Journal of Theoretical Biology 304 (2012) 172–182

Contents lists available at SciVerse ScienceDirect

Journal of Theoretical Biology

journal homepage: www.elsevier.com/locate/yjtbi

Guaranteed error bounds for structured complexity reduction of biochemical networks

Thomas P. Prescott [a,b,*], Antonis Papachristodoulou [b,c]

[a] Life Sciences Interface Doctoral Training Centre, University of Oxford, Parks Road, Oxford OX1 3QU, UK
[b] Department of Engineering Science, University of Oxford, Parks Road, Oxford OX1 3PJ, UK
[c] Oxford Centre for Integrative Systems Biology, University of Oxford, South Parks Road, Oxford OX1 3QU, UK

ARTICLE INFO

Article history:
Received 16 November 2011
Received in revised form
1 April 2012
Accepted 2 April 2012
Available online 9 April 2012

Keywords:
Model reduction
Systems biology
Mathematical biology
Singular perturbation

ABSTRACT

Biological systems are typically modelled by nonlinear differential equations. In an effort to produce high fidelity representations of the underlying phenomena, these models are usually of high dimension and involve multiple temporal and spatial scales. However, this complexity and associated stiffness makes numerical simulation difficult and mathematical analysis impossible. In order to understand the functionality of these systems, these models are usually approximated by lower dimensional descriptions. These can be analysed and simulated more easily, and the reduced description also simplifies the parameter space of the model. This *model reduction* inevitably introduces error: the accuracy of the conclusions one makes about the system, based on reduced models, depends heavily on the error introduced in the reduction process.

In this paper we propose a method to calculate the error associated with a model reduction algorithm, using ideas from dynamical systems. We first define an error system, whose output is the error between observables of the original and reduced systems. We then use convex optimisation techniques in order to find approximations to the error as a function of the initial conditions. In particular, we use the Sum of Squares decomposition of polynomials in order to compute an upper bound on the worst-case error between the original and reduced systems. We give biological examples to illustrate the theory, which leads us to a discussion about how these techniques can be used to model-reduce large, structured models typical of systems biology.

1. Introduction

Biological and biochemical processes involve multiple temporal and spatial scales, and use feedback mechanisms extensively. High fidelity mathematical and computational models are being developed (Murray, 2002) in order to understand how these features endow such large networks with robust functionality and performance guarantees (El-Samad et al., 2005). However, often these integrative models are so complicated that they form barriers to our understanding of how biological systems have evolved to function this way. For example, Chen et al. (2009) developed a model consisting of around 500 ODEs with more than 200 parameters to describe the input–output behaviour of ErbB signalling pathways. Simulations of such large-scale models can become unwieldy or impossible to carry out, and much of the behaviour is intractable to rigourous mathematical analysis. It is also impractical to identify such a large number of global system parameters. Instead, reduced-order models are usually used to draw conclusions about the functionality and structure of these networks. However, a reduced model is useful only if it is accurate enough.

Take, for example, the simple enzymatic reaction

$$E+S\underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}}ES\xrightarrow{k_2}E+P \quad (1)$$

in which a substrate S (with initial concentration s_0) is converted to a product P through the action of the enzyme E (with initial concentration e_0). Processes of this form are typical in biological systems; a metabolic pathway may have tens or even hundreds of such steps. To model them, one can consider mass-action kinetics, which results in a model of the form

$$\frac{dc}{dt}=k_1(e_0-c)s-(k_{-1}+k_2)c, \quad (2)$$

$$\frac{ds}{dt}=-k_1(e_0-c)s+k_{-1}c, \quad (3)$$

where $c=[ES]$ and $s=[S]$, and the concentrations of the other two reactants can be derived by mass conservation laws leading to algebraic equations. However, many times the so-called

[*] Corresponding author at: Department of Engineering Science, University of Oxford, Parks Road, Oxford, OX1 3PJ, UK.
E-mail address: thomas.prescott@dtc.ox.ac.uk (T.P. Prescott).

0022-5193/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.jtbi.2012.04.002

Michaelis–Menten approximation is used instead

$$\frac{ds}{dt} = -\frac{k_2 e_0 s}{K_m + s}, \tag{4}$$

where $K_m = (k_{-1}+k_2)/k_1$ is the Michaelis–Menten constant and $c = e_0 s/(K_m+s)$. If one treats (2) and (3) as the original system and (4) as its reduced-order version, what is the error introduced by this approximation? Furthermore, are the conclusions reached about the reduced order model still valid for the original model and (more importantly) for the system being modelled?

Model reduction is a well-studied field in systems theory (Zhou et al., 1996; Antoulas, 2005): it concerns the reduction of a dynamical system model in such a way that the observed dynamics remain approximately identical to those of the original system. Much of this theory concerns linear systems both in the frequency and the time domains. There have been some extensions of standard linear reduction techniques: for example, balanced truncation (Dullerud and Paganini, 2000, Chapter 4) applied to nonlinear systems (Scherpen, 1993), whereby variables with relatively unobservable and uncontrollable dynamics can be neglected. Other nonlinear reduction techniques include fitting low-dimensional dynamics to the input–output behaviour of the full system (Löhning et al., 2011); decomposition into an interconnection of linear and nonlinear systems, where the former can be reduced through well-understood techniques (Besselink et al., 2009); and Petrov–Galerkin projections (Antoulas, 2005; Matthies and Meyer, 2003).

Many of these approaches, however, alter the co-ordinates of the internal dynamics to physically meaningless quantities. This is undesirable for biological and biochemical applications, where the internal state variable dynamics are physically meaningful, and it would be preferable to keep the correspondence to the original states in a reduced order model (Borghans et al., 1996). One recently developed reduction technique, termed *zooming*, has been developed by Sunnåker et al. (2011) to deal with this problem. By assuming variables within conservation relations are at quasi-steady state and calculating their fraction parameters, the method ensures (in many cases) that any such reduction retains an explicit mapping from the state-space of the reduced system back to that of the original system, in order to compare the dynamics of the two models directly. Many other such structured model reduction techniques have been developed in recent years (Sandberg and Murray, 2009).

The main advantage of many model reduction techniques in control theory is that an error bound can be obtained, so that the user can choose how much to reduce the system by, based on the error that can be tolerated. Many reduction techniques have associated error bounds; see, for example, standard results for balanced reduction (Dullerud and Paganini, 2000). There are also techniques for bounding the error when reducing certain classes of linear time-varying (Haasdonk and Ohlberger, 2011) and nonlinear (Wirtz and Haasdonk, 2012) systems.

Model reduction is also prevalent in more traditional mathematical biology, but often these methods are not easily generalised. Additionally the lack of error quantification can mean that several reduction schemes can exist for a given model (see Tzafriri, 2003, for example). Approaches that have been identified include lumping (Huang et al., 2005) and also singular perturbation (Zagaris et al., 2004)—for an overview see Okino and Mavrovouniotis (1998). Singular perturbation is particularly attractive as there is no requirement for the transformation of state space, and thus there remains some physical intuition within the reduced system components.

The topic of this paper is the use of systems theory and Sum of Squares (SOS) programming for quantifying the error in reductions of biological models, extending similar theory from linear systems (Anderson et al., 2011; Papachristodoulou et al., 2010; Chesi, 2010). Systems theory is extended to handle nonlinearity directly while considering approaches that preserve the meaning of the variables in the reduced system. Our approach is based on convex optimisation, which can be used to bound the total output error of a model reduction method. Several examples are then presented: first, the enzyme kinetics and the quasi-steady state assumption (QSSA) discussed above, as well as a simple example from Lee and Othmer (2010) of a dimerisation/isomerisation network. We also present an example of how this method can be used to reduce larger networks and discuss how complicated systems of even greater dimension can be reduced.

2. Methodology

2.1. Structured nonlinear model reduction

Biosystem models typically take the form

$$\dot{x} = f(x), \quad y = h(x), \quad x(0) = x_0, \tag{5}$$

where $x(t) \in \mathcal{D} \subseteq \mathbb{R}^n$ is the state in some domain $\mathcal{D}$ and $y(t) \in \mathbb{R}^m$ is the output. We assume that 0 is a steady-state of this system, i.e., $f(0) = 0$. We consider systems for which this equilibrium is stable. Suppose that we have derived a reduced system of the form

$$\dot{\hat{x}} = \hat{f}(\hat{x}), \quad \hat{y} = \hat{h}(\hat{x}), \quad \hat{x}(0) = \hat{x}_0, \tag{6}$$

where $\hat{x}(t) \in \hat{\mathcal{D}} \subseteq \mathbb{R}^{\hat{n}}$ (for $\hat{n} \ll n$) and $\hat{y}(t) \in \mathbb{R}^m$, and where (again) $\hat{f}(0) = 0$. For the remainder of the paper, reductions are assumed to be structured: that is, for all times t the $\hat{n}$ components of the reduced system $\hat{x}(t)$ are identified with a subset of size $\hat{n}$ of the set of components of the full state vector $x(t)$. This structure is maintained in the initial conditions, so that $\hat{x}_0$ equals this subset of $\hat{n}$ co-ordinates of x_0. This corresponds to a reduction where the states in the reduced system keep their physical interpretation.

