Insufficiency of Chemical Network Model Integration Using a High-Order Taylor Series Method

Ryan M. Brown

Student for Master of Science in Computer Science at
California State University Channel Islands
One University Drive, Camarillo, CA 93012 USA
ryan.brown@csuci.edu

Abstract

The time evolution of chemical substances in a set of reactions is modeled by a nonlinear dynamical system, which warrants the use of numerical methods for differential equations. Previous work advocates the use of a high-order Taylor series method because of an observed reduction in computation time. Contrastingly, we demonstrate that when the model does not exhibit stiffness, the quality of output may be dramatically reduced by use of a Taylor Series Method. When the modeling function exhibits stiffness, our implementation of a Taylor Series Method fails to compute. By our provided counter example, numerical methods when applied to the given modeling function remain subject to a case by case consideration.

Keywords: kinetic model, Taylor series method, dynamic programming, nonlinear dynamical system

MSC codes: 92B05, 65P99, 65L05, 05A17

1 Introduction

Systems of chemical reactions are common in many fields, notably in biochemistry, where reactions take place within a cell. These reactions are typically presented symbolically, indicating the proportions in which a set of initial chemicals interact to form other chemical substances. For each reaction there is an associated rate that governs the change in concentration of all related chemicals over time. The rates at which reactions occur can be difficult or even impossible to measure experimentally; they can vary over many orders of magnitude and thereby posses certain difficulties for chemists.

The ability to describe the concentration of any particular chemical substance at any point in time during a reaction is valuable to both chemists and mathematicians. Doing so requires analysis of a nonlinear kinetic model, for which numerical integration is required. Previous attempts at numerically integrating the kinetic model by a Taylor series method consider only instances for which a closed form solution may be obtained. When a closed form solution is not obtainable, derivatives may be computed by an iterative procedure.

Independent of considerations regarding the step size of the Taylor series method, we address three points of computational concern; one of these is resolved by dynamic programming, though the others remain without resolution. We therefore conclude that the Taylor series approach to numerical integration requires either further analysis or abandonment.

2 Theory of Chemical Reaction Networks

Chemical reactions are traditionally written schematically by the reaction equation

$$\sum_{i=1}^{n} \alpha_i \cdot \mathrm{Reactant}_i \xrightarrow{k} \sum_{j=1}^{m} \beta_j \cdot \mathrm{Product}_j \tag{1}$$

which denotes the set of reactants chemically interacting at rate $k \in \mathbb{R}_{>0}$ to form the set of chemical products. The stoichiometric coefficient α_i, assumed in $\mathbb{Z}_{\geq 0}$, indicates the proportion in which reactant i exists relative to the other reactants, and similarly for β_j.

We formalize the schematic reaction equation 1 into more mathematical structures first by noting that a chemical reactant X with stoichiometric coefficient $\alpha(\mathrm{X})$ may also be regarded as a chemical product with coefficient 0. The converse is true for chemical products. The chemical substances that are either reactants or products collectively form the set of chemical species, which can be anything that participates in the reaction. Typically, chemical species are considered chemical elements, molecules, or proteins.

The set of reactants and products of any reaction are each called a complex, which is a set of elements with associated coefficients in $\mathbb{Z}_{\geq 0}$. In this regard, a complex is considered a multiset over the set of species, where the multiplicity of any species is given by its stoichiometric coefficient. The chemical process consumes the species on the left side of equation 1 and consequently forms those on the right.

A set theoretic interpretation of equation 1 is the chemical conversion of one complex to another by the "reaction arrow," which represents a binary relation on two complexes.

Definition 1. Given a finite set of chemical reactions, we define a chemical reaction network as a quadruple $\mathscr{N} = (\mathscr{S}, \mathscr{C}, \mathscr{R}, k)$ where

- $\mathscr{S}$ is the set of chemical species, assumed nonempty and finite;
- $\mathscr{C}$ denotes the set of complexes, having nonempty finite multisets on $\mathscr{S}$ such that for all $\mathrm{X} \in \mathscr{S}$, the stoichiometric coefficient of X is its multiplicity in its complex.
- $\mathscr{R}$ is a binary relation on $\mathscr{C}$ with the property that for all $Y \in \mathscr{C}$, there exists a dissimilar $Y' \in \mathscr{C}$ such that either $(Y, Y') \in \mathscr{R}$ or $(Y', Y) \in \mathscr{R}$ holds.
- $k : \mathscr{R} \to \mathbb{R}_{>0}$ is the rate map.

A chemical network is a mathematical structure that formalizes the schematic reaction equation 1 and generalizes to the case when many interacting chemical reactions occur. The algebraic properties of chemical networks is an area of study that complements ours [7]. For our purposes, the term largely serves to aggregate our concepts of species, complexes, and reactions.

For a chemical network $\mathscr{N}$ having $(Y, Y') \in \mathscr{R}$, it is possible that $(Y', Y) \in \mathscr{R}$ also; chemists say that these reactions are "reversible," possessing certain thermodynamic properties. We regard this situation as two independant reactions in $\mathscr{N}$.

Definition 2. For $Y \in \mathscr{C}$, we write $\alpha(\mathrm{X}) \in Y$ to mean $\alpha(\mathrm{X})$ is the multiplicity of some species X.

For any chemical network $\mathscr{N}$ having $\alpha(\mathrm{X}) \in Y$ and $\beta(\mathrm{X}) \in Y'$ for each $(Y, Y') \in \mathscr{R}$, the time evolution of species is given by the map $\mathbf{f} : \mathbb{R}_{\geq 0}^{|\mathscr{S}|} \to \mathbb{R}_{\geq 0}^{|\mathscr{S}|}$ defined by

$$\mathbf{f}([\mathbf{X}]) = \sum_{\alpha(\mathbf{X}), \beta(\mathbf{X})} (\beta(\mathbf{X}) - \alpha(\mathbf{X})) \cdot k(Y, Y') \cdot \prod_{\mathrm{X} \in \mathscr{S}} [\mathrm{X}]^{\alpha(\mathrm{X})} \tag{2}$$

where $[\mathrm{X}] = [\mathrm{X}](t)$ is the concentration of any $\mathrm{X} \in \mathscr{S}$. The scientific notion of concentration is not mathematically necessary, so we simply regard [X] as some non negative real number. The origin of our modeling function given in equation 2 may be found in [5].

