Symbolic Versus Numerical Computation and Visualization of Parameter Regions for Multistationarity of Biological Networks

Matthew England[1], Hassan Errami[2], Dima Grigoriev[3], Ovidiu Radulescu[4], Thomas Sturm[5,6], and Andreas Weber[2]

[1] Faculty of Engineering, Environment and Computing, Coventry University, UK;
Matthew.England@coventry.ac.uk
[2] Institut für Informatik II, Universität Bonn, Germany;
{errami,weber}@cs.uni-bonn.de
[3] CNRS, Mathématiques, Université de Lille, Villeneuve d'Ascq, France;
Dmitry.Grigoryev@math.univ-lille1.fr
[4] DIMNP UMR CNRS/UM 5235, University of Montpellier, France;
ovidiu.radulescu@umontpellier.fr
[5] University of Lorraine, CNRS, Inria, and LORIA, Nancy, France;
thomas.sturm@loria.fr
[6] MPI Informatics and Saarland University, Saarbrücken, Germany;
sturm@mpi-inf.mpg.de

Abstract. We investigate models of the mitogenactivated protein kinases (MAPK) network, with the aim of determining where in parameter space there exist multiple positive steady states. We build on recent progress which combines various symbolic computation methods for mixed systems of equalities and inequalities. We demonstrate that those techniques benefit tremendously from a newly implemented graph theoretical symbolic preprocessing method. We compare computation times and quality of results of numerical continuation methods with our symbolic approach before and after the application of our preprocessing.

1 Introduction

The mathematical modelling of intra-cellular biological processes has been using nonlinear ordinary differential equations since the early ages of mathematical biophysics in the 1940s and 50s [25]. A standard modelling choice for cellular circuitry is to use chemical reactions with mass action law kinetics, leading to polynomial differential equations. Rational functions kinetics (for instance the Michaelis-Menten kinetics) can generally be decomposed into several mass action steps. An important property of biological systems is their multistationarity which means having multiple stable steady states. Multistationarity is instrumental to cellular memory and cell differentiation during development or regeneration of multicellular organisms and is also used by micro-organisms in survival strategies. It is thus important to determine the parameter values for which a biochemical

model is multistationary. For mass action reactions, testing multistationarity boils down to counting real positive solutions of algebraic systems.

The models benchmarked in this paper concern intracellular signaling pathways. These pathways transmit information about the cell environment by inducing cascades of protein modifications (phosphorylation) all the way from the plasma membrane via the cytosol to genes in the cell nucleus. Multistationarity of signaling usually occurs as a result of activation of upstream signaling proteins by downstream components [2]. A different mechanism for producing multistationarity in signaling pathways was proposed by Kholodenko [24]. In this mechanism the cause of multistationarity are multiple phosphorylation/ dephosphorylation cycles that share enzymes. A simple, two steps phosphorylation/dephosphorylation cycle is capable of ultrasensitivity, a form of all or nothing response with no multiple steady states (Goldbeter–Koshland mechanism). In multiple phosphorylation/dephosphorylation cycles, enzyme sharing provides competitive interactions and positive feedback that ultimately leads to multistationarity [24, 21].

Our study is complementary to works applying numerical methods to ordinary differential equations models used for biology applications. Gross et al. [17] used polynomial homotopy continuation methods for global parameter estimation of mass action models. Bifurcations and multistationarity of signaling cascades was studied with numerical methods based on the Jacobian matrix [27].

Our work here follows [5], where it was demonstrated that determination of multistationarity of an 11-dimensional model of a mitogen-activated protein kinases (MAPK) cascade can be achieved by currently available symbolic methods when numeric values are known for all but potentially one parameter. We show that the symbolic methods used in [5], viz. real triangularization and cylindrical algebraic decomposition, and also polynomial homotopy continuation methods, benefit tremendously from a graph theoretical symbolic preprocessing method. This method has been sketched by Grigoriev et al. [16] and has been used for a "hand computation," but had not been implemented before. For our experiments we use the model already investigated in [5] and a higher dimensional model of the MAPK cascade.

2 The Systems for the Case Studies

For our investigations we use models of the MAPK cascade that can be found in the Biomodels database[7] as numbers 26 and 28 [22]. We refer to those models as Biomod-26 and Biomod-28, respectively.