Then we can define an error system using the full dynamical (5) and the reduced (6). Denoting the error state-space variable $x_e = [x^T, \hat{x}^T]^T$ and the error output z_e, the error system is given by

$$\dot{x}_e = F(x_e) = \begin{bmatrix} f(x) \\ \hat{f}(\hat{x}) \end{bmatrix},$$

$$z_e = H(x_e) = h(x) - \hat{h}(\hat{x}),$$

$$x_e(0) = \begin{bmatrix} x_0 \\ \hat{x}_0 \end{bmatrix} \tag{7}$$

so that the output of the error system is the difference in outputs between the full system and its reduction.

To quantify this error, we will use a dynamical systems representation. Suppose that there exists a differentiable function $V : \mathcal{D} \times \hat{\mathcal{D}} \to \mathbb{R}$ which satisfies

$$V(0) = 0, \tag{8}$$

$$V(x_e) > 0 \text{ on } x_e \in \mathcal{D} \times \hat{\mathcal{D}} \backslash \{0\}, \tag{9}$$

$$\frac{dV}{dt} + z_e^T(t) z_e(t) \leq 0 \text{ on } x_e \in \mathcal{D} \times \hat{\mathcal{D}}. \tag{10}$$

If these conditions are satisfied, then V is a *storage function* with respect to the supply $z_e^T(t) z_e(t)$ (Haddad and Chellaboina, 2008) in that, through integration of (10), it satisfies

$$V(x_e(t)) \leq V(x_e(0)) - \int_0^t z_e^T z_e \, dt' \tag{11}$$

for all $t \geq 0$. Thus, letting $t \to \infty$ above it can be seen that $V(x_e(0))$ is an upper bound on the norm of the output

$$V(x_e(0)) \geq \int_0^\infty z_e^T z_e \, dt = \|z_e\|^2. \tag{12}$$

A formal statement and proof can be found in Appendix A.

If a function V can be found such that (10) holds with equality then we can find the output energy exactly. This is well-characterised in linear systems with linear reductions: then we can express $F(x_e)=A_ex_e$ and $H(x_e)=C_ex_e$ in (7). The *Gramian* matrix

$$Q=\int_0^\infty e^{A_e^Tt}C_e^TC_ee^{A_et}\,\mathrm{d}t$$

is well-defined for stable linear systems and positive definite for observable ones (Zhou et al., 1996). The Gramian Q defines a quadratic function $V(x_e)=x_e^TQx_e$ for which (10) holds with equality. The difficulty of using V to bound the output in the general case (where F and H may be nonlinear functions) lies in finding such a V and optimising it such that the upper bound (12) is as tight as possible.

Note that this approach is very closely related to the search for a Lyapunov function to prove stability. If the $z_e^Tz_e$ term of (10) is ignored, conditions (8)–(10) are Lyapunov conditions. Finding a storage function proves Lyapunov stability, since $z_e^Tz_e\geq 0$. Using the error term in (10), we can now apply Sum of Squares (SOS) techniques, typically used for stability analysis, to find functions which bound the reduction error.

2.2. Finding the storage function of a nonlinear system

In general, the positive definite storage function $V(x_e)$ satisfying (12) is not of the quadratic form $V(x_e)=x_e^TQx_e$ for any matrix Q. The next step up in complexity for a storage function is a higher order polynomial. To compute this, we introduce the Sum of Squares (SOS) decomposition of multi-variable polynomials.

Definition 1 (*Sum of Squares (SOS)*). A polynomial function $p:\mathbb{R}^n\rightarrow\mathbb{R}$ is SOS if (for some $m\in\mathbb{N}$) there exist polynomial functions $q_i:\mathbb{R}^n\rightarrow\mathbb{R}$, for $i=1,\ldots,m$, such that

$$p(x)=\sum_{i=1}^m(q_i(x))^2.$$

It is clear that if p is SOS, then $p\geq 0$, but the converse is not necessarily true. It can also be shown (Powers and Wörmann, 1998) that p being SOS of degree $2d$ is equivalent to the existence of a positive semi-definite matrix Q such that

$$p(x)=Z(x)^TQZ(x),\tag{13}$$

where Z is a column vector of monomials of the variables in x up to degree d. Thus, the problem of finding a SOS function p of even degree $2d$ is equivalent to finding a positive semi-definite Q (Parrilo, 2000), which can be computed using convex optimisation, in particular semi-definite programming. SOSTOOLS (Prajna et al., 2004), implemented as a MATLAB add-on, performs the conversion of the SOS problem to the corresponding semi-definite program and then translates the result back into the terms of the original SOS problem, thus allowing a computational approach to the very difficult problem of finding a storage function.

Section 2.1 derived conditions (8)–(10) on the function V, all of which could be restated as SOS constraints. That is, if a SOS function V could be found such that

$$-\dot{V}(x_e)-z_e^Tz_e\text{ is SOS}\tag{14}$$

then V would be a storage function and the output error could be bounded above by $V(x_e(0))$. Prajna and Sandberg (2005) suggest as a heuristic for finding an optimal V, in the sense of being a close upper bound on the output norm, that the objective function to be minimised by SOSTOOLS is the trace of Q in (13).

Once a storage function V has been found, the upper bound

$$V(x_e(0))\geq\|z_e\|^2$$

is valid. Since we are assuming a structured reduction, the initial conditions of the reduced system are a projection of those of the full system; $\hat{x}_0$ is determined by x_0. Therefore, we can write

$$V(x_e(0))=V_e(x_0)$$

as a function of the initial conditions of the full system only. The inequality above means that the largest output energy from any initial condition of norm at most r, say, can be bounded above by

$$\max_{|x_0|\leq r}V_e(x_0)\geq\max_{|x_0|\leq r}\|z_e\|^2.$$

The bound is a function of the initial conditions of the full system, so in maximising the bound on the domain of initial conditions $|x_0|\leq r$ then we find a bound on the *worst-case* error: an upper bound on the largest error (measured as $\|y-\hat{y}\|$) that can be introduced by the structured reduction.

The proposition below shows that maximising $V_e(x_0)$ over a ball of n dimensions is equivalent to a one-dimensional search for the smallest parameter a such that the set P_a, defined below, is empty.

Proposition 1. *The maximum of any continuous non-negative function*[1] V_e *on a ball of radius r is equal to*

$$\gamma=\inf\{a\geq 0|P_a=\emptyset\},\tag{15}$$

where P_a for a particular $a\geq 0$ is given as

$$P_a=\{x_0\in\mathbb{R}^n||x_0|\leq r\text{ and }V_e(x_0)\geq a\}.$$

A proof can be found in Appendix B.

We can now apply SOSTOOLS again, this time to maximise the storage function $V_e(x_0)$ on a ball in $\mathbb{R}^n$. Applying a theorem from algebraic geometry known as *Positivstellensatz* (Bochnak et al., 1998), we can state that $P_a=\emptyset$ is implied by the existence of SOS polynomials s_1 and s_2 such that

$$s_1+s_2(r^2-|x_0|^2)+(V_e(x_0)-a)=0.$$

This can be formulated as a SOS constraint, and we can thus make the SOS program perform a minimisation of a subject to such a constraint. Minimising a, by exploiting the result of Proposition 1, is equivalent to maximising V.

Note that it is not guaranteed that arg max $V_e(x_0)$ is equal to the initial condition maximising $\|z_e\|^2$. However, storage functions can be used to indirectly reduce models by considering the output of an associated error system. With these tools, a greedy model reduction algorithm can be developed in the spirit of Anderson et al. (2011), which uses the error system method for model reduction. Performing the reduction state by state, we can find the single state of the system with the smallest worst-case error and reduce that state if the total error is still within tolerable limits. This continues in an iterative way until the error allowance is used up. This algorithmic approach is used in Section 3.3.

