COMBOS2: an algorithm to the input–output equations of dynamic biosystems via Gaussian elimination

Ali Kalami Yazdi a, Mehdi Nadjafikhah a and Joseph Distefano III b

a School of Mathematics, Iran University of Science and Technology, Tehran, Iran; b Biocybernetics Laboratory, Departments of Computer Science and Medicine, UCLA, Los Angeles, CA, USA

ABSTRACT
Differential algebra (DA) methods are currently being exploited for analyzing dynamic system models for their structural identifiability (SI) properties. An early step in this approach entails finding an equivalent input–output (I/O) model. A recent approach for finding these equations, based on Gröbner bases and embedded in the app COMBOS, is relatively slow and sometimes unsuccessful, even for moderate size models. In this paper, we propose a faster algorithm, using a Gaussian elimination approach for analyzing the subclass of linear dynamic system models. We apply this methodology to find the simplest set of globally SI parameter combinations of these models and show that it works effectively for linear biosystem models of simple to moderate complexity.

1. Introduction
Structural identifiability (SI) analysis deals with the problem of finding the solutions from input–output \((u(t), y(t))\) data [1, 2] for the unknown parameters \(p\) of a dynamic system model of the form (1) with state and output equations:

\[
\begin{align*}
\dot{x}(t, p) &= f(x(t, p), u(t), t; p), \quad t \in [t_0, T] \\
y(t, p) &= g(x(t, p), p),
\end{align*}
\]

where \(p\) is a \(k\)-dimensional parameter vector, \(x\) is an \(n\)-dimensional state vector, \(y\) is an \(m\)-dimensional output vector, \(u\) is an \(r\)-dimensional input vector, and \(f\) and \(g\) are rational polynomial functions of their arguments, a reasonable assumption in most applications [3]. Dynamic biosystem models typically have this form, with function \(f\) being polynomial, ratios of polynomials, or representative of linear compartmental biomodels in, e.g. tracer perturbation experiments on nonlinear biosystems.

Parameter \(p \in \mathbb{P}\) is said to be SI locally if there are finite number of distinct solutions for it, and globally SI if there is only one solution. It’s unidentifiable if there are uncountably number of solutions. Nevertheless, even when a model has unidentifiable parameters, finding useful combinations of parameters that are structurally identifiable is always possible. Furthermore, these combinations of parameters can often be utilized to reparameterize the original model into a structurally identifiable one. Practically speaking, SI is a prerequisite for finding parameter values (or values for parameter combinations) in a real “noisy” data problem, usually known as the numerical identifiability problem.

Differential algebra (DA) methods have been illustrated quite applicable in dealing with global and local SI properties of dynamic system models [4–7]; and many software tools have been elaborated and implemented using many differential algebra algorithms [7, 8], but, they are restricted by complexity of computational algebraic or other difficulties, and are then far limited in application to models of low to moderate complexity.

An early and critical step in DA identifiability analysis is finding the model input–output (I/O) equations, by first eliminating its state variables. To make the I/O equations monic, the coefficients \(c(p)\) of them are fixed uniquely by normalizing the Equation [7]. Structural identifiability is then recognized by checking the injectivity of the coefficients \(c(p)\), in other words, the model (1) is globally identifiable if and only if \(c(p) = c(p^*)\) implies \(p = p^*\) for arbitrary \(p^*\) [7]. Therefore, recognizing structural identifiability is facilitated to finding the solutions to \(c(p) = c(p^*)\) using many differential algebra algorithms.

Several methods are available for finding the I/O equations, including Ritt’s pseudodivision [9], a Gröbner Basis alternative to Ritt’s pseudodivision (GRA), developed by Meshkat et al. [3], and some other primary...
elimination methods [10–13]. All are encumbered by the computational algebraic complexity encountered in finding the I/O equations and are thus limited to relatively small to moderately complex models [14–16].

Gröbner Basis computations in current algorithmic approaches are a major encumbrance to solution speed, in some cases causing one symbolic algebra programme (COMBOS 1) to hang, thereby providing no solutions, or running on for several hours in another DA programme (DAISY 2) before providing SI solutions for a not-so-complex linear biosystem model (see Section 8 below). Gröbner Basis computations can be quite slow, even for relatively small problems; and increasing model complexity by even a small amount (the number of equations, equation terms, or unknown parameters) can have a great impact on speed and efficiency, especially for finding the I/O equations.

In this paper, we describe an algorithm that simplifies computation of and accelerates the speed of determining the I/O equations, limited only to linear ODE models. In brief, we use a Gaussian elimination approach directly applicable to linear biosystem and other ODE models [17–20], replacing the Gröbner Basis-based GRA Method in COMBOS for obtaining I/O equations when the ODEs are linear. We also extend some results associated with the GRA algorithm, finding a strict bound on the minimum number of derivatives of the model equations required for finding the I/O equations of linear ODEs. The new methodology is admittedly limited, but it is a step forward, because even some linear models present major encumbrances to ID analysis overcome by the updated algorithms in this new version of COMBOS.

2. Algorithm for finding the I/O equations

To obtain the I/O equations for (1), we need to first eliminate the state variables and their derivatives. For linear ODE models, if the number of equations (Neq) is more than the number of state variables and their derivatives (Nx), and at least two equations have dependent state variables, we can use one of them to eliminate the other, and can eliminate all state variables and their derivatives by repeating this step.

The condition (Neq ≥ Nx) ensures that at least one input–output equation is found. To find the input–output equations involving all output variables with the highest order, we also show and prove that condition Neq ≥ Nx + m is sufficient (m is the number of output equations). This does not work for nonlinear ODE models [3].

Finding the I/O equations is done in three steps: (1) Eliminate all state variables and their derivatives. (2) Assemble the coefficient matrix of the new system of equations. (3) Apply the Gaussian elimination algorithm to the coefficient matrix.

Step (1). In model (1), the equations \( y(t, p) = g(x(t, p); p) \) without derivatives of \( x \) is called the output equations. First take the derivative of the output equations, to eliminate the first derivative of \( x \) from the state variable equations, by back substitution. Every generated unknown variable then should exist in two equations of the new system. If \( \text{Neq} \geq \text{Nx} + m \), this additional equation is enough to eliminate the state variables, otherwise, the second derivative of the output equations is required. Differentiation of the corresponding state variable equations is continued until \( \text{Neq} \geq \text{Nx} + m \).

In the GRA algorithm, taking derivatives \((n−m) + 1\) times is sufficient. In Section 6, we show that this boundary is not always reached, and usually differentiating again generates more I/O equations.

Step (2). For using Gaussian elimination, we need to write all equations in matrix form, thereby assembling a coefficient matrix for the new system in which we can summarize all the information of the new system. This matrix must have as many columns as there are variables and their corresponding derivatives, and as many rows as equations. The order must be specific – first state variables and their derivatives, next output variables and input variables and their derivatives. The matrix is constructed one row at a time, writing a 1 for the coefficient if the corresponding variable is present and 0 (zero) if it is absent.

Step (3). Following Gaussian elimination of the coefficient matrix, we generate a matrix in reduced row-echelon form. Rows of the new matrix, in which elements corresponding to state variables and their derivatives are zero, are input–output equations coefficients.

Theorem 2.1: Consider a linear ODE system of the form (1) with \( m \) output equations. To find \( m \) I/O equations involving \( m \) output variables with highest order \( k \), (the number of differentiations), we must differentiate \( k \) times where the condition \( \text{Neq} \geq \text{Nx} + m \) is satisfied.

