



HAL
open science

Contribution to Integral Elimination

François Lemaire, Louis Roussel

► **To cite this version:**

| François Lemaire, Louis Roussel. Contribution to Integral Elimination. 2024. hal-04570612

HAL Id: hal-04570612

<https://hal.science/hal-04570612>

Preprint submitted on 7 May 2024

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

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

Contribution to Integral Elimination

François Lemaire¹[0000–0001–7349–4396] and Louis Rousset¹[0009–0000–2129–1706]

Univ. Lille, CNRS, Inria, Centrale Lille, UMR 9189 CRISTAL, F-59000 Lille, France
{francois.lemaire,louis.rousset}@univ-lille.fr

Abstract. We present a prototype for performing integral elimination for nonlinear integral equations. This work is motivated by the parameter estimation problem in control theory.

Keywords: Nonlinear equations · Integro-differential equations · Elimination theory

1 Introduction

This article is motivated by the parameter estimation problem in control theory for systems of nonlinear parametric ordinary differential equations. This problem has been extensively addressed in the differential algebra [18,13] context following [8] in [15,6] and many other works.

In this algebraic context, a key object is the so-called input/output (I/O) equation which is a differential equation only involving the parameters, the observed variables (i.e. variables for which experimental data can be obtained) and their derivatives, and also the (freely chosen) inputs and their derivatives. Differential I/O equations can be computed using differential elimination technics (by eliminating non observed variables) such as [5,11]. They can be sensitive to noisy experimental data especially if their order of derivation is high. Integrating those differential I/O equations [3] to obtain integral I/O equations has been successfully tested in a variety of work (see [4] and references within).

An alternative to integrating the differential I/O equations consists in developing an elimination theory for integro-differential equations. Integro-differential operators (which are suited for linear equations) has been studied in [19,16]. A Gröbner-Shirshov approach has been developed in [2,10] but has not been implemented to our knowledge.

We present a first draft of an integral elimination algorithm. Our approach is incomplete and still needs an important theoretical development. However it is already sufficient to handle nontrivial examples (see Section 7). The originality of our approach consists in using new types of reductions and introducing exponential terms during the computations. Our prototype has been implemented in Python using SymPy.

Organization of the paper. Section 2 explains how to manipulate and order the so-called integral monomials, and introduces basic rewriting rules. Section 3

presents more elaborate rewriting rules and an algorithm for reducing a polynomial. Section 4 introduces critical pairs for integral equations. Section 5 presents methods for introducing exponential terms during the elimination process. Section 6 presents Algorithm `integral_elimination` which performs the integral elimination. Finally, Section 7 presents two Biology models treated by Algorithm `integral_elimination`.

Notations. In all the paper, X is a set of indeterminates (morally representing time values functions), the set of commutative monomials on X is denoted X^* , and \mathbb{K} denotes a field of constants (such as \mathbb{Q} , $\mathbb{Q}(\alpha)$, \dots).

2 Background

In an analysis context with enough assumptions on $x(t)$, $\int_0^t x(\tau)d\tau$ is the primitive of the function $x(t)$ which cancels at $t = 0$. In an algebraic context, we can introduce a (formal) operator \int which encodes the primitive operator. It is known that this operator is a Rota-Baxter operator of weight 0 since it satisfies the integration by part $(\int f) \cdot (\int g) = \int(f \int g) + \int(g \int f)$. This last property is easily extended to iterated integrals following the shuffle product [17, section 1.4 page 23] of two words w_1 and w_2 , which is defined as the sum of all the words obtained by interlacing w_1 and w_2 . As an example (if we omit the inner parenthesis in the iterated integrals) we have $(\int f \int g) \cdot (\int a \int b) = \int f \int g \int a \int b + \int f \int a \int g \int b + \int f \int a \int b \int g + \int a \int f \int g \int b + \int a \int f \int b \int g + \int a \int b \int f \int g$.

2.1 Integral Algebra

We introduce the so-called integral monomials and the so-called integral algebra in a rather informal way. More rigorous definitions and mathematical constructions (based on tensor product and tensor algebra) can be found in [12,9] where the authors define their mixed shuffle algebra (which corresponds to our integral algebra).

Definition 1 (Integral monomial). *An integral monomial M on X is an iterated integral $m_0 \int m_1 \int \dots \int m_e$ where $e \in \mathbb{N}$, and the m_i are monomials in X^* . The length e of M is denoted $|M|$. When $e = 0$, an integral monomial is simply a monomial m_0 of X^* . The set of all integral monomials on X is denoted X^{\int} .*

Example 1. x^2 , xy , $\int y^2$, $x \int xy \int y$ are integral monomials.

Notation Monomials in X^* are written in lower case (for example m , m_0 , m_1 , \dots , n_0 , \dots) and integral monomials of X^{\int} are in upper case (M , M_1 , \dots).

One can define a commutative product on our integral monomials in the following way. Note that the product of integral monomials yield a sum of integral monomials because of the shuffle product.

Definition 2 (Product of monomials). *Let us consider two integral monomials $M = m_0 \int m_1 \int \dots \int m_e$ and $N = n_0 \int n_1 \int n_2 \int \dots \int n_f$.*

The product of the two monomials M and N , denoted $M \cdot N$ is defined as

- $m_0 n_0$ when $e = f = 0$
- $m_0 n_0 \int n_1 \int n_2 \int \dots \int n_f$ when $e = 0$ and $f \neq 0$
- $n_0 m_0 \int m_1 \int \dots \int m_e$ when $e \neq 0$ and $f = 0$
- $m_0 n_0 \cdot \sum_{i=1}^p \alpha_i I_i$ where the sum $\sum_{i=1}^p \alpha_i I_i$ is obtained by developing the product $(\int m_1 \int \dots \int m_e) \cdot (\int n_1 \int n_2 \int \dots \int n_f)$ as described above.

Definition 3 (Integral algebra). *The algebra of the so-called integral polynomials is defined as $\mathbb{K}[X^J]$ where the product is induced by the product of monomials of Definition 2. The product of two polynomials P and Q is denoted $P \cdot Q$.*

When M is a monomial in X^* , the product simply amounts to multiply the first monomials n_0 in N . In that case, we omit the \cdot in the product $M \cdot N$ to simplify the notations. The \cdot product notation allows to distinguish between the iterated integral $\int(P \int Q)$ simply written $\int P \int Q$, and the product of $\int P$ times $\int Q$ which is written $\int P \cdot \int Q$.

Example 2. $(\int x \int y \int y) \cdot (\int y \int z) = xy \int xy \int y \int z + xy \int xy \int z \int y + xy \int z \int xy \int y$.

Definition 4 (Integral ideal). *A set I is an integral ideal of $\mathbb{K}[X^J]$ if I is an ideal of the algebra $\mathbb{K}[X^J]$ and if I is stable by \int (i.e. if $P \in I$, then $\int P \in I$).*

The notion of integral ideal is consistent with the fact that the primitive of the zero function (which cancels at $t = 0$) is also the zero function.

2.2 Ordering the Integral Monomials

The elements of X^J can be totally ordered. In this paper, we define an elimination ordering on X^J . More subtle ordering are certainly possible but are left for future work.

Definition 5. *Fix an ordering on the elements of X . This ordering defines a lexicographic ordering on X^* denoted $<_E$. With the same notations as in Definition 2, we define an ordering $<_f$ on X^J in the following way: $M <_f N$ if*

- $\prod_{i=0}^e m_i <_E \prod_{i=0}^f n_i$
- or in case of equality $e < f$
- or in case of equality $(m_0, m_1, \dots, m_e) <_{\text{revlex}} (n_0, n_1, \dots, n_f)$

where $<_{\text{revlex}}$ is the reverse lexicographic order on tuples induced by $<_E$.

It can be shown that the ordering of Definition 5 is a well-ordering.