Example 1. Let $\mathbf{X}$ be the vector having components [A], [B], and [C]. The deformation of species A in the chemical reaction $2\,\mathrm{A} + \mathrm{B} \xrightarrow{k} \mathrm{C}$ is given by

$$f_{\mathrm{A}}([\mathbf{X}]) = -2k[\mathrm{A}]^2[\mathrm{B}] \tag{3}$$

Example 2. Consider the set of chemical reactions

$$\begin{gathered} \mathrm{A} \xrightarrow{k_1} \mathrm{B} \\ 2\,\mathrm{B} \xrightarrow{k_2} \mathrm{B} + \mathrm{C} \\ \mathrm{B} + \mathrm{C} \xrightarrow{k_3} \mathrm{A} + \mathrm{C} \end{gathered}$$

Let $[\mathbf{X}]$ be the vector having components [A], [B], and [C]. The time evolution for the species A, B, and C in the above reactions is enumeratively given by

$$\begin{array}{l} f_{\mathrm{A}}([\mathbf{X}]) = -k_1[\mathrm{A}] + k_3[\mathrm{B}][\mathrm{C}] \\ f_{\mathrm{B}}([\mathbf{X}]) = k_1[\mathrm{A}] - k_3[\mathrm{B}][\mathrm{C}] - k_2[\mathrm{B}]^2 \\ f_{\mathrm{C}}([\mathbf{X}]) = k_2[\mathrm{B}]^2 \end{array} \tag{4}$$

Numerical integration of the function given in equation 2 may be done by any of the standard methods, including any of the Runge-Kutta methods, an Adam-Bashforth-Moulton method, or a backward differentiation formula [3][10]; however, a Taylor series method is often advocated as having high utility for numerically integrating 2 and many implementations are known [1] [2] [8].

3 The Taylor Series Approach

If we consider

$$\frac{\mathrm{d}[\mathrm{X}]}{\mathrm{d}t} = f_{\mathrm{X}}([\mathbf{X}]) \tag{5}$$

where $f_{\mathrm{X}}([\mathbf{X}])$ are the components of $\mathbf{f}$ given in equation 2, then integration of 2 describes the concentration of each species $\mathrm{X} \in \mathscr{S}$ at any point in time during the reaction process.

Derivative computation for the Taylor series method is done in [1] by a specialized iterative procedure which reportedly reduces computation time; however a derivation is not provided. For our purposes, we compute derivatives of order n by the application of theorem 1.

Theorem 1. *Let $\mathscr{N}$ be a chemical network such that $\alpha(\mathrm{X}) \in Y$ and $\beta(\mathrm{X}) \in Y'$ for all $(Y, Y') \in \mathscr{R}$. The n^{th} derivative of $\mathbf{f}$ as in equation 2 is recursively given by*

$$\frac{\mathrm{d}^n}{\mathrm{d}t^n}\mathbf{f}\left([\mathbf{X}]\right) = \sum_{\alpha(\mathrm{X}),\beta(\mathrm{X})} \gamma \left(\sum_{\theta} \binom{n}{n_1, n_2, \ldots, n_{|\mathscr{S}|}} \prod_{i=1}^{|\mathscr{S}|} \frac{\mathrm{d}^{n_i}}{\mathrm{d}t^{n_i}} [\mathrm{X}_i]^{\alpha(\mathrm{X}_i)} \right) \tag{6}$$

$$\frac{\mathrm{d}^{n_i}}{\mathrm{d}t^{n_i}} [\mathrm{X}_i]^{\alpha(\mathrm{X}_i)} = \sum_{\pi\in\Pi} [\mathrm{X}_i]^{\alpha(\mathrm{X}_i)-|\pi|} \prod_{\ell=0}^{|\pi|-1} \left(\alpha(\mathrm{X}_i) - \ell\right) \prod_{B\in\pi} [\mathrm{X}_i]^{(|B|)} \tag{7}$$

where

- $\gamma = \big(\beta(\mathbf{X}) - \alpha(\mathbf{X})\big) \cdot k\left(Y, Y'\right)$
- *θ is the condition that $n_1 + \cdots + n_{|\mathscr{S}|} = n$.*
- *π runs through the set Π of all partitions of the set $\{1, 2, \ldots, n_i\}$.*
- *each B runs through the list of all of the "blocks" of the partition π.*

Proof. To differentiate f, we differentiate across the product-reactant stoichiometric coefficients of each reaction:

$$\frac{\mathrm{d}^n}{\mathrm{d}t^n}\mathbf{f}\left([\mathbf{X}]\right) = \sum_{\alpha(\mathbf{X}),\beta(\mathbf{X})} \big(\beta(\mathbf{X}) - \alpha(\mathbf{X})\big) \cdot k\left(Y, Y'\right) \frac{\mathrm{d}^n}{\mathrm{d}t^n} \prod_{\mathrm{X}\in\mathscr{S}} [\mathrm{X}]^{\alpha(\mathrm{X})} \tag{8}$$

thereby requiring differentiation of the product $\prod_{\mathrm{X}\in\mathscr{S}} [\mathrm{X}]^{\alpha(\mathrm{X})}$. The generalized Leibniz rule allows for differentiation of a product of functions defined on an open interval of $\mathbb{R}$. The n^{th} derivative of the product is given by

$$\frac{\mathrm{d}^n}{\mathrm{d}t^n} \prod_{\mathrm{X}\in\mathscr{S}} [\mathrm{X}]^{\alpha(\mathrm{X})} = \sum_{n_1+\cdots+n_{|\mathscr{S}|}=n} \binom{n}{n_1, n_2, \ldots, n_{|\mathscr{S}|}} \prod_{i=1}^{|\mathscr{S}|} \frac{\mathrm{d}^{n_i}}{\mathrm{d}t^{n_i}} [\mathrm{X}_i]^{\alpha(\mathrm{X}_i)} \tag{9}$$

Let $\psi : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ be defined by $\psi([\mathrm{X}]) = [\mathrm{X}]^{\alpha(\mathrm{X})}$ such that $(\psi \circ [\mathrm{X}])(t) = [\mathrm{X}](t)^{\alpha(\mathrm{X})}$, which we conveniently shorten to $[\mathrm{X}]^{\alpha(\mathrm{X})}$. Differentiation of $\psi \circ [\mathrm{X}]$ requires an extension of the chain rule up to order n. Analogous to the generalized Leibniz rule, the formula of Faà di Bruno gives an method for computing high order derivatives of a composition of functions. Applying the combinatorial form of the formula of Faà di Bruno to equation 9, we have

$$\frac{\mathrm{d}^n}{\mathrm{d}[\mathrm{X}]^n} (\psi \circ [\mathrm{X}])(t) = \sum_{\pi \in \Pi} (\psi \circ [\mathrm{X}])^{|\pi|}(t) \prod_{B \in \pi} [\mathrm{X}]^{|B|}(t) \tag{10}$$

where π runs through the set Π of all partitions of the set $\{1, 2, \ldots, n_i\}$. Each $B \in \pi$ runs through the list of all of the "blocks" of the partition π.