2.3. Discussion

Note that (11) also implies that $V(x_e(0))$ is additionally an upper bound on any *finite-time* 2-norm of z_e over [0,T], and the quality of this bound improves as T increases. However, we cannot apply the storage function methods introduced in this paper to bound L_∞ norms or for instantaneous error estimation of $z_e(t)=|y(t)-\hat{y}(t)|$.

The latter of these error norms is also important in biological applications: there may be a requirement that the system output is near a nominal level at some pre-determined finite time. Wirtz and Haasdonk (2012) have recently examined this problem in

[1] Of course, by (8) and (9) any storage function $V(x_e(0))=V_e(x_0)$ will satisfy these conditions.

relation to nonlinear systems described by kernel functions and reduced by Petrov–Galerkin methods, using improved "local" Lipschitz constants to bound the error. Previous to this, a setting allowing general error norms (although integrated through time rather than instantaneous) was investigated by Matthies and Meyer (2003) by reducing through nonlinear Galerkin methods and using duality to bound the error. Distinct from the assumptions used in the recent work on error bounds, biochemical models generally have polynomial dynamics (resulting from mass-action kinetic models) and are often reduced through singular perturbation. It will be seen in the remainder of this paper that these are therefore a natural setting for our method.

For the majority of biological systems, parameter values such as reaction rates are highly uncertain, or may vary very slowly over the timescale investigated, but in the theory above we have assumed fixed parameter values. We can extend our methods to deal with uncertainty by using constraints on auxiliary variables in the SOS program (Papachristodoulou and Prajna, 2002; Chesi, 2010). Three examples of increasing complexity are presented below where we vary fixed parameters, but we can in principle still bound the worst-case error in the uncertain case.

3. Examples for quantifying the worst-case error

The theory above is applied below to two small systems where timescales can be separated in order to quantify the error introduced by the reductions, after which we also give a third (more complex) example.

3.1. Enzyme kinetics

As a first example of calculating the worst-case error of a reduced nonlinear system, we can look at the classic equation for enzyme kinetics and the Michaelis–Menten model reduction discussed in Section 1. We will use this example to illustrate in detail how SOSTOOLS is used.

Using the law of mass action and standard notation (Murray, 2002), the dynamics of (1) can be expressed by the following four equations:

$$\frac{de}{dt} = -k_1 es + k_{-1}c + k_2 c,$$

$$\frac{ds}{dt} = -k_1 es + k_{-1}c,$$

$$\frac{dc}{dt} = k_1 es - k_{-1}c - k_2 c,$$

$$\frac{dp}{dt} = k_2 c$$

for rate constants k_i. From these equations, two conservation laws can be deduced, namely

$$e + c = e_0 \quad \forall t, \tag{16}$$

$$s + c + p = s_0 \quad \forall t \tag{17}$$

for two constants e_0 and s_0. Thus, (1) can be represented (with no error) in a two-dimensional state-space in the variables (c,s):

$$\frac{dc}{dt} = k_1(e_0 - c)s - (k_{-1} + k_2)c, \tag{18}$$

$$\frac{ds}{dt} = -k_1(e_0 - c)s + k_{-1}c. \tag{19}$$

The aim is to reduce this two-dimensional system to a single variable. The quality of the reduction can be measured by comparing the outputs and measuring the errors between them.

3.1.1. Physical realism, stability and initial conditions

All of the state-space variables in the system must be non-negative to have any biological meaning. This restricts the state-space of the full system (18) and (19) to

$$\mathcal{D} = \{(c,s) \in \mathbb{R}^2_+ \,|\, 0 \le c + s \le s_0;\ 0 \le c \le e_0\},$$

where, without any loss of generality, the initial condition of p is assumed to be $p(0) = 0$ and hence $s_0 = c(0) + s(0)$ is the initial total substrate, both free and bound.

The origin is the unique stationary point of the system. Asymptotic stability can be investigated within $\mathcal{D}$ by, for example, Lyapunov methods. Defining

$$V(c,s) = c + s$$

gives a certificate of asymptotic stability for all $(c,s) \in \mathcal{D}$. Ideally, the system would be globally asymptotically stable—that is, a trajectory of the system with initial conditions outside the domain of physical realism $\mathcal{D}$ would always tend towards 0. This would allow us to look for a global storage function from which we can estimate a worst-case error. However, there exist initial conditions for which the system is unstable. Although these are physically unrealistic and therefore have no impact on the stability of the physical system, this means that we have to be careful in finding a storage function as the search must be restricted to a bounded neighbourhood of the equilibrium state.

3.1.2. Singular perturbation to reduce the model

The classical approach (Murray, 2002) to the system described in (18) and (19) is to make the quasi-steady state assumption (QSSA), whereby a short initial transition is neglected and the complex is assumed to be in equilibrium, setting the right-hand side of (18) to zero. This gives an expression for the complex c as a function of substrate s which can be substituted into (19) to give the one-dimensional dynamics of the reduced system, hence

$$\tilde{c} = \frac{e_0 \tilde{s}}{K_m + \tilde{s}}, \tag{20}$$

$$\frac{d\tilde{s}}{dt} = -\frac{k_2 e_0 \tilde{s}}{K_m + \tilde{s}}, \tag{21}$$

where $K_m = (k_{-1} + k_2)/k_1$ is the Michaelis constant. Note that $\tilde{s} = 0$ is the unique stationary point, which is asymptotically stable for all initial conditions $\tilde{s}(0) > -K_m$, which certainly includes all physically realistic initial conditions.

Other reduction schemes have been proposed to describe (1). For example, Tzafriri (2003) reviews two others: the rQSSA (r for reverse), where ds/dt is the set equal to zero, and the tQSSA (t for total), where the substrate dynamics are replaced with those of $s + c = \bar{s}$. However, for simplicity the reduction above, known as the sQSSA (s for standard), is the example we will investigate.

3.1.3. Error estimation

The aim of this section is to demonstrate how a storage function can be constructed to give an upper bound on the worst-case error between the model and its reduction, as described in Section 2.

System 1 (*Full system*). The state space of the full system is $x = (c,s)^T$ which follows the dynamics in (18) and (19). The output of this system is assumed to be s.

System 2 (*Reduced system*). The state space of the reduced system is $\tilde{s}$ which follows the dynamics in (21). The output of this system is assumed to be $\tilde{s}$.

We take these two systems together to define the error system.

System 3 (*Error system*). The state space of the error system is $x_e = (c,s,\tilde{s})^T$ with dynamics given by (18), (19) and (21). The output of the system is $z_e = s - \tilde{s}$.

The reduction is structured, and the initial conditions of the reduced system will equal those of their corresponding states in the full system. That is, if System 1 starts from an initial condition $(c(0),s(0))$, then System 2 starts from $s(0)$, and hence System 3 starts from $(c(0),s(0),s(0))$.

There are two modifications that must be made to the method described in Section 2. First, Section 3.1.1 showed that System 1 is not globally asymptotically stable; hence, neither is System 3. Therefore, a function defined on the entire state space which satisfies condition (10) cannot be found (else this would imply global asymptotic stability). The search for the required storage function must be limited to a domain within the state space. Second, the dynamics given by (21) are described by a rational, not polynomial, function of $\tilde{s}$.

To deal with the latter, we modify (14) by multiplying by the denominator of the rational function in (21), to result in a polynomial condition

$$\left(-\frac{\mathrm{d}V}{\mathrm{d}t}(x_e) - z_e^T z_e\right)(K_m + \tilde{s}) \text{ is SOS,} \tag{22}$$

which ensures that (10) holds on $\tilde{s} > -K_m$.

We then deal with the former problem. As discussed by Papachristodoulou and Prajna (2002), suppose that we want to ensure that (22) holds in a particular region of interest $\mathbf{K} = \{x_e \mid g(x_e) \geq 0\}$ for some polynomial g. Then, we should constrain the condition further with an SOS multiplier $\sigma(x_e) \geq 0$ such that

$$\left(-\frac{\mathrm{d}V}{\mathrm{d}t}(x_e) - z_e^T z_e\right)(K_m + \tilde{s}) - \sigma(x_e)g(x_e) \text{ is SOS,} \tag{23}$$

as then the expression in (22) is non-negative for all $x_e \in \mathbf{K}$ and thus (10) holds for all $x_e \in \mathbf{K} \cap \{x_e \mid \tilde{s} > -K_m\}$. However, this loosens the upper bound, as integration now gives

$$V(x_e(0)) - \int_{t=0}^{\infty} \frac{\sigma(x_e)g(x_e)}{K_m + \tilde{s}}\, \mathrm{d}t \geq \int_{t=0}^{\infty} z_e^T z_e \, \mathrm{d}t \tag{24}$$

as the upper bound on the output error $\|z_e\|_2^2$. Thus, the additional integral term should be as small as possible in order for $V(x_e(0))$ to be a reasonable approximation to the error.