Our order is an elimination ordering and as a consequence is different from the ordering defined in [9, page 5] which first sorts the monomials with respect to their length.

Lemma 4. Consider three noncommutative words u, v, w on a alphabet Y . Let $<$ denote the lexicographic order (or the reverse lexicographic order on words). If $u < v$, then the maximum word in $u \sqcup w$ (w.r.t. $<$) is smaller than the maximum word in $v \sqcup w$ (where \sqcup denotes the shuffle product operation).

Proof. Left to the reader. □

2.3 Basic Rewriting Rules

Consider a polynomial P in $\mathbb{K}[X^J]$ with leading monomial \overline{P} . The following lemma 5 shows that any integral monomial of the form $m_0 \int \cdots \int m_e \overline{P}$ (where the m_i are in X^*) can be rewritten using P .

Lemma 5. Let $P \in \mathbb{K}[X^J]$ and monomials m_0, \dots, m_e taken in X^* . Then

$$\overline{m_0 \int \cdots \int m_e P} = m_0 \int \cdots \int m_e \overline{P}.$$

Proof. Simple consequence of Lemma 1 and Lemma 2. □

Example 4. Take $X = \{x, y\}$ with $x > y$. Let $P = \int x - y$ (with leading monomial $\overline{P} = \int x$) written as the rule $\int x \rightarrow y$. Then the polynomial $\int(yP)$ has leader $\int y \int x$, and allows to reduce $\int y \int x$ into $\int y^2$.

The following lemma 6 shows that any integral monomial of the form $\overline{\overline{P} \cdot M}$ where M is in X^J can be rewritten using P .

Lemma 6. Let $P \in \mathbb{K}[X^J]$ and $M \in X^J$. Then

$$\overline{P \cdot M} = \overline{\overline{P} \cdot M}.$$

Proof. Consequence of Lemma 3. □

Example 5. Following Example 4, take $M = \int y$. Then $P \cdot M = (\int x - y) \cdot \int y = \int x \int y + \int y \int x - y \int y$. The leading monomial $\overline{P \cdot M}$ of $P \cdot M$ is $\int y \int x$, which coincides (as expected by Lemma 6) with the leading monomial of $\overline{\overline{P} \cdot M} = (\int x) \cdot (\int y)$.

Remark 1. Examples 4 and 5 show that the polynomial $P = \int x - y$ can be used in two different ways to reduce the same monomial $\int y \int x$. In some sense, there is a critical pair between P and itself ! This will be clarified in the next section.

3 Extended Rewriting Rules and Reduction

We show in this section that a single polynomial P can be used to reduce more monomials than presented in Lemma 5 and Lemma 6. To do this, we build in the next two subsections some kind of collisions between polynomials in the ideal generated by P . Finally, last subsection presents a reducing process. Let us first introduce some notations and definitions.

Definition 7. Take an integral monomial $M = m_0 \int m_1 \int \cdots \int m_e$. We define $M_{[0]} = m_0$. When $e \geq 1$, we define $M_{[1]} = m_1$, $M_{[1+]} = m_1 \int \cdots \int m_e$, $M_{[1+]} = \int m_1 \int \cdots \int m_e$. Finally $M_{[2+]} = 1$ if $e = 1$, and $M_{[2+]} = \int m_2 \int \cdots \int m_e$ if $e \geq 2$.

Definition 8. Definition 7 can be extended to polynomials of $\mathbb{K}[X^J]$ by linearity.

Definition 9. Any polynomial $P = \sum_{i=1}^d \alpha_i M_i$ in $\mathbb{K}[X^J]$ can be uniquely written as $P = P_I + P_N$ with

$$P_I = \sum_{M_i_{[0]}=1} \alpha_i M_i \text{ and } P_N = \sum_{M_i_{[0]} \neq 1} \alpha_i M_i.$$

This just amounts to collect the monomials which directly “starts” with an integral sign in P_I (hence the suffix I), and the others in P_N .

3.1 Reduced-Product Rule

Consider a polynomial P in $\mathbb{K}[X^J]$ and a monomial M in X^J . The following lemma 7 shows that the monomial $\overline{\int(\overline{P_I}_{[1+]} \cdot \int M)}$ can be rewritten using P . This is a formalization of Remark 1.

Lemma 7. Let $M \in X^J$ and $P \in \mathbb{K}[X^J]$ such that $P_I \neq 0$ and $\overline{P} = \overline{P_I}$. Then

$$\overline{P \cdot \int M} - \overline{\int(M \cdot P)} = \overline{\int(\overline{P_I}_{[1+]} \cdot \int M)}.$$

Proof. Since $P \cdot \int M = P_I \cdot \int M + P_N \cdot \int M = \int(P_I_{[1+]} \cdot \int M) + \int(M \cdot P_I) + P_N \cdot \int M$ and since $\int(M \cdot P) = \int(M \cdot P_I) + \int(M \cdot P_N)$, we obtain $P \cdot \int M - \int(M \cdot P) = \int(P_I_{[1+]} \cdot \int M) + P_N \cdot \int M - \int(M \cdot P_N)$. Using Lemma 6, and Lemma 8 (with $M = \overline{P_N}$, $N = \overline{P_I}_{[1+]}$ and $U = \int M$), we have $\overline{P_N \cdot \int M} <_f \overline{\overline{P_N} \cdot \int M} <_f \overline{\int(\overline{P_I}_{[1+]} \cdot \int M)}$. Similarly, $\overline{\int(M \cdot P_N)} <_f \overline{\int(\overline{P_I}_{[1+]} \cdot \int M)}$, which ends the proof. \square

Lemma 8. Let M, N and U in X^J then

$$M <_f \int N \text{ implies that } \overline{M \cdot U} <_f \overline{\int(N \cdot U)} \text{ and } \overline{\int(M \cdot U)} <_f \overline{\int(N \cdot \int U)}.$$

Proof. The proof of both statements is similar to that of Lemma 3. \square

Example 6. Take $X = \{x, y\}$ with $x > y$. Let $P = \int x - y$ (with leading monomial $\overline{P} = \int x$). Take $M = y$. We have $P_I = \int x$ and $\overline{P} = \overline{P_I} = \int x$. Thus $\overline{P_I}_{[1+]} = x$.

Lemma 7 proves that the monomial $\int x \int y$ can be rewritten using P .

3.2 Reduced-Power Rule

This section introduces the Lemma 9 which amounts to apply n times Lemma 7 (where n is a positive integer) in the particular case where $\int M = \overline{P}$. To do so, we need to introduce a special power of P denoted $P^{(n)}$ which can be viewed as a simplified version of P^n modulo the ideal generated by P .

Definition 10. Consider a nonzero polynomial P in $\mathbb{K}[X^J]$ written as $P = P_I + P_N$ with $P_I \neq 0$, $P_N \neq 0$, and $\overline{P} = \overline{P_I}$. For any positive integer n , define

$$P^{(n)} = n \int \left(P_I \Big|_{1+} \cdot P_N^{n-1} \right) + P_N^n$$

Lemma 9. With the hypothesis of Definition 10, $P^{(n)}$ belongs to the integral ideal generated by P . Moreover,

$$\overline{P^{(n)}} = \int \overline{P_I \Big|_{1+}} \cdot \overline{P_N^{n-1}}.$$

Lemma 9 is proved below, since it requires some preliminary results.

Example 7. Take $X = \{x, y\}$ with $x > y$. Let $P = \int x - y$. By Definition 10, for any positive integer n , $P^{(n)} = n \int (x(-y)^{n-1}) + (-y)^n = (-1)^{n-1} (n \int (xy^{n-1}) - y^n)$. Lemma 9 then shows any term of the form $\int (xy^{n-1})$ is equal to y^n modulo the ideal generated by P .

Lemma 10. Consider A and B in $\mathbb{K}[X^J]$ and take $P = \int(B) - A$. Then for any integer $n > 0$ the polynomial $n(\int B \cdot A^{n-1}) - A^n$ is in the ideal generated by P .