Since ψ is polynomial in $[\mathrm{X}]$ with exponent $\alpha(\mathrm{X})$, its n^{th} derivative is explicitly given by

$$\frac{\mathrm{d}^n}{\mathrm{d}[\mathrm{X}]^n}\psi = [\mathrm{X}]^{\alpha(\mathrm{X})-n} \prod_{\ell=0}^{n-1} \alpha(\mathrm{X}) - \ell \tag{11}$$

Extending equation 5 to higher order cases, we compute $[\mathrm{X}]^{|B|}(t)$ recursively by

$$\frac{\mathrm{d}^{|B|}[\mathrm{X}]}{\mathrm{d}t^{|B|}} = \frac{\mathrm{d}^{|B|-1} f_{\mathrm{X}}}{\mathrm{d}t^{|B|-1}} \tag{12}$$

Equations 11 and 12 establish the existence of derivatives of the n^{th} derivative of f. Combining equations 8 through 12 defines a recursive procedure for computing the n^{th} derivative. $\square$

The formula of Faà di Bruno and generalized Leibniz rule are both infrequently encountered differentiation formulae; their derivations are outlined in [9] and [11], respectively.

Example 3. Consider the reaction $2\,\mathrm{A} + \mathrm{B} \xrightarrow{k} \mathrm{C}$ as given in example 1. The time evolution of each species' concentration is given by

$$\frac{\mathrm{d}[\mathrm{A}]}{\mathrm{d}t} = -2k[\mathrm{A}]^2[\mathrm{B}] \qquad \frac{\mathrm{d}[\mathrm{B}]}{\mathrm{d}t} = -k[\mathrm{A}]^2[\mathrm{B}] \qquad \frac{\mathrm{d}[\mathrm{C}]}{\mathrm{d}t} = k[\mathrm{A}]^2[\mathrm{B}]$$

Computing first derivatives requires partitioning $n = 1$ into $|\mathscr{S}| = 3$ nonnegative integers. Let $\varphi_{\mathrm{X}} = (\psi \circ [\mathrm{X}])$ for each $\mathrm{X} \in \mathscr{S}$ and ψ defined as in theorem 1. Application of the Leibniz rule to the product $[\mathrm{A}]^2[\mathrm{B}]$ gives

$$\frac{\mathrm{d}}{\mathrm{d}t}\left([\mathrm{A}]^2[\mathrm{B}]\right) = \varphi_{\mathrm{A}}^{(1)}\varphi_{\mathrm{B}}^{(0)}\varphi_{\mathrm{C}}^{(0)} + \varphi_{\mathrm{A}}^{(0)}\varphi_{\mathrm{B}}^{(1)}\varphi_{\mathrm{C}}^{(0)} + \varphi_{\mathrm{A}}^{(0)}\varphi_{\mathrm{B}}^{(0)}\varphi_{\mathrm{C}}^{(1)}$$

which simplifies to

$$\frac{\mathrm{d}}{\mathrm{d}t}\left([\mathrm{A}]^2[\mathrm{B}]\right) = \frac{\mathrm{d}}{\mathrm{d}t}\varphi_{\mathrm{A}}[\mathrm{B}] + [\mathrm{A}]^2\frac{\mathrm{d}}{\mathrm{d}t}\varphi_{\mathrm{B}} + [\mathrm{A}]^2[\mathrm{B}]\frac{\mathrm{d}}{\mathrm{d}t}\varphi_{\mathrm{C}}$$

We now apply equation 10, which is simply the chain rule for derivatives of order 1:

$$\frac{\mathrm{d}}{\mathrm{d}t}\varphi_{\mathrm{A}} = 2[\mathrm{A}]\frac{\mathrm{d}}{\mathrm{d}t}[\mathrm{A}] \qquad \frac{\mathrm{d}}{\mathrm{d}t}\varphi_{\mathrm{B}} = \frac{\mathrm{d}}{\mathrm{d}t}[\mathrm{B}] \qquad \frac{\mathrm{d}}{\mathrm{d}t}\varphi_{\mathrm{C}} = 0$$

The first derivative is given by combining f, the Leibniz equation, and the Faà di Bruno equation. The first derivative is given by

$$\frac{\mathrm{d}^2[\mathrm{A}]}{\mathrm{d}t^2} = -2k\left(-4k[\mathrm{A}]^3[\mathrm{B}]^2 - k[\mathrm{A}]^2[\mathrm{B}]\right)$$

$$\frac{\mathrm{d}^2[\mathrm{B}]}{\mathrm{d}t^2} = -k\left(-4k[\mathrm{A}]^3[\mathrm{B}]^2 - k[\mathrm{A}]^2[\mathrm{B}]\right)$$

$$\frac{\mathrm{d}^2[\mathrm{C}]}{\mathrm{d}t^2} = k\left(-4k[\mathrm{A}]^3[\mathrm{B}]^2 - k[\mathrm{A}]^2[\mathrm{B}]\right)$$

Derivative computation for both networks found in example 1 and 2 require partitioning n into $|\mathscr{S}| = 3$ nonnegative integer. These partitions are enumerated in table 1.

n	Distinct Partitions of n into $\lvert\mathscr{S}\rvert = 3$	Number of Partitions
1	$(0,0,1)$, $(0,1,0)$, $(1,0,0)$	3
2	$(0,0,2)$, $(0,1,1)$, $(0,2,0)$, $(1,1,0)$, $(2,0,0)$, $(1,0,1)$	6
3	$(0,0,3)$, $(2,1,0)$, $(0,2,1)$, $(2,0,1)$, $(0,3,0)$, $(1,0,2)$, $(1,2,0)$, $(0,1,2)$, $(3,0,0)$, $(1,1,1)$	10
4	$(0,0,4)$, $(0,1,3)$, $(0,2,2)$, $(0,3,1)$, $(3,0,1)$, $(4,0,0)$, $(0,4,0)$, $(1,3,0)$, $(1,2,1)$, $(1,0,3)$, $(1,1,2)$, $(2,0,2)$, $(2,1,1)$, $(2,2,0)$, $(3,1,0)$	15

Table 1: Multinomial coefficients required for the application of the Leibniz rule when $|\mathscr{S}| = 3$ for derivative orders up to $n = 4$.

