

# Recasting Nonlinear Differential Equations as S-Systems: A Canonical Nonlinear Form

MICHAEL A. SAVAGEAU AND EBERHARD O. VOIT\*

*Department of Microbiology and Immunology, The University of Michigan,  
Ann Arbor, Michigan 48109*

*Received 6 May 1987; revised 28 July 1987*

---

## ABSTRACT

An enormous variety of nonlinear differential equations and functions have been recast exactly in the canonical form called an S-system. This is a system of nonlinear ordinary differential equations, each with the same structure: the change in a variable is equal to a difference of products of power-law functions. We review the development of S-systems, prove that the minimum for the range of equations that can be recast as S-systems consists of all equations composed of elementary functions and nested elementary functions of elementary functions, give a detailed example of the recasting process, and discuss the theoretical and practical implications. Among the latter is the ability to solve numerically nonlinear ordinary differential equations in their S-system form significantly faster than in their original form through utilization of a specially designed algorithm.

---

## 1. INTRODUCTION

The purpose of this paper is to introduce the wider community of applied mathematicians to a very general canonical nonlinear form, the S-system. The current evidence in support of its generality is based upon a large collection of empirical examples and a demonstration that many functions already recognized as representing very general classes of functions are either special cases of or equivalent to this canonical form. However, many of the relevant results are unpublished, published in journals that are not readily accessible, or embedded in specific applications published in diverse areas. In providing this introduction we intend to give background and motivation rather than a series of mathematical proofs. It is our hope that this will stimulate others to pursue some of the open questions that abound in this area.

---

\*Present address: Department of Biometry, Medical University of South Carolina, Charleston, SC 29425-2503.

In the following sections we shall summarize the developments leading to S-systems, give a proof that defines the minimum size of the class of functions that can be recast in the canonical S-system form, provide a detailed example of recasting, and discuss the theoretical and practical implications of recasting. The appendices give further examples of recasting, and results demonstrating the efficiency of a numerical algorithm based upon the canonical S-system form are presented.

## 2. BACKGROUND

### *LOCAL NONLINEAR REPRESENTATIONS*

Two decades ago, the need for a tractable nonlinear description of biological and other organizationally complex systems that would capture the essential nonlinear features of synergism and saturation led to the development of the Power-Law Formalism [14–16]. In this formalism, the elemental processes of a system are defined by the hierarchical level being modelled and are assumed to be governed by logarithmically differentiable functions, i.e., functions whose logarithm is differentiable with respect to the logarithm of its arguments. When the logarithm of the function is represented as a function of the logarithms of its arguments [ $\log Z = f(\log X)$ ] one can develop the Taylor series under appropriate conditions. Then the local representation consisting of the constant and linear terms in this logarithmic space is given by

$$\log Z = \log Z_0 + \sum_{j=1}^n \left[ \partial(\log Z) / \partial(\log X_j) \right]_0 (\log X_j / X_{j0}) \quad (1)$$

which corresponds to a product of power-law functions in Cartesian space

$$Z = \alpha \prod_{j=1}^n X_j^{g_j} \quad (2)$$

where

$$g_j = (\partial Z / \partial X_j)_0 (X_{j0} / Z_0) \quad \text{and} \quad \alpha = Z_0 \prod_{j=1}^n X_{j0}^{-g_j}.$$

While this is a local representation, empirical tests show that it is valid over a surprisingly large range of variation in the variables [20, 38, 43]. Reasons for this are discussed in detail elsewhere [20, 21, 26, 43].

Each elemental process has a similar local representation in this Power-Law Formalism. However, in many situations it is natural to combine a set of elemental processes into a single aggregate process. For example,  $Z_1$  and  $Z_2$  could represent the rates of two separate reactions that produce the same product, and  $Z_{\text{net}}$  the net rate of product formation. In general,  $Z_1$  and  $Z_2$

can be any positive-valued functions whose logarithms are differentiable and with  $Z_{\text{net}}$  as their sum:

$$\begin{aligned} Z_1 &= f_1(X_1, \dots, X_n) & Z_2 &= f_2(X_1, \dots, X_n) \\ Z_{\text{net}} &= f_1(X_1, \dots, X_n) + f_2(X_1, \dots, X_n) = f_{\text{net}}(X_1, \dots, X_n) \end{aligned} \quad (3)$$

Each aggregate process also can be represented as a product of power-law functions in the Power-Law Formalism; in the example of Equation (3) we obtain

$$Z_{\text{net}} = \alpha \prod_{j=1}^n X_j^{g_j} \quad (4)$$

Depending upon the degree of aggregation assumed before formulation of the local description, there are several variants of this Power-Law Formalism [30, 43]. In the mathematically and biologically most fruitful variant, all processes that contribute to the increase in a system component and all those that contribute to its decrease are first aggregated into one net process each and then represented [15, 30, 43]. The result is a systematically structured system of nonlinear differential equations that has been called an S-system:

$$\dot{X}_i = \alpha_i \prod_{j=1}^n X_j^{g_{ij}} - \beta_i \prod_{j=1}^n X_j^{h_{ij}} \quad i = 1, 2, 3, \dots, n \quad (5)$$

where  $X_i$ ,  $\alpha_i$  and  $\beta_i$  are real nonnegative,  $g_{ij}$  and  $h_{ij}$  are real, and  $\dot{X} = dX/dt$ .

Among the obvious variants within the Power-Law Formalism are the extremes of no aggregation (Generalized Mass Action systems)

$$\dot{X}_i = \sum_k \alpha_{ik} \prod_{j=1}^n X_j^{g_{ijk}} - \sum_k \beta_{ik} \prod_{j=1}^n X_j^{h_{ijk}} \quad i = 1, 2, 3, \dots, n \quad (6)$$

and complete aggregation to a single term for each variable (Half systems)

$$\dot{X}_i = \alpha_i \prod_{j=1}^n X_j^{g_{ij}} \quad \text{or} \quad -\beta_i \prod_{j=1}^n X_j^{h_{ij}} \quad i = 1, 2, 3, \dots, n \quad (7)$$

These variants were considered in the initial development of the Power-Law Formalism and abandoned in favor of the S-system form [15]. There are three primary reasons that can be cited: mathematical tractability, efficiency of numerical solution, and accuracy. S-systems are mathematically more tractable. For example, they permit explicit symbolic solutions for nonzero steady states [15] and explicit symbolic determination of conditions for local

stability [19]. In contrast, these advantages are lost with the Generalized Mass Action representation, and Half systems are incapable of representing a nonzero steady state. S-systems have a regular structure that has permitted us to develop algorithms for their numerical solution that are extremely efficient, in many cases one to two orders of magnitude faster than other state-of-the-art methods [7; see also Appendix A]. Other variants within the Power-Low Formalism also can take advantage of such improvements in efficiency. However, the advantages in efficiency are less for Generalized Mass Action systems, for there are more computations at each iteration. In the case of Half systems, some of whose variables must go either to zero or to infinity, there are problems with scaling and computational stability. Finally, with regard to the accuracy of local representation, S-systems have been shown to be more accurate than Generalized Mass Action systems [43]. Half systems have no steady states with all  $X_i$  nonzero and hence cannot represent directly the local behavior about such a state.

#### *BIOLOGICAL APPLICATIONS*

An enormous variety of phenomena from many different areas of biology have been described and analyzed with S-systems, including biochemical systems (for reviews see [17, 20, 26, 27, 29]), genetic circuits [18, 20, 25, 28], developmental systems [22, 36, 37], immunological networks [5, 6], and ecological interactions [32, 33, 41, 42]. These applications have revealed a number of design principles used in biological systems. Some examples are given in the following paragraphs (see [27] for a brief review).

For a sequence of  $n$  consecutive reactions (i.e., a biosynthetic pathway, as represented in Figure 1a) controlled by negative feedback, there are  $2^{n(n+1)/2}$  possible patterns of feedback interactions: the product of each reaction in the sequence is capable of inhibiting each of the reactions that preceded it in the sequence. Is there a pattern of feedback interactions that is optimal in some sense? By describing such sequences with S-systems we obtained the steady-state solutions in explicit symbolic form. We then compared the steady-state behavior of systems with alternative patterns of feedback control and showed by several different criteria that the simple pattern with only the end product ( $X_n$ ) inhibiting the first reaction ( $X_0 \rightarrow X_1$ ) in the sequence is optimal. This analysis provided the first explanation for the nearly universal occurrence of this pattern in unbranched biosynthetic pathways.

There were confusing reports of intermediate products in biosynthetic pathways (i.e.,  $X_i$  in Figure 1b) that inhibited the activity of the last reaction in the sequence, thereby providing an example of a negative feedforward interaction. We modeled such reaction sequences as S-systems and again obtained the steady-state solutions in explicit symbolic form. By systematic comparison of sequences in which the relevant intermediate product oc-

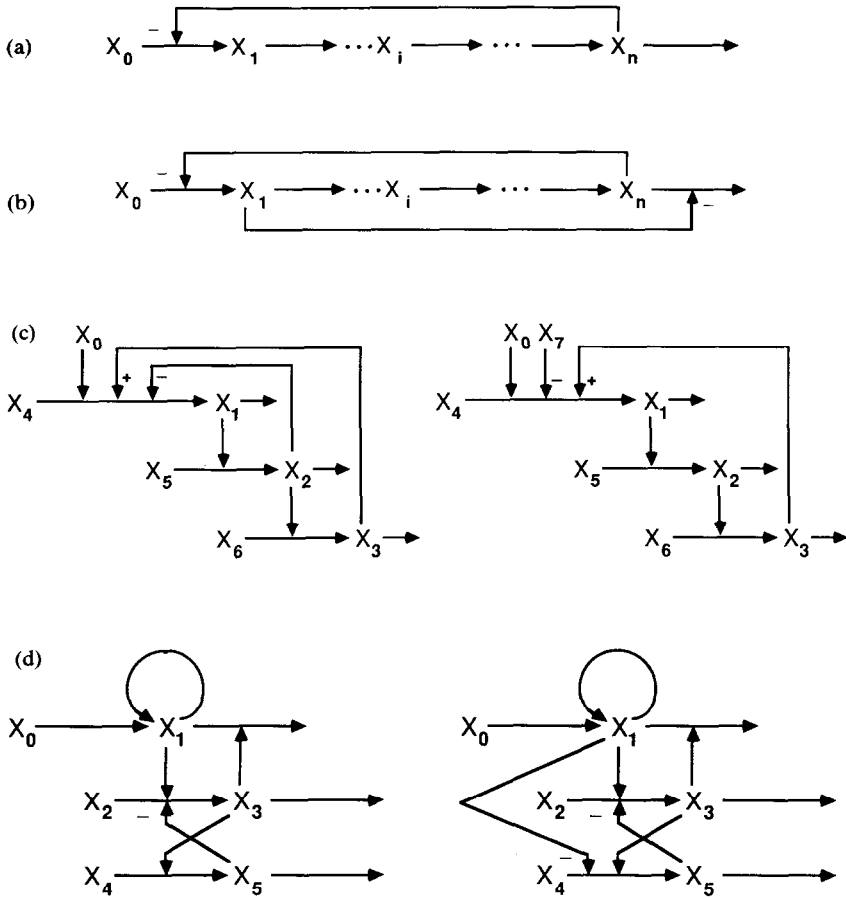


FIG. 1. (a) Feedback inhibition in biosynthetic pathways, (b) feedforward inhibition in biosynthetic pathways, (c) alternative circuits for gene regulation, and (d) alternative regulatory interactions in immune networks. See text for discussion.

cupied various positions in the sequence, we showed that the optimal design is one in which the product of the first reaction ( $X_1$ ) inhibits the last reaction. We also predicted the formation of complexes between the enzymes that catalyze the first and last reactions in the sequence. These predictions have recently been confirmed in detail by experimentalists.