Once such a $V(c,s,\tilde{s})$ is found, we seek to maximise it to give an upper bound on the worst-case error. As we have the additional constraint on initial conditions that $\tilde{s}(0) = s(0)$, we can substitute $\tilde{s}$ with s in the expression for V to make it a function $V_e(c,s)$ on $\mathbb{R}^2$ only. Hence we seek to maximise V_e on a disc. The maximum value can be found using *Positivstellensatz* as in Proposition 1.

3.1.4. Worst-case error from assuming quasi-steady state

The condition for the QSSA described in Section 3.1.2 to be valid is given (Murray, 2002) as

$$\frac{e_0}{s_0 + K_m} \ll 1, \tag{25}$$

where, without loss of generality, $s_0 = c(0) + s(0)$. If $e_0 \ll s_0$, then the only initial conditions are $c(0) \approx 0$ and $s(0) \approx s_0$, and the method of finding worst-case initial conditions is not meaningful. Analysis or a simulation would be sufficient to give us an idea of the error introduced by the single initial condition $(0,s_0)$.

However, when e_0 is of a comparable order to s_0 there is more freedom in initial conditions. Hence below, after first considering the traditional initial conditions $(0,s_0)$, we will then look at the worst-case error when taking the initial conditions $(c(0),s(0))$ to vary along the boundary of the disc of radius e_0 within the positive quadrant. Then, the size of parameter K_m will be varied, to look at the worst-case error introduced by the QSSA for various parameter regimes.

3.1.5. Traditional initial conditions

The QSSA validity regime given by (25) also assumes a fixed initial condition for the original system of

$$(c(0),s(0),e(0),p(0)) = (0,s_0,e_0,0). \tag{26}$$

We can verify that the reduced system does increase its accuracy for increasing K_m in terms of the error measure

$$\left(\int_0^{\infty} (s(t) - \tilde{s}(t))^2 \, \mathrm{d}t\right)^{1/2},$$

where we have initial conditions $c(0) = 0$, $s(0) = e_0$ and $\tilde{s}_0 = e_0$.

Recall that $K_m = (k_2 + k_{-1})/k_1$. Fig. 1 plots how upper bounds on the error vary with K_m, where K_m is varied by each of its three constituent parameters in turn. The upper bound on the error is given by V, evaluated at the fixed initial conditions. Also plotted is a simulation of the error introduced, to allow us to observe how the conservativeness of the upper bound given by V changes with K_m.

In all the three panels of Fig. 1, we can see that the error, both simulated and upper-bounded, falls with increasing K_m, which is in agreement with (25). Hence larger K_m leads to increased validity of the QSSA, at least for this initial condition.

3.1.6. Other initial conditions, and the worst case error

One strength of the method described in Section 3.1.3 is that we can estimate the biggest error in output between the full and reduced systems as the initial conditions vary. Fig. 2 shows the outcome of this, where the optimal storage function V is maximised over $|x_0| \leq e_0$ with the additional requirement of being in the positive quadrant. This gives an upper bound on the worst case error, which is compared with a lower bound on the worst-case error, where simulations were run for different initial conditions of the form $(e_0 \cos\theta, e_0 \sin\theta)$ for $\theta \in [0,\pi/2]$, and the largest of these errors is taken as the lower bound.

As in Fig. 1, in Fig. 2 we vary K_m by varying each of k_1, k_2 and k_{-1} in turn. Fig. 2(b) repeats the outcome of Fig. 1(b), in that the worst-case error between the full and reduced system decreases with increasing K_m. However, both Fig. 2(a) and (c) show the opposite results when compared to their equivalent plots in Fig. 1. When concentrating on the *worst-case* error, it actually increases with increasing K_m if it is controlled by $k_{\pm 1}$.

Hence when initial conditions are allowed to vary away from (26) and we are interested in the error introduced by the QSSA in this more general case, it can be concluded that the condition for the QSSA's validity given by (25) is no longer necessarily correct. This emphasizes the additional importance of the initial conditions in the validity of a reduction based on the QSSA.

3.2. Fast dimerisation and slow isomerisation

In their paper on model reduction through fast–slow timescale separation, Lee and Othmer (2010) consider the following reaction network:

$$2A \underset{k_{-1}/\epsilon}{\overset{k_1/\epsilon}{\rightleftharpoons}} B \underset{k_{-2}}{\overset{k_2}{\rightleftharpoons}} C, \tag{27}$$

where ϵ is small and all of the k_i are of the same order. These parameter choices mean that the reversible dimerisation reaction between A and B is fast and the isomerisation between B and C is slow. The full set of differential equations is

$$\dot{x}_1 = -\frac{2k_1}{\epsilon}x_1^2 + \frac{2k_{-1}}{\epsilon}x_2, \tag{28}$$

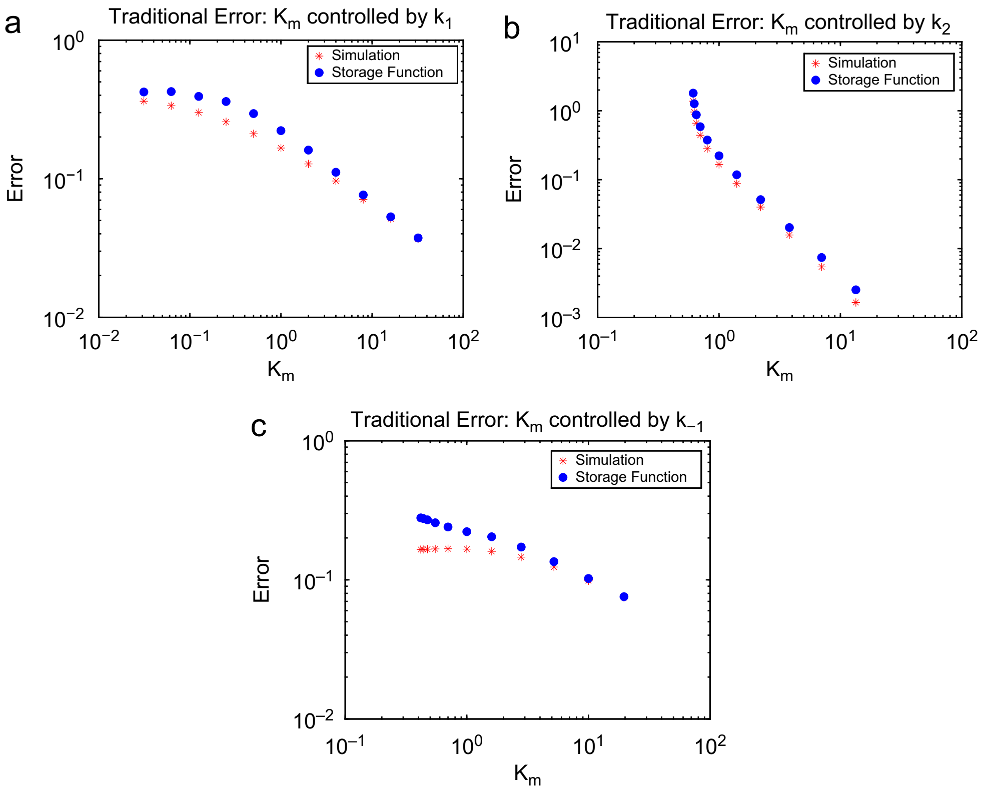

Fig. 1. Log–log plots measuring the error from the traditional initial conditions $c=0$, $s=s_0$, where error is defined as $(\int(s(t)-\tilde{s}(t))^2\,\mathrm{d}t)^{1/2}$, for base parameters $k_1=5$, $k_{-1}=3$ and $k_2=2$ where each is varied in turn. The error is estimated by both direct simulation and the evaluation of the storage function to give the upper bound (24). (a) Varying k_1, (b) varying k_2, and (c) varying k_{-1}.