As n increases, the number of partitions of n into $|\mathscr{S}|$ integers in $\mathbb{Z}_{\geq 0}$ grows rather quickly. This presents a combinatorial constraint on our analysis of equation 7. Despite having an iterative procedure for calculating n^{th} derivatives, evaluation of higher order derivatives becomes burdensome.

Definition 3. For any $\mathrm{X} \in \mathscr{S}$, the degree of a polynomial

$$\phi\left([\mathrm{X}]\right) = \sum_{n \in \mathbb{Z}_{\geq 1}} a_n \left[\mathrm{X}\right]^n + c$$

is given by

$$\deg_{\mathrm{X}}\left(\phi\right) = \exists\ a_n \neq 0\ ?\ \operatorname*{argmax}_n a_n \neq 0 : 0$$

for $c, a_i \in \mathbb{R}_{\geq 0}$.

Theorem 2. Let $\mathbf{f}$ be defined as in equation 2. If $\mathscr{N}$ is a chemical network such that $\alpha(\mathrm{X}) - \beta(\mathrm{X}) \neq 0$ for all $\alpha(\mathrm{X}) \in Y$ and $\beta(\mathrm{X}) \in Y'$ for each $(Y, Y') \in \mathscr{R}$, then

$$\deg_{\mathrm{X}}\left(f_i^{(n)}\right) = \deg_{\mathrm{X}}\left(f_j^{(n)}\right)$$

over all $\mathrm{X} \in \mathscr{S}$.

Proof. Let $\alpha(\mathrm{X}) - \beta(\mathrm{X}) \neq 0$ for all $\alpha(\mathrm{X}) \in Y$ and $\beta(\mathrm{X}) \in Y'$ where $(Y, Y') \in \mathscr{R}$. Equation 2 may be written

$$\mathbf{f}\left([\mathbf{X}]\right) = \sum_{\alpha(\mathrm{X}),\beta(\mathrm{X})} \left(\big(\beta(\mathrm{X}) - \alpha(\mathrm{X})\big) \cdot k\left(Y, Y'\right) \cdot \prod_{\mathrm{A} \in \mathscr{S} \backslash \{\mathrm{X}\}} [\mathrm{A}]^{\alpha(\mathrm{A})} \right) \mathrm{X}^{\alpha(\mathrm{X})}$$

By hypothesis, $\beta(\mathrm{X}) - \alpha(\mathrm{X}) > 0$ and $k\left(Y, Y'\right) > 0$ by definition 1. Therefore, $\Phi_{(Y,Y')} \in \mathbb{R}_{\geq 0}$, which implies that $\mathbf{f}$ is polynomial in X with coefficients

$$\Phi_{(Y,Y')} = \big(\beta(\mathrm{X}) - \alpha(\mathrm{X})\big) \cdot k\left(Y, Y'\right) \cdot \prod_{\mathrm{A} \in \mathscr{S} \backslash \{\mathrm{X}\}} [\mathrm{A}]^{\alpha(\mathrm{A})}$$

If a coefficient is zero, then the concentration of some $\mathrm{A} \in \mathscr{S} \backslash \{\mathrm{X}\}$ must be zero; if this is the case, then the coefficient will be zero in both f_i and f_j.

Since

$$\mathbf{f}\left([\mathbf{X}]\right) = \sum_{\alpha(\mathrm{X}),\beta(\mathrm{X})} \Phi_{(Y,Y')} \cdot [\mathrm{X}]^{\alpha(\mathrm{X})}$$

computing $f_i^{(n)}$ and $f_j^{(n)}$ requires applying the formula of Faà di Bruno to $[\mathbf{X}]^{\alpha(\mathrm{X})}$ only:

$$f_i^{(n)} = \sum_{\alpha(\mathbf{X}),\beta(\mathbf{X})} \Phi_{(Y,Y')} \cdot \left(\sum_{\pi \in \Pi} [\mathrm{X}]^{\alpha(\mathrm{X}) - |\pi|} \prod_{\ell=0}^{|\pi|-1} \left(\alpha(\mathrm{X}) - \ell\right) \prod_{B \in \pi} [\mathrm{X}]^{|B|}(t) \right)$$

$$f_j^{(n)} = \sum_{\alpha(\mathrm{X}),\beta(\mathrm{X})} \Phi_{(Y,Y')} \cdot \left(\sum_{\pi \in \Pi} [\mathrm{X}]^{\alpha(\mathrm{X}) - |\pi|} \prod_{\ell=0}^{|\pi|-1} \left(\alpha(\mathrm{X}) - \ell\right) \prod_{B \in \pi} [\mathrm{X}]^{|B|}(t) \right)$$

where π runs through the set Π of all partitions of n. The degrees of $f_i^{(n)}$ and $f_j^{(n)}$ are given by

$$\deg_{\mathrm{X}}\left(f_i^{(n)}\right) = \exists\, \sigma \text{ such that } \prod_{\ell=0}^{|\sigma|-1} \left(\alpha(\mathrm{X}) - \ell\right) \geq 0 \ ? \ \alpha(\mathrm{X}) - |\sigma| + \deg_{\mathrm{X}}\left(\prod_{B \in \sigma} [\mathrm{X}]^{|B|}(t)\right) : 0$$

$$\deg_{\mathrm{X}}\left(f_j^{(n)}\right) = \exists\, \sigma \text{ such that } \prod_{\ell=0}^{|\sigma|-1} \left(\alpha(\mathrm{X}) - \ell\right) \geq 0 \ ? \ \alpha(\mathrm{X}) - |\sigma| + \deg_{\mathrm{X}}\left(\prod_{B \in \sigma} [\mathrm{X}]^{|B|}(t)\right) : 0$$

for some partition σ of n. □

Whenever two polynomials f_i and f_j are defined explicitly over their respective variables such that they have common degree, their n^{th} derivatives will also have common degree. Theorem 2 extends this to the situation when f_i and f_j is implicitly defined in terms of t, even when of the form given by equation 2.

Contrastingly, whenever f_i and f_j are polynomials defined explicitly over their respective variables, differentiation creates a monotonically decreasing sequence of degrees; because of the implicit nature of $\mathbf{f}$, as given in equation 2, we form a monotonically increasing sequence.

Theorem 3. Let $\mathbf{f}$ be defined as in equation 2. If $\mathscr{N}$ is a chemical network such that $\alpha(\mathrm{X}) - \beta(\mathrm{X}) \neq 0$ for all $\alpha(\mathrm{X}) \in Y$ and $\beta(\mathrm{X}) \in Y'$ for each $(Y, Y') \in \mathscr{R}$, then the sequence

$$\left\{\deg_{\mathrm{X}}\left(f_{\mathrm{X}}\right), \deg_{\mathrm{X}}\left(f_{\mathrm{X}}^{(1)}\right), \cdots, \deg_{\mathrm{X}}\left(f_{\mathrm{X}}^{(n)}\right)\right\} \tag{13}$$

monotonically increases.