Genes ( $X_0$ ) act as templates for transcription in the synthesis of messenger RNA molecules ( $X_1$ ), which represent copies of the hereditary information in the gene. Messenger RNA molecules in turn act as templates for translation in the synthesis of protein molecules ( $X_2$ ), which represent the

specific three-dimensional encoding of the genetic information. Proteins in the case of enzymes catalyze the conversion of specific substrates ( $X_6$ ) to specific products ( $X_3$ ). The transcription of such a gene is regulated by product molecules acting in concert with protein molecules, specified either by the gene itself ( $X_2$ ) or by other genes ( $X_7$ ). These alternative circuits of gene regulation are shown schematically in Figure 1c. We have modeled these with S-systems and have shown that the usage of each regulatory pattern is based on the physiological functions of the gene product and the normal extent of its expression in the organism's natural environment. These predictions are in agreement with the prototype circuits found experimentally in the best studied organisms.

Figure 1d shows alternative regulatory interactions in model immune networks. The antigen external to the body is denoted as  $X_0$ ; when it enters the body, it is denoted as  $X_1$ . Once in the body, antigen in this case grows autocatalytically and stimulates the production of effector lymphocyte cells ( $X_3$ ) from precursor cells. Effector lymphocytes promote clearance of antigen from the system and are themselves cleared after a certain lifetime in the system. A regulatory cell type, called a suppressor lymphocyte ( $X_5$ ), suppresses the production of effector lymphocytes and is itself cleared after a certain lifetime in the system. Alternative means of modulating the production of suppressor lymphocytes have been proposed; antigen and effector lymphocyte are directly involved in one case, and the effector lymphocyte alone is directly involved in the other. We have analyzed these alternative interactions by modeling the immune networks as S-systems. Our results show that a negative interaction involving antigen modulation of suppressor lymphocyte production is superior on the basis of several criteria for functional effectiveness of the immune network. Experimental results supporting these predictions have been published by others, and we have suggested specific experiments to further test these predictions.

General conclusions were made independent of the specific parameter values in each of these applications. This was possible because the use of S-systems allowed us to obtain steady-state solutions in explicit symbolic form, to formulate specific algebraic constraints among the parameter values so as to make well-controlled comparisons, and in some cases to compare an arbitrary number of alternatives simultaneously. The systematic structure of the S-system equations also has allowed us to develop extremely efficient algorithms for their numerical solution (see Discussion section, Appendix A, and [7, 8]). These advantages are rarely encountered in most other forms of nonlinear analysis.

#### *GLOBAL NONLINEAR REPRESENTATIONS*

In the late 1970s we discovered that this Power-Law Formalism, which was initially developed as a local nonlinear representation, in fact could represent globally a variety of nonlinear differential equations and functions.

Prominent examples are given by the well-known growth laws [21, 23] and probability distribution functions [24], which include many functions already known to represent very general classes of functions.

Two of the principles of recasting are based on the chain rule and the product rule of differentiation. By using these principles to recast an arbitrary sum of variables (Voit and Savageau, unpublished) we have demonstrated that linear systems are special cases of S-systems. Similarly, we have shown that S-systems, Half systems, Generalized Mass Action systems and Generalized Volterra systems of the form

$$\dot{X}_i = X_i \left( b_i + \sum_{j=1}^n a_{ij} X_j \right) \quad i = 1, 2, 3, \dots, n \quad (8)$$

are all canonical forms that can be transformed from one to the other [42]. To this list of general canonical forms we could add Binary systems, which are identical in form to Half systems except that the exponents are either ones or zeros (Voit, unpublished; Appendix B).

In the next section we shall give the principles of recasting that have allowed us to recast as S-systems any nonlinear differential equation or function that is a composite of elementary functions. These principles permit a constructive proof of the following theorem that provides a minimum estimate of the range of functions and differential equations that can be recast in the canonical S-system form.

### 3. THEOREM

Let

$$\dot{Z}_i = f_i(Z_1, Z_2, \dots, Z_n) \quad Z_i(0) = Z_{i0} \quad i = 1, 2, 3, \dots, n \quad (9)$$

be a set of differential equations wherein each  $f_i$  is composed of sums and products of elementary functions, or nested elementary functions of elementary functions.

Then there is a smooth change of variables that recasts Equations (9) into an S-system as follows:

$$\dot{X}_i = \alpha_i \prod_{j=1}^m X_j^{g_{ij}} - \beta_i \prod_{j=1}^m X_j^{h_{ij}} \quad X_i(0) = X_{i0} \quad i = 1, 2, 3, \dots, m \quad (10)$$

where  $X_i$ ,  $\alpha_i$ , and  $\beta_i$  are real nonnegative, and  $g_{ij}$  and  $h_{ij}$  are real.

There are  $m - n$  constraints that are generated in this recasting process; they occur as one of two different types. First, there are simple algebraic constraints among new variables that take the form

$$\phi_k(X_1, X_2, \dots, X_m) = 0 \quad (11)$$

where the  $\phi_k$  are elementary functions or nested elementary functions of

elementary functions. Each of these reduces by one the dimensionality of the recast system. Second, there are constraints between new and old variables that take the form

$$X_i X_j = Z_k \quad (12)$$

The  $X_i$  and  $X_j$  variables always occur together as a product in the logarithmic derivatives  $[d(\log X_i)/dt]$  of the recast system (e.g., see Equation 25). This corresponds to a simple translation-group symmetry (along the  $-45^\circ$  direction in the  $\log X_i$  vs.  $\log X_j$  plane) in the  $m$ -dimensional logarithmic space. Thus, all the system trajectories can be translated onto a single reference manifold of dimension  $m - 1$ , without changing the behavior of the system.

The definition of the new variables, as part of the recasting process, generates the constraints, which are automatically incorporated into Equations (10); the constraints also can be given explicitly as in Equations (11) and (12). Thus, the system behavior is restricted to a manifold of the same dimension as the original problem. Since the initial conditions satisfy Equations (11) and (12) as well, the constraint set is invariant under the flow of the S-system. In the case of Equation (12), one can choose arbitrarily an initial condition— $X_{i0}$  and  $X_{j0}$ , such that their product is equal to  $Z_{k0}$ —in each basin of the system; this set of choices defines the reference manifold.

*Note 1.* By elementary functions we mean those with explicit symbolic derivatives, such as  $\exp(x)$ ,  $\log(x)$ ,  $x^a$ , trigonometric functions, and hyperbolic functions.

*Note 2.* By an invariant constraint set we mean the following: if the initial conditions satisfy the constraints, then the differential equations will automatically satisfy the constraints for all  $t$ . Thus, any solution that starts on the manifold, stays on the manifold.

*Note 3.* In the following,  $t$  represents the independent variable, upper-case symbols are dependent variables, and Greek and lower-case symbols other than  $t$  designate parameters. We shall use the notation  $\dot{X}$  for the derivative  $dX/dt$ . The independent variable  $t$  can be incorporated into the S-system form by defining a variable  $X_i = t$  with  $\dot{X}_i = 1$  and hence forth does not retain a special status among the S-system variables. The presence of  $t$  can be detected by its derivative, which is equal to unity, and by its initial condition,  $X_i(t_0) = t_0$ .

#### 4. PROOF

We shall prove the theorem for systems of first-order ordinary differential equations. Higher-order systems are a minor extension, and nonlinear algebraic functions are a special case of the procedure for differential equations. The proof involves three principles of recasting that are applied iteratively until the recasting process is complete.



A. SYSTEM EQUATIONS

Each equation of a first-order system can be written in the form

$$\dot{X}_i = \sum_j T_{ij} \quad i = 1, 2, 3, \dots, n \quad (13)$$

where the terms  $T_{ij}$  are elementary functions or composite functions.  $T_{ij}$  can be written as

$$T_{ij} = \prod_k f_{ijk} \quad (14)$$

where the factors  $f_{ijk}$  are elementary functions, or nested elementary functions of elementary functions.

B. TRANSLATION OF VARIABLES TO THE POSITIVE ORTHANT

The S-system variables  $X_i$  must be positive real. Historically, this is because the original applications dealt with biochemical variables, which are positive real. More generally, logarithms of variables and power-law functions with negative exponents are encountered, and the arguments of these functions must be positive so as not to introduce discontinuities during recasting. If all the variables in the original set of equations are positive real, then proceed to step C. If the original system has any variable that takes on negative values, then it first must be translated to the positive orthant. Similarly, if such a variable should be generated at any stage in the recasting process, it must be translated to the positive orthant. This can be done in several ways.

If a variable is always negative, a simple sign change  $X_i \rightarrow -X_i$  is sufficient. If  $X_i$  is always greater than a constant  $c < 0$ , we can write  $X_i$  as  $X_i + c - c$  and replace  $X_i - c$  with  $X_{n+1}$  which is positive. Analogously, if  $X_i$  is always less than a positive constant  $c$ , we write  $X_i$  as  $X_i - c + c$  and define  $X_{n+1}$  as  $c - X_i$ . If the range of  $X_i$  is unlimited or unknown, we can replace  $X_i$  with the difference of two new variables,  $X_i = X_{n+2} - X_{n+1}$ , where  $X_{n+1} = (1 + X_i^2)^{1/2}$  and  $X_{n+2} = X_i + X_{n+1}$  are both positive. There are different ways for defining  $X_{n+1}$ ; the definition here is chosen because it avoids computational problems when  $X_i$  changes sign. Following this translation step, one can rewrite the set of differential equations and return to step A.