Proof: As shown by Meshkat [3], after differentiating \( k \) times, we have \( \text{Neq} \geq \text{Nx} + m \) for the new system. Since all equations of a linear ODE system of the form (1) are linearly independent, the new system remains linearly independent. Then, when we apply the Gaussian elimination algorithm on the new system, we obtain the reduced row-echelon matrix (Rrem) with non-zero elements shown in Figure 1. In each row of Rrem, the leading entry is the only non-zero entry in its column. In addition, \( \text{Neq} \geq \text{Nx} + m \), and we conclude that we have \( \text{Nx} + m \) rows involving

\[ \text{http://biocyb1.cs.ucla.edu/combos/} \]
\[ \text{http://daisy.dei.unipd.it/} \]
only leading entries that correspond to one member of the set \( \{x_1^{(k)}, x_2^{(k)}, \ldots, \dot{x}_1, \dot{x}_2, \ldots, y_1^{(k)}, \ldots, y_m^{(k)} \} \). Finally, to find \( m \) I/O equations involving \( m \) output variables with order \( k \), choose the \( m \) rows that correspond to \( \{y_1^{(k)}, \ldots, y_m^{(k)}\} \).

### 3. Linear compartment models and illustrative examples

We use linear compartment models [21,22] in several examples below to illustrate our novel algorithm for finding I/O equations and identifiable parameters combinations. We also compare this with results obtained using the GRA algorithm, where available.

\[
\begin{align*}
\dot{x}(t) &= K(p)x(t) + B(p)u(t), \\
y(t, p) &= C(p)x(t),
\end{align*}
\]

where \( u \) is an \( r \)-dimensional input vector, \( x \) is an \( n \)-dimensional state vector, \( y \) is an \( m \)-dimensional output vector, and \( K \) is an \( n \times n \) compartmental matrix with rate constants \( k_{ij} \) that satisfy:

\[
k_{ij} \geq 0 \quad i \neq j, \quad k_{ii} = -\sum_{j=0}^{n} k_{ij} \quad i \neq j,
\]

where \( k_{ij} \) is the transfer rate constant from compartment \( i \) to the external environment; \( B \) is an \( n \times r \) input matrix; \( C \) is an \( m \times n \) output matrix, \( p \) is a \( p \)-dimensional parameter vector containing the \( K, B \) and \( C \) parameters (a positive subset of the real numbers).

### 4. 2-Compartment model example

SI analysis of a simple 2-compartment model (Figure 2) demonstrates all of the steps in our new algorithm.

\[
\dot{x}_1 = -(k_{01} + k_{21})x_1 + k_{12}x_2 + u,
\]

Figure 1. Reduced row-echelon form of a linear ODE system of the form (1) with \( m \) output equations. Rows \( Nx + 1 \) to \( Nx + m \) in this matrix shows the coefficients of the I/O equations.

Figure 2. 2-Compartment ODE Model. \( u \) is the input to compartment \( x_1 \), the dashed line with a circle is the output inform compartment \( x_2 \). The same notation will be used in the following figures.

\[
x_2 = k_{21}x_1 - (k_{02} + k_{12})x_2, \quad y = x_1/V_1.
\]

We first use GRA and then our algorithm for finding the I/O equations and show that the results are the same. Then we use the Meshkhat algorithm [3] for finding identifiable parameters combinations (combos) and reparametrize the model in terms of these combos. There are \( n = 2 \) state variables \( \{x_1, x_2\} \), \( m = 1 \) output equation \( y = x_1/V_1 \) and \( \text{Neq} = 3, \text{Nx} = 4 \). Then, according to the GRA Algorithm, taking derivatives \( n - (m - 1) = 2 - (1 - 1) = 2 \) times is sufficient to find the input–output equation.

1: We first differentiate \( x_1 - V_1y \) (output equation) and then add the result to the system (\( \text{Neq} = 4, \text{Nx} = 4 \)):

\[
\dot{x}_1 - V_1 \dot{y},
\]

2: Differentiate \( \dot{x}_1 - V_1 \dot{y} \) to obtain \( \ddot{y} \), and the corresponding state variable equation to obtain \( \dot{x}_1 \) and then add the result to the system (\( \text{Neq} = 6, \text{Nx} = 5 \)):

\[
\dot{x}_1 - V_1 \ddot{y}, \quad -\dot{x}_1 - (k_{01} + k_{21})x_1 + k_{12}x_2 + \dot{u},
\]
algorithm and the ranking to eliminate all unknowns. According to the GRA Algorithm, taking derivatives derivative (Nx).

Now we investigate the parameter identifiability of variables and their derivatives, \( x_1, x_2, x_1, x_2, x_1 \), enough to eliminate all unknowns. According to the GRA algorithm and the ranking \( x_1, x_2, x_1, x_2, x_1, y, y, \dot{u}, y, u \), we find the Gröbner basis of it using Mathematica. Then I/O equation is an equation only in \( \{ \dot{y}, y, \dot{u}, y, u \} \) of the following form:

\[
- (k_{o2} + k_{o1}) u + (k_{o1}k_{o2} + k_{o1}k_{o2} + k_{o2}k_{o1}) V_1 y \\
- \dot{u} + (k_{o1} + k_{o2} + k_{o1}k_{o2} + k_{o2}k_{o1}) V_1 \dot{y} + V_1 \ddot{y} = 0, \] (7)

The coefficient matrix for model (4) is the following form:

\[
\begin{bmatrix}
0 & 0 & -1 & k_{o2} & -k_{o1} - k_{21} & 0 & 0 & 0 & 0 & 1 \\
0 & -1 & 0 & -k_{o2} - k_{12} & k_{21} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & V_1 & 0 \\
0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-k_{12} & -k_{o1} - k_{21} & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
\end{bmatrix}
\]

Figure 3. Coefficient matrix of 2-compartment model.

\[
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & (k_{o1} + k_{o2} + k_{o1}k_{o2} + k_{o2}k_{o1}) V_1 \\
0 & 1 & 0 & 0 & 0 & 0 & (k_{o1} + k_{o2} + k_{o1}k_{o2} + k_{o2}k_{o1}) V_1 \\
0 & 0 & 1 & 0 & 0 & 0 & -V_1 \\
0 & 0 & 0 & 1 & 0 & 0 & -V_1 \\
0 & 0 & 0 & 0 & 1 & 0 & -V_1 \\
0 & 0 & 0 & 0 & 0 & 1 & -V_1 \\
\end{bmatrix}
\]

Figure 4. Reduced row-echelon form of 2-compartment model. The last row shows the coefficients of the I/O equations of this model.