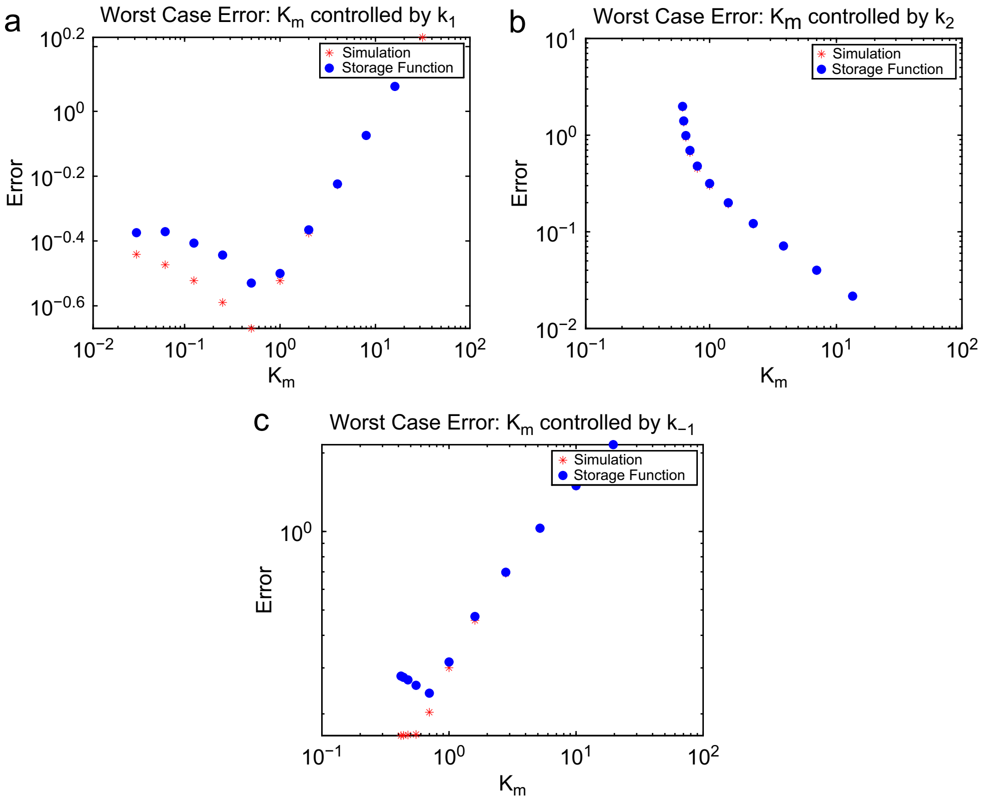

Fig. 2. Log–log plots measuring the worst-case error, where error is defined as $(\int(s(t)-\tilde{s}(t))^2\,\mathrm{d}t)^{1/2}$, for base parameters $k_1=5$, $k_{-1}=3$ and $k_2=2$ where each is varied in turn. The worst-case error is estimated by both direct simulation and the evaluation of the storage function to give the upper bound (24). (a) Varying k_1, (b) varying k_2, and (c) varying k_{-1}.

$$\dot{x}_2 = \frac{k_1}{\epsilon}x_1^2 - \left(\frac{k_{-1}}{\epsilon} + k_2\right)x_2 + k_{-2}x_3, \tag{29}$$

$$\dot{x}_3 = k_2x_2 - k_{-2}x_3, \tag{30}$$

where x_1 denotes the concentration of A, and so on. The stationary points of this system have the form $(s, \kappa_1 s^2, \kappa_2\kappa_1 s^2)$, where

$$\kappa_i = \frac{k_i}{k_{-i}}$$

for some $s \in \mathbb{R}$. The value of s is specified by the initial conditions due to the conservation equation

$$x_1(t) + 2x_2(t) + 2x_3(t) = K \quad \forall t, \tag{31}$$

which can be deduced from (28) to (30); that is, the unique physically realistic stationary point of the system is $(s, \kappa_1 s^2, \kappa_2\kappa_1 s^2)$, where

$$s = \frac{-1 + \sqrt{1 + 8K\kappa_1(1+\kappa_2)}}{4\kappa_1(1+\kappa_2)}. \tag{32}$$

Hence we can reformulate the full model in terms of the translated variables

$$(x_1, x_2, x_3) \mapsto (x_1 + s, x_2 + \kappa_1 s^2, x_3 + \kappa_2\kappa_1 s^2),$$

where s is uniquely determined by the parameter K in (32). We perform the translation so that the equilibrium of the system is at the origin. The model dimension can also be reduced from 3 to 2 with no error, as trajectories must lie on the plane defined by (31), and hence x_1 can be expressed in terms of x_2 and x_3 for a choice of model parameter K.

Thus, the full system above is a two-dimensional system with one physically realistic equilibrium point. The system is

$$\dot{x}_2 = \frac{k_1}{\epsilon}(K - 2(x_2 + x_3))^2 - \left(\frac{k_{-1}}{\epsilon} + k_2\right)x_2 + k_{-2}x_3, \tag{33}$$

$$\dot{x}_3 = k_2x_2 - k_{-2}x_3 \tag{34}$$

on $t \geq 0$ with initial condition $(x_2(0), x_3(0))$.

3.2.1. Reduced model A

The method Lee and Othmer (2010) propose justifies the following reduction of the full three-dimensional model down to two dimensions:

$$\dot{\hat{x}}_2 = \frac{4\sqrt{\hat{x}_2}}{4\sqrt{\hat{x}_2} + \sqrt{k_{-1}/k_1}}(k_{-2}\hat{x}_3 - k_2\hat{x}_2),$$

$$\dot{\hat{x}}_3 = k_2\hat{x}_2 - k_{-2}\hat{x}_3.$$

To ensure that the system dynamics can be described by rational polynomial functions, this pair of equations can be reformulated in terms of $\tilde{x}_2 = \sqrt{\hat{x}_2}$ and $\tilde{x}_3 = \hat{x}_3$ so that the reduction is

$$\dot{\tilde{x}}_2 = \frac{2(k_{-2}\tilde{x}_3 - k_2\tilde{x}_2^2)}{4\tilde{x}_2 + \sqrt{k_{-1}/k_1}}, \tag{35}$$

$$\dot{\tilde{x}}_3 = k_2\tilde{x}_2^2 - k_{-2}\tilde{x}_3. \tag{36}$$

The conservation relation in this case is found by combining (35) and (36) and integrating to give

$$\tilde{x}_3 + \tilde{x}_2\sqrt{1/4\kappa_1} + \tilde{x}_2^2 = L \tag{37}$$

for some constant L over all time $t \geq 0$. Using a similar argument to the full system above, the co-ordinates need to be translated to take into account the stationary point of the form $(\tilde{s}, \kappa_2\tilde{s}^2)$, where

$$\tilde{s} = \frac{-1 + \sqrt{1 + 16L\kappa_1(1+\kappa_2)}}{4\sqrt{\kappa_1}(1+\kappa_2)}$$

is the physically realistic stationary point.

Thus, the reduced system is only one-dimensional, as $\tilde{x}_3$ can be substituted by the re-arrangement of (37) into (35). Then, the model is simply

$$\dot{\tilde{x}}_2 = \frac{2(k_{-2}(L - \tilde{x}_2^2 - \tilde{x}_2\sqrt{1/4\kappa_1}) - k_2\tilde{x}_2^2}{4\tilde{x}_2 + \sqrt{1/\kappa_1}}. \tag{38}$$

3.2.2. Reduced model B

In Remark 7 of their paper, Lee and Othmer (2010) note that a much simpler reduction is available by letting $\epsilon \to 0$. The system then reduces to

$$\dot{\overline{x}}_2 = -\dot{\overline{x}}_3 = -k_2\overline{x}_2 + k_{-2}\overline{x}_3$$

with $k_1x_1^2 = k_{-1}x_2$ as the algebraic relation describing the fast dynamics. A simple conservation relation of $\overline{x}_2 + \overline{x}_3 = M$ for some constant M means that we can describe the reduction as the one-dimensional system

$$\dot{\overline{x}}_2 = M - (k_2 + k_{-2})\overline{x}_2. \tag{39}$$

Translating co-ordinates to take into account the stationary point of the form $(\overline{s}, \kappa_2\overline{s})$, where

$$\overline{s} = \frac{M}{1+\kappa_2}$$

gives a reduced system with unique equilibrium at 0.

3.2.3. Error between models

Lee and Othmer (2010) conclude from their analysis that reduced model A introduces less error than reduced model B. The methods of Section 2 should allow us to verify their conclusions. We have the choice of K as a system parameter, and in order to compare the reductions we choose the other conservation constants L and M so that the equilibria between the corresponding (non-translated) systems are the same.