C. DECOMPOSITION OF COMPOSITE FUNCTIONS

If each  $f_{ijk}$  factor already is a power-law function, then proceed to step D. Each  $f_{ijk}$  factor in Equation (13) that is not already a power-law function is replaced with a new variable  $X_{n+1}$ , which simplifies the existing set of

differential equations to sums and products of power-law functions. An additional differential equation is generated for each new variable. Because of the chain rule, composite functions are decomposed into factors by differentiation when these equations are generated. The functional form of the factors in the new differential equation is always simpler than the functional form of the composite function in the original equation from which it was derived. Following this decomposition step, one has rewritten and augmented the set of differential equations and can return to step A.

As an example,

$$\dot{Z} = \exp[(\ln Z)^2] \quad Z(0) = 2 \quad (15)$$

can be recast in two cycles of decomposition as follows. Rename  $Z$  as  $X_1$ , define the single composite factor ( $f_{111}$ ) as a new variable  $X_2$ , and add the differential equation for  $X_2$  to the existing set:

$$\begin{aligned} \dot{X}_1 &= X_2 & X_1(0) &= 2 \\ \dot{X}_2 &= 2 X_1^{-1} X_2^2 \ln X_1 & X_2(0) &= \exp[(\ln 2)^2] \end{aligned} \quad (16)$$

Then define a new variable  $X_3$  as  $\ln X_1$  (the third factor of the last equation,  $f_{213}$ , which is the only factor that is not already a power-law function), and add the corresponding differential equation to the existing set:

$$\begin{aligned} \dot{X}_1 &= X_2 & X_1(0) &= 2 \\ \dot{X}_2 &= 2 X_1^{-1} X_2^2 X_3 & X_2(0) &= \exp[(\ln 2)^2] \\ \dot{X}_3 &= X_1^{-1} X_2 & X_3(0) &= \ln 2 \end{aligned} \quad (17)$$

The result is in the canonical S-system form. There are two constraints embedded in Equations (17); these easily can be made explicit as

$$\begin{aligned} X_2 &= \exp[(\ln X_1)^2] \\ X_3 &= \ln X_1 \end{aligned} \quad (18)$$

These constraints define a one-dimensional manifold within the three-dimensional space of the new variables and, since the initial conditions satisfy the constraints, the constraint set is invariant.

#### D. REDUCTION OF SUMS OF PRODUCTS OF POWER-LAW FUNCTIONS

If the set of equations is in S-system form, then recasting is complete. If not, then a reduction in the number of sums of products of power-law functions is necessary. Except for opportune cases in which the dimension-

ality remains unchanged, recasting sums of products of power-law functions generally requires the replacement of an existing variable by a product of new variables. Let us assume that the differential equation under consideration contains a sum of  $m$  products of S-system variables raised to individual powers. In simplified notation, we can write this equation as

$$\dot{X}_i = SP + \gamma_i \prod_{j=1}^n X_j^{k_{ij}} \tag{19}$$

where  $SP$  denotes the sum of the first  $m - 1$  products. We replace  $X_i$  by the product of two new variables,  $X_{n+1}X_{n+2}$ , and differentiate to obtain

$$X_{n+1}\dot{X}_{n+2} + X_{n+2}\dot{X}_{n+1} = SP + \gamma_i X_{n+1}^{k_{ii}} X_{n+2} \prod_{\substack{j=1 \\ j \neq i}}^n X_j^{k_{ij}} \tag{20}$$

Each of the two terms on the right can be identified with the derivative of a new variable to yield

$$\begin{aligned} \dot{X}_{n+1} &= SPX_{n+2}^{-1} \\ \dot{X}_{n+2} &= \gamma_i X_{n+1}^{k_{ii}-1} X_{n+2} \prod_{\substack{j=1 \\ j \neq i}}^n X_j^{k_{ij}} \end{aligned} \tag{21}$$

The initial condition for one of these two equations can be chosen conveniently, and then the other initial condition is given in terms of the initial condition for the original variable  $X_i$  [e.g.,  $X_{n+1}(0) = 1$  and  $X_{n+2}(0) = X_i(0)$ ]. This choice defines the appropriate section of the reference manifold of the new system. Upon replacing  $X_i$  by  $X_{n+1}$  and  $X_{n+2}$  and renumbering variables, we reduce the sum of terms in Equation (19) by one and increase the set of equations by one [see Equation (21)]. With the same method, two products of power-law functions with different sign can be identified with the derivative of a new variable; the equation for this new variable then will contain two terms in S-system form.

This step is repeated as many times as is necessary to reduce each equation to the difference of two products of power-law functions, the canonical S-system form.

As an example,

$$\dot{Z} = 2Z^{2/3} - Z^2 - Z^{1/2} \tag{22}$$

with  $Z(0) = 4$ , can be reduced to S-system form as follows. Define  $Z$  as  $X_1 X_2$ , differentiate to obtain

$$\dot{X}_1 X_2 + X_1 \dot{X}_2 = 2(X_1 X_2)^{2/3} - (X_1 X_2)^2 - (X_1 X_2)^{1/2} \tag{23}$$

and identify terms to yield

$$\begin{aligned}\dot{X}_1 &= -X_1^{1/2}X_2^{-1/2} \\ \dot{X}_2 &= 2x_1^{-1/3}X_2^{2/3} - X_1X_2^2\end{aligned}\quad (24)$$

where  $X_1(0) = 1$  and  $X_2(0) = 4$ . The particular choice of initial conditions defines the reference manifold.

The S-system also can be written in logarithmic form as

$$\begin{aligned}\dot{Y}_1 &= -\exp[-(Y_1 + Y_2)/2] \\ \dot{Y}_2 &= 2\exp[-(Y_1 + Y_2)/3] - \exp[(Y_1 + Y_2)]\end{aligned}\quad (25)$$

where  $Y_i = \log X_i$  and  $Y_1(0) = 0$ ,  $Y_2(0) = \log(4)$ . In this form it is clear that if  $Y_1$  is incremented and  $Y_2$  decremented by a constant amount  $Q$ , the system equations remain unchanged. The trajectories in the  $Y_1$  vs.  $Y_2$  plane parallel each other and can be superimposed by a translation of  $Q\sqrt{2}$  units along the  $-45^\circ$  incline.

There are three basins for this system:  $0 < Z < 0.0158123$ ,  $0.0158123 < Z < 1$ , and  $Z > 1$ . Accordingly, there are three trajectories in the recast logarithmic space that define the three sections of the reference manifold for the system. The three basins are separated by the sets  $Y_1 + Y_2 = 0$  (attracting) and  $Y_1 + Y_2 = -4.14697$  (repelling), which correspond to the fixed points  $Z = 1$  and  $Z = 0.0158123$  of the original system.

The reference trajectories in this case define a one-dimensional manifold in the two-dimensional  $Y_1$  vs.  $Y_2$  (or  $X_1$  vs.  $X_2$ ) space. Any other choice of initial condition for  $Z$  (or for  $X_1X_2$ ) can be translated onto the reference manifold, and the subsequent behavior of the system will remain on the reference manifold as well.

This is a constructive proof demonstrating that one can always recast the given class of equations as S-systems. Each new variable introduced in the recasting procedure requires an expansion of the S-system by an additional equation; for trigonometric and hyperbolic functions, two new variables and equations have to be introduced (cf. Example A3 in Appendix A). This increase in the number of variables might be seen as a disadvantage; therefore, it is important to note that the constraints provided by the definition of new variables (and their initial conditions) restrict the system behavior to a manifold of the same dimension as the original problem. The derivatives called for in the decomposition step are always derivatives of elementary functions. Thus, no discontinuities are generated in this process that are not already present in the original equations.

#### COROLLARY 1

*Higher-order systems of ordinary differential equations can be recast in a similar fashion. Higher derivatives simply are replaced with new variables,*

which follows the conventional practice for transforming an  $n$ th order ordinary differential equation into a set of  $n$  first-order ordinary differential equations.

COROLLARY 2

Under favorable conditions, other types of differential equations — such as implicit (cf. [35] and Appendix C), partial (e.g. [1]), delay (e.g. [9]) and integro-differential equations (e.g. [9])— can be rewritten in terms of ordinary differential equations and, hence, can be recast as S-systems as well.

COROLLARY 3

Nonlinear algebraic (in contrast to differential) equations also can be recast into the canonical S-system form. One simply defines the algebraic function as a new variable  $X_1$  and differentiates the equation to obtain an expression in the form of Equation (13). The recasting procedure is then the same as described above for differential equations.

This approach might seem to be in the wrong direction, for usually the aim is to solve a differential equation in terms of explicit functions. However, this type of recasting can yield very interesting results, for it classifies functions in a new way that thereby reveals structure in an otherwise seemingly unrelated multitude of functions and it sometimes makes the interpretation of a function and its parameters more transparent.

Two classes of algebraic functions that have been unified and classified are growth laws [21, 23] and probability distributions [24]. In their algebraic form, the various growth laws often do not show much similarity. However, recasting reveals that all the well-known cases are special S-systems with one or two variables. These results, which are summarized in Tables 1 and 2 of reference [21], demonstrate the close relatedness of these growth laws and, furthermore, connect these laws with their fundamental underlying mechanisms [21]. Comparing the algebraic form of a growth law with the corresponding S-system representation often yields insight into the nature of the growth process, as was discussed in [23]. For example, the explicit form of the Bertalanffy growth law

$$W(t) = \left[ \alpha/\beta - (\alpha/\beta - W_0^{1-g}) e^{(g-1)\beta t} \right]^{1/(1-g)} \tag{26}$$

is hardly intuitive, whereas the differential form

$$\dot{W} = \alpha W^g - \beta W \tag{27}$$

suggests a direct interpretation [2]:  $\beta$  is a first-order rate constant for loss due to catabolism, which is assumed to be proportional to body mass  $W$ ;  $\alpha$  and  $g$  are parameters of a power law describing anabolism, which is assumed to be proportional to the surface area through which nutrients are absorbed, and, hence, is a power-law function of mass or volume.

The classification of probability distribution functions as S-systems reveals their mutual relatedness (see Tables 1 and 2 in reference [24]). Embedding

them in a common continuous parameter space also provides a new, and more natural, way of defining mixed or intermediate distributions just by choice of parameter values. The continuous "suprasystem of probability distributions" [24] is of practical interest, for instance, when an unknown probability distribution has to be derived from experimental data [36].

## CONJECTURE

The theorem we have given provides a minimum estimate of the range of equations that can be recast in the canonical S-system form. We know that other functions not generally considered elementary functions (e.g., the so-called Special functions, such as Elliptic, Bessel, Legendre, and Hypergeometric functions) can be recast in the S-system form. A finite Taylor series can be recast as a finite set of equations in S-system form. Thus, in an approximate sense, any function that is well represented by a finite Taylor series also is well represented within the canonical S-system form. The full range of functions that can be recast is undoubtedly much larger than we have indicated and remains to be determined, perhaps by more powerful group-theoretic methods.