Proof. Let f be defined as in equation 2. As the proof of theorem 2 suggests, the degree of f_i over each $\mathrm{X} \in \mathscr{S}$ is the largest reactant stoichiometric coefficient in $\mathscr{N}$. Mathematically, this is given by

$$\deg_{\mathrm{X}}\left(f_i\right) = \max\left(\alpha\left(\mathrm{X}\right) \middle| \forall (y, y') \in \mathscr{R} : \alpha(\mathrm{X}) \in y\right)$$

We continue with a proof by induction. The case when $n = 1$ serves as our basis. Equation 7 reduces to

$$\frac{\mathrm{d}}{\mathrm{d}t} f_{\mathrm{X}}\left([\mathbf{X}]\right) = \sum_{\alpha(\mathrm{X}), \beta(\mathrm{X})} \big(\beta(\mathrm{X}) - \alpha(\mathrm{X})\big) \cdot k\left(Y, Y'\right) \cdot \left(\sum_{\theta} [\mathrm{X}]^{\alpha(\mathrm{X})-1} \alpha(\mathrm{X}) \frac{\mathrm{d}}{\mathrm{d}t}[\mathrm{X}] \prod_{j:n_j \neq 1}^{|\mathscr{S}|} [\mathrm{A}_j]^{\alpha(\mathrm{A}_j)}\right)$$

where $\mathrm{X} = \mathrm{A}_{\mathrm{j:n_j=1}}$ and θ is the condition that $n_1 + \cdots + n_{|\mathscr{S}|} = 1$. The degree of $f_{\mathrm{X}}^{(1)}$ is given by

$$\deg_{\mathrm{X}}\left(f_{\mathrm{X}}^{(1)}\right) = \alpha(\mathrm{X}) - 1 \geq 0 \ ? \ \alpha(\mathrm{X}) - 1 + \deg_{\mathrm{X}}\left(\frac{\mathrm{d}}{\mathrm{d}t} f_{\mathrm{X}}\right) : 0$$

over any $\mathrm{X} \in \mathscr{S}$ for some other f_j. Now, suppose that for fixed n

$$\left\{\deg_{\mathrm{X}}\left(f_{\mathrm{X}}\right), \deg_{\mathrm{X}}\left(f_{\mathrm{X}}^{(1)}\right), \ldots, \deg_{\mathrm{X}}\left(f_{\mathrm{X}}^{(n)}\right)\right\}$$

increases monotonically. For $n + 1$ there must exist some tuple $\left(n_1, n_2, \ldots, n_{|\mathscr{S}|}\right)$ satisfying $n_1 + \cdots + n_{|\mathscr{S}|} = n + 1$ such that

$$\begin{aligned} \deg_{\mathrm{X}}\left(f_i^{(n+1)}\right) &= \deg_{\mathrm{X}}\left(\prod_{j=1}^{|\mathscr{S}|} \frac{\mathrm{d}^{n_j}}{\mathrm{d}t^{n_j}} [\mathrm{X}_j]^{\alpha(\mathrm{X}_j)}\right) \\ &= \sum_{j=1}^{|\mathscr{S}|} \deg_{\mathrm{X}}\left(\frac{\mathrm{d}^{n_j}}{\mathrm{d}t^{n_j}} [\mathrm{X}_j]^{\alpha(\mathrm{X}_j)}\right) \end{aligned}$$

Consider the partition $0 + 0 + \cdots + n_{\mathrm{X}} + \cdots + 0 = n + 1$, implying that $n_{\mathrm{X}} = n + 1$. We then have

$$\sum_{j=1}^{|\mathscr{S}|} \deg_{\mathrm{X}}\left(\frac{\mathrm{d}^{n_j}}{\mathrm{d}t^{n_j}} [\mathrm{X}_j]^{\alpha(\mathrm{X}_j)}\right) \geq \deg_{\mathrm{X}}\left(\frac{\mathrm{d}^{n+1}}{\mathrm{d}t^{n+1}} [\mathrm{X}]^{\alpha(\mathrm{X})}\right)$$

which warrants the use of the formula of Faà di Bruno:

$$\deg_{\mathrm{X}}\left(\frac{\mathrm{d}^{n+1}}{\mathrm{d}t^{n+1}}\,[\mathrm{X}]^{\alpha(\mathrm{X})}\right)=\deg_{\mathrm{X}}\left(\sum_{\pi\in\Pi}[\mathrm{X}]^{\alpha(\mathrm{X})-|\pi|}\prod_{\ell=0}^{|\pi|-1}(\alpha(\mathrm{X})-\ell)\prod_{B\in\pi}[\mathrm{X}]^{|B|}(t)\right)$$

such that π runs through the set Π of all partitions of the set $\{1,2,\ldots,n+1\}$. The induction hypothesis leads us to consider partitions of $n+1$ maximizing $|B|$, which will result in larger derivative degrees. The trivial partition on $n+1$ maximizes $|B|$ and minimizes $\alpha(\mathrm{X})-|\pi|$, thereby ensuring

$$\deg_{\mathrm{X}}\left(\sum_{\pi\in\Pi}[\mathrm{X}]^{\alpha(\mathrm{X})-|\pi|}\prod_{\ell=0}^{|\pi|-1}(\alpha(\mathrm{X})-\ell)\prod_{B\in\pi}[\mathrm{X}]^{|B|}(t)\right)\geq\deg_{\mathrm{X}}\left([\mathrm{X}]^{\alpha(\mathrm{X})-1}\,(\alpha(\mathrm{X})-1)\,[\mathrm{X}]^{(n+1)}\right)$$

But,

$$\begin{aligned}\deg_{\mathrm{X}}\left([\mathrm{X}]^{\alpha(\mathrm{X})-1}\,(\alpha(\mathrm{X})-1)\,[\mathrm{X}]^{(n+1)}\right) &= \alpha(\mathrm{A})-1\geq 0\ ?\ \alpha(\mathrm{A})-1+\deg_{\mathrm{X}}\left(\mathrm{X}^{(n+1)}\right):0\\ &= \alpha(\mathrm{A})-1\geq 0\ ?\ \alpha(\mathrm{A})-1+\deg_{\mathrm{X}}\left(f_r^{(n)}\right):0\end{aligned}$$

for some component f_r of $\mathbf{f}$. Using our premise, we employ theorem 2, so that

$$\alpha(\mathrm{A})-1\geq 0\ ?\ \alpha(\mathrm{A})-1+\deg_{\mathrm{X}}\left(f_r^{(n)}\right):0=\alpha(\mathrm{A})-1\geq 0\ ?\ \alpha(\mathrm{A})-1+\deg_{\mathrm{X}}\left(f_i^{(n)}\right):0$$

