iteration of the repeat loop with $N > C_0$ the probability of this iteration not being the last one is at most

$$1 - \left(1 - \frac{d_0}{N}\right)\left(1 - \frac{C_1}{\sqrt[3]{N}}\right) = \frac{d_0}{N} + \frac{C_1}{\sqrt[3]{N}} - \frac{d_0 C_1}{N^{4/3}}.$$

This value will eventually become less that $\frac{1}{2}$, so the probability of the algorithm not terminating will not exceed $\frac{1}{2} \cdot \frac{1}{2} \cdot \ldots = 0$. ∎

4.5. Factoring resultants before computing.

The algorithms described in Sections 4.1–4.4 give a complete procedure for computing a representation of the projection of an ODE system and the generators of the field of definition of this projection. These algorithms rely heavily on the resultant computation (e.g., **(Step 4)**a in Algorithm 4.3). It is well-known that resultants may give extraneous factors (especially the repeated ones [9]), and we remove these factors using the weak membership test (see Definition 4.1) for the ideal of interest. In this section we would like to present a method allowing to remove these factors *before* computing

the resultant. Although the method applies only to special cases, it turned out to be very powerful in the differential elimination context (see Table 1).

Example 5 (Simple motivating example). *Consider polynomials $f = ax+bc$ and $g = bx+ac$. The resultant $\mathrm{Res}_x(f, g)$ can be computed using the Sylvester matrix as follows:*

$$\begin{vmatrix} a & bc \\ b & ac \end{vmatrix} = (a^2 - b^2)c = c(a - b)(a + b).$$

Assume that the weak membership test says that $a+b$ is the polynomial in the ideal of interest while $a-b$ and c are extraneous factors. Note that the factor c is the gcd of the last column of the matrix, so it can be factored out before the determinant computation. Assume also that we know in advance that there will be the factor $a-b$. If $a-b=0$, then the common root of f and g will be $x = -c$. Let us make a change of variables shifting this root to be zero, that is, $x \to x - c$:

$$f = ax + bc \to ax + bc - ac, \quad g = bx + ac \to bx + ac - bc.$$

Since such a linear shift does not change the resultant, we can use the Sylvester matrix of the new polynomials for the resultant computation. This will be

$$\begin{vmatrix} a & (b-a)c \\ b & (a-b)c \end{vmatrix} = (a-b)c \begin{vmatrix} a & -1 \\ b & 1 \end{vmatrix},$$

so the extraneous factors are removed before the determinant computation.

For factoring out c in Example 5, we used a simple *observation*: if we compute a resultant of two polynomials f and g using a matrix representation, then, before computing the determinant, we can factor out gcds of rows and columns. Not all extraneous factors of the resultant can be eliminated this way (e.g., $a-b$ in Example 5), and we propose a method to force some of them to appear as such gcd by making a variables change before constructing the matrix. We formalize the transformation from Example 5 as a lemma:

Lemma 4.13. *Let k be a field of characteristic zero. Let $f, g \in k[\mathbf{a}, x]$ and $A \in k[\mathbf{a}]$ be such that $A(\mathbf{a}) \mid \mathrm{Res}_x(f, g)$. Assume that there exist $B, C \in k[\mathbf{a}]$ such that $\gcd(A, C) = 1$ and the numerators of $f(\mathbf{a}, B/C)$ and $g(\mathbf{a}, B/C)$ are divisible by A (that is, B/C is a common root of f and g modulo A). Then each entry of the last column of the Sylvester matrix for*

$$C^{\deg_x f} f(\mathbf{a}, x - B/C) \quad \text{and} \quad C^{\deg_x g} g(\mathbf{a}, x - B/C)$$

is divisible by $A(\mathbf{a})$.

Proof. The last column contains only two nonzero entries: the constant terms of the considered polynomials. The constant terms of the new polynomials are divisible by the denominators of $f(\mathbf{a}, B/C)$ and $g(\mathbf{a}, B/C)$, respectively, which are in turn divisible by A. ∎

The way the situation described in Lemma 4.13 occurs in our computation is described in the following proposition

Proposition 4.14. *Consider $f, g, h \in k[\mathbf{a}, x, y]$ such that f is of the form $A(\mathbf{a})xy + B(\mathbf{a})x + C(\mathbf{a})y + D(\mathbf{x})$, and $\gcd(F, A) = 1$, where $F := A(\mathbf{a})D(\mathbf{a}) - B(\mathbf{a})C(\mathbf{a})$. Then*

1. $F \mid \operatorname{Res}_y(\operatorname{Res}_x(f, g), \operatorname{Res}_x(f, h))$;
2. *the numerators of* $\operatorname{Res}_x(f, h)$ *and* $\operatorname{Res}_x(f, g)$ *evaluated at* $y = B/A$ *are divisible by* F.

Proof. To prove the first part, we consider any point $\mathbf{a}^*$ over the algebraic closure of k such that $F(\mathbf{a}^*) = 0$ and $A(\mathbf{a}^*) \neq 0$. Then we can write

$$(4.2) \qquad f(\mathbf{a}^*, x, y) = \left(x + \frac{C(\mathbf{a}^*)}{A(\mathbf{a}^*)} \right) (A(\mathbf{a}^*)y + B(\mathbf{a}^*))$$

Thus both $\operatorname{Res}_x(f, g)$ and $\operatorname{Res}_x(f, h)$ at $\mathbf{a}^*$ will be divisible by $A(\mathbf{a}^*)y + B(\mathbf{a}^*)$, so their resultant will vanish. Since $F(\mathbf{a}) = 0, A(\mathbf{a}) \neq 0$ is Zariski dense in $F(\mathbf{a}) = 0$, F divides the resultant.

To prove the second part, consider the factorization (4.2) in the localization of $k[\mathbf{a}]/\langle F \rangle$ with respect to A (which is well-defined since $\gcd(A, F) = 1$) with $\mathbf{a}^*$ being the image of $\mathbf{a}$ under the canonical projection. Since the images of $\operatorname{Res}_x(f, h)$ and $\operatorname{Res}_x(f, g)$ vanish at $B(\mathbf{a}^*)/A(\mathbf{a}^*)$, the numerators of their values at $B(\mathbf{a})/A(\mathbf{a})$ belong to $\langle F \rangle$, so are divisible by F. ∎

In order to explain the relation of Proposition 4.14 to Algorithms 4.2 and 4.3, we observe that $\tilde{f}_1, \ldots, \tilde{f}_{j-1}, \tilde{f}_{j+1}, \ldots, \tilde{f}_n$ in the output of Algorithm 4.3 are divisors of a resultant of some polynomial with $\tilde{f}_j$ with respect to x_j. The later runs of Algorithm 4.3 at **(Step 3)** of Algorithm 4.2 will compute resultants of these polynomials and their derivatives, this yielding repeated resultants as in Proposition 4.14. Therefore, if we see that $\tilde{f}_j$ is of the form as in Proposition 4.14 with $x = x_j$ and $y = x_s$ for some s, we perform the change of variable x_s as in Lemma 4.13 (and its derivative is updated correspondingly). Although, the further resultant computations will be performed not with $\tilde{f}_j$ but with factors of their derivatives (so the multiplicity of the common root may be lower), the efficiency gain of such a variable change is substantial as shown in Table 1 below.

Model	Time without change	Time with change
SIWR model (Example 6)	26 s.	3 s.
Pharmacokinetics (Example 7)	93 s.	18 s.
SEAIJRC model (Example 9)	> 5 h.	29 s.

Table 1: Efficiency gain by using the change of variables described in Section 4.5

Remark 9. *In practice, most multiplicities of* $A(\mathbf{a})D(\mathbf{a}) - B(\mathbf{a})C(\mathbf{a})$ *in the resultant will be factored out from the Sylvester matrix, although there is no guarantee that all of them can be eliminated this way.*

Remark 10. *Similar but more complicated variable changes can be devised for other forms of* f. *Unfortunately, in practice, while these variable changes help eliminate some extraneous factors, they introduce others that are usually even more complicated. This can already be seen in the* $f = A(\mathbf{a})xy + B(\mathbf{a})x + C(\mathbf{a})y + D(\mathbf{a})$ *case: while helping to eliminate the extraneous factor* $A(\mathbf{a})D(\mathbf{a}) - B(\mathbf{a})C(\mathbf{a})$, *the variable change actually introduces a new factor* $A(\mathbf{a})$. *Fortunately this new factor is simple enough so that the variable change is actually beneficial for the algorithm efficiency.*

5. Assessing structural identifiability using the PB-representation.

5.1. Overview. In this section we will describe how the algorithms for projection-based (PB) representation from Section 4 can be used to assess structural identifiability of the parameters (or some functions of them) in an ODE model (2.1).

The general idea of this computation follows the lines of the approach via input-output equations [45]. We will give an outline of the algorithm using the notation introduced in Section 2.2.

(Step 1) *Find multi-experiment identifiable functions.* [47, Theorem 21] implies that the field of all multi-experiment identifiable functions of a model Σ (2.1) is equal to the field of definition of the following elimination ideal

$$P := I_\Sigma \cap \mathbb{C}(\boldsymbol{\mu})[\mathbf{y}^{(\infty)}, \mathbf{u}^{(\infty)}].$$

We use Algorithm 4.2 to compute the projection-based representation of P, and then apply Algorithm 4.5 to this representation and the weak membership oracle for P provided by the original ODE system (see Algorithm 4.1) to compute polynomials $f, f_1, \ldots, f_n$ whose coefficients (after normalizing so that every nonzero polynomial has at least one coefficient that is one) generate the field of definition of P.

(Step 2) *Check: ME-identifiable = SE-identifiable ?* [49, Lemma 1] (together with [47, Theorem 21]) provides a sufficient condition for single-experiment identifiable functions to coincide with the multi-experiment ones. This condition yields the equality in many cases (for example, in all but one benchmarks we use in this paper). However, checking this criterion requires computing the rank of certain Wronskian, and such algorithms (e.g., [47, Remark 22]) have limited efficiency. They typically cannot compute the cases in which the order of the Wronskian reaches 100 and thus cannot be used in our case since the order of the Wronskian in the example considered in this paper reaches 2600 (e.g., for Example 7). We develop a fast algorithm for performing this computation described in Section 5.3.