The new system has 6 equations in 5 unknown state variables and their derivatives, \( x_1, x_2, x_1, x_2, x_1 \), enough to eliminate all unknowns. According to the GRA algorithm and the ranking \( x_1, x_2, x_1, x_2, x_1, y, y, \dot{u}, y, u \), we find the Gröbner basis of it using Mathematica. Then I/O equation is an equation only in \( \{ \dot{y}, y, \dot{u}, y, u \} \) of the following form:

\[
- (k_{o2} + k_{o1}) u + (k_{o1}k_{o2} + k_{o1}k_{o2} + k_{o2}k_{o1}) V_1 y \\
- \dot{u} + (k_{o1} + k_{o2} + k_{o1}k_{o2} + k_{o2}k_{o1}) V_1 \dot{y} + V_1 \ddot{y} = 0, \] (8)

Multiplying Equation (8) by \( V_1 \), we see our I/O equation is to the same as the GRA algorithm I/O Equation (7). Now we investigate the parameter identifiability of the system (4) using Gröbner bases for the solutions to \( c(\rho) = c(\rho^*) \) and show \( V_1, k_{12}k_{21} - k_{o2} - k_{12}, -k_{o1} - k_{21} \) are uniquely identifiable. Finally, the most appropriate identifiable combinations are \( q_1 = -k_{o1} - k_{21}, q_2 = k_{12}k_{21}, q_3 = -k_{o2} - k_{12}, q_4 = V_1 \). Let \( x_2 = k_{21}x_2 \). Then the reparameterized equations are:

\[
\begin{align*}
x_1 &= q_1 x_1 + q_2 x_2 + u, \\
\dot{x}_2 &= x_1 + q_3 x_2, \\
y &= x_1/q_4. \quad (9)
\end{align*}
\]

5. 3-Compartment model example

We analyse the 3-compartment model with one output equation shown in Figure 5, to show that we cannot find I/O equations when the number of equations (Neq) is equal to the number of state variables and their derivative (Nx).

\[
\begin{align*}
\dot{x}_1 &= -(k_{21} + k_{31}) x_1 + k_{12} x_2 + k_{13} x_3 + u, \\
\dot{x}_2 &= k_{21} x_1 - (k_{o2} + k_{12}) x_2, \\
\dot{x}_3 &= k_{31} x_1 - (k_{o3} + k_{13}) x_3, \\
y &= x_1/V_1. \quad (10)
\end{align*}
\]

Here we have \( n = 3 \) state variables \( \{x_1, x_2, x_3\} \), \( m = 1 \) output equation \( y = x_1/V_1 \) and Neq = 4, Nx = 6. Then, according to the GRA Algorithm, taking derivatives \( n - (m - 1) = 3 - (1 - 1) = 3 \) times is sufficient to find the input–output equation.
1. Differentiate $x_1 - V_1y$ and then add the result to the system (Neq = 5, Nx = 6):

$$\dot{x}_1 - V_1\dot{y},$$  \hspace{1cm} (11)

2. Differentiate $x_1 - V_1\dot{y}$ to obtain $\dot{y}$, and the corresponding state variable equation to obtain $x_1$ and then add the result to the system (Neq = 7, Nx = 7):

$$\dot{x}_1 - V_1\dot{y},$$

$$-\dot{x}_1 - (k_{21} + k_{31})\dot{x}_1 + k_{12}\dot{x}_2 + k_{13}\dot{x}_3 + \ddot{u},$$

$$-\dot{x}_2 + k_{21}\dot{x}_1 - (k_2 + k_{12})\dot{x}_2,$$

$$-\dot{x}_3 + k_{31}\dot{x}_1 - (k_3 + k_{13})\dot{x}_3.$$  \hspace{1cm} (12)

The new system now has 7 equations in 7 unknown state variables and their derivatives.

3. Differentiate $x_1 - V_1\dot{y}$ to obtain $\ddot{y}$, and the corresponding state variable equation to obtain $x_1$, $\ddot{x}_2$, and $\ddot{x}_3$. Then add the result to the system (Neq = 11, Nx = 10):

$$\ddot{x}_1 - V_1\ddot{y},$$

$$-\ddot{x}_1 - (k_{21} + k_{31})\ddot{x}_1 + k_{12}\ddot{x}_2 + k_{13}\ddot{x}_3 + \dddot{u},$$

$$-\ddot{x}_2 + k_{21}\ddot{x}_1 - (k_2 + k_{12})\ddot{x}_2,$$

$$-\ddot{x}_3 + k_{31}\ddot{x}_1 - (k_3 + k_{13})\ddot{x}_3.$$  \hspace{1cm} (13)

The new system now has 11 equations in 10 unknown state variables and their derivatives, $\{x_1, x_2, x_3, \dot{x}_1, \dot{x}_2, \dot{x}_3, x_1, x_2, x_3, x_1, \ddot{x}_1, \ddot{x}_2, \ddot{x}_3, y, \dddot{u}, \dot{u}, y, u\}$, enough to eliminate all unknowns. According to the GRA algorithm and the ranking $\{x_1, x_3, x_2, \dot{x}_1, \dot{x}_2, \dot{x}_3, x_2, x_3, x_1, \ddot{x}_1, \ddot{x}_2, \ddot{x}_3, y, \dddot{u}, \dot{u}, y, u\}$, we find the Gröbner Basis of it using Mathematica. Then the I/O equation is:

$$-(k_{02}k_{03} + k_{03}k_{12} + k_{02}k_{13} + k_{12}k_{13})u + (k_{02}k_{03}k_{21} + k_{02}k_{13}k_{21} + k_{03}k_{12}k_{31})V_1y - (k_{02} + k_{03} + k_{12} + k_{13})\dddot{u} - (k_{02}k_{03} + k_{03}k_{12} + k_{02}k_{13} + k_{03}k_{12}k_{21} + k_{03}k_{12}k_{31} + k_{02}k_{31} + k_{03}k_{31} + k_{12}k_{31})V_1\dot{y} + (k_{02} + k_{03} + k_{12} + k_{13} + k_{21} + k_{31})V_1y + V_1\dddot{y} = 0$$  \hspace{1cm} (14)

The coefficient matrix for (10) is then:

$$\begin{bmatrix}
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, k_{02} + k_{03} + k_{12} + k_{13} + k_{21} + k_{31}, k_{02}k_{03} + k_{03}k_{12} + k_{02}k_{13} + k_{03}k_{12}k_{21} + k_{03}k_{12}k_{31} + k_{02}k_{31} + k_{03}k_{31} + k_{12}k_{31}, -1/V_1, 1, k_{02}k_{03}k_{21} + k_{02}k_{13}k_{21} + k_{03}k_{12}k_{31} + k_{03}k_{13}k_{31} + k_{02}k_{31} + k_{03}k_{31} + k_{12}k_{31} + k_{13}k_{31} -1/V_1 \end{bmatrix}$$  \hspace{1cm} (15)

Multiplying this row with variables, we obtain the I/O equation. This equation is the same as that obtained with the GRA algorithm (14), after multiplying by $V_1$.

We then use Gröbner bases to find the solutions to $c(p) = c(p^*)$ and show $V_1, k_{21} + k_{31}$ are uniquely identifiable and $k_{02} + k_{12}, k_{03} + k_{13}, k_{12}k_{21}, k_{13}k_{31}$ are locally identifiable with $2$ (two) solutions. Finally, the simplest identifiable combinations $\{q_1 = -k_{21} - k_{31}, q_2 = k_{21}k_{31}, q_3 = k_{13}k_{31}, q_4 = -k_{02} - k_{12}, q_5 = -k_{03} - k_{13}, \}$

Figure 5. 3-Compartment model.