2.1 Biomod-26

Biomod-26, which we have studied also in [5], is given by the following set of differential equations. We have renamed the species names as $x_1, \ldots, x_{11}$ and the

---

[7] http://www.ebi.ac.uk/biomodels-main/

rate constants as $k_1, \ldots, k_{16}$ to facilitate reading:

$$\dot{x}_1 = k_2 x_6 + k_{15} x_{11} - k_1 x_1 x_4 - k_{16} x_1 x_5$$
$$\dot{x}_2 = k_3 x_6 + k_5 x_7 + k_{10} x_9 + k_{13} x_{10} - x_2 x_5 (k_{11} + k_{12}) - k_4 x_2 x_4$$
$$\dot{x}_3 = k_6 x_7 + k_8 x_8 - k_7 x_3 x_5$$
$$\dot{x}_4 = x_6 (k_2 + k_3) + x_7 (k_5 + k_6) - k_1 x_1 x_4 - k_4 x_2 x_4$$
$$\dot{x}_5 = k_8 x_8 + k_{10} x_9 + k_{13} x_{10} + k_{15} x_{11} - x_2 x_5 (k_{11} + k_{12}) - k_7 x_3 x_5 - k_{16} x_1 x_5$$
$$\dot{x}_6 = k_1 x_1 x_4 - x_6 (k_2 + k_3)$$
$$\dot{x}_7 = k_4 x_2 x_4 - x_7 (k_5 + k_6)$$
$$\dot{x}_8 = k_7 x_3 x_5 - x_8 (k_8 + k_9)$$
$$\dot{x}_9 = k_9 x_8 - k_{10} x_9 + k_{11} x_2 x_5$$
$$\dot{x}_{10} = k_{12} x_2 x_5 - x_{10} (k_{13} + k_{14})$$
$$\dot{x}_{11} = k_{14} x_{10} - k_{15} x_{11} + k_{16} x_1 x_5 \tag{1}$$

The Biomodels database also gives us meaningful values for the rate constants, which we generally substitute into the corresponding systems for our purposes here:

$$
\begin{array}{llll}
k_1 = 0.02, & k_2 = 1, & k_3 = 0.01, & k_4 = 0.032, \\
k_5 = 1, & k_6 = 15, & k_7 = 0.045, & k_8 = 1, \\
k_9 = 0.092, & k_{10} = 1, & k_{11} = 0.01, & k_{12} = 0.01, \\
k_{13} = 1, & k_{14} = 0.5, & k_{15} = 0.086, & k_{16} = 0.0011. \tag{2}
\end{array}
$$

Using the left-null space of the stoichiometric matrix under positive conditions as conservation constraint [12] we obtain three linear conservation constraints:

$$x_5 + x_8 + x_9 + x_{10} + x_{11} = k_{17},$$
$$x_4 + x_6 + x_7 = k_{18},$$
$$x_1 + x_2 + x_3 + x_6 + x_7 + x_8 + x_9 + x_{10} + x_{11} = k_{19}, \tag{3}$$

where k_{17}, k_{18}, k_{19} are new constants computed from the initial data. Those constants are the parameters that we are interested in here.

The steady state problem for the MAPK cascade can now be formulated as a real algebraic problem as follows. We replace the left hand sides of all equations in (1) with 0 and substitute the values from (2). This together with (3) yields a system of parametric polynomial equations with polynomials in $\mathbb{Z}[k_{17}, k_{18}, k_{19}][x_1, \ldots, x_{11}]$. Since all entities in our model are strictly positive, we add to our system positivity conditions $k_{17} > 0$, $k_{18} > 0$, $k_{19} > 0$ and $x_1 > 0$, $\ldots$, $x_{11} > 0$. In terms of first-order logic the conjunction over our equations and inequalities yields a quantifier-free Tarski formula.

2.2 Biomod-28

The system with number 28 in the Biomodels database is given by the following set of differential equations. Again, we have renamed the species names into $x_1, \ldots, x_{16}$ and the rate constants into $k_1, \ldots, k_{27}$ to facilitate reading:

$$\dot{x}_1 = k_2 x_9 + k_8 x_{10} + k_{21} x_{15} + k_{26} x_{16} - k_1 x_1 x_5 - k_7 x_1 x_5 - k_{22} x_1 x_6 - k_{27} x_1 x_6$$

$$\dot{x}_2 = k_3 x_9 + k_5 x_7 + k_{24} x_{12} - k_4 x_2 x_5 - k_{23} x_2 x_6$$

$$\dot{x}_3 = k_9 x_{10} + k_{11} x_8 + k_{16} x_{13} + k_{19} x_{14} - k_{10} x_3 x_5 - k_{17} x_3 x_6 - k_{18} x_3 x_6$$