Remark 2. Lemma 10 can be stated in the following less formal but more intuitive way using equalities. If $A = \int B$ then $A^n = n \int (B \cdot A^{n-1})$.

Proof. To make the proof easier to follow, we use equalities and we actually prove Remark 2 by induction on n . The case $n = 1$ is clear. We want to show $(n + 1) \int (B \cdot A^n) = A^{n+1}$ assuming $n \int (B \cdot A^{n-1}) = A^n$. Multiplying both sides of the last equality by $\int B$ and developing the product yields $n(\int((B \cdot A^{n-1}) \cdot \int B)) + n(\int(B \cdot \int(B \cdot A^{n-1}))) = A^n \cdot \int B = A^{n+1}$. Using $A = \int B$ and the induction hypothesis ends the proof. \square

Lemma 11. Let A and B in $\mathbb{K}[X^J]$. If $\overline{A} <_J \overline{\int B}$ and $n \geq 1$ then

$$n \int (B \cdot A^{n-1}) - A^n = \int \overline{B \cdot A^{n-1}}.$$

Proof. It is straightforward to prove by induction on n using Lemma 8 that $\overline{A^n} <_J \overline{\int(B \cdot A^{n-1})}$. Since $n \int(B \cdot A^{n-1}) = \int(B \cdot \overline{A^{n-1}}) > \overline{A^n}$, we have

$$n \int (B \cdot A^{n-1}) - A^n = \int (B \cdot A^{n-1}) = \int (\overline{B \cdot A^{n-1}}).$$

\square

Proof (of Lemma 9). The first claim of the lemma is a direct consequence of Lemma 10 by taking $A = -P_N$ and $B = P_{I_{1+}}$. Indeed, $(\int B) - A$ equals $P_I + P_N = P$. Moreover, $n(\int B \cdot A^{n-1}) - A^n$ is equal to $P^{(n)}$ up to $(-1)^{n-1}$. The second claim of the lemma is a consequence of Lemma 11. \square

3.3 Reduction

This section presents a reduction process for rewriting an integral monomial M of X^J by an integral polynomial P . This reduction process is not complete yet and would need to be extended in a future work. In particular our reduction process does not handle the case where M is equal to $\overline{P \cdot N}$ for some integral monomial N in X^J .

The following lemma is straightforward (its proof is left to the reader).

Lemma 12 (Reduction of a monomial M by P (simple case)). *Consider M in X^J and P in $\mathbb{K}[X^J]$ with P nonzero. Assume that the leading coefficient of P is equal to one. If there exists a monomial m in X^* such that $m\overline{P} = M$, then introduce $R = M - mP$. Otherwise, if \overline{P} is in X^* and if there exists a monomial m in X^* such that $m\overline{P} = M_{[0]}$, then introduce $R = M - mP \cdot M_{[1+]}$.*

Then $\overline{R} <_J M$ when $R \neq 0$. Moreover R is equal to M modulo the ideal generated by P .

Next lemma handles a case where a power of P can be used to rewrite M .

Lemma 13 (Reducing a monomial M by P when $\overline{P} = \overline{P_I}$ and $|M| \geq 1$). *Consider M in X^J and M in $\mathbb{K}[X^J]$ with P and P_I nonzero with $\overline{P} = \overline{P_I}$. Assume that the leading coefficient of P is equal to one.*

If there exists a positive integer n such that $\overline{P_I}_{[1]} \overline{P_N}_{[0]}^{n-1} = M_{[1]}$ and $\overline{(\overline{P_I}_{[2+]} \cdot (\overline{P_N}_{[1+]})^{n-1})} = M_{[2+]}$ then introduce

$$R = M - \frac{1}{n\ell_P^{n-1}} M_{[0]} P^{(n)}$$

where ℓ_P is the leading coefficient of P_N .

Then $\overline{R} <_J M$ when $R \neq 0$. Moreover R is equal to M modulo the ideal generated by P .

Proof. Lemma 9 implies $\overline{P^{(n)}} = \overline{\int (\overline{P_I}_{[1+]} \cdot \overline{P_N}^{n-1})} = \int (\overline{P_I}_{[1]} \overline{P_N}_{[0]}^{n-1} B_P)$ where $B_P = \overline{(\overline{P_I}_{[2+]} \cdot (\overline{P_N}_{[1+]})^{n-1})}$. Using the two equality assumptions, $M_{[0]} \overline{P^{(n)}} = M_{[0]} \int (M_{[1]} M_{[2+]}) = M$. Finally, it is easy to prove that the leading coefficient of $P^{(n)}$ is equal to $n\ell_P^{n-1}$, which proves the lemma. \square

Example 8. Recall Example 7 where $X = \{x, y\}$ with $x > y$ and $P = \int x - y$. Take $M = \int (xy^{n-1})$ for some positive integer n . Applying Lemma 13 yields $R = \frac{y^n}{n}$.

The following lemma simply extends Lemma 12 and Lemma 13 by considering the “suffixes” of M .

Lemma 14 (Reduction of M by P). *Consider M in X^J and P in $\mathbb{K}[X^J]$ with P nonzero. Denote $M = m_0 \int \cdots \int m_e$. If a suffix $m_i \int \cdots \int m_e$ for $0 \leq i \leq e$ can be reduced by P using either Lemma 12 or Lemma 13, then there exists R in $\mathbb{K}[X^J]$ such that R is equal to M modulo P , with $\overline{R} <_f M$ when $R \neq 0$.*

Proof. Direct consequence of Lemma 5, 12, and 13. □

Previous Lemmas can be used to write Algorithm 1 which reduces a polynomial by a set of polynomials. Note that Algorithm 1 terminates since $<_f$ is a well-ordering.

Algorithm 1: reduce(Q, T, X)

Input: Q an integral polynomial in $\mathbb{K}[X^J]$;
 T a set of integral polynomials of $\mathbb{K}[X^J]$
Result: a polynomial R equal to Q modulo the ideal generated by T
begin
 $R \leftarrow Q$;
 while R contains a monomial M that can be reduced by some polynomial
 P of T using Lemma 14 **do**
 | reduce M by P in R using Lemma 14 ;
 end
 return R
end

4 Critical Pairs

Given two integral polynomials P and Q , we want to find an equivalent of the classical S-polynomials used in the Gröbner basis context. To do so, we look for monomials M that can be reduced by both P and Q .

Using results from the previous sections, we define three different critical pairs detailed in Lemmas 15, 16 and 17. More critical pairs could be considered and are left for future work. Finally, we present Algorithm 2 which extracts all possible S-polynomials from a set of polynomials R .

In all this section, P and Q are non zero integral polynomials such that

$$\begin{aligned}
 &P_I \neq 0, P_N \neq 0, Q_I \neq 0 \text{ and } Q_N \neq 0 \\
 &P_I \text{ and } Q_I \text{ have a leading coefficient equal to 1} \\
 &\text{the leading coefficient of } P_N \text{ (resp. } Q_N) \text{ is denoted } \ell_P \text{ (resp } \ell_Q)
 \end{aligned}
 \tag{1}$$

Since the three lemmas below are a bit technical, we first illustrate each lemma with an example.

Example 9 (for Lemma 15). Take $X = \{x, y\}$ with $x > y$. Let $P = \int x^2 - y \int y$ and $Q = \int xy - x$. Both equations P and Q can be used to reduce the monomial $\int x^2 y \int y$. Indeed, $P^{(2)} = (y \int y)^2 - 2 \int x^2 y \int y$. Moreover, $Q^{(2)} = x^2 - 2 \int x^2 y$, and $(Q^{(2)} \cdot \int y - \int y Q^{(2)}) = -2 \int x^2 y \int y + x^2 \int y - \int x^2 y$. Finally introduce $S(P, Q) = -\frac{1}{2} P^{(2)} - \left(-\frac{1}{2} Q^{(2)} \cdot \int y - \int y Q^{(2)}\right) = \frac{1}{2} (x^2 \int y - \int x^2 y - (y \int y)^2)$ whose leading monomial is less than $\int x^2 y \int y$.