Figure 6. Coefficient matrix of 3-compartment model.
\( q_6 = V_1 \) are choosen so all model terms are positive after reparametrizing. Let \( x_2 = k_{21}X_2, x_3 = k_{31}X_3 \). Then the reparameterized equations are:

\[
\begin{align*}
\dot{x}_1 &= q_1X_1 + q_2X_2 + q_3X_3 + u, \\
\dot{x}_2 &= x_1 + q_4X_2, \\
\dot{x}_3 &= x_1 + q_5X_3, \\
y &= x_1/q_6.
\end{align*}
\]  

(16)

6. 4-Compartment thyroid system model example

The model shown in Figure 7, which represents the linear dynamics of two thyroid hormones (T3 and T4) in blood (1 and 2) and tissues (3 and 4), with conversion of T4 into T3 in the lumped tissue compartment. It has one input and two output. For this example, we find sufficient input–output equations for finding identifiable combinations, with fewer derivatives than with the GRA algorithm.

\[
\begin{align*}
\dot{x}_1 &= -k_{31}x_1 + k_{13}x_3 + u, \\
\dot{x}_2 &= -k_{24}x_2 + k_{24}x_4, \\
\dot{x}_3 &= k_{31}x_1 - (k_{03} + k_{13} + k_{43})x_3, \\
\dot{x}_4 &= k_{42}x_2 + k_{43}x_3 - (k_{04} + k_{24})x_4, \\
y_1 &= x_1, \\
y_2 &= x_2.
\end{align*}
\]  

(17)

We use the same method as above, but the minimum bound for the number of taking derivatives is 2 using our algorithm, while it is 3 in GRA algorithm. There are \( n = 4 \) state variables \( \{x_1, x_2, x_3, x_4\} \) and \( m = 2 \) output equations \( \{y_1 = x_1, y_2 = x_2\} \). Then, according to Meshkat [14], taking derivatives \( n - (m - 1) = 4 - (2 - 1) = 3 \) times is sufficient for finding the I/O equation.

1: Differentiate \( y_1 - x_1, y_2 - x_2 \) (output equations) and then add the result to the system (\( \text{Neq} = 8, \text{Nx} = 8 \)):

\[
\begin{align*}
\dot{y}_1 &= x_1, \\
\dot{y}_2 &= x_2.
\end{align*}
\]  

(18)

The number of equations is still not enough to eliminate all unknowns.

2: Differentiate \( y_1 \) and \( y_2 \) to obtain \( \dot{y}_1 \) and \( \dot{y}_2 \), and the corresponding state variable equation to obtain \( \dot{x}_1 \) and \( \dot{x}_2 \). Then add the result to the system (\( \text{Neq} = 12, \text{Nx} = 10 \)):

\[
\begin{align*}
\dot{y}_1 &= \dot{x}_1, \\
\dot{y}_2 &= \dot{x}_2, \\
-\dot{x}_1 &= k_{31}x_1 + k_{13}x_3 + u, \\
-\dot{x}_2 &= k_{42}x_2 + k_{24}x_4.
\end{align*}
\]  

(19)

Then new system now has 12 equations in 10 unknown state variables and their derivatives, \( \{x_1, x_2, x_3, x_4, x_1, x_2, x_3, x_4, x_1, x_2\} \), enough to eliminate all unknowns. So, we do not need to take another derivative and the algorithm terminates here. According to the GRA algorithm and the ranking \( \{x_2, x_1, x_4, x_3, x_2, x_1, x_4, x_3, x_2, x_1, y_1, y_2, y_1, u, y_2, y_1, u\} \), we find the Gröbner Basis of it using Mathematica. Then I/O equations have the following forms:

1) \( k_{24}k_{43}u - k_{24}k_{31}k_{43}y_1 + k_{04}k_{13}k_{24}y_2 - k_{24}k_{43}\dot{y}_1 \\
+ (k_{04}k_{13} + k_{13}k_{24} + k_{13}k_{42})y_2 + k_{13}\dot{y}_2 = 0 \\
2) -(k_{03} + k_{13} + k_{43})u + (k_{03}k_{31} + k_{31}k_{43})y_1 - \dot{u} \\
+ (k_{03} + k_{13} + k_{31} + k_{43})y_1 + \dot{y}_1 = 0
\]  

(20)

Coefficient matrix for model (17) is:

We then use Gaussian elimination and acquire a reduced row-echelon form (not shown, too big). We select two rows with 0 (zero) elements corresponding to state variables and their derivatives (Figure 8).

1) \( \{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, k_{04} + k_{24} + k_{42}, \\
- \frac{k_{24}k_{43}}{k_{13}}, 0, k_{04}k_{24} - k_{24}k_{43} - k_{24}k_{43} \frac{k_{13}k_{43}}{k_{13}} \frac{k_{24}k_{43}}{k_{13}} \}
\)  

2) \( \{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, k_{03} + k_{13} + k_{31} \\
+ k_{43}, -1, 0, k_{03}k_{31} + k_{31}k_{43} - k_{03} - k_{13} - k_{43} \}
\)

Multiplying these rows with variables, we obtain two I/O equations:

1) \( k_{24}k_{43}u - \frac{k_{24}k_{31}k_{43}}{k_{13}}y_1 + k_{04}k_{42}y_2 - \frac{k_{24}k_{43}}{k_{13}}\dot{y}_1 \\
+ (k_{04} + k_{24} + k_{42})\dot{y}_2 + \dot{y}_2 = 0 \\
2) -(k_{03} + k_{13} + k_{43})u + (k_{03}k_{31} + k_{31}k_{43})y_1 \\
- \dot{u} + (k_{03} + k_{13} + k_{31} + k_{43})y_1 + \dot{y}_1 = 0
\)

(20)

Our input–output equations are the same as GRA algorithm I/O Equation (20). Finally, we investigate parameter identifiability of the system (17) using Gröbner bases for solutions to \( c(p) = c(p^+\}) \) and show \( k_{13}, k_{31}, k_{03} + k_{43}, k_{04}k_{42}, k_{24}k_{43} \) and \( k_{04} + k_{24} + k_{42} \) are uniquely identifiable.

![Figure 7. 4-Compartment model.](image-url)
7. Evaluation of the third derivative

We investigate the result of obtaining a third derivative using the GRA algorithm. We show that the third derivative gives more input–output equations and, consequently, more coefficients. But identifiable parameters combination results are the same as using the Meshkat algorithm [3].

Let’s explore the relationships among the coefficients of the I/O equations obtained from the third and second steps, which do not change the results. A coefficient table of these input–output equations for the second derivative are:

We did not repeat the same coefficients because they do not affect the exhaustive summary or the Gröbner Basis.

For the third derivative, we differentiate \( y'_1 \) and \( y'_2 \) to obtain \( y''_1 \) and \( y''_2 \), and the corresponding state variable equation to acquire \( x''_1, x''_2, x''_3 \) and \( x''_4 \). Add these to the system (Neq = 18, Nx = 14):

\[
\begin{align*}
\ddot{y}_1 & = \ddot{x}_1, \\
\ddot{y}_2 & = \ddot{x}_2, \\
-\dot{x}_1 & = k_{31}x_1 + k_{13}x_3 + u, \\
-\dot{x}_2 & = k_{42}x_2 + k_{24}x_4, \\
-\dot{x}_3 & = k_{31}x_1 - (k_{03} + k_{13} + k_{43})x_3, \\
-\dot{x}_4 & = k_{42}x_2 + k_{43}x_3 - (k_{04} + k_{24})x_4, \\
\end{align*}
\]

(21)

Using the GRA algorithm and according to the ranking (\( \dot{x}_2, \dot{x}_1, \dot{x}_4, \dot{x}_3, \dot{x}_2, \dot{x}_1, \dot{x}_4, \dot{x}_3, \dot{x}_2, \dot{x}_1, \dot{x}_4, \dot{x}_3, \dot{x}_2, \dot{x}_1, \dot{y}_2, \dot{y}_1, \dot{y}_1, \ddot{u}, \ddot{y}_2, y_1, y_1, u, \dot{u}, \dot{y}_2, y_1, \dot{y}_1, \ddot{u}, \ddot{y}_2, y_1, y_1, u), \) we find the Gröbner Basis of the new system. Then I/O equations are:

1) \( k_{24}k_{43}u - k_{24}k_{31}k_{43}y_1 + k_{42}k_{13}k_{42}y_2 - k_{24}k_{43}y_1 + (k_{04}k_{13} + k_{13}k_{24} + k_{13}k_{42})y_2 + k_{13}y_2 = 0 \)

2) \( - (k_{03} + k_{13} + k_{43})y_1 + (k_{03}k_{31} + k_{31}k_{43})y_1 + y_1 = 0 \)

3) \( (k_{03}^2 + 2k_{03}k_{13} + k_{13}^2 + k_{31}k_{43} - k_{13}k_{43} + 2k_{03}k_{43} + k_{43})u \)

\( - (k_{03}k_{31} + k_{31}k_{43} + k_{43} - k_{13}k_{43} - k_{13}k_{43} - k_{03}k_{43} + k_{03}k_{43} + k_{43})y_1 + k_{31}u - \ddot{u} \)

\( - (k_{03}^2 + 2k_{03}k_{13} + k_{13}^2 + k_{31}k_{43} + 2k_{13}k_{43} + k_{31}k_{43} + k_{43})y_1 + y_1 = 0 \)

4) \( - (k_{24}k_{43}) + (k_{03}k_{42} + k_{43})y_2 \)

\( + (k_{24}k_{43})y_1 + (k_{03}k_{42} + k_{43})y_2 + (k_{03} + k_{43})y_2 + (k_{03} + k_{43})y_2 = 0 \)

(22)

Here we summarize some differences in the procedure of finding identifiable parameter combinations using the second and third derivative. For the second derivative, we have equations (Neq = 12) and state variables and their derivatives (Nx = 10), while for three, Neq = 18, Nx = 14. Because of this, use of the Gröbner Basis or other methods becomes more computationally complex. Also we have 2 (two) input–output equations after two differentiations. After three, we have 4 input–output equations two of which are the same as after two differentiations. The two other input–output equations do not effect the final answer and can be eliminated. We should show that eliminating their coefficient does not effect the exhaustive summary and Gröbner Basis. We use a coefficient table of input–output equations, where we write distinct coefficients, because
repeating a coefficient does not effect the exhaustive summary and Gröbner Basis.

**Theorem 7.1:** Let $a_1, a_2, \ldots, a_n, b_1, b_2, \ldots, b_m, p_1, p_2, \ldots, p_{n+m} \in \mathbb{R} - \{0\}$, and $n, m \in \mathbb{N}$, then solution set, \(\{a_1 a_2 \cdots a_n b_1 b_2 \cdots b_m = p_1 p_2 \cdots p_{n+m}, a_1 = p_1, a_2 = p_2, \ldots, a_n = p_n\}\) is equal to solution set, \(\{b_1 b_2 \cdots b_m = p_{n+1} p_{n+2} \cdots p_{n+m}, a_1 = p_1, a_2 = p_2, \ldots, a_n = p_n\}\).

\[
\begin{align*}
|a_1 a_2 \cdots a_n b_1 b_2 \cdots b_m &= p_1 p_2 \cdots p_{n+m} \\
a_1 &= p_1 \\
a_2 &= p_2 \\
&\vdots \\
a_n &= p_n
\end{align*}
\]

\[
\begin{align*}
b_1 b_2 \cdots b_m &= p_{n+1} p_{n+2} \cdots p_{n+m} \\
a_1 &= p_1 \\
a_2 &= p_2 \\
&\vdots \\
a_n &= p_n
\end{align*}
\]

\(\iff\)

\(\alpha \in \mathbb{R}^n\)

**Proof:** By induction.

**Theorem 7.2:** Let $a_1, a_2, \ldots, a_n, b_1, b_2, \ldots, b_m, p_1, p_2, \ldots, p_{n+m} \in \mathbb{R} - \{0\}$, and $n, m \in \mathbb{N}$, then solution set, \(\{a_1 + a_2 + \cdots + a_n + b_1 + b_2 + \cdots + b_m = p_1 + p_2 + \cdots + p_{n+m}, a_1 = p_1, a_2 = p_2, \ldots, a_n = p_n\}\) is equal to solution set, \(\{b_1 + b_2 + \cdots + b_m = p_{n+1} + p_{n+2} + \cdots + p_{n+m}, a_1 = p_1, a_2 = p_2, \ldots, a_n = p_n\}\).

\[
\begin{align*}
a_1 + a_2 + \cdots + a_n + b_1 + b_2 + \cdots + b_m &= p_1 + p_2 + \cdots + p_{n+m} \\
a_1 &= p_1 \\
a_2 &= p_2 \\
&\vdots \\
a_n &= p_n
\end{align*}
\]

\[
\begin{align*}
b_1 + b_2 + \cdots + b_m &= p_{n+1} + p_{n+2} + \cdots + p_{n+m} \\
a_1 &= p_1 \\
a_2 &= p_2 \\
&\vdots \\
a_n &= p_n
\end{align*}
\]

**Proof:** By induction.

**Theorem 7.3:** Let $A, B$ and $C$ be subsets of the exhaustive summary in linear model, if $A \cup B = C$ and $A \cong A', B \cong B'$, then $C \cong C' = A' \cup B'$ (The symbol $\cong$ means that two sets have the same distinct solutions).

**Proof:** Let $S_A, S_A', S_B, S_B', S_C, S_C'$ denote the sets of solutions of $A, A', B, B', C$ and $C'$ respectively. We should show $S_C \subseteq S_{C'}$ and $S_C \supseteq S_{C'}$. For $S_C \subseteq S_{C'}$, Suppose $s \in S_C$. Then $s \in S_{A'}$ or $s \in S_{B'}$; if it is in $S_{A'}$, then by relation $A \cong A'$, we have $s \in S_{A'} \subseteq S_C \Rightarrow S_C \subseteq S_{C'}$, by symmetry $s \in S_{B'}$ leads to the same conclusion. We also can show $S_C \supseteq S_{C'}$ by using this method.

Theorems 7.1, 7.2 and 7.3 are true for Gröbner Basis of the same model, that is, we can perform Gröbner Basis on each side of above expressions without changing the equality [23]. For theorem 7.1, we need to get the coefficient domain of polynomials as rational functions instead of integers numbers to simplify the solutions. We use these theorems for comparing between the obtained coefficients of I/O equations by taking derivatives two times (Table 1) and three times (Table 2).

We introduce a method to show the relation between the coefficients in Table 1 ($T_1$) and Table 2 ($T_2$).

First, we specify each coefficient according to its position number in $T_1$ or $T_2$. For example, the relation number of coefficient $k_{03} + k_{13} + k_{43}$ is 7 in $T_1$. Second, the coefficient in each table may have a different relation number. For example, the relation number of coefficient $k_{03} + k_{13} + k_{31} + k_{43}$ is 9 in $T_1$ and 11 in $T_2$.

Third, Suppose coefficient $\alpha$ is in $T_2$, but not in $T_1$. To find relation number in $T_1$, we find a subset $A$ including $\alpha$ of $T_2$ and a subset $B$ of $T_1$ such that $A \cong B$. That is:

\[
\begin{align*}
\exists \alpha &\in A \subset T_1 \\
B &\subset T_2 \\
\text{Such that } A &\cong B
\end{align*}
\]

The relation number of $\alpha \in A$ in $T_1$ is equal to all the relation number of the coefficients in $B$. For example

\[
\begin{align*}
\{k_{03} + k_{13} + k_{43}(7), k_{13}(2)\} \cong \{k_{03} + k_{43}(2,7), k_{13}(2)\} \\
\{k_{03} + k_{43}(2,7), k_{24}k_{43}(3)\} \\
\cong \{k_{04}k_{42} + k_{04}k_{42}k_{43}(2,3,7), k_{24}k_{43}(3)\}
\end{align*}
\]

Then the relation number of $k_{03}k_{04}k_{42} + k_{04}k_{42}k_{43}$ is 2, 3, 7 in $T_1$. The relation number is not unique and we use them only to show related coefficients do not effect the final answer and can be eliminated. The best way to find these subsets is to write the coefficients in $T_2$.