(Step 3) *Check identifiability.* We use Theorem 3.3 to check whether a parameter or a given function of parameters belong to the field of definition of P using the generators of this field computed at **(Step 1)**.

The rest of the section is organized as follows. Section 5.2 contains the proof of Theorem 3.3 which yields a probabilitic algorithm for field membership. Section 5.3 describes an algorithm for checking whether multi-experiment and single-experiment identifiable functions coincide. Finally, in Section 5.4, we give a complete algorithm for assessing structural identifiability detailing the above outline.

5.2. Assessing field membership. In this section, we will prove Theorem 3.3 and give a randomized Algorithm 5.1 for testing membership in rational function fields using it.

Lemma 5.1. *Let $X \subset \mathbb{A}^{n+1}$ be an irreducible algebraic variety over $\mathbb{C}$. We denote the coordinate functions in the ambient space by $x_1, \ldots, x_{n+1}$. Assume that x_{n+1} is algebraic over $x_1, \ldots, x_n$ modulo $\mathcal{I}(X)$ of degree d. Then there exists a polynomial $P \in \mathbb{C}[x_1, \ldots, x_n]$ such that*

- $P \in \mathcal{I}(X)$;

- $L \notin \mathcal{I}(X)$, where L is the leading coefficient of P with respect to x_{n+1};
- $\deg_{x_{n+1}} P = d$ and $\deg P \leqslant 2 \deg X$.

Proof. Relabeling $x_1, \ldots, x_n$ if necessary, we will further assume that there exists $r \leqslant n$ such that $x_1, \ldots, x_r$ form a transcendence basis of $\mathbb{C}[x_1, \ldots, x_{n+1}]$ modulo $\mathcal{I}(X)$. By [17, Theorem 2] X can be represented by a triangular set $P_{n+1}, P_n, \ldots, P_{r+1}$ with respect to the ordering $x_{n+1} > \ldots > x_1$ such that, for every $r < i \leqslant n + 1$,

- $P_i \in \mathbb{C}[x_1, \ldots, x_i]$;
- the total degree of P_i with respect to $x_1, \ldots, x_r$ does not exceed $\deg X$;
- for every $r < j < i$, we have $\deg_{x_j} P_i < \deg_{x_j} P_j$.

  For every $r < i \leqslant n + 1$, we set $d_i := \deg_{x_i} P_i$. Due to the construction of this triangular set [17, Definition 2] (P_i's correspond to N_i's in the notation of [17]), we have that

- none of the leading coefficients of $P_{n+1}, \ldots, P_{r+1}$ belongs to $\mathcal{I}(X)$;
- the degrees $d_{n+1}, \ldots, d_{r+1}$ are the degrees of the polynomials in the lexicographic Gröbner basis with $x_{n+1} > \ldots > x_{r+1}$ of the localization of $\mathcal{I}(X)$ with respect to $\mathbb{C}[x_1, \ldots, x_r]$ (see [17, Assumption 1]).

  The latter implies that $d_{n+1} = d$ and the degree of the variety defined by the localization is equal to $d_{n+1} \ldots d_{r+1}$. Therefore, $d_{n+1} \ldots d_{r+1} \leqslant \deg X$. Thus, we can bound the total degree of P_{n+1} as follows

$$\deg P_{n+1} \leqslant \deg X + d_{n+1} + (d_n - 1) + \ldots + (d_{r+1} - 1) \leqslant \deg X + \prod_{i=r+1}^{n+1} d_i \leqslant 2 \deg X.$$

Hence, we can take $P = P_{n+1}$. ∎

Proposition 5.2. *Let $X \subset \mathbb{A}^{n+1}$ be an irreducible algebraic variety over $\mathbb{C}$. Let $\pi \colon \mathbb{A}^{n+1} \to \mathbb{A}^n$ be the projection to the first n coordinates, and assume that the generic fiber of the restriction $\pi|_X$ is finite. Then there exists a hypersurface $H \subset \mathbb{A}^{n+1}$ of degree at most $4 \deg X$ not containing X with the following properties.*

1. *If the size of the generic fiber of $\pi|_X$ is one, then, for every $p \in X \setminus H$, we have*

$$\mathcal{I}(X) + \mathcal{I}(\pi(p) \times \mathbb{C}) = \mathcal{I}(p).$$

2. *If the size of the generic fiber of $\pi|_X$ is finite but greater than 1, then, for every $p \in X \setminus H$, we have $|\pi^{-1}(\pi(p)) \cap X| > 1$.*

Proof. Let d be the size of the generic fiber of $\pi|_X$. We will give separate constructions for H for the cases $d = 1$ and $d > 1$.

If $d = 1$, then we apply the last part of [32, Lemma 4.3] with $X = X$ and $r = 1$, and obtain a desired hypersurface $H \subset \mathbb{A}^{n+1}$.

Now assume that $d > 1$. We denote the coordinate functions by $x_1, \ldots, x_{n+1}$. We apply Lemma 5.1 to X and obtain a polynomial P with the leading coefficient with respect to x_{n+1} denoted by L. We set $Q := L \cdot \frac{\partial P}{\partial x_{n+1}}$. Note that $\deg Q < 2 \deg P = 4 \deg X$ and Q does not vanish on X. We will prove that the hypersurface H defined by $Q = 0$ satisfied the requirements of the proposition. We denote the images of $x_1, \ldots, x_{n+1}$ in the ring of regular functions of X (that is, in $\mathbb{C}[x_1, \ldots, x_{n+1}]/\mathcal{I}(X)$) by $a_1, \ldots, a_{n+1}$. Let $R := \mathbb{C}[a_1, \ldots, a_n]$.

Then a_{n+1} is algebraic of degree d over R. Consider a polynomial $p(X) := P(a_1, \ldots, a_n, X) \in R[X]$. Then $\deg p = d$ and $p(a_{n+1}) = 0$. We define $\ell = L(a_1, \ldots, a_n) \in R$, the leading coefficient of p. Then [33, Propostion 3.2] implies that

$$(5.1) \qquad \{q(X) \in R[X] \mid q(a_{n+1}) = 0\} = \langle p(X) \rangle \colon \ell^\infty.$$

Consider a homomorphism $\varphi \colon R \to \mathbb{C}$ such that $\varphi(\ell) \neq 0$ and consider any root α of $\varphi(p)(X)$. (5.1) implies that, for every $q(X) \in R[X]$ vanishing at a_{n+1}, there exists N such that $\ell^N q(X)$ is divisible by $p(X)$ over R. Therefore, α is also a root of $\varphi(q)(X)$. This implies that φ can be extended to a homomorphism $R[a_{n+1}] \to \mathbb{C}$ by setting $\varphi(a_{n+1}) = \alpha$. In other words, $(\varphi(x_1), \ldots, \varphi(x_n), \alpha) \in X$.

Now we consider any point $\mathbf{x}^* = (x_1^*, \ldots, x_{n+1}^*)$ such that $Q(\mathbf{x}^*) \neq 0$. We define a homomorphism $\varphi \colon R \to \mathbb{C}$ by $\varphi(x_i) = x_i^*$ for every $1 \leqslant i \leqslant n$. $Q(\mathbf{x}^*) \neq 0$ implies $\varphi(\ell) \neq 0$ and $\varphi(p)'(x_{n+1}^*) \neq 0$. Therefore, $\varphi(p)(X)$ is a polynomial of degree > 1 having at least one simple root, so it must have at least one extra root, let us call it x_{n+1}°. The previous paragraph implies that $(x_1^*, \ldots, x_n^*, x_{n+1}^\circ) \in X$, so $|\pi^{-1}(\pi(\mathbf{x}^*)) \cap X| > 1$. ∎

Lemma 5.3 (Geometric resolution). *Let $r_1(\mathbf{X}) = \frac{f_1(\mathbf{X})}{g(\mathbf{X})}, \ldots, r_N(\mathbf{X}) = \frac{f_N(\mathbf{X})}{g(\mathbf{X})} \in \mathbb{C}(\mathbf{X})$ generate a field F over $\mathbb{C}$. Set $d := \max(1 + \deg g, \deg f_1, \ldots, \deg f_N)$. Then there exists an invertible matrix $M \in \mathbb{C}^{N \times N}$ such that rational functions $\tilde{r}_i := M(r_1, \ldots, r_N)^T$ have the following properties:*

1. *there exists $1 \leqslant \ell \leqslant N$ such that $\tilde{r}_1, \ldots, \tilde{r}_\ell$ form a $\mathbb{C}$-transcendence basis of F;*
2. *$F = \mathbb{C}(\tilde{r}_1, \ldots, \tilde{r}_{\ell+1})$, where ℓ is defined in the previous item;*
3. *there exists a nonzero polynomial $q \in \mathbb{C}[T_1, \ldots, T_{\ell+1}]$ with $\deg q \leqslant d^{\ell+1}$ such that $q(\tilde{r}_1, \ldots, \tilde{r}_{\ell+1}) = 0$;*
4. *there exist polynomials $w_{\ell+1}, \ldots, w_N \in \mathbb{C}[T_1, \ldots, T_{\ell+1}]$ such that*

$$\tilde{r}_i = \frac{w_i(\tilde{r}_1, \ldots, \tilde{r}_{\ell+1})}{\frac{\partial q}{\partial T_{\ell+1}}(\tilde{r}_1, \ldots, \tilde{r}_{\ell+1})} \qquad \text{for every } \ell + 1 < i \leqslant N.$$

Proof. Consider an ideal

$$I := \{p \in \mathbb{C}[T_1, \ldots, T_N] \mid p(r_1, \ldots, r_N) = 0\}.$$

Since I a prime ideal, it admits a geometric resolution due to [25, Proposition 3] which yields the matrix M and polynomials $q, w_{\ell+2}, \ldots, w_N$ such that q is the minimal polynomial of $\tilde{r}_{\ell+1}$ over $\mathbb{C}(\tilde{r}_1, \ldots, \tilde{r}_\ell)$. All the properties of M and the polynomials stated in the lemma follow from [25, Proposition 3] except for the degree bound for q which we will now establish. We write $\tilde{r}_1 = \frac{\hat{f}_1}{g}, \ldots, \tilde{r}_{\ell+1} = \frac{\hat{f}_N}{g}$ for $\hat{f}_1, \ldots, \hat{f}_N \in \mathbb{C}[\mathbf{X}]$. We introduce new variables $\mathbf{y} = (y_1, \ldots, y_{\ell+1})$ and consider an ideal

$$J := \langle \hat{f}_1(\mathbf{X}, \mathbf{y}) - y_1 g(\mathbf{X}), \ldots, \hat{f}_{\ell+1}(\mathbf{X}, \mathbf{y}) - y_{\ell+1} g(\mathbf{X}) \rangle \colon g(\mathbf{X})^\infty \subset \mathbb{C}[\mathbf{X}, \mathbf{y}].$$