□

Example 2 demonstrates that theorem 3 is not always applicable, since product-reactant differences of stoichiometric coefficients of C in reactions $\mathrm{A}\xrightarrow{k_1}\mathrm{B}$ and $\mathrm{B}+\mathrm{C}\xrightarrow{k_3}\mathrm{A}+\mathrm{C}$ are zero. If we remove the premise of theorem 3, we may achieve a similar results by considering $\deg_{\mathrm{X}}(f_{\mathrm{X}}([\mathrm{X}]))$ where $f_{\mathrm{X}}([\mathrm{X}])$ describes the time evolution of species $\mathrm{X}\in\mathscr{S}$.

Theorem 4. Let $f_{\mathrm{X}}([\mathrm{X}])=[\mathrm{X}]^{(1)}(t)$. The sequence

$$\left\{\deg_{\mathrm{X}}(f_{\mathrm{X}}),\deg_{\mathrm{X}}\left(f_{\mathrm{X}}^{(1)}\right),\cdots,\deg_{\mathrm{X}}\left(f_{\mathrm{X}}^{(n)}\right)\right\}\tag{14}$$

monotonically increases.

Proof. The proof of theorem 4 by induction is nearly identical to that of theorem 3. The proofs become slightly different at the point

$$\begin{aligned}\deg_{\mathrm{X}}\left(f_{\mathrm{X}}^{(n+1)}\right) &\geq \deg_{\mathrm{X}}\left([\mathrm{X}]^{\alpha(\mathrm{X})-1}\,(\alpha(\mathrm{X})-1)\,[\mathrm{X}]^{(n+1)}\right)\\ &= \alpha(\mathrm{A})-1\geq 0\ ?\ \alpha(\mathrm{A})-1+\deg_{\mathrm{X}}\left(\mathrm{X}^{(n+1)}\right):0\end{aligned}$$

since we cannot employ theorem 2. Rather, our premise ensures $\deg_{\mathrm{X}}\left(\mathrm{X}^{(n+1)}\right)=\deg_{\mathrm{X}}\left(f_{\mathrm{X}}^{(n)}\right)$ because $\mathrm{X}^{(n+1)}=f_{\mathrm{X}}^{(n)}$, and therefore

$$\deg_{\mathrm{X}}\left(f_{\mathrm{X}}^{(n+1)}\right)\geq\alpha(\mathrm{A})-1\geq 0\ ?\ \alpha(\mathrm{A})-1+\deg_{\mathrm{X}}\left(f_{\mathrm{X}}^{(n)}\right):0$$

□

Theorems 3 and 4 reveal computational challenges when derivatives for the Taylor series are calculated by equation 7 by virtue of the implicitly nature of 7 [12].

Suppose we aim to minimize the truncation error of the Taylor series by including terms up to order $n = 100$. Let P_n be the set of all partitions of n and $P_{n,|\mathscr{S}|}$ be the set of all distinct partitions of n into $|\mathscr{S}|$ non negative integers; then, $|P_n| \leq \left|P_{n,|\mathscr{S}|}\right|$. When truncating the Taylor series at $n = 100$, we must compute P_j for all $1 \leq j \leq 100$ when applying the formula of Faà di Bruno. When we get to $j = 100$, we must compute the $|P_{100}| = 190,569,292$ partitions of 100. Clearly, combinatorial explosion is an inevitable consequence of differentiation as outlined in theorem 7.

Moreover, when evaluating $f_{\mathrm{X}}^{(100)}$ for each $\mathrm{X} \in \mathscr{S}$, theorem 4 requires the evaluation of $[\mathrm{X}]^{100}$. If our implementation of equation 7 is done using double-precision (64-bit) numbers, then the boundary of computation is defined by

- The positive and negative numbers closest to zero are $\pm 2^{-1074} \approx \pm 5 \cdot 10^{-324}$.
- The positive and negative normalized numbers closest to zero are $\pm 2^{-1022} \approx \pm 2.2250738585072020 \cdot 10^{-308}$.
- The finite positive and finite negative numbers furthest from zero are $\pm\bigl(1 - (1/2)^{53}\bigr)2^{1024} \approx \pm 1.7976931348623157 \cdot 10^{308}$.

Suppose that $[\mathrm{X}] = \pm 5 \cdot 10^4$; then $\pm\left(5 \cdot 10^4\right)^{100} = \pm 5 \cdot 10^{400}$ ensures numerical overflow; likewise, if $[\mathrm{X}] = \pm 3 \cdot 10^{-4}$, then $\pm\left(3 \cdot 10^{-4}\right)^{100} = \pm 3 \cdot 10^{-400}$ guarantees numerical underflow. Although $n = 100$ is a rather large derivative order to compute, overflow and underflow may occur at lower order derivatives, since $\mathbf{f}^{(n)}$ contains products of species' concentrations and products of rate coefficients, each raised to a monotonically increasing exponent.

4 A Dynamic Programming Implementation

Because the combinatorics of computing high-order derivatives are too burdensome, we aim to exploit the structure of the equations of theorem 1 to yield more efficient results. As an initial attempt, a naïve implementation is given in algorithm 1.

Algorithm 1: Deriv

```
input  : $X$, an array of species' concentrations
         $k$, an array containing the reaction rates
         $\alpha$, an array containing reactant stoichiometric coefficients
         $\beta$, an array containing product stoichiometric coefficients
         $n$, the derivative order to be calculated
output: $D$, an array of the $n^{\text{th}}$ derivative components evaluated using $X$

if $n < 0$ then return 0;
$L \leftarrow$ length($X$);
init $V \leftarrow 0$;
for $\ell \leftarrow 1$ to length($k$) do
    $W \leftarrow (\beta[\ell] - \alpha[\ell]) \cdot k[\ell]$;
    foreach distinct partition $P$ of $n$ into $L$ non negative integers do
        $U \leftarrow$ multinomial($n, P$);
        for $i \leftarrow 1$ to $L$ do
            init $d \leftarrow 0$;
            foreach partition $\pi$ in partitions($P$) do
                $a \leftarrow$ exponentiation($X[i], \alpha[i]-$length($\pi$));
                for $j \leftarrow 0$ to length($\pi$)$-1$ do
                    $a \leftarrow a \cdot (\alpha[i] - j)$;
                end
                foreach block $B$ in $\pi$ do
                    $a \leftarrow a \cdot$deriv($X, k, \alpha, \beta,$length($B$)$-1$);
                end
                $d \leftarrow d + a$;
            end
            $U \leftarrow Ud$;
        end
        $W \leftarrow W + U$;
    end
    $V \leftarrow V + W$
end
return $V$
```

Algorithm 1 calls functions `Multinomial`, `Partitions`, and `Exponentiation`. The functions `Multinomial` and `Exponentiation` are easily implemented and do not present any particular challenges; `Partitions` is a significanly more challenging routine to efficiently implement, and remains a topic unto itself [14].

