

Finding All Steady State Solutions of Chemical Kinetic Models

Jason W. Zwolak*, John J. Tyson**, and Layne T. Watson*

Departments of Computer Science* and Biology**

Virginia Polytechnic Institute and State University

Blacksburg, Virginia 24061-0106

e-mail: jzwolak@vt.edu

Keywords: Chemical kinetic models, finding steady states, POLSYS_PLP, homotopy, biological models

Abstract

Ordinary differential equations are used frequently by theoreticians to model kinetic process in chemistry and biology. These systems can have stable and unstable steady states and oscillations. This paper presents an algorithm to find all steady state solutions to a restricted class of ODE models, for which the right hand sides are linear combinations of rational functions of variables and parameters. The algorithm converts the steady state equations into a system of polynomial equations and uses a globally convergent homotopy method to find all the roots of the system of polynomials. All steady state solutions of the original ODEs are guaranteed to be present as roots of the polynomial equations. The conversion may generate some spurious roots that do not correspond to steady state solutions. The stability properties of the steady states are not revealed. This paper explains the algorithms used and gives results for a cell cycle modeling problem.

1. INTRODUCTION

The rate of change of concentration x_i of a chemical involved in a number of reaction in a well-stirred solution is described by an ordinary differential equation of the form $dx_i/dt = \sum_j v_{ij} r_j$, where v_{ij} is the stoichiometric coefficient of species i in reaction j ($v_{ij} > 0$ for products and $v_{ij} < 0$ for reactants) and r_j is the rate of the j th reaction. These rates are typically polynomial functions (mass-action kinetics) or rational functions (Michaelis-Menten or Hill functions) of the concentrations of the chemical species taking part in a a reaction. A steady state solution of the rate equations is a constant vector $(x_1^*, x_2^*, \dots, x_n^*)$ that satisfies the system of algebraic equations $\sum v_{ij} r_j = 0$, $i = 1, \dots, n$. For a chemical (or biochemical) reaction system, we are interested in finding all feasible steady state solutions ($x_i^* \geq 0$ for all i). Identification of steady states is usually the first step in a qualitative analysis of the kinetic properties of a reaction network. After finding all steady states,

one typically analyzes their stability properties and then tracks the steady states as parameters are varied, looking for changes in stability (bifurcations). To get a complete picture of the qualitative dynamics of a nonlinear reaction network, it is crucial to know, at the start, all feasible steady states. In this paper, we provide a computational method that is guaranteed to find all steady state solutions of a set of chemical reaction equations, provided every rate law is a polynomial function or rational function. In this case, the steady state equation can be transformed into a system of polynomial equations. Probability-one homotopy theory provides a practical algorithm to find all roots of polynomial systems.

In Section 2 we describe a simple example of biochemical reaction network (for control of the cell cycle in frog eggs) that we use to illustrate the method. The structure of the ODE right hand sides (required for the methods described in this paper) is described in Section 3. The algorithm for finding all the steady states is described in Section 4. The claim of finding all steady states in a model, guaranteed, relies on some technical assumptions about the ODE right hand side that are generically true, and a mathematical software package POLSYS_PLP [11] (Section 5). Parallel methods for this problem are discussed in Section 6. The results from the example problem and the conclusion of this paper are in Section 7 and Section 8, respectively.

2. EXAMPLE ODE MODEL

The example problem is a simplified version of the cell cycle model presented in Marlovits [6]. The network of proteins can be seen in Figure 1. This is a simple model of the MPF activity in frog egg extracts. The system of ODEs for this model are

$$\frac{dM}{dt} = v'_d(1-D)(C_T - M) + v''_d D(C_T - M) - v'_w(1-W)M + v''_w WM, \quad (1)$$

$$\frac{dD}{dt} = \frac{v_d M(1-D)}{K_{md} + (1-D)} - \frac{v_{dr} D}{K_{mdr} + D}, \quad (2)$$

$$\frac{dW}{dt} = -\frac{v_w MW}{K_{mw} + W} + \frac{v_{wr}(1-W)}{K_{mwr} + (1-W)}, \quad (3)$$