## 5. DETAILED EXAMPLE OF RECASTING

As an example illustrating several of these methods, consider the following set of nonlinear differential equations representing normalized material and energy balances in a continuous stirred tank reactor [34]

$$\begin{aligned} dZ_1/dt &= -Z_1 + A(1 - Z_1)\exp[cZ_2/(c + Z_2)] \\ dZ_2/dt &= -Z_2 + B(1 - Z_1)\exp[cZ_2/(c + Z_2)] - b(Z_2 - a) \end{aligned} \quad (28)$$

with initial conditions  $Z_1(0) = Z_{10}$  and  $Z_2(0) = Z_{20}$ .

This set of differential equations already is written in the form of Equation (13). The variables  $Z_1$  and  $Z_2$  have the potential to go to zero, so the first task is translation of these variables into the positive orthant. This can be done by adding to each variable a positive constant. For instance, we can define

$$\begin{aligned} X_1 &= Z_1 + 1 \\ X_2 &= Z_2 + 2 \end{aligned} \quad (29)$$

which transforms the original equations into

$$\begin{aligned} \dot{X}_1 &= 1 + 2A \exp[c(X_2 - 2)/(c - 2 + X_2)] \\ &\quad - X_1 - AX_1 \exp[c(X_2 - 2)/(c - 2 + X_2)] \end{aligned}$$

$$\begin{aligned}\dot{X}_2 = C + 2B \exp\left[ c(X_2 - 2)/(c - 2 + X_2) \right] \\ - (1 + b)X_2 - BX_1 \exp\left[ c(X_2 - 2)/(c - 2 + X_2) \right]\end{aligned}\quad (30)$$

where  $C = ab + 2(1 + b)$ .

Next is decomposition. Each factor that is not already a power-law function is replaced by a new variable and the corresponding differential equation is added to the set. Thus, we define

$$X_3 = \exp\left[ c(X_2 - 2)/(c - 2 + X_2) \right]\quad (31)$$

which is always positive, and the set of differential equations becomes

$$\begin{aligned}\dot{X}_1 &= 1 + 2AX_3 - X_1 - AX_1X_3 \\ \dot{X}_2 &= C + 2BX_3 - (1 + b)X_2 - BX_1X_3 \\ \dot{X}_3 &= CX_3\left[ (c - 2 + X_2)/c \right]^{-2} + 2BX_3^2\left[ (c - 2 + X_2)/c \right]^{-2} \\ &\quad - (1 + b)X_2X_3\left[ (c - 2 + X_2)/c \right]^{-2} - BX_1X_3^2\left[ (c - 2 + X_2)/c \right]^{-2}\end{aligned}\quad (32)$$

There is still one factor that is not a power-law function, so we repeat the decomposition step. We define

$$X_4 = (c - 2 + X_2)/c\quad (33)$$

which is always positive, and the new set of equations becomes

$$\begin{aligned}\dot{X}_1 &= 1 + 2AX_3 - X_1 - AX_1X_3 \\ \dot{X}_2 &= C + 2BX_3 - (1 + b)X_2 - BX_1X_3 \\ \dot{X}_3 &= CX_3X_4^{-2} + 2BX_3^2X_4^{-2} - (1 + b)X_2X_3X_4^{-2} - BX_1X_3^2X_4^{-2} \\ \dot{X}_4 &= C/c + (2B/c)X_3 - [(1 + b)/c]X_2 - (B/c)X_1X_3\end{aligned}\quad (34)$$

Each equation now consists of a sum of four terms. These sums are reduced by defining each variable as a product of two new variables:

$$\begin{aligned}X_1 &= X_5X_6 \\ X_2 &= X_7X_8 \\ X_3 &= X_9X_{10} \\ X_4 &= X_{11}X_{12}\end{aligned}\quad (35)$$

Differentiating each of these products, identifying the appropriate terms in Equations (34) for each new derivative, and permuting the indices of the

variables ( $5 \rightarrow 1$ ,  $6 \rightarrow 2, \dots$ ,  $12 \rightarrow 8$ ,  $1 \rightarrow 9, \dots$ ,  $4 \rightarrow 12$ ) yields the final S-system

$$\begin{aligned}
 \dot{X}_1 &= X_2^{-1} - X_1 \\
 \dot{X}_2 &= 2AX_1^{-1}X_5X_6 - AX_2X_5X_6 \\
 \dot{X}_3 &= CX_4^{-1} - (1+b)X_3 \\
 \dot{X}_4 &= 2BX_3^{-1}X_5X_6 - BX_1X_2X_3^{-1}X_5X_6 \\
 \dot{X}_5 &= CX_3X_7^{-2}X_8^{-2} - (1+b)X_3X_4X_5X_7^{-2}X_8^{-2} \\
 \dot{X}_6 &= 2BX_5X_6^2X_7^{-2}X_8^{-2} - BX_1X_2X_5X_6^2X_7^{-2}X_8^{-2} \\
 \dot{X}_7 &= (C/c)X_8^{-1} - [(1+b)/c]X_3X_4X_8^{-1} \\
 \dot{X}_8 &= (2B/c)X_5X_6X_7^{-1} - (B/c)X_1X_2X_5X_6X_7^{-1}
 \end{aligned} \tag{36}$$

with initial conditions  $X_1(0) = 1$ ,  $X_2(0) = Z_{10} + 1$ ,  $X_3(0) = 1$ ,  $X_4(0) = Z_{20} + 2$ ,  $X_5(0) = 1$ ,  $X_6(0) = \exp[cZ_{20}/(c + Z_{20})]$ ,  $X_7(0) = 1$ ,  $X_8(0) = (c + Z_{20})/c$ . Note that the differential equations for  $X_9$  through  $X_{12}$  and the variables themselves are automatically eliminated in this process.

The result is in the canonical S-system form with eight variables. There are six constraints embedded in Equations (36), which restrict the behavior to a two-dimensional manifold. Two constraints among the new variables in Equations (36) are found by rewriting Equations (31) and (33) in terms of the new variables

$$\begin{aligned}
 X_5X_6 &= \exp[c(X_3X_4 - 2)/(c + X_3X_4 - 2)] \\
 X_7X_8 &= (c + X_3X_4 - 2)/c
 \end{aligned} \tag{37}$$

Four additional constraints between new and old variables are given by

$$\begin{aligned}
 X_1X_2 &= X_9 \quad (= \text{old } X_1) \\
 X_3X_4 &= X_{10} \quad (= \text{old } X_2) \\
 X_5X_6 &= X_{11} \quad (= \text{old } X_3) \\
 X_7X_8 &= X_{12} \quad (= \text{old } X_4)
 \end{aligned} \tag{38}$$

These define four translation-group symmetries. Since these six constraints are satisfied by the initial conditions for Equations (36), the constraint set is invariant.

The original variables easily are reconstructed as

$$\begin{aligned}
 Z_1(t) &= X_1(t)X_2(t) - 1 \\
 Z_2(t) &= X_3(t)X_4(t) - 2
 \end{aligned} \tag{39}$$



The benefits of such recasting include simplification of the nonlinearities and reduction to a standard canonical form; the cost is the increase in number of variables. Our experience to date indicates that the benefits greatly outweigh the cost when it comes to computational efficiency. For instance, the above system, in which the number of variables increases fourfold, can be solved with reasonable tolerances 3 to 11 times faster in the recast form using our algorithm than in its original form using conventional Runge-Kutta methods. The global accuracy of these two solutions is the same.

Further examples of recasting can be found in earlier references and also in Appendix A, where a standard set of benchmark problems used in testing numerical methods for solving differential equations [4] has been recast as S-systems.

## 6. DISCUSSION

We have shown how ordinary initial-value problems can be transformed into the canonical S-system form. Currently there are no conditions that define the boundary of the class of equations and functions that can be recast as S-systems. Nevertheless, the large body of empirical evidence, and the fact that functions generally regarded as among the most general have been shown to be special cases of or equivalent to S-systems, strongly suggest that most, if not all, ordinary differential equations can be represented exactly as S-systems. Other types of differential equations, such as implicit, partial, delay and integro-differential equations under favorable conditions can be rewritten in terms of ordinary differential equations and, hence, can be recast as S-systems as well.

This is not to say that a fully developed nonlinear theory is imminent. However, the identification of a canonical form for nonlinearities seems to be an important step in initiating the development of such a theory, which would provide the basis for a general classification of differential equations and for the development of general techniques for treating nonlinear differential equations with analytical or numerical methods. As yet, such general techniques are known only for linear systems, one reason being that no unifying structural form was known that included arbitrary nonlinear equations as special cases. In analogy with linear analysis one can now focus upon a canonical nonlinear structure and hope to see the rapid development of appropriate methods and techniques, and perhaps eventually a theory rivaling the power of linear mathematics, but applicable to the much wider and more important domain of nonlinear phenomena.

A variety of methods have been developed for analyzing S-systems that are used for local descriptions of biological phenomena (e.g., [20, 21, 39, 7]). These methods provide a framework for the analysis of recast equations as

well. Important special cases of S-systems have been solved analytically [40], but since functions like elliptic integrals and Bessel functions can be represented as S-systems, the general S-system is known to have no finite closed-form solution in terms of elementary functions. Thus, one important goal in dealing with S-systems is efficient numerical solution.

A set of numerical algorithms called ESSYNS (Evaluation and Simulation of Synergistic Systems) has been developed in our laboratory and is based on the S-system form [8]. It is generally superior to existing methods for solving this class of problems, often one to two orders of magnitude faster [7, 8; see also examples A1, A2, A4, B1, C1, C2 and E4 in Appendix A and Table 1]. This vast improvement was possible because each part of the program could be optimized exclusively for the very rigid S-system form and did not have to account for the multitude of features that could arise in arbitrary nonlinear differential equations. The recasting procedure presented here makes the advantages of these efficient algorithms available to any differential equation that can be recast as an S-system.

Even though recasting usually requires the introduction of additional variables, there is generally a net gain in efficiency and often it is immense, as can be seen with standard benchmark problems (see Table 1 and Appendix A). The algorithm of Gear, as implemented in the IMSL software library,\* and a Runge-Kutta-Fehlberg (RKF) method were chosen for these comparisons because of their wide availability and common use as references in this type of comparison (e.g., [3, 4, 31]). All comparisons were executed with 8 Hz IBM PC/AT compatible microcomputers equipped with numerical coprocessors. The results (Table 1) show that the cost of recasting (identified with decreases in efficiency due to the introduction of additional variables) is generally recouped at reasonable error tolerances and the net improvements are often remarkable: at stringent error tolerances, several problems were solved 50 or even 100 times faster with ESSYNS than with RKF45 and up to 50 times faster than with the Gear algorithm. For lax error tolerances RKF45 or the Gear algorithm performed better than ESSYNS in some cases, although the difference was usually less than twofold. The global accuracy of ESSYNS, the Gear algorithm, and RKF45 was similar for these problems.