A naïve implementation requires transversing the directed acyclic graph (DAG) of distinct tuples given by partitioning n into $|\mathscr{S}|$ non negative integers; each tuple is associated with a set of iterations. This approach is rather inefficient, since we may compute the same derivative term for multiple tuples.

Example 4. Both the reaction $2\,\mathrm{A} + \mathrm{B} \xrightarrow{k} \mathrm{C}$ as given in example 1 and set of reactions in example 2 have $|\mathscr{S}| = 3$. For each $\mathrm{X} \in \mathscr{S}$, $f_{\mathrm{X}}^{(2)}$ when derived via the algorithm of theorem 1, DAG representation given in figure 1.

Because each element of a tuple corresponds to the order of a derivative that must be calculated, branches of the DAG must be transversed multiple times. The total number of times $\mathbf{f}$ must be evaluated is 45: 15 as depicted in figure 1 for each of the 3 species. When computing $f_{\mathrm{X}}^{(3)}$ each $\mathrm{X} \in \mathscr{S}$, the entire DAG of figure 1 is transversed 6 times, as well as transversing various subgraphs.

One possible resolution to the inefficiency of algorithm 1 is by "memoization," where derivative terms are stored in a map object. If the map does not contain the needed derivative term, then it is computed and stored in the map, otherwise the value is accessed.

Let **map** be an object defined by $(\mathrm{X}, n) \mapsto f_{\mathrm{X}}^{(n-1)}$ for each $\mathrm{X} \in \mathscr{S}$ such that $f_{\mathrm{X}}^{(0)} = f_{\mathrm{X}}$. Algorithm 2 constitues a more efficient algorithm than that of 1, since values are only computed when needed, and are otherwise accessed in the map object.

Even when using a map object for a dynamic programming implementation, we do not avoid the combinatorial difficulties presented by equation 7. Wheather we choose a "top-down" or a "bottom-up" approach, increasing the derivative order restricts our computational abilities, particularly for large $|\mathscr{S}|$. It is therefore critical that we employ calculus methods to solve instances of 7 whenever possible [13].

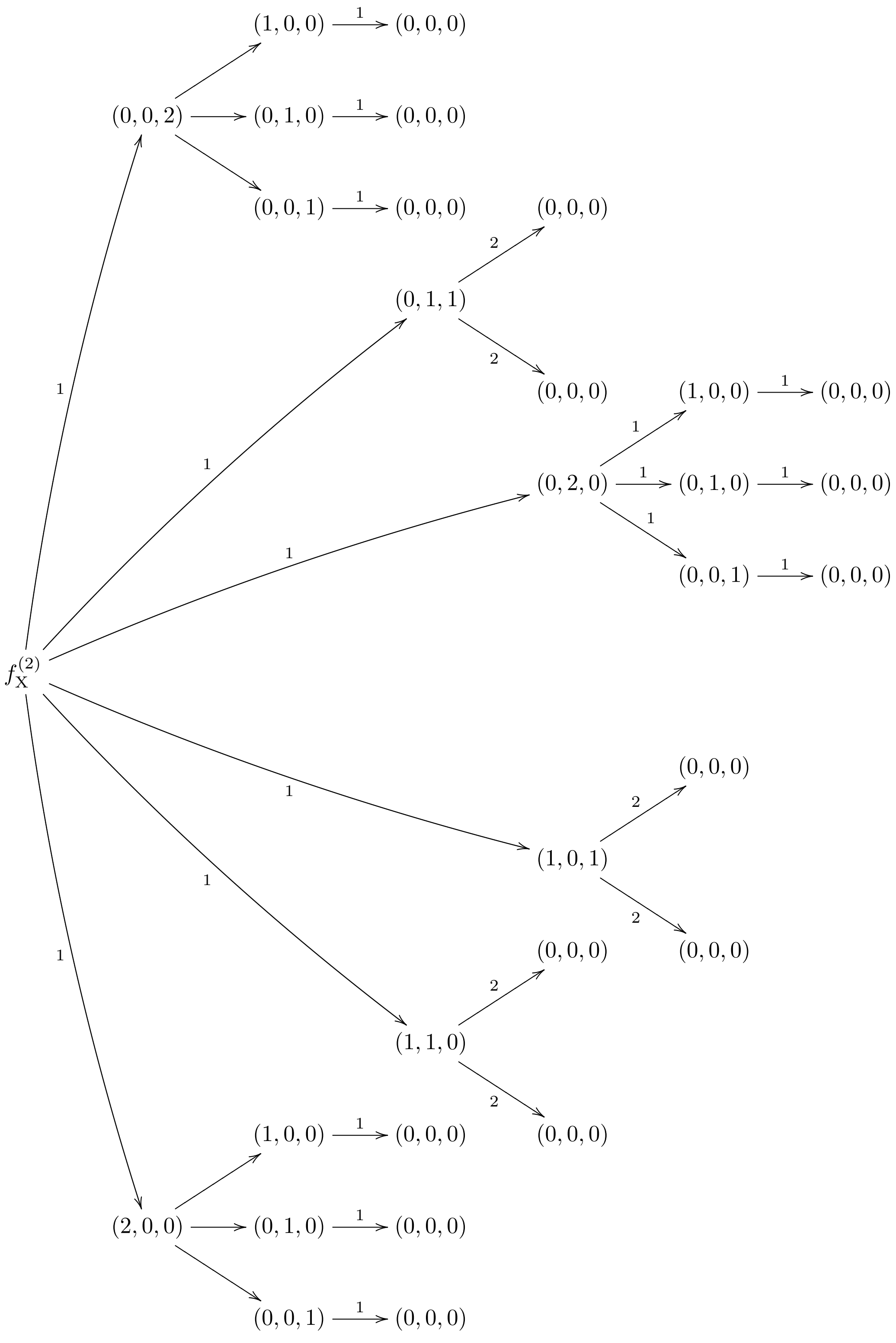

Figure 1: Tuples needed for recursive differentiation when computing $f_{\mathrm{X}}^{(2)}$ for each $\mathrm{X} \in \mathscr{S}$, where $\mathscr{S} = \{\mathrm{A}, \mathrm{B}, \mathrm{C}\}$. Arrows indicate recurssion; superscripts represent number of equivelant recurssive calls.