Example 10 (for Lemma 16). Take $X = \{x, y\}$ with $x > y$. Let $P = \int x^2 - y \int y$ and $Q = x \int y - x$. Both equations P and Q can be used to reduce the monomial $\int x^2 y \int y$. Indeed, $P^{(2)} = (y \int y)^2 - 2 \int x^2 y \int y$. Moreover, $\int xy Q = \int x^2 y \int y - \int xy^2$. Finally introduce $S(P, Q) = -\frac{1}{2} P^{(2)} - \int xy Q = \int x^2 y - \frac{1}{2} (y \int y)^2$ whose leading monomial is less than $\int x^2 y \int y$.

Example 11 (for Lemma 17). Take $X = \{x, y\}$ with $x > y$. Let $P = x^2 y \int y + x$ and $Q = xy^3 \int x + y^2$. Both equations P and Q can be used to reduce the monomial $x^2 y^3 \int y \int x$.

Indeed, $\frac{x^2 y^3}{x^2 y} P \cdot \int x = y^2 (x^2 y \int y + x) \cdot \int x = x^2 y^3 (\int y \int x + \int x \int y) + xy^2 \int x$.

Moreover, $\frac{x^2 y^3}{xy^3} Q \cdot \int y = x (xy^3 \int x + y^2) \cdot \int y = x^2 y^3 (\int y \int x + \int x \int y) + xy^2 \int y$.

Finally, introduce $S(P, Q) = \frac{x^2 y^3}{x^2 y} P \cdot \int x - \frac{x^2 y^3}{xy^3} Q \cdot \int y = xy^2 \int x - xy^2 \int y$, whose leading monomial is less than $x^2 y^3 \int y \int x$.

Lemma 15 (Critical pairs: $\overline{P} = \overline{P_I}$ and $\overline{Q} = \overline{Q_I}$ satisfying (1)).

Let $P, Q \in \mathbb{K}[X^J]$ such that $\overline{P} = \overline{P_I}$ and $\overline{Q} = \overline{Q_I}$. If there exist positive integers α and β such that $\overline{P_I}_{[1]} \overline{P_N}_{[0]}^{\alpha-1} = \overline{Q_I}_{[1]} \overline{Q_N}_{[0]}^{\beta-1}$, then we define

$$S(P, Q) = \frac{1}{\alpha \ell_P^{\alpha-1}} \underbrace{\left(P^{(\alpha)} \cdot B_Q - \int (C_Q \cdot P^{(\alpha)}) \right)}_U - \frac{1}{\beta \ell_Q^{\beta-1}} \underbrace{\left(Q^{(\beta)} \cdot B_P - \int (C_P \cdot Q^{(\beta)}) \right)}_V$$

where $B_P = \overline{(\overline{P_I}_{[2+]}) \cdot (\overline{P_N}_{[1+]})^{\alpha-1}}$, $B_Q = \overline{(\overline{Q_I}_{[2+]}) \cdot (\overline{Q_N}_{[1+]})^{\beta-1}}$,

$$C_P = \begin{cases} 0 & \text{if } B_P = 1 \\ B_P]_{1+} & \text{if } B_P \neq 1 \end{cases} \text{ and } C_Q = \begin{cases} 0 & \text{if } B_Q = 1 \\ B_Q]_{1+} & \text{if } B_Q \neq 1. \end{cases}$$

Then $\overline{U} = \overline{V}$. Moreover, $\overline{S(P, Q)} <_J \overline{U}$ when $S(P, Q) \neq 0$.

Proof. We only prove the general case where $B_P \neq 1$ and $B_Q \neq 1$ (the other cases are in fact particular cases that are not difficult to prove when the general case is understood).

Since both B_P and B_Q start with an integral (more precisely $B_P|_{[0]} = B_Q|_{[0]} = 1$), we have $B_P = \int C_P$ and $B_Q = \int C_Q$. Applying Lemma 9 yields $\overline{P^\ominus} = \int \left(\overline{P_I|_{1+}} \cdot \overline{P_N^{\alpha-1}} \right) = \int (\overline{P_I|_{[1]}} \overline{P_N^{\alpha-1}}|_{[0]} B_P)$. Applying Lemma 7 with P^\ominus and $\int C_Q$, we get $\overline{U} = \int (\overline{P_I|_{[1]}} \overline{P_N^{\alpha-1}}|_{[0]} B_P \cdot (\int C_Q)) = \int (\overline{P_I|_{[1]}} \overline{P_N^{\alpha-1}}|_{[0]} \overline{B_P \cdot B_Q})$.

By symmetry on P and Q , we obtain $\overline{V} = \int (\overline{Q_I|_{[1]}} \overline{Q_N^{\alpha-1}}|_{[0]} B_Q \cdot B_P)$. Using the assumption $\overline{P_I|_{[1]}} \overline{P_N^{\alpha-1}}|_{[0]} = \overline{Q_I|_{[1]}} \overline{Q_N^{\beta-1}}|_{[0]}$, we get $\overline{U} = \overline{V}$. It is easy to prove that the leading coefficient of U (resp. V) is $\alpha \ell_P^{\alpha-1}$ (resp. $\beta \ell_Q^{\beta-1}$). As a consequence, the leaders of U and V cancel each other in $S(P, Q)$, thus proving that $\overline{S(P, Q)} <_f \overline{U}$ when $S(P, Q) \neq 0$. \square

Lemma 16 (Critical pairs: $\overline{P} = \overline{P_I}$ and $\overline{Q} = \overline{Q_N}$ satisfying (1)).

Let $P, Q \in \mathbb{K}[X^J]$ such that $\overline{P} = \overline{P_I}$ and $\overline{Q} = \overline{Q_N}$. If there exist a positive integer α and a monomial m in X^* such that $\overline{P_I|_{[1]}} \overline{P_N^{\alpha-1}}|_{[0]} = m \overline{Q_N|_{[0]}}$, then we define

$$S(P, Q) = \frac{1}{\alpha \ell_P^{\alpha-1}} \underbrace{\left(P^\ominus \cdot B_Q - \int (C_Q \cdot P^\ominus) \right)}_U - \frac{1}{\ell_Q} \underbrace{\int (mQ \cdot B_P)}_V$$

where

$$B_P = \overline{(\overline{P_I|_{2+}}) \cdot (\overline{P_N|_{1+}})^{\alpha-1}}, B_Q = \overline{Q_N|_{1+}} \text{ and } C_Q = \begin{cases} 0 & \text{if } B_Q = 1 \\ B_Q|_{1+} & \text{if } B_Q \neq 1. \end{cases}$$

Then $\overline{U} = \overline{V}$. Moreover $\overline{S(P, Q)} <_f \overline{U}$ when $S(P, Q) \neq 0$.

Proof. We only prove the lemma in the general case where $B_P \neq 1$ and $B_Q \neq 1$, so $B_Q = \int C_Q$. From Lemma 15 proof, we have $\overline{U} = \int (\overline{P_I|_{[1]}} \overline{P_N^{\alpha-1}}|_{[0]} \overline{B_P \cdot B_Q})$. On the other hand $\overline{V} = \int (\overline{mQ \cdot B_P}) = \int \left(\overline{mQ_N|_{[0]}} \overline{B_Q \cdot B_P} \right)$. Using the assumption $\overline{P_I|_{[1]}} \overline{P_N^{\alpha-1}}|_{[0]} = m \overline{Q_N|_{[0]}}$, we obtain $\overline{U} = \overline{V}$. Since the leading coefficient of V is ℓ_Q , the leaders of U of V cancel each other in $S(P, Q)$ and $\overline{S(P, Q)} <_f \overline{U}$ when $S(P, Q) \neq 0$. \square

The following lemma is directly inspired by the critical pair defined in the Gröbner basis context.