Table 1. Coefficients of the I/O Equations for the second derivative.

| Coefficients | No. in $T_2$ | No. in $T_1$ |
|--------------|--------------|--------------|
| $k_{03}$     | 1            | 1            |
| $k_{13}$     | 2            | 2            |
| $k_{24}$     | 3            | 3            |
| $k_{04}k_{42}$ | 5            | 4            |
| $k_{24}k_{43}$ | 6            | 5            |
| $k_{03}k_{31} + k_{31}k_{43}$ | 7            | 6            |
| $k_{03} + k_{13} + k_{43}$ | 8            | 7            |
| $k_{04}k_{31} + k_{31}k_{43} + k_{13}k_{42}$ | 10           | 8            |
| $k_{03} + k_{13} + k_{31} + k_{43}$ | 11           | 9            |

Note: No. in $T_1$ is the position of the coefficient in Table 1, No. in $T_2$ is the position of the coefficient in Table 2.
as a combination of coefficients in $T_1$. Some of these combinations are in the following forms:

\[
k_{03}k_{04} + k_{03}k_{24} + k_{03}k_{42} + k_{04}k_{42} + k_{04}k_{43} + k_{24}k_{43} + k_{24}k_{42} + k_{24}k_{43} = (k_{03} + k_{43})(k_{04} + k_{24} + k_{42}) + k_{04}k_{42}
\]
\[
k_{02} + 2k_{03}k_{13} + k_{13}^2 + k_{03}k_{31} + k_{13}k_{31} + 2k_{03}k_{43} + 2k_{13}k_{43} + k_{31}k_{43} + k_{43}^2 = (k_{03} + k_{13} + k_{43})(k_{03} + k_{13} + k_{43}) + k_{03}k_{31}
\]
\[
k_{02} + 2k_{03}k_{13} + k_{13}^2 + k_{03}k_{31} + 2k_{13}k_{31} + k_{31}^2 + 2k_{03}k_{43} + 2k_{13}k_{43} + k_{31}k_{43} + k_{43}^2 = (k_{03} + k_{13} + k_{43})(k_{03} + k_{13} + k_{43}) + k_{03}k_{31}
\]
\[
k_{03}k_{13} + 2k_{03}k_{13} + 2k_{03}^2 + 2k_{03}k_{31} + 2k_{03}k_{43} + k_{13}k_{43} + k_{31}k_{43} + k_{43}^2 = k_{31}k_{03} + k_{43}k_{03} + k_{31}k_{43} + k_{43}^2
\]

\[\text{(24)}\]

8. Linear thyroid hormone distribution and metabolism (D&M) model example (Figure 9)

This more complex version of a well-studied biosystem model [24–26] illustrates the power of our algorithm.

\[
x_1 = k_{12}x_2 + k_{13}x_3 - (k_{31} + k_{21})x_1 + u_1,
\]
\[
x_2 = k_{21}x_1 - (k_{12} + k_{02} + k_{25})x_2,
\]
\[
x_3 = k_{31}x_1 - (k_{13} + k_{03} + k_{63})x_3,
\]
\[
x_4 = k_{45}x_5 + k_{46}x_6 - (k_{64} + k_{54})x_4 + u_2,
\]
\[
x_5 = k_{54}x_4 + k_{52}x_2 - (k_{45} + k_{05})x_5,
\]
\[
x_6 = k_{64}x_4 + k_{63}x_3 - (k_{64} + k_{06})x_6,
\]
\[
y_1 = x_1,
\]
\[
y_2 = x_4.
\]

\[\text{(25)}\]

Table 2. Coefficients of the I/O Equations for the third derivative.

| Coefficients | No. in $T_2$ | RNo. |
|--------------|--------------|------|
| $1$          | $1$          | $1$  |
| $k_{13}$     | $2$          | $2$  |
| $k_{31}$     | $3$          | $3$  |
| $k_{24}k_{43}$ | $4$          | $4$  |
| $k_{04}k_{42}$ | $5$          | $4$  |
| $k_{24}k_{42}$ | $6$          | $5$  |
| $k_{03}k_{11} + k_{11}k_{43}$ | $7$          | $6$  |
| $k_{03} + k_{13} + k_{43}$ | $8$          | $7$  |
| $k_{03}k_{04} + k_{03}k_{42} + k_{04}k_{42} + k_{04}k_{43}$ | $9$          | $2,3,7$ |
| $k_{03} + k_{13} + k_{43}$ | $10$         | $8$  |
| $k_{03} + k_{13} + k_{43}$ | $11$         | $9$  |
| $k_{03}k_{04} + k_{03}k_{42} + k_{04}k_{42} + k_{04}k_{43} + k_{24}k_{43}$ | $12$         | $2,3,4,8,9$ |
| $k_{03}k_{04} + k_{03}k_{42} + k_{04}k_{42} + k_{04}k_{43} + k_{24}k_{43}$ | $13$         | $2,3,4,8,9$ |
| $k_{03} + k_{13} + k_{43}$ | $14$         | $3,4,9$ |
| $k_{03} + 2k_{03}k_{13} + k_{13}^2 + k_{03}k_{31} + k_{13}k_{31} + 2k_{03}k_{43} + 2k_{13}k_{43} + k_{31}k_{43} + k_{43}^2$ | $15$         | $2,3,4,9$ |
| $k_{03} + 2k_{03}k_{13} + k_{13}^2 + k_{03}k_{31} + 2k_{13}k_{31} + k_{31}^2 + 2k_{03}k_{43} + 2k_{13}k_{43} + k_{31}k_{43} + k_{43}^2$ | $16$         | $2,3,4,9$ |

Note: RNo. is relation number.

Figure 9. Linear thyroid D&M model.

We use the method of Section 4. This model has $n = 6$ state variables $\{x_1, x_2, x_3, x_4, x_5, x_6\}$ and $m = 2$ output equations $\{y_1, y_2, y_4, y_4\}$. Then, according to the GRA algorithm, taking derivatives $n - (m - 1) = 6 - (2 - 1) = 5$ times is sufficient to find the input–output equation (Figure 9).