Recently we have learned that Peschel and Mende have come independently to similar conclusions regarding the transformation of differential equations into standard forms [10–13]. In their work they have focused primarily on Generalized Volterra systems but also have considered so-called

---

\*As standard parameter settings we used  $H = 0.00001$ ,  $INDEX = 1$ ,  $METH = 2$ , and  $MITER = 0$ . We also tested some examples with  $MITER = 3$  and found very similar results.

TABLE 1  
CPU-Time in Seconds for Solving Benchmark Problems in Original Form and in Recast S-system Form\*

Problem	Tolerance ( $-\log T$ )											
	3			6			9			12		
	Original RKF 45	Original GEAR	S-System ESSYNS	Original RKF 45	Original GEAR	S-System ESSYNS	Original RKF 45	Original GEAR	S-System ESSYNS	Original RKF 45	Original GEAR	S-System ESSYNS
A1	2.2	1.5	0.5	2.2	3.0	0.5	2.2	4.9	0.5	2.2	10.4	0.5
2	2.8	1.2	0.9	3.4	2.4	1.3	5.4	5.2	1.9	17.6	9.8	2.8
3	2.1	2.4	3.9	6.6	6.8	8.5	30.7	16.3	15.6	163.0	33.6	25.0
4	2.5	1.2	0.7	2.5	1.8	0.9	7.4	3.6	1.3	35.0	7.4	1.7
5	3.4	1.1	2.0	3.4	2.1	3.6	7.8	4.1	5.6	37.0	8.6	8.5
B1	6.5	4.4	3.0	18.7	14.8	6.1	92.6	41.8	10.9	484.4	84.9	16.5
2	4.8	2.6	2.7	7.1	7.3	3.8	25.4	18.7	6.0	126.1	35.2	9.0
3	34.2	2.9	2.7	38.1	5.7	5.6	149.2	13.2	10.0	678.7	30.3	15.5
4	7.4	4.8	15.1	27.0	13.9	34.7	136.3	39.3	62.0	734.7	2652.3	102.5
5	5.6	3.6	6.6	17.4	13.0	14.8	77.5	32.0	27.3	412.7	74.9	42.5
C1	39.5	8.0	5.4	69.0	17.0	10.4	601.1	41.6	16.8	3696.6	117.3	24.9
2	133.6	9.3	11.1	155.2	29.9	14.9	732.5	74.1	20.9	3813.6	126.4	31.0
3	55.9	17.6	14.3	89.3	28.5	19.8	462.9	114.2	31.0	2502.5	176.5	43.8
D1	12.9	10.4	17.0	40.0	28.3	32.5	183.3	83.6	53.1	989.4	4437.7	81.0
2	12.4	12.2	19.4	40.8	37.8	38.6	200.8	113.8	63.1	1092.0	4421.1	95.9
3	12.1	15.8	25.3	44.9	51.8	53.8	216.5	138.6	90.9	1193.5	4315.2	139.0
4	14.3	18.6	30.6	49.7	66.1	68.2	244.2	174.6	117.5	1313.7	4714.2	180.1
5	18.8	26.5	38.4	60.2	86.1	94.0	289.8	238.8	167.1	1581.5	4716.8	260.2
E1	4.2	2.9	6.1	15.5	7.9	13.2	77.8	18.2	24.0	424.8	36.6	37.5
2	6.5	5.2	15.9	24.1	20.1	33.8	119.0	47.2	58.8	649.9	117.5	89.5
3	6.4	5.3	19.0	22.7	14.9	47.0	110.5	39.8	89.1	593.0	75.1	144.0
4	22.5	1.5	1.5	23.9	2.2	1.5	90.8	4.4	2.3	373.3	7.4	3.1
5	3.1	1.9	2.2	3.1	2.8	3.7	6.0	5.4	5.6	31.0	10.4	8.2

\*The benchmark problems are described in Appendix A. See text for further discussion.

Riccati systems and Multinomial systems. Their Riccati systems are identical to our Half systems—i.e., to S-systems with only one product of power-law functions in each equation—and their Multinomial systems are identical to our Generalized Mass Action systems—i.e., they contain the sum of several products of power-law functions in each equation. They also have demonstrated that these various forms are equivalent in the sense that they can be transformed into one another and then exhibit exactly the same solution [11, 13].

Our approaches are fundamentally similar, although because of their different histories they differ in a number of details. For example, Peschel and Mende's transformation of differential equations into Generalized Volterra systems tends to create discontinuities that aggravate further analysis [13 Ch 3]. Although they find that some of these difficulties are removed through transformation to the Riccati form, the resulting systems have the disadvantage of computational instabilities [13], as we also have noted. A method of rescaling to overcome this computational problem has been proposed [13]. Since these difficulties are circumvented in our procedure using S-systems, and since it has been established that these are indeed equivalent canonical forms, it should be possible for these difficulties to be removed in the approach of Peschel and Mende as well.

The different standard forms can be interchanged by transformation and are equivalent with respect to solution, though they can differ with respect to interpretability (cf. [42]), mathematical tractability, and computational efficiency. Each of these standard forms has its appeal and may be most appropriate for particular classes of problems; further work will be required to explore and to weigh the advantages and disadvantages of each form.

#### APPENDIX A: RECASTING STANDARD BENCHMARK PROBLEMS FOR NUMERICAL SOLUTION

Hull et al. [4] have provided a set of benchmark problems that have become a standard for evaluating numerical methods for the solution of mildly stiff ordinary nonlinear differential equations [e.g. 3]. In order to evaluate ESSYNS [8], a set of algorithms developed in this laboratory based on the S-system form, we have recast this set of benchmark problems as S-systems. In addition to providing numerous examples of recasting and further evidence of its generality, test results shown in Table 1 demonstrate the practical advantages of solution in the S-system form using ESSYNS. We have analyzed all of the benchmark tests of Hull et al. [4] with the exception of two problems (C4 and C5) whose numbers of equations exceeded the current capacity of our microcomputer version of ESSYNS. In the following, we shall present the original set of equations, the definitions of S-system variables, and the S-system form we used for comparison.

*General Specification of Problems.* Each problem is characterized by the set of differential equations and initial conditions, by the initial and final times, which were always chosen to be 0 and 20, and by each of the four local error tolerances  $10^{-3}$ ,  $10^{-6}$ ,  $10^{-9}$ , or  $10^{-12}$ . Using the same definitions as Hull et al. [4], we have compared efficiency and accuracy for 92 problems in total.

*PROBLEM CLASS A. SINGLE EQUATIONS.*

$$A1: \quad \dot{Z} = -Z \quad Z(0) = 1$$

This negative exponential decay is already in S-system form.

$$A2: \quad \dot{Z} = -Z^3/2 \quad Z(0) = 1$$

This special case of the Riccati equation also is already in S-system form.

$$A3: \quad \dot{Z} = Z \cos(t) \quad Z(0) = 1$$

We define  $X_1 = Z$ ,  $X_2 = \cos(t) + 2$ ,  $X_3 = \sin(t) + 2$ ,  $X(0) = (1, 3, 2)$  and obtain the S-system

$$\begin{aligned} \dot{X}_1 &= X_1 X_2 - 2 X_1 \\ \dot{X}_2 &= 2 - X_3 \\ \dot{X}_3 &= X_2 - 2 \end{aligned}$$

$$A4: \quad \dot{Z} = 0.25Z(1 - Z/20) \quad Z(0) = 1$$

This is the logistic growth law and has S-system form.

$$A5: \quad \dot{Z} = (Z - t)/(Z + t) \quad Z(0) = 4$$

A5 describes a spiral curve. The S-system form can be obtained upon defining  $X_1 = Z + p$ ,  $X_2 = t + p$ ,  $X_3 = X_1 + X_2 - 2p$ , where  $p$  is an arbitrary parameter greater than 4; we have chosen  $p = 5$ ,  $X(0) = (9, 1, 4)$ .

$$\begin{aligned} \dot{X}_1 &= X_1 X_3^{-1} - X_2 X_3^{-1} \\ \dot{X}_2 &= 1 \\ \dot{X}_3 &= 2 X_1 X_3^{-1} - 2 p X_3^{-1} \end{aligned}$$

*PROBLEM CLASS B. SMALL SYSTEMS.*

$$\begin{aligned} B1: \quad \dot{Z}_1 &= 2(Z_1 - Z_1 Z_2) \\ \dot{Z}_2 &= -(Z_2 - Z_1 Z_2) \end{aligned}$$

$Z(0) = (1, 3)$ . These Lotka-Volterra equations, describing the population dynamics of conflicting species, are already in S-system form.

$$\begin{aligned} \text{B2:} \quad \dot{Z}_1 &= -Z_1 + Z_2 \\ \dot{Z}_2 &= Z_1 - 2Z_2 + Z_3 \\ \dot{Z}_3 &= Z_2 - Z_3 \end{aligned}$$

$Z(0) = (2, 0, 1)$ . We obtain the S-system form for this linear system of chemical reactions by defining  $X_1 = Z_1 + p$ ,  $X_2 = Z_2 + p$ ,  $X_3 = Z_3 + p$ ,  $X_4 = X_1 + X_3$ ; where  $p$  arbitrarily has been chosen to be 2.

$$\begin{aligned} \dot{X}_1 &= X_2 - X_1 \\ \dot{X}_2 &= X_4 - 2X_2 \\ \dot{X}_3 &= X_2 - X_3 \\ \dot{X}_4 &= 2X_2 - X_4 \end{aligned}$$

$X(0) = (4, 2, 3, 7)$ .

$$\begin{aligned} \text{B3:} \quad \dot{Z}_1 &= -Z_1 \\ \dot{Z}_2 &= Z_1 - Z_2^2 \\ \dot{Z}_3 &= Z_2^2 \end{aligned}$$

$Z(0) = (1, 0, 0)$ . This represents a nonlinear system of chemical reactions. Since two of the initial values are zero, we define  $X_1 = Z_2 + p$ ,  $X_2 = Z_3 + q$ ,  $X_3 = X_1 + p$ , and  $X_4 = Z_1 + 4pX_1$ ,  $X(0) = (p, q, 2p, 1 + 4p^2)$ . We have chosen  $p = 0.25$ ,  $q = 1$ .