$$\dot{x}_4 = k_6 x_7 + k_{12} x_8 + k_{14} x_{11} - k_{13} x_4 x_6$$

$$\dot{x}_5 = k_2 x_9 + k_3 x_9 + k_5 x_7 + k_6 x_7 + k_8 x_{10} + k_9 x_{10} + k_{11} x_8 + k_{12} x_8 - $$
$$k_1 x_1 x_5 - k_4 x_2 x_5 - k_7 x_1 x_5 - k_{10} x_3 x_5$$

$$\dot{x}_6 = k_{14} x_{11} + k_{16} x_{13} + k_{19} x_{14} + k_{21} x_{15} + k_{24} x_{12} + k_{26} x_{16} - $$
$$k_{13} x_4 x_6 - k_{17} x_3 x_6 - k_{18} x_3 x_6 - k_{22} x_1 x_6 - k_{23} x_2 x_6 - k_{27} x_1 x_6$$

$$\dot{x}_7 = k_4 x_2 x_5 - k_6 x_7 - k_5 x_7$$

$$\dot{x}_8 = k_{10} x_3 x_5 - k_{12} x_8 - k_{11} x_8$$

$$\dot{x}_9 = k_1 x_1 x_5 - k_3 x_9 - k_2 x_9$$

$$\dot{x}_{10} = k_7 x_1 x_5 - k_9 x_{10} - k_8 x_{10}$$

$$\dot{x}_{11} = k_{13} x_4 x_6 - k_{15} x_{11} - k_{14} x_{11}$$

$$\dot{x}_{12} = k_{23} x_2 x_6 - k_{25} x_{12} - k_{24} x_{12}$$

$$\dot{x}_{13} = k_{15} x_{11} - k_{16} x_{13} + k_{17} x_3 x_6$$

$$\dot{x}_{14} = k_{18} x_3 x_6 - k_{20} x_{14} - k_{19} x_{14}$$

$$\dot{x}_{15} = k_{20} x_{14} - k_{21} x_{15} + k_{22} x_1 x_6$$

$$\dot{x}_{16} = k_{25} x_{12} - k_{26} x_{16} + k_{27} x_1 x_6$$

The estimates of the rate constants given in the Biomodels database are:

$k_1 = 0.005,$	$k_2 = 1,$	$k_3 = 1.08,$	$k_4 = 0.025,$
$k_5 = 1,$	$k_6 = 0.007,$	$k_7 = 0.05,$	$k_8 = 1,$
$k_9 = 0.008,$	$k_{10} = 0.005,$	$k_{11} = 1,$	$k_{12} = 0.45,$
$k_{13} = 0.045,$	$k_{14} = 1,$	$k_{15} = 0.092,$	$k_{16} = 1,$
$k_{17} = 0.01,$	$k_{18} = 0.01,$	$k_{19} = 1,$	$k_{20} = 0.5,$
$k_{21} = 0.086,$	$k_{22} = 0.0011,$	$k_{23} = 0.01,$	$k_{24} = 1,$
$k_{25} = 0.47,$	$k_{26} = 0.14,$	$k_{27} = 0.0018.$	

Again, using the left-null space of the stoichiometric matrix under positive conditions as a conservation constraint [12] we obtain the following:

$$x_6 + x_{11} + x_{12} + x_{13} + x_{14} + x_{15} + x_{16} = k_{28},$$
$$x_5 + x_7 + x_8 + x_9 + x_{10} = k_{29},$$
$$x_1 + x_2 + x_3 + x_4 + x_7 + x_8 + x_9 + x_{10} + x_{11} + $$
$$x_{12} + x_{13} + x_{14} + x_{15} + x_{16} = k_{30},$$

where k_{28}, k_{29}, k_{30} are new constants computed from the initial data. We formulate the real algebraic problem as described at the end of Sect. 2.1. In particular, note that we need positivity conditions for all variables and parameters.

3 Graph-Theoretical Symbolic Preprocessing

The complexity, primarily in terms of dimension, of polynomial systems obtained with steady-state approximations of biological models plus conservation laws is comparatively high for the application of symbolic methods. It this therefore highly relevant for the success of such methods to identify and exploit particular structural properties of the input. Our models have remarkably low total degrees with many linear monomials after some substitutions for rate constants. This suggests to preprocess with essentially Gaussian elimination in the sense of solving single suitable equations with respect to some variable and substituting the corresponding solution into the system.

Generalizing this idea to situations where linear variables have parametric coefficients in the other variables requires, in general, a parametric variant of Gaussian elimination, which replaces the input system with a finite case distinction with respect to the vanishing of certain coefficients and one reduced system for each case. With Biomod-26 and Biomod-28 considered here it turns out that the positivity assumptions on the variables are strong enough to effectively guarantee the non-vanishing of all relevant coefficients so that case distinctions are never necessary. On the other hand, those positivity conditions establish an apparent obstacle, because we are formally not dealing with a parametric system of linear equations but with a parametric linear programming problem. However, here the theory of real quantifier elimination tells us that it is sufficient that the inequality constraints play a passive role. Those constraints must be considered when substituting Gauss solutions from the equations, but otherwise can be ignored [23, 20].