1. We differentiate $y_1 - x_1$ and $y_4 - x_4$ and then add the result to the system (Neq = 10, Nx = 12):

\[
y_1 - x_1,
\]
\[
y_4 - x_4,
\]

\[\text{(26)}\]

2. Differentiate $y_1$ and $y_2$ to obtain $\dot{y}_1$ and $\dot{y}_2$, and the corresponding state variable equation to acquire $x_1$ and $x_4$. Add these to the system (Neq = 14, Nx = 14):

\[
\dot{y}_1 - \dot{x}_1,
\]
\[
\dot{y}_4 - \dot{x}_4,
\]

\[\dot{x}_1 + k_{12}x_2 + k_{13}x_3 - (k_{31} + k_{21})x_1 + u_1,
\]
\[
\dot{x}_4 + k_{45}x_5 + k_{46}x_6 - (k_{64} + k_{54})x_4 + u_2,
\]

\[\text{(27)}\]

3. Differentiate $y_1$ and $y_2$ to obtain $\ddot{y}_1$ and $\ddot{y}_2$, and the corresponding state variable equation to acquire $x_1$, $x_4$, $x_3$ and $\dot{x}_4$. Add these to the system (Neq = 22, Nx = 20):

\[
\ddot{y}_1 - \ddot{x}_1,
\]
\[
\ddot{y}_4 - \ddot{x}_4,
\]

\[\ddot{x}_1 + k_{12}x_2 + k_{13}x_3 - (k_{31} + k_{21})x_1 + \ddot{u}_1,
\]
the new system now has 22 equations in 20 unknown state variables and their derivatives, \( \{x_1, x_2, x_3, x_4, x_5, x_6, x_1, x_2, x_3, x_4, x_5, x_6, x_1, x_2, x_3, x_4, x_5, x_6, x_1, x_2, x_3, x_4, x_5, x_6, x_1\} \), enough to eliminate all unknowns. The algorithm terminates here. (In comparison the GRA algorithm requires 5 (instead of 3) steps, giving 38 equations in 32 unknown (instead of 22 in 20 unknowns) state variables and their derivatives). This gives two input–output equations.

To test structural identifiability of this model, we adopt the quasi-numerical method used in DAISY \[7\]. We provide the exhaustive summary \( c(p) \), using coefficients of the I/O equations plus random numbers. For every function of the unknown \( p \) in the exhaustive summary we assign a pseudo-randomly chosen numerical value for the elements of vector \( p \), providing a system of nonlinear algebraic equations in the unknown \( p \). This quasi-numerical approach allows significantly reduced symbolic computations and hence broadens the class of testable models. For this example, the resulting nonlinear algebraic equations (with random numbers) were solved using Mathematica, which returns solutions for each unknown parameter. When these solutions indicate that the parameter is equal to its assigned number, the model is globally structurally identifiable, otherwise it is not. We chose numerical parameter values as follows:

\[
\begin{align*}
k_{02} & \rightarrow 2, k_{03} \rightarrow 3, k_{05} \rightarrow 5, k_{06} \rightarrow 7, k_{12} \rightarrow 11, \\
k_{13} & \rightarrow 13, k_{21} \rightarrow 17, k_{31} \rightarrow 19, k_{45} \rightarrow 23, \\
k_{46} & \rightarrow 29, k_{52} \rightarrow 31, k_{54} \rightarrow 37, k_{63} \rightarrow 41, k_{64} \rightarrow 43
\end{align*}
\]

After solving exhaustive summary, none are equal to their numerical value, so the model is not structurally identifiable (Appendix 1).

9. Implementation of method as web application

Our new algorithm is implemented as web application COMBOS2, available online at: http://biocyt1.cs.ucla.edu/combos2. To show the speed of our algorithm compared with the Gröbner Basis approach in COMBOS, we give some examples of analyzing linear models for identifiable combinations, which DAISY does not provide. COMBOS \[8\] provides identifiable parameter combinations for the 4-Compartment biosystem model in about 17 seconds, decreasing to 5 seconds with COMBOS2. COMBOS2 shows that a 5-Compartment identifiable model \[27\] is identifiable in about 9 seconds, whereas COMBOS needs 28 seconds. Each is at least 3 times faster with COMBOS2. For the 6-compartment linear thyroid model above, COMBOS did not provide any answer because it hung during the I/O phase of the computation. COMBOS2 took only 2 seconds for finding I/O equations, but it also hangs when attempting to find identifiable parameter combinations, because it uses the Gröbner Basis approach in its parameter combinations algorithm. An alternative method for solving this problem using Mathematica is described in the Appendix 2. DAISY does not compute identifiable parameters combinations, but it did analyse for structural identifiability of individual parameters, and this took several hours to complete.

10. Conclusion

We introduced a novel algorithm for finding the I/O equations for systems of linear biosystem models and other linear ODEs of the form (1) with \( m \) output equations, by taking derivatives of the system and then using a Gaussian elimination approach. The new algorithm runs much faster and is simpler to implement than a Grobner basis approach. We also showed the number of derivatives needed for computing I/O equations is smaller when the condition \( N_{eq} \geq N_x + m \) is satisfied. We demonstrated the utility of the new algorithmic approach in a new app COMBOS2, applying it to several example models and, in particular, to a 6-dimensional thyroid hormone dynamics model, shown in 2 seconds to be not structurally identifiable, while COMBOS provided no solution. Another app DASIY needed several hours to provide only a structural identifiability analysis for the original individual parameters, and no parameter combination results. The main limitation of our new algorithm is, of course, that it is not applicable to nonlinear biosystem models. But we believe it gets a bit closer.

Disclosure statement

No potential conflict of interest was reported by the authors.

ORCID

Ali Kalami Yazdi http://orcid.org/0000-0002-1354-9786

References

[1] DiStefano III J. Dynamic systems biology modeling and simulation. Waltham (MA): Academic Press; 2015; Jan 10.
[2] Cobelli C, Distefano 3rd JJ. Parameter and structural identifiability concepts and ambiguities: a critical review and analysis. Am J Physiol Regul Integr Comp Physiol. 1980;239(1):R7–R24.
[3] Meshkat N, Anderson C, DiStefanoll JJ. Alternative to Ritt’s pseudodivision for finding the input-output equations of multi-output models. Math Biosci. 2012;239(1):117–123.
Appendices
Appendix 1

1) \[ k_{02} \rightarrow \frac{1333k_{63} - 51619}{1517}, k_{03} \rightarrow 44 - k_{63}, \]
   \[ k_{05} \rightarrow \frac{1204k_{63} - 56949}{43k_{63} - 3280}, k_{06} \rightarrow \frac{36k_{63} - 1189}{k_{63}}, \]
   \[ k_{12} \rightarrow 11, k_{13} \rightarrow 13, k_{21} \rightarrow 17, k_{31} \rightarrow 19, \]
   \[ k_{45} \rightarrow \frac{34891}{43k_{63} - 3280}, k_{46} \rightarrow \frac{1189}{k_{63}}, \]
   \[ k_{52} \rightarrow \frac{31(43k_{63} - 3280)}{1517}, k_{54} \rightarrow \frac{1}{41}(3280 - 43k_{63}), \]
   \[ k_{64} \rightarrow \frac{43k_{63}}{41}. \]

2) \[ k_{02} \rightarrow \frac{385237k_{63} - 14136513}{547637}, k_{03} \rightarrow \frac{1}{17}(722 - 17k_{63}), \]
   \[ k_{05} \rightarrow \frac{20468k_{63} - 1082031}{731k_{63} - 62320}, \]
   \[ k_{06} \rightarrow \frac{612k_{63} - 22591}{17k_{63}}, k_{12} \rightarrow 187 - \frac{19}{17}k_{13} \rightarrow 247, \]
   \[ k_{21} \rightarrow 19, k_{31} \rightarrow 17, k_{45} \rightarrow \frac{662929}{731k_{63} - 62320}, \]
   \[ k_{46} \rightarrow \frac{22591}{17k_{63}}, k_{52} \rightarrow \frac{527(731k_{63} - 62320)}{547637}, \]
   \[ k_{54} \rightarrow \frac{1}{779}(62320 - 731k_{63}), k_{64} \rightarrow \frac{731k_{63} - 779}{779}, \]