The degree of the variety defined by J is $\leqslant d^{\ell+1}$ due to [30, Theorem 1]. The ideal J is the vanishing ideal of the point $(\mathbf{X}, \tilde{r}_1, \ldots, \tilde{r}_{\ell+1})$ over $\mathbb{C}$. Therefore, its projection on the $\mathbf{y}$-coordinates will be the ideal of relations between $\tilde{r}_1, \ldots, \tilde{r}_{\ell+1}$ over $\mathbb{C}$ which is principal due to independence of $\tilde{r}_1, \ldots, \tilde{r}_\ell$ and generated by q. Therefore, the degree of q is bounded by $d^{\ell+1}$. ∎

Proof of Theorem 3.3. We recall the following notation from the statement of the theorem

$$I := \langle f_1(\mathbf{X})g(\mathbf{a}) - f_1(\mathbf{a})g(\mathbf{X}), \dots, f_N(\mathbf{X})g(\mathbf{a}) - f_N(\mathbf{a})g(\mathbf{X}) \rangle \colon g(\mathbf{X})^\infty \subset \mathbb{C}[\mathbf{X}]$$

and denote $F := \mathbb{C}\left(\frac{f_1(\mathbf{X})}{g(\mathbf{X})}, \dots, \frac{f_N(\mathbf{X})}{g(\mathbf{X})}\right)$.

We start with replacing $\frac{f_1}{g}, \dots, \frac{f_N}{g}$ with the result of applying Lemma 5.3 to them. Note that this will not change the ideal I and the field F. Let integer ℓ and polynomials $q, w_{\ell+2}, \dots, w_N \in \mathbb{C}[T_1, \dots, T_{\ell+1}]$ be as in the lemma.

We introduce new variables $y_1, \dots, y_{\ell+1}, z$, and denote by X the variety in $\mathbb{A}^{n+\ell+2}$ defined by the ideal

$$(5.2) \qquad \langle f_1(\mathbf{X}) - y_1 g(\mathbf{X}), \dots, f_{\ell+1}(\mathbf{X}) - y_{\ell+1} g(\mathbf{X}), f(\mathbf{X}) - z g(\mathbf{X}) \rangle \colon g(\mathbf{X})^\infty \subset \mathbb{C}[\mathbf{X}, \mathbf{y}, z].$$

Denote the projection of $\mathbb{A}^{n+\ell+2}$ to the $(\mathbf{y}, z)$-coordinates by π. Let $X_0 := \overline{\pi(X)}$. Denote the projection $(\mathbf{y}, z) \to \mathbf{y}$ by π_0. Note that $\frac{f(\mathbf{X})}{g(\mathbf{X})} \in F$ if and only if the generic fiber of $\pi_0|X_0$ is of cardinality one. If the generic fiber of $\pi_0|_{X_0}$ is finite, let P_0 be the defining polynomial of the hypersurface given by Proposition 5.2 applied to variety X_0 and projection π_0. Then

$$(5.3) \qquad\qquad\qquad\qquad \deg P_0 \leqslant 4 \deg X_0 \leqslant 4 \deg X.$$

We apply [32, Lemma 4.4] to the variety X and projection π, and denote the resulting subvariety of X_0 by Y, we have $\deg Y \leqslant \deg X$.

We will now define a polynomial $Q \in \mathbb{C}[y_1, \dots, y_{\ell+1}, z]$ depending on the geometric situation such that

$$(5.4) \quad Q\left(\frac{f_1(\mathbf{a})}{g(\mathbf{a})}, \dots, \frac{f_{r+1}(\mathbf{a})}{g(\mathbf{a})}, \frac{f(\mathbf{a})}{g(\mathbf{a})}\right) \neq 0 \implies \left[\frac{f(\mathbf{X})}{g(\mathbf{X})} \in F \iff f(\mathbf{X})g(\mathbf{a}) - f(\mathbf{a})g(\mathbf{X}) \in I\right]$$

- *If the cardinality of the generic fiber of $\pi_0|_{X_0}$ is one.* We set $Q := P_0$.
- *If the cardinality of the generic fiber of $\pi_0|_{X_0}$ if greater than one.* We define polynomial $P_1 \in \mathbb{C}[y_1, \dots, y_\ell]$ with $\deg P_1 \leqslant \deg X$ as described below and set $Q := q P_0 P_1$ (where polynomial q comes from Lemma 5.3, geometric resolution, as explained in the beginning of the proof).
  - *If the generic fiber of $\pi_0|_{X_0}$ is finite.* Since Y is a proper subvariety of X_0 and the generic fiber of $\pi_0|_{X_0}$ is finite, $\overline{\pi_0(Y)}$ is a proper subvariety of $\overline{\pi_0(X_0)}$. Since $\overline{\pi_0(Y)}$ has degree at most $\deg Y \leqslant \deg X$, it can be defined by polynomials of degree at most $\deg X$. Therefore, due to [30, Proposition 3], there exists a polynomial $P_1 \in \mathbb{C}[\mathbf{y}]$ of degree at most $\deg X$ vanishing on $\overline{\pi_0(Y)}$ but not vanishing on $\overline{\pi_0(X_0)}$.
  - *If the generic fiber of $\pi_0|_{X_0}$ is infinite.* Since the dimension of a fiber of π_0 does not exceed one and the dimension of a fiber is upper semicontinuous [22, Theorem 14.8], each fiber of $\pi_0|_{X_0}$ is one-dimensional, and, consequently, a line. Since $\dim Y < \dim X_0$, either $\overline{\pi_0(Y)} \subsetneq \overline{\pi_0(X)}$ or the generic fiber of $\pi_0|_Y$ is finite.
    * If $\overline{\pi_0(Y)} \subsetneq \overline{\pi_0(X)}$, let P_1 be a polynomial vanishing on $\overline{\pi_0(Y)}$ but not on $\overline{\pi_0(X_0)}$. Since $\deg \overline{\pi_0(Y)} \leqslant \deg Y$, [30, Proposition 3] implies that P_1 can be chosen so that $\deg P_1 \leqslant \deg Y \leqslant \deg X$.

∗ If $\overline{\pi_0(Y)} = \overline{\pi_0(X)}$, then the generic fiber of $\pi_0|_Y$ is finite. We apply [32, Lemma 4.3(i)] with $X = Y$ and $\pi = \pi_0$, and denote the resulting subvariety of $\overline{\pi_0(Y)} = \overline{\pi_0(X)}$ by Z. By [32, Lemma 4.3(i)], $\deg Z \leqslant \deg X$, so, by [30, Proposition 3], we can choose P_1 to vanish on Z and not vanish on $\pi_0(X)$ such that $\deg P_1 \leqslant \deg Z \leqslant \deg X$.

Consider any point $\mathbf{a} \in \mathbb{C}^n$ such that

$$Q\left(\frac{f_1(\mathbf{a})}{g(\mathbf{a})}, \ldots, \frac{f_{r+1}(\mathbf{a})}{g(\mathbf{a})}, \frac{f(\mathbf{a})}{g(\mathbf{a})}\right) \neq 0 \text{ and is well-defined.}$$

Again, consider cases:

- *the cardinality of the generic fiber of $\pi_0|_{X_0}$ is one.* Then we have (by the choice of P_0)

$$z - \frac{f(\mathbf{a})}{g(\mathbf{a})} \in \mathcal{I}(X_0) + \left\langle y_1 - \frac{f_1(\mathbf{a})}{g(\mathbf{a})}, \ldots, y_{r+1} - \frac{f_{r+1}(\mathbf{a})}{g(\mathbf{a})} \right\rangle.$$

Therefore,

$$f(\mathbf{X})g(\mathbf{a}) - f(\mathbf{a})g(\mathbf{X}) \in \mathcal{I}(X) + \left\langle y_1 - \frac{f_1(\mathbf{a})}{g(\mathbf{a})}, \ldots, y_{r+1} - \frac{f_{r+1}(\mathbf{a})}{g(\mathbf{a})} \right\rangle.$$

Together with (5.2) this yields

$$f(\mathbf{X})g(\mathbf{a}) - f(\mathbf{a})g(\mathbf{X}) \in \langle f_1(\mathbf{X})g(\mathbf{a}) - f_1(\mathbf{a})g(\mathbf{X}), \ldots, f_{r+1}(\mathbf{X})g(\mathbf{a}) - f_{r+1}(\mathbf{a})g(\mathbf{X})\rangle : g(\mathbf{X})^\infty \subset I.$$

- *the cardinality of the generic fiber of $\pi_0|_{X_0}$ is greater than one.* We will show that there exists $\alpha \in \mathbb{C}$ such that $\alpha \neq \frac{f(\mathbf{a})}{g(\mathbf{a})}$ and

$$p := \left(\frac{f_1(\mathbf{a})}{g(\mathbf{a})}, \ldots, \frac{f_{r+1}(\mathbf{a})}{g(\mathbf{a})}, \alpha\right) \in X_0 \setminus Y.$$

Consider cases:

  − *if the generic fiber $\pi_0|_{X_0}$ is finite,* then the choice of P_0 implies that there exists $\alpha \in \mathbb{C}$ such that $\alpha \neq \frac{f(\mathbf{a})}{g(\mathbf{a})}$ and $p := (\frac{f_1(\mathbf{a})}{g(\mathbf{a})}, \ldots, \frac{f_{r+1}(\mathbf{a})}{g(\mathbf{a})}, \alpha) \in X_0$. Since $P_1\left(\frac{f_1(\mathbf{a})}{g(\mathbf{a})}, \ldots, \frac{f_{r+1}(\mathbf{a})}{g(\mathbf{a})}\right) \neq 0$, we have $p \notin Y$.