Parametric Gaussian elimination can increase the degrees of variables in the parametric coefficient, in particular destroying their linearity and suitability to be used for further reductions. As an example consider the steady-state approximation, i.e., all left hand sides replaced with 0, of the system in (1), solving the last equation for x_5, and substituting into the first equation. The natural question for an optimal strategy to Gauss-eliminate a maximal number of variables has been answered positively only recently [16]: draw a graph, where vertices are variables and edges indicate multiplication between variables within some monomial. Then one can Gauss-eliminate a *maximum independent set*, which is the complement of a *minimum vertex cover*. Fig. 1 shows that graph for Biomod-26, where $\{x_4, x_5\}$ is a minimal vertex cover, and all variables can be linearly eliminated. Similarly, for Biomod-28 we find $\{x_5, x_6\}$ as a minimum vertex cover. Recall that minimum vertex cover is one of Karp's 21 classical NP complete problems [19]. However, our instances considered here and instances to be expected from other biological models are so small that the use of existing approximation algorithms [15] appears unnecessary. We have used real quantifier elimination, which did not consume measurable CPU time; alternatively one could use integer linear programming or SAT-solving.

It is a most remarkable fact that a significant number of biological models in the databases have that property of loosely connected variables. This phenomenon resembles the well-known *community structure* of propositional satisfiability
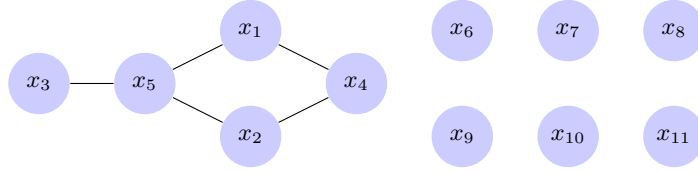
Fig. 1. The graph for Biomod-26 is loosely connected. Its minimum vertex cover $\{x_4, x_5\}$ is small. All other variables form a maximum independent set, which can be eliminated with linear methods.

problems, which has been identified as one of the key structural reasons for the impressive success of state-of-the-art CDCL-based SAT solvers [14].

We conclude this section with the reduced systems as computed with our implementation in Redlog [9]. For Biomod-26 we obtain $x_5 > 0$, $x_4 > 0$, $k_{19} > 0$, $k_{18} > 0$, $k_{17} > 0$ and

$$
\begin{aligned}
& 1062444k_{18}x_4^2x_5 + 23478000k_{18}x_4^2 + 1153450k_{18}x_4x_5^2 + 2967000k_{18}x_4x_5 \\
& \quad + 638825k_{18}x_5^3 + 49944500k_{18}x_5^2 - 5934k_{19}x_4^2x_5 - 989000k_{19}x_4x_5^2 \\
& \quad - 1062444x_4^3x_5 - 23478000x_4^3 - 1153450x_4^2x_5^2 - 2967000x_4^2x_5 \\
& \hspace{6cm} - 638825x_4x_5^3 - 49944500x_4x_5^2 = 0, \\
& 1062444k_{17}x_4^2x_5 + 23478000k_{17}x_4^2 + 1153450k_{17}x_4x_5^2 + 2967000k_{17}x_4x_5 \\
& \quad + 638825k_{17}x_5^3 + 49944500k_{17}x_5^2 - 1056510k_{19}x_4^2x_5 - 164450k_{19}x_4x_5^2 \\
& \quad - 638825k_{19}x_5^3 - 1062444x_4^2x_5^2 - 23478000x_4^2x_5 - 1153450x_4x_5^3 \\
& \hspace{5cm} - 2967000x_4x_5^2 - 638825x_5^4 - 49944500x_5^3 = 0.
\end{aligned}
$$