Lemma 17 (Critical pairs: $\overline{P} = \overline{P_N}$ and $\overline{Q} = \overline{Q_N}$ satisfying (1)).

Let $P, Q \in \mathbb{K}[X^J]$ such that $\overline{P} = \overline{P_N}$ and $\overline{Q} = \overline{Q_N}$. We denote by L the least common multiple (LCM) of $\overline{P_N|_{[0]}}$ and $\overline{Q_N|_{[0]}}$, i.e. $L = \text{LCM}(\overline{P_N|_{[0]}}|_{[0]}, \overline{Q_N|_{[0]}}|_{[0]})$.

Let us define

$$\begin{cases} S(P, Q) = \frac{L}{\ell_P \overline{P_N}_{[0]}} \underbrace{P \cdot \overline{Q_N}_{[1+]}}_U - \frac{L}{\ell_Q \overline{Q_N}_{[0]}} \underbrace{Q \cdot \overline{P_N}_{[1+]}}_V & \text{if } \overline{Q_N}_{[1+]} \neq \overline{P_N}_{[1+]} \\ S(P, Q) = \frac{L}{\ell_P \overline{P_N}_{[0]}} \underbrace{P}_U - \frac{L}{\ell_Q \overline{Q_N}_{[0]}} \underbrace{Q}_V & \text{if } \overline{Q_N}_{[1+]} = \overline{P_N}_{[1+]} \end{cases}$$

Then $\overline{U} = \overline{V}$, and $\overline{S(P, Q)} <_f \overline{U}$ when $S(P, Q) \neq 0$.

Proof. The proof is left to the reader. □

Algorithm 2: critical_pairs(R, X)

Input: R a set of integral polynomials in $\mathbb{K}[X^f]$;

Result: the set S of all S -polynomials that can be obtained from R . Note that S can be empty.

begin

$S \leftarrow \emptyset$;

for every pair (P, Q) of polynomials in R **do**

 | add $S(P, Q)$ to S if Lemma 15, 16 or 17 applies to P and Q

end

return S ;

end

5 Exponentials

In this section, we explain how to encode exponentials of integral polynomials by introducing new indeterminates (called u_i and v_i). This proves useful in many examples we have treated. After presenting a small example illustrating the interest of exponentials, we present some lemmas allowing to encode the exponentials in an algebraic way. We introduce Algorithm 3 which checks if an exponential can be introduced, and present Algorithm 4 which updates the sets of exponentials used in the elimination algorithm (Algorithm 6) presented in the next section.

Example 12. Consider the following system (written in both differential and integral form) where x is unknown, and y is known

$$\begin{cases} \dot{x} = \theta x \\ \dot{y} = xy \end{cases} \iff \begin{cases} x = x_0 + \theta \int x \\ y = y_0 + \int xy. \end{cases} \quad (2)$$

System (2) admits the following differential I/O equation $-\theta^2 y + \theta \dot{y} - \dot{y}^2$. We believe (personal unpublished proof) that no integral I/O equation in y only can

be deduced from System (2). However, by (informally) introducing $u_1 = e^{\theta t}$, one obtains $x = x_0 u_1$ (by solving the equation in x) and (by a simple substitution)

$$y = y_0 + x_0 \int (u_1 y)$$

which is an integral I/O equation in y .

The following lemma will prove useful later to compute a kind of product of two polynomials.

Lemma 18. *Consider polynomials A_1, A_2, C_1 and C_2 in $\mathbb{K}[X^J]$ and two constants k_1 and k_2 in \mathbb{K} . Define $P_1 = A_1 - (k_1 + \int C_1)$ and $P_2 = A_2 - (k_2 + \int C_2)$. Then the polynomial*

$$A_1 \cdot A_2 - (k_1 k_2 + \int (C_1 \cdot A_2 + A_1 \cdot C_2))$$

belongs to the ideal generated by P_1 and P_2 .

Remark 3. Lemma 18 can be stated in the following less formal but more intuitive way using equalities. If $A_1 = k_1 + \int C_1$ and $A_2 = k_2 + \int C_2$, then $A_1 \cdot A_2 = k_1 k_2 + \int (C_1 \cdot A_2 + A_1 \cdot C_2)$.

Proof. To make the proof easier to follow, we use equalities and we actually prove Remark 3. Developing $A_1 \cdot A_2 = (k_1 + \int C_1)(k_2 + \int C_2)$ gives $A_1 \cdot A_2 = k_1 k_2 + k_1 \int C_2 + k_2 \int C_1 + \int (C_1 \cdot \int C_2) + \int (C_2 \cdot \int C_1)$. Using $\int C_1 = A_1 - k_1$ and $\int C_2 = A_2 - k_2$ in the previous expression yields $A_1 \cdot A_2 = k_1 k_2 + \cancel{k_1 \int C_2} + \cancel{k_2 \int C_1} + \int (C_1 \cdot A_2) - \cancel{k_2 \int C_1} + \int (C_2 \cdot A_1) - \cancel{k_1 \int C_2} = k_1 k_2 + \int (C_1 \cdot A_2 + A_1 \cdot C_2)$. \square

Example 13. If $u = 1 + \int(uG)$ and $v = 1 - \int(vG)$, then $uv = 1$ by Lemma 18.

The following Lemma 19 presented in an analysis context serves as a preparation for Lemma 20.

Lemma 19. *For any continuous function $f(t)$ on \mathbb{R} , denote by $I(f(t))$ the primitive of f that cancels at $t = 0$.*

Consider three real continuous functions $A(t), G(t)$ and $F(t)$ defined on \mathbb{R} . Assume that $A(t) = A(0) + I(A(t)G(t) + F(t))$. Let us introduce $u(t) = e^{I(G(t))}$ and $v(t) = e^{-I(G(t))}$. Then the following hold

$$A(t)v(t) = A(0) + I(v(t)F(t)) \tag{3}$$

$$A(t) = A(0)u(t) + u(t)I(v(t)F(t)). \tag{4}$$

Proof. This lemma is easy to prove using derivations: we have $A'(t) = A(t)G(t) + F(t)$, $u'(t) = G(t)u(t)$ and $v'(t) = -G(t)v(t)$. Equation (3) is satisfied at $t = 0$ since $v(0) = 1$. It suffices to prove that the derivative of (3) is equal to 0. We have $(A(t)v(t) - (A(0) + I(v(t)F(t))))' = A'(t)v(t) + A(t)v'(t) - v(t)F(t) = (A(t)G(t) + F(t))v(t) - A(t)G(t)v(t) - v(t)F(t) = 0$ which proves Equation (3). Since $u(t)v(t) = 1$, Equation (4) is proved by multiplying Equation (3) by $u(t)$. \square

The exponential $u(t) = e^{I(G(t))}$ in Lemma 19 can be encoded in an algebraic way by introducing the integral equation $u = 1 + \int(uG)$. The following Lemma 20 restates Lemma 19 in an algebraic context.

Lemma 20. *Take three polynomials A , G and F in $\mathbb{K}[X^J]$ and a constant A_0 in \mathbb{K} . Define $P = A - (A_0 + \int(A \cdot G + F))$.*

Introduce two new fresh indeterminates u and v (and add them to X), and consider the two polynomials $P_u = u - (1 + \int(uG))$ and $P_v = v - (1 - \int(vG))$.

Then the polynomials $Av - (A_0 + \int(vF))$ and $A - (A_0u + u \int(vF))$ belong to the integral ideal generated by P , P_u and P_v .