  − *if generic fiber of $\pi_0|_{X_0}$ is infinite,* the inequation $P_1\left(\frac{f_1(\mathbf{a})}{g(\mathbf{a})}, \ldots, \frac{f_{r+1}(\mathbf{a})}{g(\mathbf{a})}\right) \neq 0$ implies that $\pi_0^{-1}(\frac{f_1(\mathbf{a})}{g(\mathbf{a})}, \ldots, \frac{f_{r+1}(\mathbf{a})}{g(\mathbf{a})}) \cap Y$ is finite, so one can choose a point p to avoid this finite set.

Since $p \in X_0 \setminus Y$, there exists $\mathbf{b} \in \mathbb{C}^n$ such that

$$p = \left(\frac{f_1(\mathbf{b})}{g(\mathbf{b})}, \ldots, \frac{f_{r+1}(\mathbf{b})}{g(\mathbf{b})}, \frac{f(\mathbf{b})}{g(\mathbf{b})}\right).$$

Since $q\left(\frac{f_1(\mathbf{a})}{g(\mathbf{a})}, \ldots, \frac{f_{\ell+1}(\mathbf{a})}{g(\mathbf{a})}\right) \neq 0$, the values of $\frac{f_{\ell+2}(\mathbf{a})}{g(\mathbf{a})}, \ldots, \frac{f_N(\mathbf{a})}{g(\mathbf{a})}$ are uniquely defined by $\frac{f_1(\mathbf{a})}{g(\mathbf{a})}, \ldots, \frac{f_{\ell+1}(\mathbf{a})}{g(\mathbf{a})}$. Therefore, since $\frac{f_i(\mathbf{a})}{g(\mathbf{a})} = \frac{f_i(\mathbf{b})}{g(\mathbf{b})}$ for $1 \leqslant i \leqslant r+1$, we have $\frac{f_i(\mathbf{a})}{g(\mathbf{a})} = \frac{f_i(\mathbf{b})}{g(\mathbf{b})}$ for $\ell + 1 < i \leqslant N$ as well. Since $f(\mathbf{X})g(\mathbf{a}) - f(\mathbf{a})g(\mathbf{X})$ and $g(\mathbf{X})$ do not vanish at $\mathbf{b}$, the polynomial $f(\mathbf{X})g(\mathbf{a}) - f(\mathbf{a})g(\mathbf{X})$ does not belong to I.

Since the conclusion of the implication (5.4) implies both conclusions of the theorem, it is sufficient to prove that

$$P\left(Q\left(\frac{f_1(\mathbf{a})}{g(\mathbf{a})}, \ldots, \frac{f_{\ell+1}(\mathbf{a})}{g(\mathbf{a})}\right) \neq 0 \text{ and well-defined}\right) \geqslant p.$$

We start with bounding the degree of Q:

$$\deg Q \leqslant \deg P_0 + \deg P_1 + \deg q \leqslant 5 \deg X + \deg q \leqslant 5d^{\ell+2} + d^{\ell+1},$$

where the last inequality follows from the Bezout bound [30, Theorem 1]. Then the degree of the denominator $Q_d(\mathbf{X})$ of $Q\left(\frac{f_1(\mathbf{X})}{g(\mathbf{X})}, \ldots, \frac{f_{\ell+1}(\mathbf{X})}{g(\mathbf{X})}\right)$ does not exceed $5d^{\ell+3} + d^{\ell+2}$. Therefore,

$$P\left(Q\left(\frac{f_1(\mathbf{a})}{g(\mathbf{a})}, \ldots, \frac{f_{\ell+1}(\mathbf{a})}{g(\mathbf{a})}\right) \neq 0 \text{ well-defined}\right) \geqslant P(Q_d(\mathbf{a})g(\mathbf{a}) \neq 0) \geqslant 1 - \frac{5d^{\ell+3} + d^{\ell+2} + d}{M}.$$

where the second inequality is due to the Demillo-Lipton-Schwartz-Zippel lemma [66, Proposition 98]. Since $d \geqslant 2$, the latter expression is greater than p. ∎

Algorithm 5.1 Checking membership for rational function fields

Input   rational functions $f, f_1, \ldots, f_N \in \mathbb{C}(\mathbf{x})$, where $\mathbf{x} = (x_1, \ldots, x_n)$, and a real number $0 < \varepsilon < 1$;

Output   YES if $f \in \mathbb{C}(f_1, \ldots, f_N)$ and NO otherwise.
   The result is guaranteed to be correct with probability at least $1 - \varepsilon$.

(Step 1) Write $f, f_1, \ldots, f_N$, where $f = \frac{F}{Q}$ and $f_i = \frac{F_i}{Q}$ for every $i = 1, \ldots, N$ and $Q, F, F_1, \ldots, F_N \in \mathbb{C}[x_1, \ldots, x_n]$.

(Step 2) Compute $d := \max(\deg Q + 1, \deg F, \deg F_1, \ldots, \deg F_N)$ and $M := 6d^{n+3}/\varepsilon$.

(Step 3) Sample integers $\mathbf{a} = (a_1, \ldots, a_n)$ from $[0, M]$ uniformly at random.

(Step 4) Set $I := \langle F_1(\mathbf{x})Q(\mathbf{a}) - F_1(\mathbf{a})Q(\mathbf{x}), \ldots, F_N(\mathbf{x})Q(\mathbf{a}) - F_N(\mathbf{a})Q(\mathbf{x})\rangle \colon Q(\mathbf{x})^{\infty} \subset \mathbb{C}[\mathbf{x}]$.

(Step 5) Use Gröbner bases to check $F(\mathbf{x})Q(\mathbf{a}) - F(\mathbf{a})Q(\mathbf{x}) \in I$. If yes, return YES, otherwise return NO.

Lemma 5.4. *The output of Algorithm 5.1 is correct with probability at least $1 - \varepsilon$.*

Proof. Follows directly from Theorem 3.3. ∎

5.3. Checking whether SE-identifiable = ME-identifiabile. We will use the notation from Section 5.1. As described in **(Step 1)** of the approach outlined there, for a given model Σ (2.1), one can combine Algorithms 4.2 and 4.5 in order to compute $f, f_1, \ldots, f_n$ such that the coefficients of $f, f_1, \ldots, f_n$ (after normalizing so that every nonzero polynomial has at least one coefficient is equal to one) generate the field of definition of the ideal of input-output relations, $I_\Sigma \cap \mathbb{C}(\boldsymbol{\mu})[\mathbf{y}^{(\infty)}, \mathbf{u}^{(\infty)}]$, which is equal to the field of all multi-experiment identifiable functions due to [47, Theorem 21]. In this section, we will present an algorithm allowing one to conclude in some cases (in practice, very often) that the coefficients also generate the field of single-experiment identifiable functions. The algorithm will be based on Lemma 5.6 below which, is a slight modification of [49, Lemma 1].

Definition 5.5 (Wronskian). *Let $a_1, \ldots, a_m \in R$ be elements of a differential ring R. Then we defined the* Wronskian *of $a_1, \ldots, a_m$ to be the following matrix*

$$\mathrm{Wr}(a_1, \ldots, a_m) := \begin{pmatrix} a_1 & \ldots & a_m \\ a_1' & \ldots & a_m' \\ \vdots & \ddots & \vdots \\ a_1^{(m-1)} & \ldots & a_m^{(m-1)} \end{pmatrix}.$$

Lemma 5.6. *Let Σ and I_Σ be as above. Let $g \in I_\Sigma \cap \mathbb{C}(\boldsymbol{\mu})[\mathbf{y}^{(\infty)}, \mathbf{u}^{(\infty)}]$ be a nonzero element such that at least one of the coefficients of g is 1. Assume that g is written as $g = z_0 + \sum_{i=1}^{N} a_i z_i$ such that $z_0, \ldots, z_N \in \mathbb{C}[\mathbf{y}^{(\infty)}, \mathbf{u}^{(\infty)}]$ and $a_1, \ldots, a_N \in \mathbb{C}(\boldsymbol{\mu})$. If the rank of the $\mathrm{Wr}(z_0, \ldots, z_N)$ modulo I_Σ is equal to N, then $a_1, \ldots, a_N$ are single-experiment identifiable.*

Proof. Since the proof of [49, Lemma 1] does not use the fact that $z_0, \ldots, z_N$ are monomials but only uses that they belong to $\mathbb{C}[\mathbf{y}^{(\infty)}, \mathbf{u}^{(\infty)}]$, the same argument proves this lemma. ∎

If one can verify that each of $f, f_1, \ldots, f_n$ satisfy the requirements of Lemma 5.6, then one can conclude that the fields of single-experiment and multi-experiment functions coincide. However, in the benchmarks we consider in this paper, the number N in Lemma 5.6 can be as large as 2600 (in Example 7). For the problems of this size, the algorithms computing the derivatives of the outputs up to this order symbolically (such as in [47, 35]) are not efficient enough. We propose Algorithm 5.2, which, together with the enhancement from Remark 11, can compute the rank of a specialization of the Wronskian from Lemma 5.6 in several minutes for all the examples we consider.

Algorithm 5.2 Checking single-experiment identifiability of coefficients

---

Input   an ODE model Σ as in (2.1) and a differential polynomial $g \in \mathbb{C}(\boldsymbol{\mu})[\mathbf{y}^{(\infty)}, \mathbf{u}^{(\infty)}] \cap I_\Sigma$ with at least one coefficient being one.

Output Single-experiment if all the coefficients of g are single-experiment identifiable or Not sure if single-experiment identifiability could not be concluded.

(Step 1) Write $g = z_0 + \sum_{i=1}^{N} a_i z_i$, where $z_0, \ldots, z_N \in \mathbb{C}[\mathbf{y}^{(\infty)}, \mathbf{u}^{(\infty)}]$ and $a_1, \ldots, a_N \in \mathbb{C}(\boldsymbol{\mu})$ such that N is minimal possible using [47, Algorithm 3].

(Step 2) Let h be the order of g, and pick a prime number $p > N$.

(Step 3) Sample random $\boldsymbol{\mu}^* \in \mathbb{F}_p^\ell$, $\mathbf{x}^* \in \mathbb{F}_p^n$, and truncated up to order $N + h + 1$ power series $\mathbf{u}^* \in (\mathbb{F}_p[\![t]\!])^s$.