3) \[ k_{02} \rightarrow \frac{385237}{\left(1 - 19k_{63} + 11\sqrt{647} + 1034\right)}, \]
   \[ k_{05} \rightarrow \frac{38657\sqrt{647}}{\left(2480\sqrt{647} + 44640 + 703\right)k_{63}}, \]
   \[ \rightarrow \frac{695826}{\left(2480\sqrt{647} + 44640 + 703\right)k_{63}} + 36, \]
   \[ k_{06} \rightarrow \frac{532k_{63} + 713\sqrt{647} + 12834}{19k_{63}}, \]
   \[ k_{12} \rightarrow \frac{13}{17}\left(\sqrt{647} - 18\right), k_{13} \rightarrow \frac{11}{19}\left(-\sqrt{647} - 18\right), \]
   \[ k_{21} \rightarrow \sqrt{647} + 18, k_{31} \rightarrow 18 - \sqrt{647}, \]
substituting some of them with numbers.

ing this problem is to reduce the number of parameters by

\( k \rightarrow 38657 \left( \frac{\sqrt{647} + 18}{k_{63}} \right) \),
\[ k_{45} \]
\[ \frac{2480\sqrt{647}}{k_{63}} + 44640 \]
\[ + 703 \] ,
\[ 713 \left( \frac{\sqrt{647} + 18}{k_{63}} \right) \],
\[ k_{46} \]
\[ \frac{19k_{63}}{41} \left( 1332\sqrt{647}k_{63} - 35927k_{63} - 42160\sqrt{647} + 758880 \right) \],
\[ k_{52} \]
\[ \frac{1}{527} \left( 37\sqrt{647}k_{63} - 666k_{63} + 42160 \right) \],
\[ k_{54} \]
\[ \frac{1}{527} \left( -37 \left( \frac{\sqrt{647}k_{63} - 18k_{63}}{k_{63}} \right) \right) ,
\[ k_{64} \]
\[ -54612\sqrt{647}k_{63} + 1473007k_{63} + 2023153 \sqrt{647} + 58375263 \]
\[ 4 \] ,
\[ k_{02} \]
\[ \frac{385237}{695826} \]
\[ + \frac{2480\sqrt{647}}{k_{63}} - 44640 \]
\[ \frac{1}{k_{63}} - 703 \]
\[ k_{06} \]
\[ \frac{532k_{63} - 713\sqrt{647} + 12834}{19k_{63}} \],
\[ k_{12} \]
\[ \frac{13}{17} \left( -\sqrt{647} - 18 \right) ,
\[ k_{13} \]
\[ \frac{11}{19} \left( \sqrt{647} - 18 \right) ,
\[ k_{21} \]
\[ 18 - \sqrt{647} ,
\[ k_{31} \]
\[ \sqrt{647} + 18 ,
\[ k_{45} \]
\[ 38657 \left( \frac{\sqrt{647} - 18}{k_{63}} \right) \],
\[ k_{46} \]
\[ \frac{2480\sqrt{647}}{k_{63}} - 44640 \]
\[ \frac{1}{k_{63}} - 703 \]
\[ k_{52} \]
\[ \frac{1}{527} \left( -37\sqrt{647}k_{63} - 666k_{63} + 42160 \right) \],
\[ k_{54} \]
\[ \frac{1}{527} \left( 37\sqrt{647}k_{63} + 18k_{63} \right) \].

### Appendix 2

In this section, we introduced a method for finding identifiable parameters combinations of the linear thyroid model by using Mathematica. COMBOS2 hangs when attempting to find identifiable parameter combinations. A technique for solving this problem is to reduce the number of parameters by substituting some of them with numbers.

**First Step.** According to the calculations for the linear thyroid model (A1), (A2), (A3), (A4), the best choice is as follows:

\[ \{ k_{12} \rightarrow 11, k_{13} \rightarrow 13, k_{21} \rightarrow 17, k_{31} \rightarrow 19 \} \]

\[ (A5) \]

| \( k_{11} = -k_{21} - k_{31} \) |
| \( k_{22} = -(k_{022} + k_{12}) = -k_{12} - k_{02} - k_{22} \) |
| \( k_{33} = -(k_{033} + k_{13}) = -k_{13} - k_{03} - k_{33} \) |
| \( k_{44} = -k_{044} = -k_{44} \) |
| \( k_{55} = -k_{055} = -k_{55} \) |
| \( k_{66} = -k_{066} = -k_{66} \) |
| \( T_{22} = k_{12721} = k_{1221} \) |
| \( T_{33} = k_{13731} = k_{1331} \) |
| \( T_{52} = k_{45752712} = k_{45722112} = k_{54}k_{24}k_{12} \) |
| \( T_{55} = k_{45754} = k_{4554} \) |
| \( T_{63} = k_{46763713} = k_{4675313} = k_{66}k_{34}k_{13} \) |
| \( T_{66} = k_{46764} = k_{64}k_{64} \) |

**Table A1.** The identifiable parameters combinations of the linear thyroid model.

**Uniquely Identifiable Parameters Combinations for Linear Thyroid Model**

\[ (A4) \]

After substituting these parameters in \( c(p) = c(p^*) \), we investigate the parameter identifiability of the system (25) using Gröbner bases for the solutions to new \( c(p) = c(p^*) \) and have:

\[ \{ k_{54} + k_{64}, k_{60}k_{64}, k_{64}k_{65}, k_{64}k_{66}, k_{64}k_{52}, \]
\[ k_{06} + k_{46}, k_{05} + k_{45}, k_{03} + k_{63}, k_{02} + k_{52} \} \]
\[ (A6) \]

Let \( k_{5464} = k_{54} + k_{64}, k_{46764} = k_{46}k_{64}, k_{46763} = k_{46}k_{63}, k_{46756} = k_{45}k_{64}, k_{45752} = k_{45}k_{52}, k_{064} = k_{06} + k_{46}, k_{0545} = k_{05} + k_{45}, k_{0363} = k_{03} + k_{63}, k_{0252} = k_{02} + k_{52} \)

\[ \text{and also } x_{2} = k_{12}x_{2} ,
\[ x_{3} = k_{13}x_{3} ,
\[ x_{4} = k_{4}x_{4} \]
\[ x_{5} = k_{5}x_{5} ,
\[ x_{6} = k_{6}x_{6} \]
\[ y_{1} = x_{1} ,
\[ y_{2} = x_{4} \]
\[ (A7) \]

**Second Step.** Using Mathematica we investigate the parameter identifiability of this new system (A7) and find the following combinations:

\[ \{ k_{5464}, k_{46764}, k_{4676713}, k_{45754}, k_{45752712}, \]
\[ k_{4505}, k_{21} + k_{31}, k_{13731}, k_{12721}, k_{0464} ,
\[ k_{0363} + k_{13}, k_{0252} + k_{12} \} \]

\[ (A8) \]

By combining the identifiable parameter combinations obtained from the first (A6) and second (A8) steps for the linear thyroid model, we have the following Table A1:

Then the reparameterized linear thyroid model equations are:

\[ x_{1} = x_{2} + x_{3} + k_{11}x_{1} + u_{1} ,
\[ X_{2} = T_{22}x_{1} + k_{22}x_{2} ,
\[ X_{3} = T_{33}x_{1} + k_{33}x_{3} ,
\[ X_{4} = x_{4} + x_{5} + k_{44}x_{4} + u_{2} ,
\[ \ X_{5} = T_{55}x_{4} + T_{52}x_{2} + k_{55}x_{5} ,
\[ X_{6} = T_{66}x_{4} + T_{63}x_{3} + k_{66}x_{6} ,
\[ y_{1} = x_{1} ,
\[ y_{2} = x_{4} \]
\[ (A9) \]