Proof. Applying Lemma 18 on P and P_u implies that $Av - (A_0 + \int(-vA \cdot G + v(A \cdot G + F))) = Av - (A_0 + \int(vF))$ is in the ideal generated by P and P_u . Applying Lemma 18 on P_u and P_v implies that $uv - 1$ is in the ideal generated by P_u and P_v . As a consequence, $u(Av - (A_0 + \int(vF))) - A(uv - 1) = A - (A_0u + u \int(vF))$ is in the ideal generated by P , P_u and P_v . \square

5.1 Algorithms `find_A_A0_G_F` and `update_exp`

Algorithm 3 takes an integral polynomial P and tries to write it in the form $P = A - (A_0 + \int(A \cdot G + F))$ where A_0 is in \mathbb{K} , A, G, F are in $\mathbb{K}[X^J]$, and A and G nonzero. When this succeeds, by Lemma 20, a new polynomial $A - (A_0u + u \int(vF))$ can be introduced where u and v are new indeterminates encoding $u = e^{\int G}$ and $v = e^{-\int G}$. Note the cases $A = 0$ and $G = 0$ are excluded since they do not introduce any useful equations.

Example 14. Let $P = x + y - 2 - \int xfy - \int yfy - \int y^2$. Algorithm 3 succeeds and returns $A = x + y$, $A_0 = 2$, $G = \int y$ and $F = y^2$.

Algorithm 4 takes as input a set T' of integral polynomials and a set E of triples (u_i, v_i, Q_i) . Algorithm 4 is used by Algorithm 6 which performs the integral elimination. The set T' is the set of generators under construction. The set E describes the exponential terms introduced so far, where u_i (resp. v_i) encodes e^{Q_i} (resp. e^{-Q_i}). Algorithm 4 mainly updates T' and E by finding new interesting exponentials.

6 Integral Elimination Prototype

We present Algorithm 6 which performs the integral elimination. It follows a classical Knuth-Bendix approach by completing a set T with critical pairs until the set T stabilises. An originality of Algorithm 6 is that exponential terms are detected and added during the process.

Our algorithm still has many flaws. We do not know whether it terminates or not. The reduction part and the detection of critical pairs are not exhaustive. Moreover, we miss a Composition-Diamond lemma like in [2,9]. As a consequence, our algorithm may miss some equations.

Algorithm 3: find_A_A0_G_F(P, X)

Input: a polynomial P in $\mathbb{K}[X^f]$
Result: FAIL or (A, A_0, G, F) such that $P = A - (A_0 + \int (A \cdot G + F))$ with A_0 is in \mathbb{K} , A, G, F are in $\mathbb{K}[X^f]$, and A and G nonzero.

```

begin
    // We only deal with a simple case where A is in K[X]
    try to write P as A - (A_0 + \int Q) where A is a nonzero polynomial in K[X]
    with no constant term, A_0 is in K and Q is in K[X^f];
    return FAIL if it is not possible;
    Write Q = \sum \alpha_i M_i with \alpha_i in K and M_i in X^f;
    // Since {A} is a Gröbner basis (for the ordering <_E), one can
    decompose each M_i_{[0]} in the following way:
    Decompose each M_i_{[0]} as M_i_{[0]} = q_i A + r_i where r_i is the normal form of
    M_i_{[0]} w.r.t. {A};
    G \leftarrow \sum \alpha_i q_i M_i_{[1+]} ; F \leftarrow \sum \alpha_i r_i M_i_{[1+]} ;
    if G = 0 then return FAIL;
    return (A, A_0, G, F);
end
    
```

However, our algorithm produces encouraging results (see Section 7). Moreover, it is to our knowledge the only implemented algorithm which performs integral elimination. Our algorithm has been coded in Python using SymPy.

We terminate this section by running Algorithm 6 on System (2) by taking $F = \{x - (x_0 + \theta \int x), y - (y_0 + \int xy)\}$, $X = x > y$ and $\mathbb{K} = \mathbb{Q}(\theta, x_0, y_0)$. We write the polynomials of T , T' , T_E and C as rewriting rules. The triplets (u_i, v_i, Q_i) will be written as $u_i = e^{Q_i}$.

After Line 6, we have

$$T' : \begin{cases} \theta \int x \xrightarrow{R_1} x - x_0 \\ \int xy \xrightarrow{R_2} y - y_0 \end{cases} \text{ and } X' = x > y.$$

After Lines 7 and 8, we have

$$T_E : \begin{cases} x \xrightarrow{R_3} x_0 u_1 \\ y_0 u_2 \xrightarrow{R_4} y \end{cases}, E : \begin{cases} u_1 = e^{\theta \int 1} \\ u_2 = e^{\int x} \end{cases} \text{ and } X' = u_2 > v_2 > x > y > u_1 > v_1.$$

The set T' gets enlarged with T_E at Line 9. After Line 10, we have

$$C : \begin{cases} x \rightarrow x_0 + \theta x_0 \int u_1 \\ -x_0 \int u_1 y \rightarrow y_0 - y. \end{cases}$$

After Line 11, T' is enlarged with the two new equations

$$\begin{cases} -\theta x_0 \int u_1 \xrightarrow{R_5} x_0 - x_0 u_1 \\ -x_0 \int u_1 y \xrightarrow{R_6} y_0 - y. \end{cases}$$

Algorithm 4: update_exp(T', E, X)

Input: T' set of integral polynomials of $\mathbb{K}[X^J]$;
 E set of triples (u_i, v_i, Q_i) in $X \times X \times \mathbb{K}[X^J]$
Result: T_E, E'

```

begin
   $T_E \leftarrow \emptyset$  ;  $E' \leftarrow E$  ;
  for every  $P \in T'$  do
    if find_AA0_G_F( $P, X$ ) succeeds and returns  $(A, A_0, G, F)$  then
      if  $\int G$  is different from all  $Q_i$  then
        // The following lines are based on Lemma 20, and
        encodes  $u_i = e^{\int G}$ ,  $v_i = e^{-\int G}$ 
        introduce two new variables  $u_k$  and  $v_k$  which are not in  $E'$  ;
        add  $(u_k, v_k, \int G)$  to  $E'$  ;
         $P_{\text{exp}} \leftarrow A - (A_0 u_k + u_k \int (v_k F))$  ;
        add  $P_{\text{exp}}$  to  $T_E$  ;
      end
    end
  return  $T_E, E'$ 
end

```

After Lines 12 and 13 (u_2 is reduced by R_1 then R_3), we have

$$E' : \begin{cases} u_1 = e^{\theta \int 1} \\ u_2 = e^{\frac{1}{\theta}(x_0 u_1 - x_0)} \end{cases} \text{ and } X' = x > y > u_2 > v_2 > u_1 > v_1.$$

Consequently, we have found the following two equations involving y and θ

$$y - y_0 u_2 = 0 \text{ and } y - y_0 - x_0 \int u_1 y = 0.$$

7 Examples

We present two examples handled by our implementation in Python. Both examples take less than two minutes on a computer equipped with a single CPU Ryzen 7 Pro 5875U. Note that the kind of integral I/O equations we obtain seem suitable for parameter estimation [14] even if they contain complicated exponential terms.