Figs. 3 and 4 show the output of two approaches to quantifying the error between the full model and its reductions. Each figure corresponds to a different parameter set. The first method of comparison is a simulation of the error, in the top panel of each

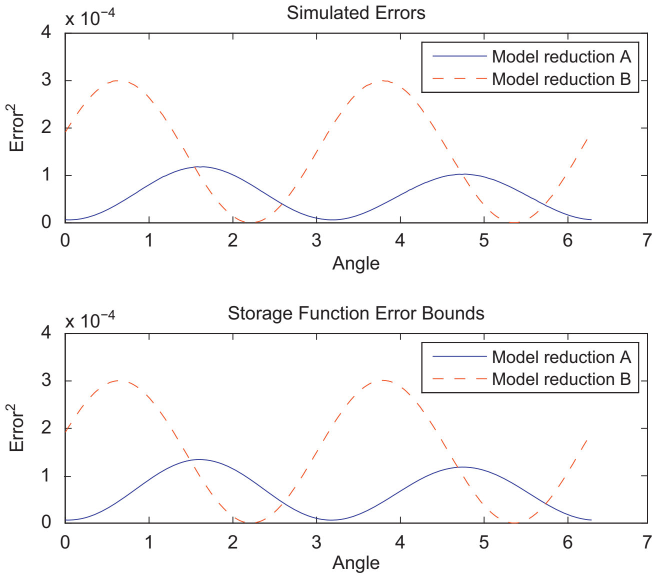

Fig. 3. Error varying along the boundary of a circle around the equilibrium point of the error system. Parameters used here are $k_1 = k_2 = 1$, $k_{-1} = k_{-2} = 5$, $\epsilon = 0.1$ and $K = 10$. The errors are $\|x_2 - \tilde{x}_2^2\|^2$ for reduction A and $\|x_2 - \overline{x}_2\|^2$ for reduction B. Here, we compare the simulated (top) and bounded (bottom) error between the two reduction schemes.

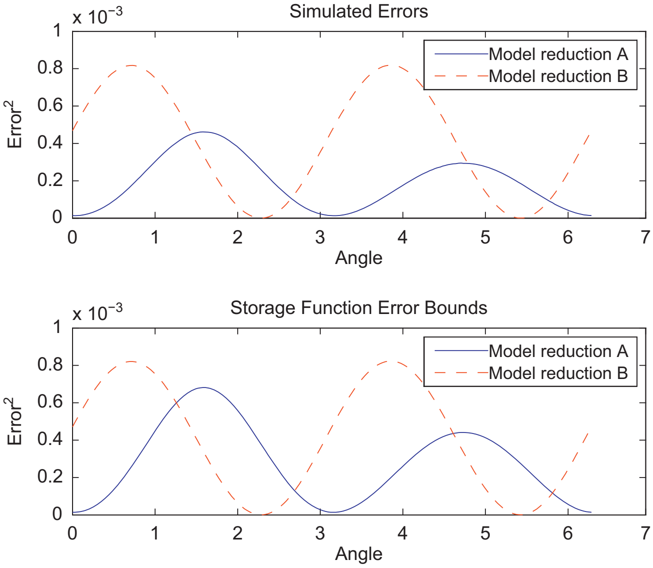

Fig. 4. Error varying along the boundary of a circle around the equilibrium point of the error system. Parameters used here are $k_1=k_2=1$, $k_{-1}=k_{-2}=1.5$, $\epsilon=0.1$ and K=10. The errors are $\|x_2-\tilde{x}_2^2\|^2$ for reduction A and $\|x_2-\bar{x}_2\|^2$ for reduction B. Here, we compare the simulated (top) and bounded (bottom) error between the two reduction schemes.

figure, which shows that model reduction A results in a smaller worst-case error. Note, however, that this is not the case for all initial conditions. The lower panel of each figure shows the outcome of the methods of Section 2 in approximating the reduction error.

Fig. 3 shows that the storage function gives an approximation to the error, for both reduction schemes, which is close enough for us to conclude (without needing to simulate) that model reduction A gives a smaller error. For a different set of parameter values, Fig. 4 shows that the storage function is not as close an approximation to the simulated error of model reduction A. The reason for the variable quality of the bound is that, due to (35) being a rational function, the SOS program requires that

$$\left(-\frac{\mathrm{d}V}{\mathrm{d}t}-z_e^T z_e\right)(4\tilde{x}_2+\sqrt{k_{-1}/k_1})-\sigma(x)(R-(x_2^2+x_3^2+\tilde{x}_2^2)) \text{ is SOS}$$

for error output z_e and SOS multiplier $\sigma(x)$, where the domain is restricted to $|x_e|^2 \le R$. This condition is the equivalent of (23), where we have again multiplied by the denominator of the rational function in the dynamics to ensure a polynomial condition, and restricted our search to a local region around the steady point such that $|x_e|^2 < R$. Integrating this equation, similar to (24), results in a slack term in the upper bound. Making $(4\tilde{x}_2+\sqrt{k_{-1}/k_1})$ large is an easy way to ensure that the slack in the bound caused by the term

$$\int_{t=0}^{\infty}\frac{\sigma(x)(R-(x_2^2+x_3^2+\tilde{x}_2^2))}{4\tilde{x}_2+\sqrt{k_{-1}/k_1}}\,\mathrm{d}t$$

is small. Thus, the storage function becomes a better approximation to the error for smaller values of $\kappa_1=k_1/k_{-1}$, as can be seen by comparing Figs. 3 and 4.

3.3. A kinetic proofreading model

The last example we provide is a simple model of a T-cell signalling pathway. McKeithan's model represents the decision-making process as a network of cascading enzymatic reactions (McKeithan, 1995)—see Fig. 5. As there are now N complexes, this model can be of much higher dimension than the previous two examples. Nevertheless, our method still yields an upper bound on the worst-case error much more efficiently than simulations.

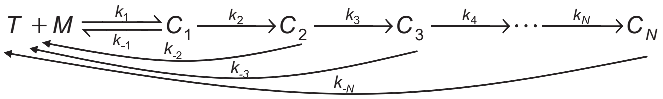

Fig. 5. McKeithan's network for kinetic proofreading.

The differential equations describing the system are

$$\dot{T}=\sum_{i=1}^{N}k_{-i}C_i-k_1TM,$$

$$\dot{M}=\dot{T},$$

$$\dot{C}_1=k_1TM-(k_{-1}+k_2)C_1,$$

$$\dot{C}_i=k_iC_{i-1}-(k_{-i}+k_{i+1})C_i,\quad i=2,\ldots,N-1,$$

$$\dot{C}_N=k_NC_{N-1}-k_{-N}C_N.$$

There are two conservation relations in the above model, so two variables can be removed. Integrating the second equation, we obtain $M-M^*=T-T^*$, where T^* and M^* are the total concentrations of T and M, respectively. Summing up the $\dot{C}_i$ equations we obtain

$$\sum_{i=1}^{N}\dot{C}_i=-\left(\sum_{i=1}^{N}k_{-i}C_i-k_1TM\right)$$

and integrating the first differential equation we obtain

$$-\sum_{i=1}^{N}C_i+C^*=T-T^*=M-M^*,$$

where C^* is a constant. In total, given (T^*,M^*) a simplified, exact model is

$$\dot{C}_1=k_1\left(T^*-\sum_{i=1}^{N}C_i\right)\left(M^*-\sum_{i=1}^{N}C_i\right)-(k_{-1}+k_2)C_1,$$

$$\dot{C}_i=k_iC_{i-1}-(k_{-i}+k_{i+1})C_i,\quad i=2,\ldots,N-1,$$

$$\dot{C}_N=k_NC_{N-1}-k_{-N}C_N.$$

The advantage of using this example is that the size of the network can be varied. The iterative greedy algorithm (Papachristodoulou et al., 2010) described at the end of Section 2.2 was used to show that the order that states need to be removed to minimise the worst-case error is $C_1,C_2,\ldots,C_N$ (where the output signal in this case was the variable C_N). This is to be expected, as the operation of the network depends mainly on the states that are key for proofreading: these are found at the end of the pathway. We note, however, that in this example we cannot back-substitute C_1 when we set $\dot{C}_1=0$; we rather need to keep this as an algebraic constraint when searching for a storage function. Fig. 6 shows how the error increases as more states are removed from a 7-state (i.e., 8-stage) network—the states $C_1,C_2,C_3,\ldots$ are removed sequentially, and the output is the last state, C_7. We have used forward reaction rates which are about 10 times larger than backward ones as it was suggested in McKeithan (1995).