$$\begin{aligned} \dot{X}_1 &= X_4 - X_3^2 \\ \dot{X}_2 &= X_3^2 - X_1 \\ \dot{X}_3 &= X_4 - X_3^2 \\ \dot{X}_4 &= X_1 - X_3^2 \end{aligned}$$

$$\begin{aligned} \text{B4:} \quad \dot{Z}_1 &= -Z_2 - Z_1 Z_3 (Z_1^2 + Z_2^2)^{-1/2} \\ \dot{Z}_2 &= Z_1 - Z_2 Z_3 (Z_1^2 + Z_2^2)^{-1/2} \\ \dot{Z}_3 &= Z_1 (Z_1^2 + Z_2^2)^{-1/2} \end{aligned}$$

$Z(0) = (3, 0, 0)$ . B4 represents the integral surface of a torus. We define  $X_1 = Z_1 + p$ ,  $X_2 = Z_2 + q$ ,  $X_3 = Z_3 + r$ ,  $X_4 = (Z_1^2 + Z_2^2)^{-1/2}$ , and subse-

quently replace  $X_1$  by the product  $X_5 X_6$  and  $X_2$  by  $X_7 X_8 X_9$  which, upon renumbering, yields the S-system

$$\begin{aligned}\dot{X}_1 &= X_2 X_3 X_4 X_5 - p X_2 \\ \dot{X}_2 &= X_1 X_2^2 - r X_2^2 \\ \dot{X}_3 &= p X_1 X_2 X_4^{-1} X_5^{-1} - X_1 X_2 X_3 \\ \dot{X}_4 &= r X_2 X_4 - pr X_2 X_3^{-1} X_5^{-1} \\ \dot{X}_5 &= q X_3^{-1} X_4^{-1} - X_3^{-1} X_4^{-1} X_6 X_7 X_8 \\ \dot{X}_6 &= q X_1 X_2 X_7^{-1} X_8^{-1} - X_1 X_2 X_6 \\ \dot{X}_7 &= r X_2 X_7 - qr X_2 X_6^{-1} X_8^{-1} \\ \dot{X}_8 &= X_3 X_4 X_5 X_6^{-1} X_7^{-1} - p X_6^{-1} X_7^{-1}\end{aligned}$$

$$X(0) = (r, 1/3, p + 3, 1, 1, q, 1, 1).$$

$$\begin{aligned}\text{B5:} \quad \dot{Z}_1 &= Z_2 Z_3 \\ \dot{Z}_2 &= -Z_1 Z_3 \\ \dot{Z}_3 &= -0.51 Z_1 Z_2\end{aligned}$$

$Z(0) = (0, 1, 1)$ . These are Euler's equations of motion for a rigid body without external forces. Since  $Z_1$  and  $Z_2$  become negative in the time interval  $[0, 20]$ , we define S-system variables as  $X_1 = Z_1 + p$ ,  $X_2 = Z_2 + q$ , and  $X_3 = Z_3$  and obtain the S-system

$$\begin{aligned}\dot{X}_1 &= X_2 X_3 X_4 - q X_3 X_4 \\ \dot{X}_2 &= p X_3 X_4 - X_1 X_3 X_4 \\ \dot{X}_3 &= 0.51 p X_2 X_4^{-1} - 0.51 p q X_4^{-1} \\ \dot{X}_4 &= 0.51 q X_1 X_3^{-1} - 0.51 X_1 X_2 X_3^{-1}\end{aligned}$$

$$X(0) = (p, 1 + q, 1, 1). \text{ For our comparison we chose } p = q = 2.$$

#### PROBLEM CLASS C. MODERATE SYSTEMS

$$\begin{aligned}\text{C1:} \quad \dot{Z}_1 &= -Z_1 \\ \dot{Z}_i &= Z_{i-1} - Z_i \\ \dot{Z}_{10} &= Z_9\end{aligned}$$

( $i = 2, \dots, 9$ ),  $Z(0) = (1, 0, \dots, 0)$ . This represents a radioactive decay chain.

Translation of all variables to the positive orthant by defining  $X_i = Z_i + p$  yields the corresponding S-system

$$\begin{aligned}\dot{X}_1 &= p - X_1 \\ \dot{X}_i &= X_{i-1} - X_i \\ \dot{X}_{10} &= X_9 - p\end{aligned}$$

$(i = 2, \dots, 9)$ ,  $X(0) = (1 + p, p, \dots, p)$ .

C2:

$$\begin{aligned}\dot{Z}_1 &= -Z_1 \\ \dot{Z}_i &= Z_{i-1} - 2Z_i \\ \dot{Z}_{10} &= Z_9\end{aligned}$$

$(i = 2, \dots, 9)$ ,  $Z(0) = (1, 0, \dots, 0)$ . This again represents a radioactive decay chain. Here convenient translation parameters are  $p_i = p_1/i$  ( $i = 2, \dots, 10$ ), where  $p_1$  is positive. The S-system then is

$$\begin{aligned}\dot{X}_1 &= p_1 - X_1 \\ \dot{X}_i &= (i-1)X_{i-1} - iX_i \\ \dot{X}_{10} &= 9X_9 - p_1\end{aligned}$$

$X(0) = (1 + p_1, p_i/i, p_1)$ , ( $i = 2, \dots, 9$ ).

C3:

$$\begin{aligned}\dot{Z}_1 &= -2Z_1 + Z_2 \\ \dot{Z}_i &= Z_{i-1} - 2Z_i + Z_{i+1} \\ \dot{Z}_{10} &= Z_9 - 2Z_{10}\end{aligned}$$

$(i = 2, \dots, 9)$ ,  $Z(0) = (1, 0, \dots, 0)$ . C3 is derived from a parabolic partial differential equation. Its recasting requires two steps. First, we translate the variables  $X_i = Z_i + ip_1$ , where  $p_1$  is positive, and secondly we replace  $X_i$  ( $i = 2, \dots, 9$ ) by a product of two new variables and obtain the S-system

$$\begin{aligned}\dot{X}_1 &= X_2 X_3 - 2X_1 \\ \dot{X}_2 &= X_1 X_3^{-1} - X_2 \\ \dot{X}_i &= X_{i-1}^{-1} X_{i+1} X_{i+2} - X_i \\ \dot{X}_j &= X_{j-2} X_{j-1} X_{j+1}^{-1} - X_j \\ \dot{X}_{19} &= 11p_1 X_{18}^{-1} - X_{19}\end{aligned}$$

$(i = 3, 5, 7, \dots, 17; j = 4, 6, 8, \dots, 18)$ ,  $x(0) = (2, 2, 1, 3, 1, 4, 1, 5, 1, 6, 1, 7, 1, 8, 1, 9, 1, 10, 1)$ .



## PROBLEM CLASS D. ORBIT EQUATIONS.

$$\begin{aligned}\dot{Z}_1 &= Z_3 \\ \dot{Z}_2 &= Z_4 \\ \dot{Z}_3 &= -Z_1(Z_1^2 + Z_2^2)^{-3/2} \\ \dot{Z}_4 &= -Z_2(Z_1^2 + Z_2^2)^{-3/2}\end{aligned}$$

$Z(0) = (1 - e, 0, 0, ((1 + e)/(1 - e))^{1/2})$ , where  $e$  is the eccentricity of the orbit.

D1:	$e = 0.1$
D2:	$e = 0.3$
D3:	$e = 0.5$
D4:	$e = 0.7$
D5:	$e = 0.9$

We define  $X_1 = Z_1 + 3$ ,  $X_2 = Z_2 + 3$ ,  $X_3 = Z_3 + 5$ ,  $X_4 = Z_4 + 5$ ,  $X_5 = Z_1^2 + Z_2^2$  and replace  $X_5$  by the product  $X_6 X_7 X_8 X_9$ . Upon renumbering, the result is an eight-dimensional S-system:

$$\begin{aligned}\dot{X}_1 &= X_3 - 5 \\ \dot{X}_2 &= X_4 - 5 \\ \dot{X}_3 &= 3(X_5 X_6 X_7 X_8)^{-3/2} - X_1(X_5 X_6 X_7 X_8)^{-3/2} \\ \dot{X}_4 &= 3(X_5 X_6 X_7 X_8)^{-3/2} - X_2(X_5 X_6 X_7 X_8)^{-3/2} \\ \dot{X}_5 &= 2X_1 X_3(X_6 X_7 X_8)^{-1} - 6X_3(X_6 X_7 X_8)^{-1} \\ \dot{X}_6 &= 2X_2 X_4(X_5 X_7 X_8)^{-1} - 6X_4(X_5 X_7 X_8)^{-1} \\ \dot{X}_7 &= 30(X_5 X_6 X_8)^{-1} - 10X_2(X_5 X_6 X_8)^{-1} \\ \dot{X}_8 &= 30(X_5 X_6 X_7)^{-1} - 10X_1(X_5 X_6 X_7)^{-1}\end{aligned}$$

$X(0) = (4 - e, 3, 5, 5 + ((1 + e)/(1 - e))^{1/2}, (1 - e)^2, 1, 1, 1)$ .

## PROBLEM CLASS E. HIGHER ORDER EQUATIONS.

E1:	$\dot{Z}_1 = Z_2$
	$\dot{Z}_2 = 0.25(t + 1)^{-2} Z_1 - Z_2(t + 1)^{-1} - Z_1$

$Z(0) = (0.671396707141803, 0.09540051444747446)$ . This problem is derived from Bessel's equation of order  $1/2$  with the origin shifted one unit to the

left. The S-system form can be obtained by first defining  $X_1 = Z_1 + p$  and  $X_2 = Z_2 + q$ , where  $p$  and  $q$  are positive, and then replacing  $X_2$  by the product of four new variables. The result is

$$\begin{aligned}\dot{X}_1 &= X_2 X_3 X_4 - q \\ \dot{X}_2 &= q X_3^{-1} X_4^{-1} X_5^{-1} - X_2 X_5^{-1} \\ \dot{X}_3 &= p X_2^{-1} X_4^{-1} - X_1 X_2^{-1} X_4^{-1} \\ \dot{X}_4 &= 0.25 X_1 X_2^{-1} X_3^{-1} X_5^{-2} - 0.25 p X_2^{-1} X_3^{-1} X_5^{-2} \\ \dot{X}_5 &= 1\end{aligned}$$

$$X(0) = (Z_1(0) + p, Z_2(0) + q, 1, 1, 1).$$

$$\begin{aligned}\text{E2:} \quad \dot{Z}_1 &= Z_2 \\ \dot{Z}_2 &= (1 - Z_1^2) Z_2 - Z_1\end{aligned}$$