Algorithm 2: Deriv

```
input  : $X$, an array of species' concentrations
         $k$, an array containing the reaction rates
         $\alpha$, an array containing reactant stoichiometric coefficients
         $\beta$, an array containing product stoichiometric coefficients
         $n$, the derivative order to be calculated
output: $D$, an array of the $n^{\text{th}}$ derivative components evaluated using $X$

$L \leftarrow$ length($X$);
var $M \leftarrow$ map for $\ell \leftarrow 1$ to length($k$) do
    map $(\ell, 0) \leftarrow X[\ell]$
end
init $V \leftarrow 0$;
for $\ell \leftarrow 1$ to length($k$) do
    $W \leftarrow (\beta[\ell] - \alpha[\ell]) \cdot k[\ell]$;
    foreach distinct partition $P$ of $n$ into $L$ non negative integers do
        $U \leftarrow$ multinomial($n, P$);
        for $i \leftarrow 1$ to $L$ do
            init $d \leftarrow 0$;
            foreach partition $\pi$ in partitions($P$) do
                $a \leftarrow$ exponentiation($X[i], \alpha[i]-$length($\pi$));
                for $j \leftarrow 0$ to length($\pi$)$-1$ do
                    $a \leftarrow a \cdot (\alpha[i] - j)$;
                end
                foreach block $B$ in $\pi$ do
                    if $M$ does not contain key $X_i$,length($B$)$-1$ then
                        $M[i,$length($B$)$-1] \leftarrow$deriv($X, k, \alpha, \beta,$length($B$)$-1$)
                    end
                    $a \leftarrow aM[i,$length($B$)$-1]$;
                end
                $d \leftarrow d + a$;
            end
            $U \leftarrow Ud$;
        end
        $W \leftarrow W + U$;
    end
    $V \leftarrow V + W$
end
return $V$
```

5 Conclusion

We herein consider the deterministic approach of modeling the time evolution of species' concentrations in a chemical system, where they are considered as a continuous, wholly predictable process which is governed by a set of differential equations. Because applications of this effort include biopharmaceutical, accuracy in the numerical method is of great importance. We identify three numerical consequences from theorem 7 that restrict our numerical and compoutational integrity:

- the number of partitions of n into $|\mathscr{S}|$ non negative integers grows quickly.
- derivative terms for a high-order Taylor series approach our boundary of compoutational abilty.
- implementations of theorem 7 may be inefficient, requiring redundant derivative term computation.

A dynamic programming implementation is able to resolve our third concern about theorem 7. However, the results of theorems 3 and 4 are unavoidable, thereby ensuring a trade-off between minimizing truncation error and maximizing numerical accuracy.

Additional topics regarding the Taylor series method include choosing an adjustible step size and the stiffness of the system. Both of these do not address or resolve the first two points of numerical complexity; rather, they add to our list of concerns.

From this perspective, it seems reasonable to employ the Taylor series approach only when the number of species is relatively low, and when the derivative order is not unreasonablly large.

Acknowledgments

Ryan M. Brown thanks Gilles Gnacadja and Graham Matthews for continued support and guidance. This work is indirectly supported by Geoffery Buhl and Bill Wolfe, as well as other faculty members at California State University Channel Islands.

References

[1] Baeza Baeza, J. J., Pérez Plá, F., and Ramis Ramos, G. (1992). "On the Integration of Kinetic Models Using a High-Order Taylor Series Method." *Journal of Chemometrics.* Vol 6, 231-246.

[2] Baeza Baeza, J. J., Pérez Plá, F., and Ramis Ramos, G. (1992). "Stiffness-Adaptive Taylor Method for the Integration of Non-Stiff and Stiff Kinetic Models." *Journal of Computational Chemistry.* Vol 13, Issue 7, 810-820.

[3] Butcher, J. C. (2003). *Numerical Methods for Ordinary Differential Equations.* Chichester, West Sussex, England: Wiley.

[4] Cornish-Browden, A. and Hofmeyr, JH. (2002). "The Role of Stoichiometric Analysis in Studies of Metabolism: An Example." *Journal of Theoretical Biology.* Vol 216

Complex Biochemical Reaction Mechanisms." *Progress in Biophysics & Molecular Biology.* Volume 86, 77-112.

[5] Feinberg, M. (1995). "The Existence and Uniqueness of Steady States for a Class of Chemical Reaction Networks." Archive for Rational Mechanics and Analysis: Springer-Verlag. Volume 132, 311-370.

[6] Gavalas, G. (1968). *Nonlinear Differential Equations of Chemically Reacting Systems.* Berlin: Springer-Verlag.

[7] Gunawardena, J. (2003). "Chemical Reaction Network Theory for *in-silico* Biologists." Harvard University: Bauer Center for Genomics Research.

[8] Kennealy, J. and Moore, W. (1977). "A Numerical Method for Chemical Kinetics Modeling Based on the Taylor Series Expansion." *The Journal of Physical Chemistry.* Vol. 81, No. 25,

[9] Leipnik, Roy and Pearce, C. (2007). "The Multivariate Faà di Bruno Formula and Multivariate Taylor Expansions with Explicit Integral Remainder Term." Vol 48, 327341.

[10] Hairer E., Nørsett S.P. and Wanner G. (1993). *Solving Ordinary Differential Equations I.* (2nd Ed.) Berlin: Springer-Verlag.

[11] Mazkewitsch, D. (1963). "The n-th Derivative of a Product." *The American Mathematical Monthly.* Vol. 70, No. 7.

[12] Reverter, F., and Oller J. M. (1997). "A Modified Taylor Series Method for Solving Initial-Value Problems in Ordinary Differential Equations." *International Journal of Computer Mathematics.* Volume 65 Issue 3-4, 231-246.

[13] Steinfeld, J., Francisco, J, and Hase, W. (1989). *Chemical Kinetics and Dynamics.* New Jersey: Prentice Hall.

[14] Zoghbiu, A. and Stojmenović, I. (1998). "Fast Algorithms for Generating Integer Partitions". *International Journal of Computer Mathematics.* Vol 70. pp. 319 332