(Step 4) Compute the truncated power series solution for Σ in $\mathbb{F}_p[\![t]\!]$ up to order $N + h + 1$ with the parameter value, initial conditions, and inputs being $\boldsymbol{\mu}^*$, $\mathbf{x}^*$, and $\mathbf{u}^*$, repectively (using the algorithm from [5]). If division by zero occurs, go to **(Step 2)** and choose a larger prime.

(Step 5) Denote the $\mathbf{y}$-components of the computed solution by $\mathbf{y}^*$. Compute $g_1 = z_1(\mathbf{y}^*, \mathbf{u}^*), \ldots, g_N = z_N(\mathbf{y}^*, \mathbf{u}^*)$ (see Remark 11).

(Step 6) Form an $N \times N$-matrix M with the i-th column being the first N Taylor coefficients in g_i for $i = 1, \ldots, N$.

(Step 7) If M is nonsingular, return Single-experiment, otherwise return Not sure.

Lemma 5.7. *Algorithm 5.2 is correct.*

Proof. Let $\widetilde{M}$ be the matrix obtained from M by multiplying the j-th row by $(j-1)!$ for every $j = 1, \ldots, N$. Since $p > N$, $\det M \neq 0 \iff \det \widetilde{M} \neq 0$. We observe that $\widetilde{M}$ is the evaluation of the matrix $\mathrm{Wr}(g_1, \ldots, g_N)$ at $t = 0$. If the evaluation is nonsingular, then the Wronskian as well as the Wronskian $\mathrm{Wr}(z_1, \ldots, z_N) \pmod{I_\Sigma}$ are nonsingular, too. Therefore, the output value `Single-experiment` in this case is justified by Lemma 5.6. ∎

Algorithm 5.3 Assessing identifiability

Input
- an ODE model Σ as in (2.1) over $\mathbb{Q}$;
- a rational function $h(\boldsymbol{\mu}) \in \mathbb{Q}(\boldsymbol{\mu})$;
- a real number $0 < \varepsilon < 1$.

Output
- One of
  - `NO` if $h(\boldsymbol{\mu})$ is not multi-experiment identifiable;
  - `Locally` if $h(\boldsymbol{\mu})$ is locally but not globally mutli-experiment identifiable;
  - `Globally` if $h(\boldsymbol{\mu})$ is globally mutli-experiment identifiable
- `True` if the result is also valid for single-experiment identifiability, and `False` if such a conclusion could not be made by the algorithm.

The result is guaranteed to be correct with probability at least $1 - \varepsilon$.

(Step 1) *Assess local identifiability*
  - (a) Use [46, Algorithm 1] (with probability of error at most $\varepsilon/4$) to compute r such that SE-idenitfiability in Σ_r implies ME-identifiability.
  - (b) Use the algorithm from [54] (with probability of error at most $\varepsilon/4$) to check whether $h(\boldsymbol{\mu})$ is Σ_r locally-identifiable. If the answer is no, return (`NO`, `True`) if $r = 1$ or (`NO`, `False`) if $r > 1$.

(Step 2) *Compute input-output projections*
  - (a) Apply Algorithm 4.2 to obtain the parametric profile $\mathbf{h}$ and the projections $\mathbf{f}$ for $I_\Sigma \cap \mathbb{C}(\bar{\mu})[\mathbf{y}^{(\infty)}, \mathbf{u}^{(\infty)}]$.
  - (b) Apply Algorithm 4.5 to the projections $\mathbf{f}$ with the profile $\mathbf{h}$ and the weak membership test from Algorithm 4.1 to obtain the polynomial f.

(Step 3) *SE-identifiable = ME-identifiable?* Apply Algorithm 5.2 to Σ and each of $f, \mathbf{f}$ (after the normalization so that at least one coefficient is equal to one). If any of the results will be `Not sure`, set F to be `False`, otherwise set it to be `True`.

(Step 4) *Assess global identifiability* Let $c_1, \ldots, c_N$ be the coefficients of $f, \mathbf{f}$ after the normalization. Use Algorithm 5.1 (with probability of error at most $\varepsilon/2$) to check whether $h \in \mathbb{Q}(c_1, \ldots, c_N)$. If the result is YES, return (`Globally`, F), otherwise return (`Locally`, F).

5.4. Algorithm for assessing identifiability.

Remark 11 (Simultaneous evaluation). *The most computationally demanding step of Algorithm 5.2 is typically **(Step 5)**. This is because N is often of order of hundreds or thousands, and $\mathbf{y}^*$ and $\mathbf{u}^*$ are truncated power series of order more than N, so performing many arithmetic operations with them (especially, multiplication) may take long time.*

The problem can be formulated in a general form as follows: given N polynomials $q_1, \ldots, q_N \in R[x_1, \ldots, x_r]$ over ring R and a tuple $\mathbf{a} \in R^r$, evaluate $q_1(\mathbf{a}), \ldots, q_N(\mathbf{a})$ in a way that would make the number of multiplications in R smaller. We did this by first evaluating all the monomials appearing in $q_1, \ldots, q_N$ at $\mathbf{a}$, and then taking linear combinations of these results. For evaluating the monomials, we iterate through them in the ascending order with respect to degree and maintain a trie with the exponent vectors of already evaluated monomials. For each next monomial m, we find a monomial m_0 in the trie of largest degree such that $m_0 \mid m$ (this can be done in the linear time in the size of the trie), and reduce evaluation of m to the evaluation of m/m_0. This approach leads to substantial speedup (e.g., 5 min. instead of 10 h. for Example 7).

Proposition 5.8 (Proof of Theorem 3.4). *The output of Algorithm 5.3 is correct with probability at least $1 - \varepsilon$.*

Proof. If the function $h(\boldsymbol{\mu})$ is not identifiable, this will be detected at **(Step 1)**. Moreover, this will be done correctly with probability at least $1 - 2\varepsilon/4 = 1 - \varepsilon/2$.

If the function $h(\boldsymbol{\mu})$ is locally identifiable, then the cases "globally" and "locally not globally" will be distinguished by **(Step 2)** and **(Step 4)**. Summing up the probability of errors, we see that the result will be correct with probability at least $1 - \varepsilon$. Finally, the returned value of F will satisfy the specification due to the specification of Algorithm 5.2. ∎

6. Implementation and performance. We have implemented Algorithm 5.3 (and all the algorithms it relies on) in Julia language as a part of StructuralIdentifiability package. The package is publicly available at https://github.com/SciML/StructuralIdentifiability.jl as a part of the SciML ecosystem[2]. We use the symbolic computation libraries Nemo [23], AbstractAlgebra.jl, Singular.jl, and GroebnerBases.jl. In this section we will demonstrate the performance of our implementation (version 0.2) on a set of benchmark problems. The problems and their parameters are summarized in Table 2 below and the explicit equations for most of them are collected in the Appendix. All the timing in this paper are CPU times (not elapsed) measured on a laptop with Mac OS, 16 Gb RAM, and 4 cores 1.60GHz each. The benchmark models in the formats of all the used software tools are available at https://github.com/SciML/StructuralIdentifiability.jl/tree/master/benchmarking.

The rest of the section is structured as follows. In Section 6.1 we report and discuss the runtimes of the main steps of the algorithms on the benchmark problems. Section 6.2 contains the comparison of the performance of our implementation with several state-of-the art software tools for assessing global structural parameter identifiability, DAISY [4], SIAN [31], COMBOS [44], and GenSSI [39]. Finally, in Section 6.3, we compare the performance of the elimination algorithm we use (Algorithm 4.2) for ODE models with general-purpose differential elimination algorithms (Rosenfeld-Gröbner and Differential Thomas from Maple).

---

[2]https://sciml.ai/

Name	Equations	states	params	outputs	inputs
SIWR model (Example 6)	(6.1) & (6.2)	4	7	1	0
SIWR model - 2 (Example 6)	(6.1) & (6.3)	4	7	2	0
Pharmacokinetics (Example 7)	(6.4)	4	7	1	0
MAPK pathway - 1 (Example 8)	(6.5) & (6.6)	12	22	6	0
MAPK pathway - 2 (Example 8)	(6.5) & (6.7)	12	22	5	0
MAPK pathway - 3 (Example 8)	(6.5) & (6.8)	12	22	5	0
SEAIJRC model (Example 9)	(6.9)	6	8	2	0
Goodwin oscillator (Example 10)	(6.11)	4	7	1	0
Akt pathway (Example 11)	(6.12) & (6.13)	9	16	3	1
NFκB ([31, Section B.5])	[31, Section B.5]	15	13	6	1
Mass-action ([31, Section B.1])	[31, Section B.1]	6	6	2	0
SIRS w. forcing ([31, Section B.3])	[31, Section B.1]	5	6	2	0

Table 2: Summary of the used benchmark problems

6.1. Performance and its discussion. For each of the benchmark models, Table 3 below contains

- the runtime of our implementation with a breakdown for the steps of Algorithms 5.3;
- indication whether the computed projection-based representation was actually a characteristic set of the elimination ideal as described in Remark 4;
- whether or not Algorithm 5.2 could conclude that the coefficients of the computed input-output equations are single-experiment identifiable.

Name	Runtimes (sec.)					Char. set?	SE=ME?
	(Step 1)	(Step 2)	(Step 3)	(Step 4)	Total		
SIWR model (Example 6)	0.1	3	9.5	5.3	18	✓	✓
SIWR model - 2 (Example 6)	0.1	0.2	0.2	0.2	0.7	✓	✓
Pharmacokinetics (Example 7)	0.1	18.4	344.4	43	406	✓	✓
MAPK pathway - 1 (Example 8)	14.2	2.5	11.4	11.4	39.5	✓	✓
MAPK pathway - 2 (Example 8)	4.9	27.6	14.7	11	58	✓	✓
MAPK pathway - 3 (Example 8)	32.8	397	226	429	1084	✓	✗
SEAIJRC model (Example 9)	0.1	28.6	78	25.2	131.3	✓	✓
Goodwin oscillator (Example 10)	< 0.1	< 0.1	< 0.1	< 0.1	0.2	✓	✓
Akt pathway (Example 11)	1.5	0.2	2.3	1	5	✓	✓
NFκB ([31, Section B.5])	2	> 5h.	N/A	N/A	> 5h.	N/A	N/A
Mass-action ([31, Section B.1])	< 0.1	< 0.1	0.3	< 0.1	0.5	✓	✓
SIRS w. forcing ([31, Section B.3])	< 0.1	1	28.2	1	30.3	✓	✓

Table 3: Details on the performance of the implemented algorithm

We would like to make the following observations:

1. For the majority of the models, computing input-output equations (i.e., **(Step 2)**) is not the most time-consuming step (unlike, for example, DAISY [4]).