3.3.1. Bound comparison

The kinetic proofreading model has linear dynamics for $N-1$ of its N variables. We can therefore compare the error bounds found using SOSTOOLS to others in the literature.

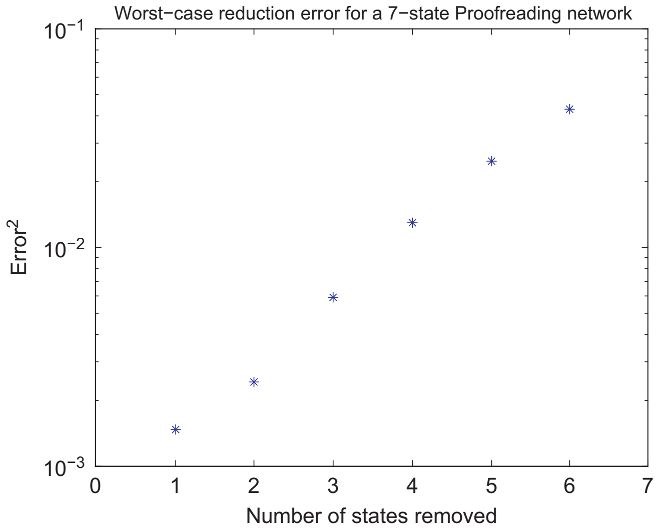

Fig. 6. Worst-case errors for a seven-node network, which is sequentially reduced according to the minimum worst-case error. Parameter values are $T^* = 5$, $M^* = 5.1$, $k_i = 1$ and $k_{-i} = 10$, $i = 1, \ldots, 7$.

We first considered a reduction where the full system with $N=7$ was first linearised, and then reduced by truncating $C_1, \ldots, C_4$. Using the method in this paper, we can determine an upper bound on the worst-case error between this reduction and the full nonlinear system, giving an L_2 norm of 0.6046. The L_2 norm of the worst-case error between the reduction and the linearisation of the full system can also be found, using the methods of Papachristodoulou et al. (2010), to be 0.5076. That is, there is an error introduced when reducing the linearised system, but additional error when we linearise the full system in the first place.

One alternative error bound for this approach is given by Haasdonk and Ohlberger (2011), who use Lipschitz constants to bound the instantaneous error between the linearised and reduced linear system. To search for an approximation to the worst case error, we randomised 100 initial conditions of norm 1 and chose the largest resulting bound at time $T=5$, giving an instantaneous error of $|C_7(5) - \hat{C}_7(5)| \leq 1.876$. The bounds are not directly comparable. The instantaneous error is sometimes the important question asked of a reduction, to which the storage function method is not applicable.

4. Conclusions

When reducing a mathematical model of a system, our aim is to make it more amenable to analysis and simulation. However, for this to have any value we need to guarantee that the reduction has not introduced enough error to invalidate the conclusions of the reduced model. In this paper we have described a method for using Sum of Squares (SOS) techniques to obtain an upper bound on the error introduced by a model reduction. The techniques discussed are a development, for nonlinear polynomial systems, of those of Anderson et al. (2011) for bounding the error resulting from the reduction of linear systems. Biochemical systems are often modelled with polynomial dynamics, and are thus a key application area for the error bounding techniques discussed in this paper. The emphasis on bounding the error of *structured* reductions in the theory of Section 2 reinforces the applicability of these methods to models where we seek to maintain the correspondence of the reduced state space to physical quantities.

We examined three examples of singular perturbation and timescale separation: first, enzyme kinetics and the quasi-steady state assumption (QSSA); second, a simple example from Lee and Othmer (2010) of a dimerisation/isomerisation network; and third, a more complicated network of kinetic proofreading. These examples illustrate the importance of quantifying worst-case error. The first gave a detailed account of how SOSTOOLS can be used to calculate the worst-case error. The second example demonstrated how the method can be used to compare two competing model reduction approaches, while the third showed how our methods can be applied to error quantification for larger, generalised systems which are less amenable to simulation.

Our method works reliably for small to medium-scale systems, but the type of models that urgently need such tools are typically high-dimensional. In the example of Section 3.1, where a two-dimensional system was reduced to one, it took 6.23 s to find a storage function and 0.3 s to maximise it. The reductions consideredn u in Section 3.2 took 1.38 s and 0.63 s to find storage functions for. The two full models are the same dimension, but the variation in times is due to the storage function in Section 3.1 being sixth order, while both in Section 3.2 are fourth order, with Reduction A leading to rational dynamics requiring an additional multiplier. In the final example, the time taken to estimate the error from reducing from 7 states ranges between 12.9 s (reducing to 2 states) and over 3 min (reducing to 6 states) even though we only use a second order storage function.

One way of allowing these methods to scale better is to take advantage of (or impose) a modular approach to model reduction. In particular, one can ‘break them down’ into a network of smaller sub-systems, where the outputs of each form the inputs of their neighbours. A “naïve” structured reduction approach would be to define modules in an ad hoc way, reduce each module as an independent system, and then reconnect the modules together: however, the aggregate error can be amplified significantly this way (Sandberg and Murray, 2009). A better approach would be to develop a method of identifying how to partition the system into sub-systems, each of which is reduced, in order that the error introduced in the full system is as small as possible.

This idea builds on the fact that several biological systems are (to some extent) modular: their behaviour resembles that of a collection of sub-systems with specific functions which are connected to one another in feed-forward and feedback loops. For instance, El-Samad et al. (2005) illustrate the hierarchical, modular viewpoint of the cell with a particular example: the heat shock response is one of many functional modules within a cell, but this individual module can itself be broken down into a collection of “submodules with … predefined functionalities”, such as a “temperature sensor”, “actuator” and so on. Another example of modular decomposition is given by Conzelmann et al. (2004), where the EGF receptor signalling pathway – a high-dimensional system, consisting of more than 30 states – is reduced to a number of interconnected modules, each of which has its dynamics approximated by a model reduction.

The above concept can be used to address another interesting topic: cell-based models which can also be scaled up as far as the tissue level, where individual cells are interconnected with one another based on physiological properties, such as signalling or the propagation of action potential. Simulations of multi-scale systems usually use reduced-order versions of systems at lower organisational scales when simulating higher organisational scales. If the system in which we are interested can be modelled as a hierarchical interconnection of modules (i.e., cells), then reducing models at the smallest scales will introduce error that will be distributed across the resolution levels of the multi-scale model. The challenge is to understand how errors propagate through layers and how they can be calculated. For example, Conzelmann et al. (2004) show that the quality of their reductions, once connected together, are vastly affected by the degree of *retroactivity*, or feedback, between the modules. Even a very small amount of feedback appears to cause a great deal of error. As another example, Ciliberto et al. (2007) discuss the interconnection of a network of Goldbeter–Koshland switches, each of which is a pair of enzymatic reactions, and conclude that the tQSSA is a better model reduction than the sQSSA in terms of how well the dynamics of the full system are reproduced: the error of the individual reductions propagates less.

In conclusion, there is significant potential to apply the nonlinear theory in Section 2 to networks of interacting systems, which would allow a quantitative bound on the errors introduced by the simplifications of modular and multi-scale models.

Appendix A. Error bound statement and proof

Proposition 2. *Using the notation of Section 2.1, suppose that there exists a function $V:\mathcal{D}\times\hat{\mathcal{D}}\to\mathbb{R}$ which satisfies the three conditions* (8)–(10):

$$V(0)=0, \tag{A.1}$$

$$V(x_e)>0 \quad \text{on } x_e\in\mathcal{D}\times\hat{\mathcal{D}}\backslash\{0\}, \tag{A.2}$$

$$\frac{\mathrm{d}V}{\mathrm{d}t}+z_e^T(t)z_e(t)\le 0 \quad \text{on } x_e\in\mathcal{D}\times\hat{\mathcal{D}}. \tag{A.3}$$

Then, if the error system given by (7) *is released from any given initial condition $x_e(0)$, the norm of the observed error output z_e can be bounded above such that*

$$V(x_e(0))\ge\|z_e\|^2. \tag{A.4}$$

Thus, if the error output norm $\|z_e\|$ is treated as a function of the initial conditions of the system, the function V is an upper bound.