7.1 Intra-host Model of Malaria (taken from [1])

$$\begin{cases} \dot{x} = 1 - x - \beta x m \\ \dot{y} = \beta x m - y \\ \dot{m} = y - m - \beta x m \end{cases}$$

We assume that the only known indeterminate is y . As a consequence, we want to eliminate x and m in order to obtain an I/O equation in y and β . The

Algorithm 5: extend_X_with_exp(X, E)

Input: $X = [x_1, \dots, x_n]$ ordered by $x_1 > \dots > x_n$;
 $E = [(u_1, v_1, Q_1), \dots, (u_m, v_m, Q_m)]$ where the u_i and v_i are indeterminates,
 and where the Q_i are integral polynomials over $x_1, \dots, x_n, u_1, v_1, \dots, u_m, v_m$.
Result: a list X' ordered decreasingly of $n + 2m$ indeterminates obtained by
 inserting the u_i and v_i in X at the rightmost positions such that each
 u_i (resp. v_i) is greater than any variable in Q_i .

```

begin
  If needed, reorder  $E$  such that each  $Q_i$  does not involve any  $u_j$  or  $v_j$  with
   $j > i$ . This can be done by a topological sort. ;
   $X' \leftarrow X$  ;
  for  $i$  from 1 to  $m$  do
    if  $Q_i$  involves at least one indeterminate in  $X'$  then
      find the highest indeterminate  $w$  in  $Q_i$  w.r.t. the ordering on  $X'$  ;
      insert  $u_i$  and  $v_i$  just before  $w$  in the list  $X'$  ;
    else
      insert  $u_i$  and  $v_i$  at the end of  $X'$  ;
    end
  end
  return  $X'$  ;
end
    
```

differential I/O equation (computed with the Differential Algebra package [5])
 is

$$\begin{aligned}
 & 4\beta^2 y^3 \dot{y}^3 + 12\beta^2 y^2 \dot{y}^4 + 12\beta^2 y \dot{y}^5 + 4\beta^2 \dot{y}^6 - 12\beta^2 y^2 \ddot{y}^3 \\
 & - 24\beta^2 y \dot{y}^4 - 12\beta^2 \dot{y}^5 + 12\beta^2 y \ddot{y}^3 + 12\beta^2 \dot{y}^4 - 8\beta y^3 \dot{y}^2 + 4\beta y^3 \dot{y} \ddot{y} \\
 & + \beta y^3 \ddot{y}^2 - 32\beta y^2 \dot{y}^3 - 4\beta y^2 \dot{y}^2 \ddot{y} + 3\beta y^2 \dot{y} \ddot{y}^2 - 37\beta y \dot{y}^4 - 6\beta y \dot{y}^3 \ddot{y} \\
 & - 8\beta y \dot{y}^3 \ddot{y} + 6\beta y \dot{y}^2 \ddot{y}^2 - 13\beta \dot{y}^5 - 2\beta \dot{y}^4 \ddot{y} - 4\beta \dot{y}^4 \ddot{y} \\
 & + 4\beta \dot{y}^3 \ddot{y}^2 - 4\beta^2 \dot{y}^3 + 16\beta y^2 \dot{y}^2 - 8\beta y^2 \dot{y} \ddot{y} - 2\beta y^2 \ddot{y}^2 + 52\beta y \dot{y}^3 \\
 & + 10\beta y \dot{y}^2 \ddot{y} + 8\beta y \dot{y}^2 \ddot{y} - 6\beta y \dot{y} \ddot{y}^2 + 34\beta \dot{y}^4 + 14\beta \dot{y}^3 \ddot{y} \\
 & + 8\beta \dot{y}^3 \ddot{y} - 6\beta \dot{y}^2 \ddot{y}^2 - 8\beta y \dot{y}^2 + 4\beta y \dot{y} \ddot{y} + \beta y \ddot{y}^2 - 20\beta \dot{y}^3 \\
 & - 10\beta \dot{y}^2 \ddot{y} - 4\beta \dot{y}^2 \ddot{y} + 3\beta \dot{y} \ddot{y}^2 + 4y^3 \dot{y} - 4y^3 \ddot{y} + 20y^2 \dot{y}^2 - 4y^2 \dot{y} \ddot{y} \\
 & + 4y^2 \dot{y} \ddot{y} - 6y^2 \ddot{y}^2 - 2y^2 \dot{y} \ddot{y} + 29y \dot{y}^3 + 11y \dot{y}^2 \ddot{y} + 12y \dot{y}^2 \ddot{y} \\
 & - 5y \dot{y} \ddot{y}^2 + y \dot{y} \ddot{y} \ddot{y} + y \dot{y} \ddot{y}^2 - y \ddot{y}^2 \ddot{y} + 10y^4 + 2y^3 \dot{y} + 7y^3 \ddot{y} \\
 & - 7y^2 \dot{y}^2 + y^2 \dot{y} \ddot{y} + y^2 \ddot{y}^2 - y \dot{y}^3 - 2y \dot{y}^2 \ddot{y} \\
 & + \dot{y}^4 - 4y^2 \dot{y} + 4y^2 \ddot{y} - 20y \dot{y}^2 - 4y \dot{y} \ddot{y} + 10y \ddot{y}^2 + 2y \ddot{y} \ddot{y} - 25y^3 - 25y^2 \ddot{y} \\
 & - 10y^2 \ddot{y} + y \ddot{y}^2 - 5y \ddot{y} \ddot{y} - y \ddot{y}^2 + 4y^3 + y^2 \ddot{y} = 0
 \end{aligned} \tag{5}$$

Algorithm 6: `integral_elimination(F, X)`

Input: F a set of polynomials in $\mathbb{K}[X^J]$;
 X a list of known indeterminates ordered decreasingly.
Result: if the algorithm stops, it returns an encoding of exponentials E , and a set of polynomials T that generates an ideal containing the one generated by F . Moreover, F probably involves polynomials with small leading monomials.

```

1 begin
2    $T \leftarrow F$  ;  $E \leftarrow \emptyset$  ;  $X' \leftarrow X$  ;
3   finished  $\leftarrow$  false ;
4   while not(finished) do
5     auto reduce  $T$  ;
6      $T' \leftarrow T$  ;
7      $T_E, E' \leftarrow$  update_exp( $T', E, X'$ ) ;
8      $X' \leftarrow$  extend_X_with_exp( $X, E'$ ) ;
9      $T' \leftarrow T' \cup T_E$  ;
10     $C \leftarrow$  critical_pairs( $T', X'$ ) ;
11    For each  $Q \in C$ , add reduce( $Q, T', X'$ ) to  $T'$  if reduce( $Q, T', X'$ )  $\neq 0$  ;
12    replace in  $E'$  each  $(u_i, v_i, Q_i)$  by  $(u_i, v_u, \text{reduce}(Q_i, T', X'))$  ;
13     $X' \leftarrow$  extend_X_with_exp( $X, E'$ ) ;
14     $E \leftarrow E'$  ;
15    if  $T = T'$  then finished  $\leftarrow$  true;
16    else  $T \leftarrow T'$ ;
17  end
18  return ( $E, T$ ) ;
19 end
```

After introducing $u_1 = e^{-t}$ and $v_1 = e^t$, Algorithm 6 computes

$$\begin{aligned}
& -\beta m_0 \int (u_1 y) - \beta x_0 \int (u_1 y) - 2\beta y_0 \int (u_1 y) + \beta m_0 x_0 \int (u_1^2) \\
& \quad + \beta m_0 y_0 \int (u_1^2) + \beta y_0 x_0 \int (u_1^2) + \beta y_0^2 \int (u_1^2) + \beta \int (y^2) \\
& -\beta \int \left(u_1 y \int v_1 y \right) - \beta \int \left(u_1 y \int v_1 \right) + \beta x_0 \int \left(u_1^2 \int v_1 y \right) \\
& + \beta y_0 \int \left(u_1^2 \int v_1 y \right) + \beta m_0 \int \left(u_1^2 \int v_1 \right) + \beta y_0 \int \left(u_1^2 \int v_1 \right) \\
& + \beta \int \left(u_1^2 \int v_1 y \int v_1 \right) + \beta \int \left(u_1^2 \int v_1 \int v_1 y \right) + y_0 - y - \int y = 0.
\end{aligned} \tag{6}$$

The prototype was not able to find any I/O equations without introducing exponentials. A non trivial step during the computation consists in introducing an exponential thanks to an intermediate equation $y + m = y_0 + m_0 - \int m$ which produces $y + m = (y_0 + m_0)u_1 + u_1 \int v_1 y$. This last equation permits to rewrite m . The same process also occurs for the x variable. Rewriting x and m terminates the elimination process quickly.