2. In all the benchmark models the computed input-output projections form a characteristic set of the elimination ideal as described in Remark 4.

3. In all but one case (in Example 8), the algorithm was able to conclude that the coefficients of the input-output equations are SE-identifiable. To the best of our knowledge, this is the first case of a practically relevant model without constant states exhibiting such a behavior. Using SIAN [31] we have verified that all the parameters are in fact SE-identifiable.

6.2. Comparison with identifiability software. In this section we compare the performance of our implementation of Algorithm 5.3 with popular software packages for assessing local and global structural parameter identifiability:

- SIAN: software writte in MAPLE [31]. The algorithm uses an improved version of the Taylor series approach (for details, see [32]) and can assess single-experiment identifiability of both parameters and initial conditions. The algorithm is a randomized Monte-Carlo algorithm, and we were running it with the default probability 99%. We used version 1.5 of SIAN with MAPLE 2021. The reported runtimes were measured using the `CPUTime` function in MAPLE.

- DAISY: package written in REDUCE [4]. It uses the approach via input-output equations but does not check that their coefficients are SE-identifiable [32, Example 2.14]. The algorithm is randomized but no probability bound is provided. We used DAISY 2.0 and REDUCE REV. 4859. The reported runtimes were measured using the `Showtime` function in REDUCE.

- COMBOS: web-application described in [44]. Similarly to DAISY, it uses input-output equations but does not check that their coefficients are SE-identifiable and does not provide a probability bound.

- GenSSI 2.0: package written in MATLAB [39]. It uses the generating series approach.

Table 4 contains the timings of our implementation, DAISY, and SIAN on the set of benchmarks. We did not include GenSSI and COMBOS into the table because, for all the example, the computation either did not finish in 5 hours or returned an error (*"Warning: Unable to find explicit solution"* for GenSSI and *"Model may have been entered incorrectly or cannot be solved with COMBOS algorithms"* for COMBOS).

From the table, one can see that the performance of our algorithm compares favorably to the state-of-the-art software and performs identifiability analysis for systems that were previously out of reach.

6.3. Comparison with general-purpose differential elimination algorithms. In this section we compare the performance of Algorithm 4.2 (which is used as a subroutine in the main Algorithm 5.3) for computing a PB-representation of the ideal of input-output relations of an ODE system to two state-of-the-art general purpose MAPLE packages for differential elimination: DIFFERENTIALALGEBRA (built on top of the BLAD library) and DIFFERENTIALTHOMAS representing the result of elimination via a characteristic set and simple subsystems, respectively. Note that, for all the considered benchmarks, the PB-representation

is actually a characteristic set (see Remark 4). We used MAPLE 2021 and all the reported runtimes were measured using the `CPUTime` function. Table 5 contains the runtimes. One can see that many models which are out of reach for general-purpose algorithms can be tackled by our dedicated algorithm.

Model	DAISY	SIAN	Algorithm 5.3 (our)
SIWR model (Example 6)	OOM	> 5 h.	18 s.
SIWR model - 2 (Example 6)	OOM	213 s.	0.7 s.
Pharmacokinetics (Example 7)	> 5 h.	> 5 h.	406 s.
MAPK pathway - 1 (Example 8)	OOM	31 s.	39.5 s.
MAPK pathway - 2 (Example 8)	> 5 h.	> 5 h.	58 s.
MAPK pathway - 3 (Example 8)	> 5 h.	35 s.	1084 s.
SEAIJRC model (Example 9)	OOM	> 5 h.	131.3 s.
Goodwin oscillator (Example 10)	18 s.*	4920 s.	0.2 s.
Akt pathway (Example 11)	182 s.	28 s.	5 s.
NFκB ([31, Section B.5])	> 5 h.	2018 s.	> 5 h.
Mass-action ([31, Section B.1])	> 5 h.	3 s.	0.5 s.
SIRS w. forcing ([31, Section B.3])	OOM	5 s.	30.3 s.

Table 4: Comparison with other identifiability software

OOM: "out of memory";   ∗: the output is different from the one by SIAN and Algorithm 5.3

Model	DIFFALGEBRA	DIFFTHOMAS	Algorithm 4.2 (our)
SIWR model (Example 6)	> 5 h.	> 5 h.	3 s.
SIWR model - 2 (Example 6)	> 5 h.	> 5 h.	0.2 s.
Pharmacokinetics (Example 7)	OOM	> 5 h.	18.4 s.
MAPK pathway - 1 (Example 8)	13 s.	7 s.	2.5 s.
MAPK pathway - 2 (Example 8)	> 5 h.	> 5 h.	27.6 s.
MAPK pathway - 3 (Example 8)	> 5 h.	> 5 h.	397 s.
SEAIJRC model (Example 9)	> 5 h.	> 5 h.	28.6 s.
Goodwin oscillator (Example 10)	0.2 s.	0.4 s.	< 0.1 s.
Akt pathway (Example 11)	0.2 s.	> 5 h.	0.2 s.
NFκB ([31, Section B.5])	> 5 h.	> 5 h.	> 5 h.
Mass-action ([31, Section B.1])	4.7 s.	> 5 h.	< 0.1 s.
SIRS w. forcing ([31, Section B.3])	> 5 h.	> 5 h.	1 s.

Table 5: Comparison with general purpose libraries for differential elimination

OOM: "out of memory"

REFERENCES

[1] T. BÄCHLER, V. GERDT, M. LANGE-HEGERMANN, AND D. ROBERTZ, *Thomas decomposition of algebraic and differential systems*, in Computer Algebra in Scientific Computing, 2010, pp. 31–54, https://doi.org/10.1007/978-3-642-15274-0_4.

[2] D. J. BATES, J. D. HAUENSTEIN, AND N. MESHKAT, *Identifiability and numerical algebraic geometry*, PLOS ONE, 14 (2019), p. e0226299, https://doi.org/10.1371/journal.pone.0226299.

[3] R. BELLMAN AND K. ÅSTRÖM, *On structural identifiability*, Mathematical Biosciences, 7 (1970), pp. 329–339, http://dx.doi.org/10.1016/0025-5564(70)90132-X.

[4] G. BELLU, M. P. SACCOMANI, S. AUDOLY, AND L. D'ANGIÒ, *DAISY: A new software tool to test global identifiability of biological and physiological systems*, Computer methods and programs in biomedicine, 88 (2007), pp. 52–61, https://doi.org/10.1016/j.cmpb.2007.07.002.

[5] A. BOSTAN, F. CHYZAK, F. OLLIVIER, B. SALVY, E. SCHOST, AND A. SEDOGLAVIC, *Fast computation of power series solutions of systems of differential equations*, in Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms, 2007, p. 1012–1021.

[6] F. BOULIER, *BLAD: Bibliothèques Lilloises d'Algèbre Différentielle*, https://pro.univ-lille.fr/francois-boulier/logiciels/blad/.

[7] F. BOULIER, D. LAZARD, F. OLLIVIER, AND M. PETITOT, *Computing representations for radicals of finitely generated differential ideals*, Applicable Algebra in Engineering, Communication and Computing, 20 (2009), pp. 73–121, https://doi.org/10.1007/s00200-009-0091-7.

[8] F. BOULIER, F. LEMAIRE, AND M. M. MAZA, *Computing differential characteristic sets by change of ordering*, Journal of Symbolic Computation, 45 (2010), pp. 124–149, https://doi.org/10.1016/j.jsc.2009.09.004.

[9] L. BUSÉ AND B. MOURRAIN, *Explicit factors of some iterated resultants and discriminants*, Mathematics of Computation, 78 (2009), pp. 345–386, http://www.jstor.org/stable/40234778.

[10] A. CASTILLO AND R. DIETMANN, *On Hilbert's irreducibility theorem*, Acta Arithmetica, 180 (2017), pp. 1–14, https://doi.org/10.4064/aa8380-2-2017.

[11] O.-T. CHIS, J. R. BANGA, AND E. BALSA-CANTO, *Structural identifiability of systems biology models: A critical comparison of methods*, PLoS ONE, 6 (2011), p. e27755, https://doi.org/10.1371/journal.pone.0027755.

[12] S. D. COHEN, *The distribution of Galois groups and Hilbert's irreducibility theorem*, Proceedings of the London Mathematical Society, s3-43 (1981), pp. 227–250, https://doi.org/10.1112/plms/s3-43.2.227.

[13] S. COLLART, M. KALKBRENNER, AND D. MALL, *Converting bases with the gröbner walk*, Journal of Symbolic Computation, 24 (1997), pp. 465–469, https://doi.org/10.1006/jsco.1996.0145.

[14] G. CONTE, C. H. MOOG, AND A. M. PERDON, *Algebraic Methods for Nonlinear Control Systems*, Springer London, 2007, https://doi.org/10.1007/978-1-84628-595-0.

[15] D. COX, J. LITTLE, AND D. O'SHEA, *Using Algebraic Geometry*, Springer New York, NY, 1998, https://doi.org/10.1007/978-1-4757-6911-1.

[16] X. DAHAN, X. JIN, M. M. MAZA, AND É. SCHOST, *Change of order for regular chains in positive dimension*, Theoretical Computer Science, 392 (2008), pp. 37–65, https://doi.org/10.1016/j.tcs.2007.10.003.

[17] X. DAHAN AND E. SCHOST, *Sharp estimates for triangular sets*, in Proceedings of the 2004 International Symposium on Symbolic and Algebraic Computation, ISSAC '04, ACM, 2004, pp. 103–110, https://doi.org/10.1145/1005285.1005302.

[18] W. DECKER, G.-M. GREUEL, AND G. PFISTER, *Primary decomposition: Algorithms and comparisons*, in

Algorithmic Algebra and Number Theory, B. H. Matzat, G.-M. Greuel, and G. Hiss, eds., Springer Berlin Heidelberg, 1999, pp. 187–220, https://doi.org/10.1007/978-3-642-59932-3_10.