Proof. Consider the system given by (7) running with initial condition $x_e(0)$. Condition (A.3) can be re-written as

$$-\frac{\mathrm{d}V}{\mathrm{d}t}\ge z_e^Tz_e$$

and then integrated between 0 and any finite time τ to give

$$V(x_e(0))-V(x_e(\tau))\ge\int_0^\tau z_e^Tz_e\,\mathrm{d}t.$$

Conditions (A.1) and (A.2) mean that $V(x_e(\tau))\ge 0$, so the inequality above can be loosened to

$$V(x_e(0))\ge\int_0^\tau z_e^Tz_e\,\mathrm{d}t.$$

The left-hand side of this equation is independent of τ, and hence this inequality also holds in the limit as $\tau\to\infty$. The definition of $\|z_e\|$ such that

$$\|z_e\|^2=\int_0^\infty z_e^Tz_e\,\mathrm{d}t$$

completes the proof. □

Appendix B. Proof of Proposition 1

Proposition 1. *The maximum of any continuous non-negative function V_e on a ball of radius r is equal to*

$$\gamma=\inf\{a\ge 0|P_a=\emptyset\},$$

where P_a for a particular $a\ge 0$ is given as

$$P_a=\{x_0\in\mathbb{R}^n||x_0|\le r \text{ and } V_e(x_0)\ge a\}.$$

Proof. The set $P_a=\emptyset$ for some $a\ge 0$ if and only if

$$|x|\le r\Rightarrow V(x)<a \tag{B.1}$$

is true.

We want to show that

$$\sup\{V(x)||x|\le r\}=\gamma.$$

For any a such that $P_a=\emptyset$, by (B.1) a is an upper bound on $X:=\{V(x)||x|\le r\}$ and hence $a\ge\sup X$. Thus, the supremum of X is a lower bound for the set $Y:=\{a\ge 0|P_a=\emptyset\}$, and hence

$$\sup X\le\inf Y=\gamma.$$

To show equality, if $\sup X=\gamma-2\epsilon$ for some $\epsilon>0$, then $\gamma-\epsilon>\gamma-2\epsilon\ge V(x)$ for all x such that $|x|\le r$. Hence $\gamma-\epsilon\in Y$, contradicting the definition of $\gamma=\inf Y$. Thus, $\gamma=\sup X$. □

Appendix C. Implementation of storage function method in SOSTOOLS

An implementation of our method for the first example is online, and can be found at http://sysos.eng.ox.ac.uk/JTBMMSOSTOOLS.zip

References

Anderson, J., Chang, Y.C., Papachristodoulou, A., 2011. Model decomposition and reduction tools for large-scale networks in systems biology. Automatica 47 (6), 1165–1174.

Antoulas, A.C., 2005. Approximation of Large-scale Dynamical Systems. Society for Industrial Mathematics.

Besselink, B., van de Wouw, N., Nijmeijer, H., 2009. An error bound for model reduction of Lur'e-type systems. In: Proceedings of the 48th IEEE Conference on Decision and Control (CDC). IEEE, pp. 3264–3269.

Bochnak, J., Coste, M., Roy, M.F., 1998. Real Algebraic Geometry. Springer-Verlag.

Borghans, J.A.M., de Boer, R.J., Segel, L.A., 1996. Extending the quasi-steady state approximation by changing variables. Bull. Math. Biol. 58 (1), 43–63.

Chen, W.W., Schoeberl, B., Jasper, P.J., Niepel, M., Nielsen, U.B., Lauffenburger, D.A., Sorger, P.K., 2009. Input–output behavior of ErbB signaling pathways as revealed by a mass action model trained against dynamic data. Mol. Syst. Biol. 5, 239.

Chesi, G., 2010. LMI techniques for optimization over polynomials in control: a survey. IEEE Trans. Autom. Control 55 (11), 2500–2510.

Ciliberto, A., Capuani, F., Tyson, J.J., 2007. Modeling networks of coupled enzymatic reactions using the total quasi-steady state approximation. PLoS Comput. Biol. 3 (3), e45.

Conzelmann, H., Saez-Rodriguez, J., Sauter, T., Bullinger, E., Allgöwer, F., Gilles, E.D., 2004. Reduction of mathematical models of signal transduction networks: simulation-based approach applied to EGF receptor signalling. Syst. Biol. 1 (1), 159–169.

Dullerud, G.E., Paganini, F.G., 2000. A Course in Robust Control Theory: A Convex Approach. Springer-Verlag.

El-Samad, H., Kurata, H., Doyle, J.C., Gross, C.A., Khammash, M., 2005. Surviving heat shock: control strategies for robustness and performance. Proc. Nat. Acad. Sci. 102 (8), 2736–2741.

Haasdonk, B., Ohlberger, M., 2011. Efficient reduced models and *a posteriori* error estimation for parametrized dynamical systems by offline/online decomposition. Math. Comput. Modell. Dyn. Syst. 17 (2), 145–161.

Haddad, W.M., Chellaboina, V.S., 2008. Nonlinear Dynamical Systems and Control: A Lyapunov-based Approach. Princeton University Press.

Huang, H., Fairweather, M., Griffiths, J.F., Tomlin, A.S., Brad, R.B., 2005. A systematic lumping approach for the reduction of comprehensive kinetic models. Proc. Combust. Inst. 30 (1), 1309–1316.

Lee, C.H., Othmer, H.G., 2010. A multi-time-scale analysis of chemical reaction networks: I. Deterministic systems. J. Math. Biol. 60 (3), 387–450.

Löhning, M., Hasenauer, J., Algöwer, F., 2011. Trajectory-based model reduction of nonlinear biochemical networks employing the observability normal form. In: Proceedings of the 18th IFAC World Congress. IFAC, pp. 10442–10447.

Matthies, G.H., Meyer, M., 2003. Nonlinear Galerkin methods for the model reduction of nonlinear dynamical systems. Comput. Struct. 81 (12), 1277–1286.

McKeithan, T.W., 1995. Kinetic proofreading in T-cell receptor signal transduction. Proc. Nat. Acad. Sci. 92 (11), 5042–5046.

Murray, J.D., 2002. Mathematical Biology: An Introduction. Springer.

Okino, M.S., Mavrovouniotis, M.L., 1998. Simplification of mathematical models of chemical reaction systems. Chem. Rev. 98 (2), 391–408.

Papachristodoulou, A., Prajna, S., 2002. On the construction of Lyapunov functions using the Sum of Squares decomposition. In: Proceedings of the 41st IEEE Conference on Decision and Control (CDC). IEEE, pp. 3482–3487.

Papachristodoulou, A., Chang, Y.C., August, E., Anderson, J., 2010. Structured model reduction for dynamical networked systems. In: Proceedings of the 49th IEEE Conference on Decision and Control (CDC). IEEE, pp. 2670–2675.

Parrilo, P.A., 2000. Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization. Ph.D. Thesis, Caltech, Pasadena, CA, 2000.

Powers, V., Wörmann, T., 1998. An algorithm for sums of squares of real polynomials. J. Pure Appl. Algebra 127 (1), 99–104.

Prajna, S., Sandberg, H., 2005. On model reduction of polynomial dynamical systems. In: Proceedings of the 44th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC). IEEE, pp. 1666–1671.

Prajna, S., Papachristodoulou, A., Seiler, P., Parrilo, P.A., 2004. SOSTOOLS: Sum of Squares optimization toolbox for MATLAB. Available from 〈http://www.cds.caltech.edu/sostools〉 and 〈http://www.mit.edu/~parrilo/sostools〉.

Sandberg, H., Murray, R.M., 2009. Model reduction of interconnected linear systems. Optimal Control Appl. Methods 30 (3), 225–245.

Scherpen, J.M.A., 1993. Balancing for nonlinear systems. Syst. Control Lett. 21 (2), 143–153.

Sunnåker, M., Cedersund, G., Jirstrand, M., 2011. A method for zooming of nonlinear models of biochemical systems. BMC Syst. Biol. 5, 140.

Tzafriri, A.R., 2003. Michaelis–Menten kinetics at high enzyme concentrations. Bull. Math. Biol. 65 (6), 1111–1129.

Wirtz, D., Haasdonk, B., 2012. Efficient a-posteriori error estimation for nonlinear kernel-based reduced systems. Syst. Control Lett. 61 (1), 203–211.

Zagaris, A., Kaper, H.G., Kaper, T.J., 2004. Analysis of the computational singular perturbation reduction method for chemical kinetics. J. Nonlinear Sci. 14 (1), 59–91.

Zhou, K., Doyle, J.C., Glover, K., 1996. Robust and Optimal Control. Prentice Hall, Upper Saddle River, NJ.