7.2 SIWR Model - Cholera (taken from [7])

$$\begin{cases} \dot{S} = \mu - \beta_I SI - \beta_W Sw - \mu S + \alpha R \\ \dot{I} = \beta_W SW + \beta_I SI - \gamma I - \mu I \\ \dot{W} = \xi(I - W) \\ \dot{R} = \gamma I - \mu R - \alpha R \\ y = \kappa I \end{cases}$$

We assume that the only known indeterminate is y . As a consequence, we want to eliminate S , I , W and R in order to obtain an I/O equation in y and the parameters. A differential I/O equation is computed in [7] using the `StructuralIdentifiability` package in Julia. This equation is huge and is a sum of 209 350 terms containing $y(t)^{10}$, $\dot{y}(t)^{10}$, $\ddot{y}(t)^8$, $\dddot{y}(t)^5$, $\ddot{\ddot{y}}(t)^4$...

After introducing the following exponentials

$$\begin{aligned} -u_1 &= e^{-\zeta \int 1} \\ -u_2 &= e^{-(\alpha+\mu) \int 1} \\ -u_3 &= e^{-\left(\mu \int 1 + \frac{\beta_I}{\kappa} \int y + \frac{\beta_W}{\kappa} \int y - \frac{\beta_W}{\kappa} e^{-\zeta \int 1} \left(\int y e^{\zeta \int 1}\right) + \frac{\beta_W w(0)}{\zeta} - \frac{\beta_W w(0)}{\zeta} e^{-\zeta \int 1}\right)} \end{aligned}$$

and v_1, v_2, v_3 their respective inverses, Algorithm 6 computes 4 equations only involving y . The biggest of them involves all parameters and is equal to

$$\begin{aligned} & -\alpha\beta_I\gamma \int \left(u_3 y \int u_2 v_3 \int v_2 y\right) - \alpha\beta_I\kappa r_0 \int \left(u_3 y \int u_2 v_3\right) \\ & -\alpha\beta_W\gamma\kappa w_0 \int \left(u_1 u_3 \int u_2 v_3 \int v_2 y\right) - \alpha\beta_W\gamma\zeta \int \left(u_1 u_3 \int u_2 v_3 \int v_1 y \int v_2 y\right) \\ & -\alpha\beta_W\gamma\zeta \int \left(u_1 u_3 \int u_2 v_3 \int v_2 y \int v_1 y\right) - \alpha\beta_W\gamma\zeta \int \left(u_1 u_3 \int v_1 y \int u_2 v_3 \int v_2 y\right) \\ & -\alpha\beta_W\kappa^2 r_0 w_0 \int \left(u_1 u_3 \int u_2 v_3\right) - \alpha\beta_W\kappa\zeta r_0 \int \left(u_1 u_3 \int u_2 v_3 \int v_1 y\right) \\ & -\alpha\beta_W\kappa\zeta r_0 \int \left(u_1 u_3 \int v_1 y \int u_2 v_3\right) - \beta_I\kappa m u \int \left(u_3 y \int v_3\right) - \beta_I\kappa s_0 \int \left(u_3 y\right) \\ & -\beta_W\kappa^2 m u w_0 \int \left(u_1 u_3 \int v_3\right) - \beta_W\kappa^2 s_0 w_0 \int \left(u_1 u_3\right) - \beta_W\kappa m u \zeta \int \left(u_1 u_3 \int v_1 y \int v_3\right) \\ & -\beta_W\kappa m u \zeta \int \left(u_1 u_3 \int v_3 \int v_1 y\right) - \beta_W\kappa\zeta s_0 \int \left(u_1 u_3 \int v_1 y\right) \\ & + \gamma\kappa \int (y) - \kappa^2 i_0 + \kappa m u \int (y) + \kappa y = 0 \end{aligned}$$

Note that our I/O equations are much more compact than the differential ones.

Acknowledgments. This work has been partially supported by the THIA ANR Program « AI_PhD@Lille », from the French National Research Agency, grant ANR-20-THIA-0014.

References

1. Anderson, R., May, R., Gupta, S.: Non-linear phenomena in host–parasite interactions. *Parasitology* **99 Suppl**, S59–79 (02 1989). <https://doi.org/10.1017/S0031182000083426>

2. Bokut, L.A., Chen, Y., Deng, X.: Gröbner-shirshov bases for rota-baxter algebras. *Siberian Mathematical Journal* **51**(6), 978–988 (Nov 2010). <https://doi.org/10.1007/s11202-010-0097-1>
3. Boulier, F., Lallemand, J., Lemaire, F., Regensburger, G., Rosenkranz, M.: Additive Normal Forms and Integration of Differential Fractions. *Journal of Symbolic Computation* (2016), <https://hal.archives-ouvertes.fr/hal-01245378>
4. Boulier, F., Lemaire, F., Rosenkranz, M., Ushirobira, R., Verdière, N.: On Symbolic Approaches to Integro-Differential Equations, pp. 161–182. Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-38356-5_6, <https://hal.archives-ouvertes.fr/hal-01367138v2/file/BLRUV.pdf>
5. Boulier, F.: The DifferentialAlgebra project (2023), <https://codeberg.org/francois.boulier/DifferentialAlgebra>
6. Denis-Vidal, L., Joly-Blanchard, G., Noiret, C.: System identifiability (symbolic computation) and parameter estimation (numerical computation). In: *Numerical Algorithms*. vol. 34, pp. 282–292 (2003)
7. Dong, R., Goodbrake, C., Harrington, H.A., Pogudin, G.: Differential elimination for dynamical models via projections with applications to structural identifiability (2022)
8. Fliess, M.: Automatique et corps différentiels. *Forum Mathematicum - FORUM MATH* **1**, 227–238 (01 1989). <https://doi.org/10.1515/form.1989.1.227>
9. Gao, X., Guo, L.: Constructions of Free Commutative Integro-Differential Algebras. In: Barkatou, M., Cluzeau, T., Regensburger, G., Rosenkranz, M. (eds.) *Algebraic and Algorithmic Aspects of Differential and Integral Operators*. LNCS, vol. 8372, pp. 1–22 (2014)
10. Gao, X., Guo, L., Zhang, H.: Rota’s program on algebraic operators, rewriting systems and gröbner-shirshov bases (2021)
11. Gerdt, V.P., Lange-Hegermann, M., Robertz, D.: The MAPLE package TDDS for computing Thomas decompositions of systems of nonlinear PDEs. *Computer Physics Communications* **234**, 202–215 (2019). <https://doi.org/https://doi.org/10.1016/j.cpc.2018.07.025>
12. Guo, L.: An Introduction to Rota-Baxter Algebra (vol. 4 in the *Surveys of Modern Mathematics* series). International Press of Boston, Incorporated, paperback edn. (10 2012)
13. Kolchin, E.R.E.R.: *Differential algebra and algebraic groups*. Acad. Press (1973)
14. Lemaire, F., Roussel, L.: Parameter Estimation Using Integral Equations (2024), <https://hal.science/hal-04560846>, to appear in *Maple Transactions*
15. Noiret, C.: Utilisation du calcul formel pour l’identifiabilité de modèles paramétriques et nouveaux algorithmes en estimation de paramètres. Ph.D. thesis, Université de Technologie de Compiègne (2000)
16. Quadrat, A., Regensburger, G.: Computing Polynomial Solutions and Annihilators of Integro-Differential Operators with Polynomial Coefficients, pp. 87–114. Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-38356-5_3
17. Reutenauer, C.: *Free Lie Algebras*. Oxford University Press (05 1993). <https://doi.org/10.1093/oso/9780198536796.001.0001>
18. Ritt, J.: *Differential Algebra*, vol. 33. American Mathematical Society (1950). <https://doi.org/10.1090/coll/033>
19. Rosenkranz, M., Regensburger, G.: Integro-differential polynomials and operators. In: *ISSAC’08: Proceedings of the twenty-first international symposium on Symbolic and algebraic computation*. pp. 261–268. ACM, New York, NY, USA (2008), <http://doi.acm.org/10.1145/1390768.1390805>