[19] S. Demignot and D. Domurado, *Effect of prosthetic sugar groups on the pharmacokinetics of glucose-oxidase*, Drug Design and Delivery, 1 (1987), pp. 333–348.

[20] L. Denis-Vidal, G. Joly-Blanchard, and C. Noiret, *System identifiability (symbolic computation) and parameter estimation (numerical computation)*, Numerical Algorithms, 34 (2003), pp. 283–292, https://doi.org/10.1023/b:numa.0000005366.05704.88.

[21] L. Denis-Vidal, G. Joly-Blanchard, C. Noiret, and M. Petitot, *An algorithm to test identifiability of non-linear systems*, IFAC Proceedings Volumes, 34 (2001), pp. 197–201, https://doi.org/10.1016/S1474-6670(17)35173-X.

[22] D. Eisenbud, *Commutative Algebra with a View Toward Algebraic Geometry*, Springer-Verlag New York, 1995, https://doi.org/10.1007/978-1-4612-5350-1.

[23] C. Fieker, W. Hart, T. Hofmann, and F. Johansson, *Nemo/Hecke: Computer algebra and number theory packages for the Julia programming language*, in Proceedings of the 2017 ACM on International Symposium on Symbolic and Algebraic Computation, ISSAC '17, ACM, 2017, pp. 157–164, http://doi.acm.org/10.1145/3087604.3087611.

[24] K. A. Fujita, Y. Toyoshima, S. Uda, Y. i. Ozaki, H. Kubota, and S. Kuroda, *Decoupling of receptor and downstream signals in the akt pathway by its low-pass filter characteristics*, Science Signaling, 3 (2010), pp. ra56–ra56, https://doi.org/10.1126/scisignal.2000810.

[25] M. Giusti, G. Lecerf, and B. Salvy, *A Gröbner free alternative for polynomial system solving*, Journal of Complexity, 17 (2001), pp. 154–211, https://doi.org/10.1006/jcom.2000.0571.

[26] A. Glumineau, C. Moog, and F. Plestan, *New algebraic-geometric conditions for the linearization by input-output injection*, IEEE Transactions on Automatic Control, 41 (1996), pp. 598–603, https://doi.org/10.1109/9.489283.

[27] O. Golubitsky, M. Kondratieva, and A. Ovchinnikov, *Algebraic transformation of differential characteristic decompositions from one ranking to another*, Journal of Symbolic Computation, 44 (2009), pp. 333–357, https://doi.org/10.1016/j.jsc.2008.07.002.

[28] B. C. Goodwin, *Oscillatory behavior in enzymatic control processes*, Advances in Enzyme Regulation, 3 (1965), pp. 425–437, https://doi.org/10.1016/0065-2571(65)90067-1.

[29] H. A. Harrington, K. L. Ho, and N. Meshkat, *A parameter-free model comparison test using differential algebra*, Complexity, 2019 (2019), pp. 1–15, https://doi.org/10.1155/2019/6041981.

[30] J. Heintz, *Definability and fast quantifier elimination in algebraically closed fields*, Theoretical Computer Science, 24 (1983), pp. 239–277, https://doi.org/10.1016/0304-3975(83)90002-6.

[31] H. Hong, A. Ovchinnikov, G. Pogudin, and C. Yap, *SIAN: software for structural identifiability analysis of ODE models*, Bioinformatics, 35 (2019), pp. 2873–2874, https://doi.org/10.1093/bioinformatics/bty1069.

[32] H. Hong, A. Ovchinnikov, G. Pogudin, and C. Yap, *Global identifiability of differential models*, Communications on Pure and Applied Mathematics, 73 (2020), pp. 1831–1879, https://doi.org/10.1002/cpa.21921.

[33] E. Hubert, *Notes on triangular sets and triangulation-decomposition algorithms I: Polynomial systems*, in Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2003, pp. 1–39, https://doi.org/10.1007/3-540-45084-x_1.

[34] E. Hubert, *Notes on triangular sets and triangulation-decomposition algorithms II: Differential systems*, in Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2003, pp. 40–87, https://doi.org/10.1007/3-540-45084-x_2.

[35] I. Ilmer, A. Ovchinnikov, and G. Pogudin, *Web-based structural identifiability analyzer*, in Computational Methods in Systems Biology, 2021, pp. 254–265, https://doi.org/10.1007/978-3-030-85633-5_17.

[36] Z. Jiafan, *Nonlinear systems fault diagnosis with differential elimination*, in 2009 International Conference on Computational Intelligence and Natural Computing, 2009, https://doi.org/10.1109/cinc.2009.38.

[37] M. Komatsu, T. Yaguchi, and K. Nakajima, *Algebraic approach towards the exploitation of "softness": the input-output equation for morphological computation*, The International Journal of Robotics Research, (2020), https://doi.org/10.1177/0278364920912298.

[38] E. C. Lee, M. R. Kelly, B. M. Ochocki, S. M. Akinwumi, K. E. Hamre, J. H. Tien, and M. C. Eisenberg, *Model distinguishability and inference robustness in mechanisms of cholera transmission and loss of immunity*, Journal of Theoretical Biology, 420 (2017), pp. 68–81, http://dx.doi.org/10.1016/j.jtbi.2017.01.032.

[39] T. Ligon, F. Fröhlich, O. T. Chiş, J. Banga, E. Balsa-Canto, and J. Hasenauer, *GenSSI 2.0: multi-experiment structural identifiability analysis of SBML models*, Bioinformatics, 34 (2018), pp. 1421–1423, http://dx.doi.org/10.1093/bioinformatics/btx735.

[40] L. Ljung and T. Glad, *On global identifiability for arbitrary model parametrizations*, Automatica, 30 (1994), pp. 265–276, https://doi.org/10.1016/0005-1098(94)90029-9.

[41] A. K. Manrai and J. Gunawardena, *The geometry of multisite phosphorylation*, Biophysical journal, 95 (2008), pp. 5533–5543.

[42] G. Massonis, J. R. Banga, and A. F. Villaverde, *Structural identifiability and observability of compartmental models of the COVID-19 pandemic*, Annual Reviews in Control, (2020), https://doi.org/10.1016/j.arcontrol.2020.12.001.

[43] N. Meshkat, M. Eisenberg, and J. DiStefano, *An algorithm for finding globally identifiable parameter combinations of nonlinear ODE models using Gröbner bases*, Mathematical Biosciences, 222 (2009), pp. 61–72, https://doi.org/10.1016/j.mbs.2009.08.010.

[44] N. Meshkat, C. Kuo, and J. DiStefano, *On finding and using identifiable parameter combinations in nonlinear dynamic systems biology models and COMBOS: A novel web implementation*, PLoS ONE, 9 (2014), p. e110261, https://doi.org/10.1371/journal.pone.0110261.

[45] F. Ollivier, *Le problème de l'identifiabilité structurelle globale: approche th éorique, méthodes effectives et bornes de complexité*, PhD thesis, École polytechnique, 1990, https://www.theses.fr/1990EPXX0009.

[46] A. Ovchinnikov, A. Pillay, G. Pogudin, and T. Scanlon, *Multi-experiment parameter identifiability of ODEs and model theory*, (2020), https://arxiv.org/abs/2011.10868.

[47] A. Ovchinnikov, A. Pillay, G. Pogudin, and T. Scanlon, *Computing all identifiable functions of parameters for ODE models*, Systems & Control Letters, 157 (2021), p. 105030, https://doi.org/10.1016/j.sysconle.2021.105030.

[48] A. Ovchinnikov, G. Pogudin, and P. Thompson, *Input-output equations and identifiability of linear ODE models*. 2020, https://arxiv.org/abs/1910.03960.

[49] A. Ovchinnikov, G. Pogudin, and P. Thompson, *Parameter identifiability and input-output equations*, Applicable Algebra in Engineering, Communication and Computing, (2021), https://doi.org/10.1007/s00200-021-00486-8.

[50] G. Pogudin, *A differential analog of the Noether normalization lemma*, International Mathematics Research Notices, (2017), dx.doi.org/10.1093/imrn/rnw275.

[51] C. H. Remien, M. J. Eckwright, and B. J. Ridenhour, *Structural identifiability of the generalized Lotka-Volterra model for microbiome studies*, Royal Society Open Science, 8 (2021), p. 201378, https://doi.org/10.1098/rsos.201378.

[52] J. F. Ritt, *Differential Equations from the Algebraic Standpoint*, Colloquium Publications, American Mathematical Society, 1932.

[53] K. Roosa and G. Chowell, *Assessing parameter identifiability in compartmental dynamic models using a computational approach: application to infectious disease transmission models*, Theoretical Biology and Medical Modelling, 16 (2019), https://doi.org/10.1186/s12976-018-0097-6.

[54] A. Sedoglavic, *A probabilistic algorithm to test local algebraic observability in polynomial time*, Journal of Symbolic Computation, 33 (2002), pp. 735–755, https://doi.org/10.1006/jsco.2002.0532.

[55] A. Seidenberg, *An elimination theory for differential algebra*, University of California publications in Mathematics, III (1956), pp. 31–66.

[56] E. D. Sontag, *Mathematical Control Theory*, Springer New York, 1998, https://doi.org/10.1007/978-1-4612-0577-7.

[57] M. Staroswiecki and G. Comtet-Varga, *Analytical redundancy relations for fault detection and isolation in algebraic dynamic systems*, Automatica, 37 (2001), pp. 687–699, https://doi.org/10.1016/s0005-1098(01)00005-x.

[58] N. Tuncer and M. Martcheva, *Determining reliable parameter estimates for within-host and within-vector models of zika virus*, Journal of Biological Dynamics, 15 (2021), pp. 430–454, https://doi.org/

10.1080/17513758.2021.1970261.

[59] J. van der Hoeven, *Relax, but don't be too lazy*, J. of Symbolic Computation, 34 (2002), pp. 479–542, https://doi.org/10.1006/jsco.2002.0562.

[60] J. van der Hoeven, *Newton's method and FFT trading*, J. of Symbolic Computation, 45 (2010), pp. 857–878, https://doi.org/10.1016/j.jsc.2010.03.005.