$Z(0) = (2, 0)$ . The problem is derived from van der Pol's equation. We translate the variables with the positive parameters  $p$  and  $q$  and replace the second variable by the product of four new variables. These transformations yield the S-system

$$\begin{aligned}\dot{X}_1 &= X_2 X_3 X_4 X_5 - q \\ \dot{X}_2 &= (p^2 q + p - q) X_3^{-1} X_4^{-1} X_5^{-1} - p^2 X_2 \\ \dot{X}_3 &= 2p X_1 X_3 - 2pq X_1 X_2^{-1} X_4^{-1} X_5^{-1} \\ \dot{X}_4 &= q X_1^2 X_2^{-1} X_3^{-1} X_5^{-1} - X_1^2 X_4 \\ \dot{X}_5 &= X_5 - X_1 X_2^{-1} X_3^{-1} X_4^{-1}\end{aligned}$$

$$X(0) = (2 + p, q, 1, 1, 1).$$

$$\begin{aligned}\text{E3:} \quad \dot{Z}_1 &= Z_2 \\ \dot{Z}_2 &= (1/6) Z_1^3 - Z_1 + 2 \sin(2.7835t)\end{aligned}$$

$Z(0) = (0, 0)$ . The problem is derived from Duffing's equation. We define  $X_1 = Z_1 + p$ ,  $X_2 = Z_2 + q$ ,  $X_3 = \sin(2.78525t)$ . The equation for  $X_2$  contains a sum of terms, which can be reduced by replacing  $X_2$  with the product of three variables. The definition of  $X_3$  as a sine function requires the additional introduction of the corresponding cosine. The final S-system con-

tains six equations:

$$\begin{aligned}\dot{X}_1 &= X_4 X_5 X_6 - q \\ \dot{X}_2 &= 5.5707r - 5.5707X_3 \\ \dot{X}_3 &= 1.392675 X_2 - 1.392675p \\ \dot{X}_4 &= X_2 X_5^{-1} X_6^{-1} - X_1 X_5^{-1} X_6^{-1} \\ \dot{X}_5 &= (p^2/2) X_1 X_4^{-1} X_6^{-1} - (p^3/6) X_4^{-1} X_6^{-1} \\ \dot{X}_6 &= (1/6) X_1^3 X_4^{-1} X_5^{-1} - (p/2) X_1^2 X_4^{-1} X_5^{-1}\end{aligned}$$

$X(0) = (p, p, r+1, q, 1, 1)$ . For our comparison we have chosen  $p = q = r = 4$ .

E4:

$$\begin{aligned}\dot{Z}_1 &= Z_2 \\ \dot{Z}_2 &= 0.032 - 0.4Z_2^2\end{aligned}$$

$Z(0) = (30, 0)$ . E4 describes the motion of a falling body. We redefine  $X_2 = Z_2 + p$ . For the particular choice  $p = 0.08^{1/2}$ , the S-system is

$$\begin{aligned}\dot{X}_1 &= X_2 - p \\ \dot{X}_2 &= 0.8pX_2 - 0.4X_2^2\end{aligned}$$

$X(0) = (30, p)$ .

E5:

$$\begin{aligned}\dot{Z}_1 &= Z_2 \\ \dot{Z}_2 &= (1 + Z_2^2)^{1/2} (25 - t)^{-1}\end{aligned}$$

$Z(0) = (0, 0)$ . Definition of  $X_1 = Z_1 + p$ ,  $X_2 = Z_2 + p$ ,  $X_3 = 1 + Z_2^2$ , and  $X_4 = 25 - t$ , yields the S-system form in a straightforward manner:

$$\begin{aligned}\dot{X}_1 &= X_2 - p \\ \dot{X}_2 &= X_3^{1/2} X_4^{-1} \\ \dot{X}_3 &= 2 X_2 X_3^{1/2} X_4^{-1} - 2pX_3^{1/2} X_4^{-1} \\ \dot{X}_4 &= -1\end{aligned}$$

$X(0) = (p, p, 1, 25)$ .

## APPENDIX B: TRANSFORMATION OF A HALF SYSTEM INTO A BINARY SYSTEM

At the cost of additional equations, every Half system can be transformed into the even simpler structure of a Binary system that has the form of a Half system but whose exponents are all unity or zero. Since S-systems, Generalized Mass Action systems, and Generalized Volterra systems can be recast as Half systems, they also can be transformed into Binary systems.

For simplicity of discussion, let us use for the representation of a Half system the notation

$$\dot{Z}_i = \gamma_i Z_i \prod_{j=1}^n Z_j^{f_{ij}} \quad i=1, \dots, n \quad (\text{B1})$$

in which  $\gamma_i$  and  $f_{ij}$  are real numbers [cf. Equation (7)]. System (B1) is transformed by replacing in equation  $i$  each variable  $Z_j$ , raised to  $f_{ij}$ , by a new variable  $V_{ij}$  if the exponent  $f_{ij}$  is different from zero. These new variables satisfy the equations

$$\begin{aligned} \dot{V}_{ij} &= f_{ij} Z_j^{(f_{ij}-1)} \dot{Z}_j \\ &= \gamma_j f_{ij} V_{ij} \prod_{k \in K} V_{jk} \quad K = \{k \in \{1, \dots, n\} : f_{jk} \neq 0\} \end{aligned} \quad (\text{B2})$$

The new Equations (B2) do not contain explicitly any old variables  $Z_j$  and, thus, all  $Z_j$  can be eliminated. A trivial exception appears with any variable  $Z_j$  whose exponents  $f_{ij}$  ( $i=1, \dots, n$ ) are all equal to zero; but simply renaming  $Z_j = V_{jj}$  and setting  $f_{jj} = 1$  yield the unified notation of (B2) and  $Z_j$  can be eliminated. The original variables can always be retrieved from the definitions  $Z_j^{f_{ij}} = V_{ij}$ . In cases where  $f_{ii} \neq 0$ , the variable  $V_{ii}$  appears twice in Equations (B2) with  $i = j$ . These square powers are eliminated readily by introducing additional variables  $U_i = V_{ii}^2$ . Finally, all variables  $V_{ij}$  and  $U_j$  are renamed as variables  $X_k$  and we obtain a Binary system that is equivalent to Equations (B1)

$$\dot{X}_i = \eta_i \prod_{j=1}^m X_j^{e_{ij}} \quad (e_{ij} \in \{0, 1\}) \quad (\text{B3})$$

The dimension,  $m$ , of the system is usually greater than  $n$ , and in the extreme, the number of equations increases from  $n$  to  $m = n^2 + n$ .

One should note that Equations (B2), upon dividing them by  $V_{ij}$ , represent changes in logarithms of  $V_{ij}$  as simple products of the system variables  $V_{ij}$ . This fact is interesting in comparison with Generalized Volterra systems, which represent changes in logarithms as sums of system variables [cf.

Equation (8)] but which, nevertheless, are equivalent to Binary systems with respect to solution, as we have shown above (see also [42]).

*Example.* A simple Half system may elucidate the methods of transformation into the corresponding Binary system.

$$\begin{aligned}
 \dot{Z}_1 &= 3Z_1^{2/3}Z_2^{1/2} \\
 \dot{Z}_2 &= -Z_1^4Z_2 \\
 \dot{Z}_3 &= 0.5Z_1Z_3
 \end{aligned} \tag{B4}$$

with initial conditions  $Z_1(0) = Z_{10}$ ,  $Z_2(0) = Z_{20}$ ,  $Z_3(0) = Z_{30}$ . Defining  $V_{11} = Z_1^{-1/3}$ ,  $V_{12} = Z_2^{1/2}$ ,  $V_{21} = Z_1^4$ ,  $V_{31} = Z_1$  yields

$$\begin{aligned}
 \dot{Z}_1 &= 3Z_1V_{11}V_{12} \\
 \dot{Z}_2 &= -Z_2V_{21} \\
 \dot{Z}_3 &= 0.5Z_3V_{31} \\
 \dot{V}_{11} &= -V_{11}V_{11}V_{12} \\
 \dot{V}_{12} &= -0.5V_{12}V_{21} \\
 \dot{V}_{21} &= 12V_{11}V_{12}V_{21} \\
 \dot{V}_{31} &= 3V_{11}V_{12}V_{31}
 \end{aligned} \tag{B5}$$

The original variables  $Z_1$  and  $Z_2$  are retrievable from the definitions (e.g., from  $V_{11}$  and  $V_{12}$ );  $Z_3$  is not, and so we formally replace it with  $V_{33}$  and then eliminate  $Z_1, Z_2, Z_3$ . Further, we introduce  $U_1 = V_{11}^2$  to eliminate the square power of  $V_{11}$  in the fourth equation. Finally, we rename  $X_1 = V_{11}$ ,  $X_2 = V_{12}$ ,  $X_3 = V_{21}$ ,  $X_4 = V_{31}$ ,  $X_5 = V_{33}$  and  $X_6 = U_1$  and obtain the binary form of system (B4)

$$\begin{aligned}
 \dot{X}_1 &= -X_2X_6 \\
 \dot{X}_2 &= -0.5X_2X_3 \\
 \dot{X}_3 &= 12X_1X_2X_3 \\
 \dot{X}_4 &= 3X_1X_2X_4 \\
 \dot{X}_5 &= 0.5X_4X_5 \\
 \dot{X}_6 &= -2X_1X_2X_6
 \end{aligned} \tag{B6}$$

with initial conditions  $X_1(0) = Z_{10}^{-1/3}$ ,  $X_2(0) = Z_{20}^{1/2}$ ,  $X_3(0) = Z_{10}^4$ ,  $X_4(0) = Z_{10}$ ,  $X_5(0) = Z_{30}$ ,  $X_6(0) = Z_{10}^{-2/3}$ .

### APPENDIX C: TRANSFORMATION OF IMPLICIT DIFFERENTIAL EQUATIONS

Texts on differential equations seldom discuss the problem of implicit differential equations and it is difficult to find helpful references. Thus we will present a recipe for the transformation of implicit into explicit differential equations that is suitable in many cases. The idea of this method is sketched briefly in a paper by Verner [35], who presents a more general albeit more complicated method.

We can assume that the system contains only first derivatives; if that is not the case, we introduce new variables for all higher-order derivatives (cf. Corollary 1). The transformation requires three steps. If the system cannot be rewritten directly in explicit form, a new variable  $X_j$  is defined for each derivative  $\dot{Z}_k$  that cannot be expressed explicitly in terms of existing variables  $Z_i$ . This yields a set of implicit algebraic equations. With standard methods, one has to solve this set numerically for initial conditions of the new variables  $X_j$ . Next, one differentiates all implicit (algebraic) equations symbolically with respect to the independent variable. This yields a system of implicit differential equations that is linear in terms of derivatives of the new variables  $X_j$ . The "coefficients" are functions of old and newly-defined variables  $Z_i$  and  $X_j$ . This linear system has to be solved symbolically for  $X_j$ ; if a solution exists, it will be a system of explicit ordinary first-order differential equations.