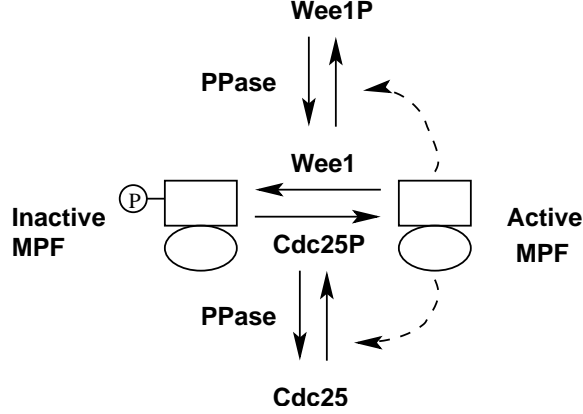


Figure 1. The network of proteins for the example problem used in this paper. The example problem is a simplified version of the model presented in Marlovits [6]. In this network active MPF phosphorylates Wee1 and Cdc25. There is an enzyme, PPase, that dephosphorylates Wee1 and Cdc25 (the name of the actual enzyme that catalyzes the dephosphorylation reactions is not important, the enzyme concentration is assumed to be constant). Wee1 phosphorylates MPF causing MPF to become inactive. Cdc25 dephosphorylates MPF causing MPF to become active.

where M is active MPF, $C_T - M$ is inactive MPF, D is active Cdc25 (phosphorylated), $1 - D$ is inactive Cdc25, W is active Wee1 (not phosphorylated), and $1 - W$ is inactive Wee1. The v s and K s are rate constants and Michaelis constants, respectively. Note that the reverse enzyme in the D and W equations does not appear. This enzyme is assumed to have a constant value and is absorbed into the rate constants v_{dr} and v_{wr} .

3. STRUCTURE OF ODE RHS

The algorithm described in this paper will only work with systems of ODEs

$$\frac{dx}{dt} = H(x)$$

of a certain form. Ultimately the steady state problem $H(x) = 0$ must be transformed into a system of polynomials. For each system $H(x) = 0$ there must be an algorithm to convert the right hand sides (RHS) to polynomial equations. One such algorithm is described in Section 4 and works on problems of the nature described below. The restriction imposed in Section 4 is that each component $H_i(x)$ of the right hand side of the system of ODEs be a linear combination of rational functions.

Most biological models of molecular networks have linear combinations of rational functions for the right hand side of their system of ODEs. In fact, the

right hand sides are usually even more restricted to mass action and Michaelis-Menten kinetics. Mass action kinetics have the form

$$k * S_1 * S_2 * \dots * S_n,$$

where k is a rate constant (parameter) and S_i represents the concentration of a protein (variable). Michaelis-Menten kinetics have the form

$$\frac{k * S * E}{K_m + S},$$

where k is a rate constant (parameter), K_m is a Michaelis constant (parameter), S is the substrate concentration (variable), and E is the enzyme concentration (variable). Clearly these kinetics (illustrated in (1)–(3)) are rational functions.

4. ALGORITHM

The algorithm presented here covers the type of kinetics described in the previous section. The steady state system must first be converted to a system of polynomials. The system of polynomials can then be given to a homotopy algorithm (POLSYS_PLP), which finds *all* zeros of the system. The roots must be checked in the steady state equations to verify that they are not extraneous roots introduced by the conversion to a system of polynomials. The verified steady states constitute *all* steady states of the ODE, assuming there are finitely many steady states. The case of infinitely many steady states is a degenerate situation, and is not considered further here.

These steps are now illustrated on the example model. First, set up the steady state equations by setting the right hand side of the ODE equal to zero. The example problem yields

$$0 = v'_d(1 - D)(C_T - M) + v''_d D(C_T - M) \quad (4)$$

$$- v'_w(1 - W)M + v''_w WM,$$

$$0 = \frac{v_d M(1 - D)}{K_{md} + (1 - D)} - \frac{v_{dr} D}{K_{mdr} + D}, \quad (5)$$

$$0 = -\frac{v_w MW}{K_{mw} + W} + \frac{v_{wr}(1 - W)}{K_{mwr} + (1 - W)}. \quad (6)$$

Second, multiply each equation by a constructed polynomial to eliminate the denominators. Note that the steady states are preserved by this transformation. Multiplying through by a polynomial may add extraneous zeros to the system, but they can be eliminated by substituting into the original steady state equations. The constructed polynomials for the example problem are

$$1, \quad (7)$$

$$(K_{md} + (1 - D))(K_{mdr} + D), \quad (8)$$

$$(K_{mw} + W)(K_{mwr} + (1 - W)). \quad (9)$$

Multiplying the right hand sides by these polynomials yields the polynomial system of equations

$$0 = v'_d(1 - D)(C_T - M) + v''_d D(C_T - M) - v'_w(1 - W)M + v''_w WM, \quad (10)$$

$$0 = v_d M(1 - D)(K_{mdr} + D) - v_{dr} D(K_{md} + (1 - D)), \quad (11)$$

$$0 = -v_w MW(K_{mwr} + (1 - W)) + v_{wr}(1 - W)(K_{mw} + W). \quad (12)$$

A globally convergent, probability-one homotopy algorithm (implemented in POLSYS_PLP) is then applied to find all the zeros of (10)–(12), among which the steady states of the ODE (1)–(3) lie. The theoretical justification for the claim that *all* the zeros are found is given in the next section.