[61] N. Verdiere, L. Denis-Vidal, G. Joly-Blanchard, and D. Domurado, *Identifiability and estimation of pharmacokinetic parameters for the ligands of the macrophage Mannose receptor*, International Journal of Applied Mathematics and Computer Science, 15 (2005), pp. 517–526, http://eudml.org/doc/207763.

[62] D. Wang, *EPSILON: A library of software tools for polynomial elimination*, in Mathematical Software, 2002, https://doi.org/10.1142/9789812777171_0040.

[63] Y. Wang and E. D. Sontag, *Orders of input/output differential equations and state-space dimensions*, SIAM Journal on Control and Optimization, 33 (1995), pp. 1102–1126, https://doi.org/10.1137/s0363012993246828.

[64] X. Xia and C. Moog, *Identifiability of nonlinear systems with application to HIV/AIDS models*, IEEE Transactions on Automatic Control, 48 (2003), pp. 330–336, https://doi.org/10.1109/tac.2002.808494.

[65] E. Yeung, S. McFann, L. Marsh, E. Dufresne, S. Filippi, H. A. Harrington, S. Y. Shvartsman, and M. Wühr, *Inference of multisite phosphorylation rate constants and their modulation by pathogenic mutations*, Current Biology, 30 (2020), pp. 877–882.e6, https://doi.org/10.1016/j.cub.2019.12.052.

[66] R. Zippel, *Effective Polynomial Computation*, Springer, 1993, http://dx.doi.org/10.1007/978-1-4615-3188-3.

Appendix: Benchmark models used in Section 6. In the appendix, we collect some of the models used as benchmarks in the paper. We did not include some relatively long models already explicitly formulated in the literature.

Example 6 (SIWR model). *The following extension of the classical SIR model was proposed by Lee et al [38, Eq. (3)] to model Cholera:*

$$(6.1) \qquad \begin{cases} \dot{s} &= \mu - \beta_I si - \beta_W sw - \mu s + \alpha r, \\ \dot{i} &= \beta_W sw + \beta_I si - \gamma i - \mu i, \\ \dot{w} &= \xi(i - w), \\ \dot{r} &= \gamma i - \mu r - \alpha r, \end{cases}$$

where the state variables s, i, and r denote the fractions of the population that are susceptible, infectious, and recovered, respectively. The variable w represents the concentration of the bacteria in the environment. The following output was considered in the original paper by Lee et al [38]

$$(6.2) \qquad y_1 = \kappa i.$$

In [32], the following extended set of outputs was used

$$(6.3) \qquad y_1 = \kappa i \quad and \quad y_2 = s + i + r.$$

Example 7 (Pharmacokinetics). *This model was first presented by Demignot and Domu-*

rado in [19], and its identifiability was studied in [11, Case study 2]:

$$(6.4) \quad \begin{cases} \dot{x}_1 = a(x_2 - x_1) - \frac{k_a V_m x_1}{k_c k_a + k_c x_3 + k_a x_1}, \\ \dot{x}_2 = a(x_1 - x_2), \\ \dot{x}_3 = b_1(x_4 - x_3) - \frac{k_c V_m x_3}{k_c k_a + k_c x_3 + k_a x_1}, \\ \dot{x}_4 = b_2(x_3 - x_4), \\ y = x_1. \end{cases}$$

Its simplified version has been used in [31, Section B.6] and [32, Example 6.4].

   Example 8 (MAPK pathway). *The following model describing mitogen-activated protein kinase (MAPK) was presented and analuzed by Manrai and Gunawardenahas [41]*

$$(6.5)$$
$$\begin{cases} KS'_{00} = -a_{00}K \cdot S_{00} + b_{00}KS_{00} + \gamma_{0100}FS_{01} + \gamma_{1000}FS_{10} + \gamma_{1100}FS_{11}, \\ KS'_{01} = -a_{01}K \cdot S_{01} + b_{01}KS_{01} + c_{0001}KS_{00} - \alpha_{01}F \cdot S_{01} + \beta_{01}FS_{01} + \gamma_{1101}FS_{11}, \\ KS'_{10} = -a_{10}K \cdot S_{10} + b_{10}KS_{10} + c_{0010}KS_{00} - \alpha_{10}F \cdot S_{10} + \beta_{10}FS_{10} + \gamma_{1110}FS_{11}, \\ FS'_{01} = -\alpha_{11}F \cdot S_{11} + \beta_{11}FS_{11} + c_{0111}KS_{01} + c_{1011}KS_{10} + c_{0011}KS_{00}, \\ FS'_{10} = a_{00}K \cdot S_{00} - (b_{00} + c_{0001} + c_{0010} + c_{0011})KS_{00}, \\ FS'_{11} = a_{01} \cdot S_{01} - (b_{01} + c_{0111})KS_{01}, \\ K' = a_{10}K \cdot S_{10} - (b_{10} + c_{1011})KS_{10}, \\ F' = \alpha_{01}F \cdot S_{01} - (\beta_{01} + \gamma_{0100})FS_{01}, \\ S'_{00} = \alpha_{10}F \cdot S_{10} - (\beta_{10} + \gamma_{1000})FS_{10}, \\ S'_{01} = \alpha_{11}F \cdot S_{11} - (\beta_{11} + \gamma_{1101} + \gamma_{1110} + \gamma_{1100})FS_{11}, \\ S'_{10} = -a_{00}K \cdot S_{00} + (b_{00} + c_{0001} + c_{0010} + c_{0011})KS_{00} - a_{01}K \cdot S_{01} + \\ \qquad (b_{01} + c_{0111})KS_{01} - a_{10}K \cdot S_{10} + (b_{10} + c_{1011})KS_{10}, \\ S'_{11} = -\alpha_{01}F \cdot S_{01} + (\beta_{01} + \gamma_{0100})FS_{01} - \alpha_{10}F \cdot S_{10} + (\beta_{10} + \gamma_{1000})FS_{10} - \alpha_{11}F \cdot S_{11} + \\ \qquad (\beta_{11} + \gamma_{1101} + \gamma 1110 + \gamma 1100)FS_{11}. \end{cases}$$

We will consider three sets of outputs: the one from [2, Example 24]:

$$(6.6) \qquad y_1 = K, \quad y_2 = F, \quad y_3 = S_{00}, \quad y_4 = S_{01}, \quad y_5 = S_{10}, \quad y_6 = S_{11},$$

a restricted version of (6.6)

$$(6.7) \qquad y_1 = F, \quad y_2 = S_{00}, \quad y_3 = S_{01}, \quad y_4 = S_{10}, \quad y_5 = S_{11},$$

and a "symmetrized" version of (6.6)

$$(6.8) \qquad y_1 = K, \quad y_2 = F, \quad y_3 = S_{00}, \quad y_4 = S_{01} + S_{10}, \quad y_5 = S_{11}.$$

   Example 9 (SEAIJRC model). *The following extension of the SIR model was originally designed for studying infuenza, and its identifiability was studied by Roosa and Chowell [53,*

Model 2] and Massonis et al [42, Model 41]:

$$(6.9) \quad \begin{cases} S' = -\beta S(I + J + qA)/N, \\ E' = \beta S(I + J + qA)/N - kE, \\ A' = k(1-\rho)E - \gamma_1 A, \\ I' = k\rho E - (\alpha + \gamma_1)I, \\ J' = \alpha I - \gamma_2 J, \\ C' = \alpha I, \\ R' = \gamma_1(A + I) + \gamma_2 J, \end{cases}$$

where $N = S + E + A + I + J + C + R$ is a known constant. Since R does not appear in the right-hand sides and not present in the output (presented below), we omit it. We use the observable $y_1 = C$ proposed in [42] and add $y_2 = N$ to indicate that the total population is known.

Example 10 (Goodwin oscillator). *The following model describes the oscillations in enzyme kinetics [28] (see also [11, Case study 1]):*

$$(6.10) \quad \begin{cases} \dot{x}_1 = -bx_1 + \frac{a}{A+x_3^\sigma}, \\ \dot{x}_2 = \alpha x_1 - \beta x_2, \\ \dot{x}_3 = \gamma x_2 - \delta x_3, \\ y_1 = x_1. \end{cases}$$

The system is not polynomial but the following way of polynomializing without increasing the dimension of the parameter space has been suggested in [31, Supplementary materials A.3]: we introduce a new parameter c and a new state variable x_4 defined by $c = \frac{A}{a}$ and $x_4 = \frac{x_3^\sigma}{a}$, respectively. Then the system (6.10) becomes

$$(6.11) \quad \begin{cases} \dot{x}_1 = -bx_1 + \frac{1}{c+x_4}, \\ \dot{x}_2 = \alpha x_1 - \beta x_2, \\ \dot{x}_3 = \gamma x_2 - \delta x_3, \\ \dot{x}_4 = \sigma x_4 \frac{\gamma x_2 - \delta x_3}{x_3}, \\ y_1 = x_1. \end{cases}$$

Example 11 (Akt pathway). *The following model for Akt pathway was developed by Fujita*

et al [24]:

$$(6.12) \quad \begin{cases} x'_1 = (k_{12} - k_{11})x_9 + a(u - x_1), \\ x'_2 = k_{91}x_9 - k_{41}x_2 + k_{22}x_3 + k_{31}x_3 - k_{21}x_2x_4, \\ x'_3 = k_{21}x_2x_4 - k_{31}x_3 - k_{22}x_3, \\ x'_4 = k_{71}x_5 + k_{22}x_3 - k_{21}x_2x_4, \\ x'_5 = k_{52}x_7 - k_{71}x_5 + k_{61}x_7 + k_{31}x_3 - k_{51}x_5x_6, \\ x'_6 = k_{52}x_7 + k_{81}x_8 - k_{51}x_5x_6, \\ x'_7 = k_{51}x_5x_6 - k_{61}x_7 - k_{52}x_7, \\ x'_8 = k_{61}x_7 - k_{81}x_8, \\ x'_9 = k_{11}x_9 - k_{91}x_9 - k_{12}x_9, \end{cases}$$

where u is an input variable. The outputs used in [24] were

$$(6.13) \quad \begin{cases} y_1 = a_1(x_2 + x_3), \\ y_2 = a_2(x_5 + x_7), \\ y_3 = a_3x_6. \end{cases}$$