*Example*

$$\begin{aligned}(\ln \dot{Z}_1)^2 + \dot{Z}_2 \exp(\dot{Z}_1) - t Z_1 &= 0 \\ \dot{Z}_1^2 + \dot{Z}_1 \dot{Z}_2 + \dot{Z}_2^2 - Z_2 - t &= 0\end{aligned}\tag{C1}$$

We define  $X_1 = \dot{Z}_1$  and  $X_2 = \dot{Z}_2$ ; the resulting set of implicit algebraic equations is

$$\begin{aligned}(\ln X_1)^2 + X_2 \exp(X_1) - t Z_1 &= 0 \\ X_1^2 + X_1 X_2 + X_2^2 - Z_2 - t &= 0\end{aligned}\tag{C2}$$

Upon differentiating this set with respect to  $t$  we obtain

$$\begin{aligned}2(\ln X_1) \dot{X}_1 X_1^{-1} + \dot{X}_2 \exp(X_1) + \dot{X}_1 X_2 \exp(X_1) - t X_1 - Z_1 &= 0 \\ 2 X_1 \dot{X}_1 + \dot{X}_1 X_2 + X_1 \dot{X}_2 + 2 X_2 \dot{X}_2 - \dot{Z}_2 - 1 &= 0\end{aligned}\tag{C3}$$

This system is linear with respect to  $\dot{X}_1$  and  $\dot{X}_2$ :

$$\begin{aligned}\dot{X}_1 &= -\dot{X}_2 \left[ (X_1 + 2X_2) / (2X_1 + X_2) \right] + (1 + X_2) / (2X_1 + X_2) \\ \dot{X}_2 &= -\dot{X}_1 \left[ 2(\ln X_1) X_1^{-1} + X_2 \exp(X_1) \right] \exp(-X_1) + (Z_1 + t X_1) \exp(-X_1)\end{aligned}\tag{C4}$$

In general, the final calculation requires symbolic matrix inversion. The resulting set of explicit ordinary differential equations then is

$$\begin{aligned}\dot{X}_1 &= \frac{1 + X_2 - (X_1 + 2X_2)(X_3 + X_1X_5)\exp(-X_1)}{2X_1 + X_2 - (X_1 + 2X_2)[2X_1^{-1}\ln X_1 + X_2\exp(X_1)]\exp(-X_1)} \\ \dot{X}_2 &= \frac{(2X_1 + X_2)(X_3 + X_1X_5) - (1 + X_2)[2X_1^{-1}\ln X_1 + X_2\exp(X_1)]}{(2X_1 + X_2)\exp(X_1) - (X_1 + 2X_2)[2X_1^{-1}\ln X_1 + X_2\exp(X_1)]} \\ \dot{X}_3 &= X_1 \\ \dot{X}_4 &= X_2 \\ \dot{X}_5 &= 1\end{aligned}\tag{C5}$$

where we have defined  $X_3 = Z_1$ ,  $X_4 = Z_2$ , and  $X_5 = t$ . The initial conditions for  $X_1$  and  $X_2$  are calculated from the initial values for  $X_3$ ,  $X_4$ , and  $X_5$  by solving once the set of implicit algebraic equations (C2).

*We thank Dr. Douglas Irvine for helpful suggestions throughout this work, and Dr. Carl Simon for constructive criticism of the manuscript. This work was supported in part by U.S. Public Health Service grant GM 30054 from the National Institutes of Health.*

#### REFERENCES

- 1 P. W. Berg and J. L. McGregor, *Elementary Partial Differential Equations*, Holden-Day, San Francisco, 1966.
- 2 L. von Bertalanffy, Principles and theory of growth, in W. W. Nowinski, (Ed.), *Fundamental Aspects of Normal and Malignant Growth*, Elsevier, New York, 1960, pp. 137–259.
- 3 P. Deuffhard, Recent progress in extrapolation methods for ordinary differential equations, *SIAM Review* 27:505–535 (1985).
- 4 T. E. Hull, W. H. Enright, B. M. Fellen, and A. E. Sedgwick, Comparing numerical methods for ordinary differential equations, *SIAM J. Numer. Anal.* 9:603–637 (1972).
- 5 D. H. Irvine and M. A. Savageau, Network regulation of the immune response: Alternative control points for suppressor modulation of effector lymphocytes, *J. Immunol.* 134:2100–2116 (1985).
- 6 D. H. Irvine and M. A. Savageau, Network regulation of the immune response: Modulation of suppressor lymphocytes by alternative signals including contrasuppression, *J. Immunol.* 134:2117–2130 (1985).
- 7 D. H. Irvine and M. A. Savageau, Numerical solution of nonlinear ordinary differential equations expressed in canonical forms, submitted.
- 8 D. H. Irvine and M. A. Savageau, ESSYNS—Evaluation and simulation of synergistic systems—an interactive program for computer assisted mathematical analysis and simulation of organizationally complex system, submitted.
- 9 N. MacDonald, Time lags in biological models, *Lecture Notes in Biomathematics, Vol. 27*, Springer Verlag, Berlin, Heidelberg, New York, 1978.
- 10 M. Peschel and W. Mende, Volterra-Modelle für nichtlineare und instationäre Systeme und deren Äquivalenztransformationen, *Messen-Steuern-Regeln* 25:102–104 (1982).

- 11 M. Peschel and W. Mende, Riccati-Darstellung nichtlinearer Systeme und einige messtechnische Konsequenzen, *Messen-Steuern-Regeln* 26:67–69 (1983).
- 12 M. Peschel and W. Mende, A unified modelling concept for nonlinear systems with Lotka-Volterra equations, *Syst. Anal. Model. Simul.* 1:17–26 (1984).
- 13 M. Peschel and W. Mende, *The Predator–Prey Model: Do We Live in a Volterra World?* Akademie-Verlag, Berlin, 1986.
- 14 M. A. Savageau, Biochemical systems analysis: I. Some mathematical properties of the rate law for the component enzymatic reactions, *J. Theor. Biol.* 25:365–369 (1969).
- 15 M. A. Savageau, Biochemical systems analysis: II. The steady-state solutions for an  $n$ -pool system using a power-law approximation, *J. Theor. Biol.* 25:370–379 (1969).
- 16 M. A. Savageau, Biochemical systems analysis: III. Dynamic solutions using a power-law approximation, *J. Theor. Biol.* 26:215–226 (1970).
- 17 M. A. Savageau, The behavior of intact biochemical control systems, *Curr. Top. Cell Reg.* 6:63–130 (1972).
- 18 M. A. Savageau, Comparison of the classical and autogenous system of regulation in inducible operons, *Nature* 252:546–549 (1974).
- 19 M. A. Savageau, Optimal design of feedback control by inhibition: dynamic considerations, *J. Mol. Evol.* 5:199–222 (1975).
- 20 M. A. Savageau, *Biochemical Systems Analysis: A Study of Function and Design in Molecular Biology*, Addison-Wesley, Reading, Massachusetts, 1976.
- 21 M. A. Savageau, Growth of complex systems can be related to the properties of their underlying determinants, *Proc. Nat. Acad. Sci.* 76:5413–5417 (1979).
- 22 M. A. Savageau, Allometric morphogenesis of complex systems: derivation of the basic equation from first principles, *Proc. Nat. Acad. Sci.* 76:6023–6025 (1979).
- 23 M. A. Savageau, Growth equations: A general equation and a survey of special cases, *Math. Biosci.* 48:267–278 (1980).
- 24 M. A. Savageau, A suprasystem of probability distributions, *Biometr. J.* 24:323–330 (1982).
- 25 M. A. Savageau, Models of gene function: General methods of kinetic analysis and specific ecological correlates, in H. W. Blanch, E. T. Papoutsakis, and G. N. Stephanopoulos (Eds.), *Foundations of Biochemical Engineering*, American Chemical Society, Washington, D.C., 1983, pp. 3–25.
- 26 M. A. Savageau, Mathematics of organizationally complex systems, *Biomed. Biochim. Acta* 44:839–844 (1985).
- 27 M. A. Savageau, A theory of alternative designs for biochemical control systems, *Biomed. Biochim. Acta.* 44:875–880 (1985).
- 28 M. A. Savageau, Coupled circuits of gene regulation, in R. Calendar and L. Gold (Eds.), *Sequence Specificity in Transcription and Control*, Alan R. Liss, New York, 1985, pp. 633–642.
- 29 M. A. Savageau and E. O. Voit, Power-law approach to modeling biological systems: I. Theory, *J. Ferment. Technol.* 60:221–228 (1982).
- 30 M. A. Savageau, E. O. Voit, and D. H. Irvine, Biochemical systems theory and metabolic control theory: I. Fundamental similarities and differences, *Math. Biosci.* 86:127–145 (1987).
- 31 L. F. Shampine, H. A. Watts, and S. M. Davenport, Solving nonstiff ordinary differential equations—the state of the art, *SIAM Review* 18:376–411 (1976).
- 32 D. F. Smith, Quantitative analysis of the functional relationships existing between ecosystem components: II. Analysis of non-linear relationships, *Oecologia* 16:107–117 (1974).



- 33 D. F. Smith, Quantitative analysis of the functional relationships existing between ecosystem components: III. Analysis of ecosystem stability, *Oecologia* 21:17–29 (1975).
- 34 A. Uppal, W. H. Ray, and A. B. Poore, On the dynamic behavior of continuous stirred tank reactors, *Chem. Eng. Sci.* 29:967–985 (1974).
- 35 J. H. Verner, Quadratures for implicit differential equations, *SIAM J. Num. Anal.* 7:373–385 (1970).
- 36 E. O. Voit, Cell cycles and growth laws: The CCC-model, *J. Theor. Biol.* 114:589–599 (1985).
- 37 E. O. Voit and H. J. Anton, Derivation of the frequency distribution of cycle durations from continuous labeling curves, *J. Theor. Biol.* 112:575–588 (1985).
- 38 E. O. Voit and M. A. Savageau, Power-law approach to modeling biological systems: II. Application to ethanol production, *J. Ferment. Technol.* 60:229–232 (1982).
- 39 E. O. Voit and M. A. Savageau, Power-law approach to modeling biological systems: III. Methods of analysis, *J. Ferment. Technol.* 60:233–241 (1982).
- 40 E. O. Voit and M. A. Savageau, Analytical solutions to a generalized growth equation, *J. Math. Anal. Appl.* 103:380–386 (1984).
- 41 E. O. Voit and M. A. Savageau, S-system analysis of biological systems. in V. Capasso, E. Grosso, and S. L. Paveri-Fontana (Eds.), *Lecture Notes in Biomathematics, Vol. 57*, Springer-Verlag, Berlin, 1985, pp. 517–524.
- 42 E. O. Voit and M. A. Savageau, Equivalence between S-systems and Volterra-systems, *Math. Biosci.* 78:47–55 (1986).
- 43 E. O. Voit and M. A. Savageau, Accuracy of alternative representations for integrated biochemical systems, *Biochemistry*, to appear.