Finally, it is necessary to verify which zeros of the polynomial system are steady states of the model. All the zeros of the polynomial system are checked by substituting them into the steady state equations.

Pseudocode for finding all the steady states follows:

begin

read ODE system right hand side $H = (H_1, \dots, H_n)^t$;

for $i := 1$ to n **do**

$L_i := H_i * \text{LCM}(\text{denominators of terms comprising } H_i)$;

done

apply POLSYS_PLP to the polynomial system $L(x) = 0$;

check whether each zero z returned by POLSYS_PLP satisfies $H(z) = 0$;

end

5. POLSYS_PLP

The mathematical software package POLSYS_PLP makes this work possible and deserves some explanation. POLSYS_PLP efficiently finds all zeros of a system of polynomials (assumed to have finitely many zeros, the nondegenerate case). The zeros are found by tracking zero curves of a homotopy map from a start system to a target system. The start system is another system of polynomials with easily found zeros and a structure similar to the target system. This section describes in some detail the method implemented by the code POLSYS_PLP as described in Wise et al. [11] and cites theorems (also from Wise et al. [11]) showing that all the zeros can be reliably found.

Polynomial systems of equations have the form

$$F_i(z) = \sum_{j=1}^{n_i} [c_{ij} \prod_{k=1}^n z_k^{d_{ijk}}] = 0, \quad i = 1, \dots, n, \quad (13)$$

where the c_{ij} are complex, and the d_{ijk} are nonnegative integers. The degree of $F_i(z)$ is

$$d_i = \max_{1 \leq j \leq n_i} \sum_{k=1}^n d_{ijk},$$

and the total degree of the system (13) is

$$d = \prod_{i=1}^n d_i. \quad (14)$$

Following Wise et al. [11], define a homotopy map $\rho : [0, 1] \times \mathbf{C}^n \rightarrow \mathbf{C}^n$ by

$$\rho(\lambda, z) = (1 - \lambda)G(z) + \lambda F(z). \quad (15)$$

$\lambda \in [0, 1]$ is the homotopy parameter, $G(z) = 0$ is the start system, and $F(z) = 0$ is the target system. The start system is constructed with a partitioned linear product (PLP) structure, explained next.

PLP structure refers to a particular way of constructing the start system, and matching it to the target system. Following Wise et al. [11], let $P = (P_1, P_2, \dots, P_n)$ be an n -tuple of partitions P_i of the set $\{z_1, z_2, \dots, z_n\}$. That is, for $i = 1, 2, \dots, n$, $P_i = \{S_{i1}, S_{i2}, \dots, S_{im_i}\}$, where S_{ij} has cardinality $n_{ij} \neq 0$, $\cup_{j=1}^{m_i} S_{ij} = \{z_1, z_2, \dots, z_n\}$, and $S_{ij_1} \cap S_{ij_2} = \emptyset$ for $j_1 \neq j_2$. For $i = 1, 2, \dots, n$ and $j = 1, 2, \dots, m_i$ define d_{ij} to be the degree of the component F_i in only the variables of the set S_{ij} , that is, considering the variables of $\{z_1, z_2, \dots, z_n\} \setminus S_{ij}$ as constants. For convenience of defining the start system, rename the variables z_k such that $S_{ij} = \{z_{ij1}, z_{ij2}, \dots, z_{ijn_{ij}}\}$. The start system is represented mathematically by $G_i(z) = \prod_{j=1}^{m_i} G_{ij}$, where

$$G_{ij} = \begin{cases} \left(\sum_{k=1}^{n_{ij}} c_{ijk} z_{ijk} \right)^{d_{ij}} - 1, & \text{if } d_{ij} > 0; \\ 1, & \text{if } d_{ij} = 0, \end{cases} \quad (16)$$

for $i = 1, 2, \dots, n$, where the numbers $c_{ijk} \in \mathbf{C}_0 = \mathbf{C} \setminus \{0\}$ are chosen at random. Note that all the start system's zeros can be easily found by solving systems of linear equations [11].

THEOREM 1 [5]. Let $f : \mathbf{C}^n \rightarrow \mathbf{C}^n$ be a system of polynomials and $U \subset \mathbf{C}^n$ be (Zariski) open. Define $N(f, U)$ to be the number of nonsingular solutions to $f = 0$ that are in U . Assume that there are positive integers r_1, \dots, r_n and m_1, \dots, m_n and finitely