For Biomod-28 we obtain $x_6 > 0$, $x_5 > 0$, $k_{30} > 0$, $k_{29} > 0$, $k_{28} > 0$ and

$$
\begin{aligned}
& 3796549898085k_{29}x_5^3x_6 + 71063292573000k_{29}x_5^3 + 106615407090630k_{29}x_5^2x_6^2 \\
& \quad + 479383905861000k_{29}x_5^2x_6 + 299076127852260k_{29}x_5x_6^3 \\
& \quad + 3505609439955600k_{29}x_5x_6^2 + 91244417457024k_{29}x_6^4 \\
& \quad + 3557586742819200k_{29}x_6^3 - 598701732300k_{30}x_5^3x_6 \\
& \quad - 83232870778950k_{30}x_5^2x_6^2 - 185019487578700k_{30}x_5x_6^3 \\
& \quad - 3796549898085x_5^4x_6 - 71063292573000x_5^4 - 106615407090630x_5^3x_6^2 \\
& \quad - 479383905861000x_5^3x_6 - 299076127852260x_5^2x_6^3 - 3505609439955600x_5^2x_6^2 \\
& \hspace{4cm} - 91244417457024x_5x_6^4 - 3557586742819200x_5x_6^3 = 0, \\
& 3796549898085k_{28}x_5^3x_6 + 71063292573000k_{28}x_5^3 + 106615407090630k_{28}x_5^2x_6^2 \\
& \quad + 479383905861000k_{28}x_5^2x_6 + 299076127852260k_{28}x_5x_6^3 \\
& \quad + 3505609439955600k_{28}x_5x_6^2 + 91244417457024k_{28}x_6^4 \\
& \quad + 3557586742819200k_{28}x_6^3 - 3197848165785k_{30}x_5^3x_6 \\
& \quad - 23382536311680k_{30}x_5^2x_6^2 - 114056640273560k_{30}x_5x_6^3 \\
& \quad - 91244417457024k_{30}x_6^4 - 3796549898085x_5^3x_6^2 - 71063292573000x_5^3x_6 \\
& \quad - 106615407090630x_5^2x_6^3 - 479383905861000x_5^2x_6^2 - 299076127852260x_5x_6^4 \\
& \hspace{3.5cm} - 3505609439955600x_5x_6^3 - 91244417457024x_6^5 - 3557586742819200x_6^4 = 0.
\end{aligned}
$$

Notice that no complex positivity constraints come into existence with these examples. All corresponding substitution results are entailed by the other constraints, which is implicitly discovered by using the standard simplifier from [10] during the elimination.

4 Determination of Multiple Steady States

We aim to identify via grid sampling regions of parameter space where bistability occurs. Our focus is on the parameters introduced with the conservation laws. We consider two different approaches: first a polynomial homotopy continuation method implemented in Bertini, and second a combination of symbolic computation methods implemented in Maple. We compare the approaches with respect to performance and quality of results for both the reduced and the unreduced systems.

4.1 Numerical Approach

We use the homotopy solver Bertini [1] in its standard configuration to compute complex roots. We parse the output of Bertini using Python, and determined numerically, which of the complex roots are real and positive using a threshold of 10^{-6} for positivity. Computations are done in Python with Bertini embedded.

For System Biomod-26 we produced the two plots in Fig. 2 using the original system and the two in Fig. 3 using the reduced system. The sampling range for k_{19} was from 200 to 1000 by 50. In the left plots the sampling range for k_{17} is from 80 to 200 by 10 with k_{18} fixed at 50. In the right plots the sampling range for k_{18} is 5 to 75 by 5 with k_{17} fixed to 100. The colours indicate the computed number of positive real fixed points: blue for 3, green for 2, yellow for 1 and red for 0. We analyse these further in the comparison in Sect. 4.3.

For Biomod-28 we produced the two plots in Fig. 5 using the original system. The sampling range for k_{30} was from 100 to 1600 by 100. In the left plots the sampling range for k_{28} is from 40 to 160 by 10 with k_{29} fixed at 180. In the right plots the sampling range for k_{29} is from 120 to 240 by 10 with k_{28} fixed to 100. The colours indicate the number of fixed points as before. In this case Bertini found no fixed points for the reduced system. For the reduced system Bertini (wrongly) could not find any roots—not even complex ones—for any of the parameter settings. The situation did not change when going from adaptive precision to a very high fixed precision. However, we have not attempted more sophisticated techniques like providing user homotopies. We analyse these results further in the comparison in Sect. 4.3.

4.2 Symbolic Approach

Our next approach will still use grid sampling, but each sample point will undergo a symbolic computation. The result will still be an approximate identification of the region (since the sampling will be finite) but the results at those sample

points will be guaranteed free of numerical errors. The computations follow the strategy introduced in [5, Section 2.1.2]. This combined tools from the Regular Chains Library[8] available for use in Maple. Regular chains are the triangular decompositions of systems of polynomial equations (triangular in terms of the variables in each polynomial). Highly efficient methods for working in complex space have been developed based on these (see [26] for a survey).

We make use of recent work by Chen et al. [6] which adapts these tools to the real analogue: semi-algebraic systems. They describe algorithms to decompose any real polynomial system into finitely many regular semi-algebraic systems: both directly and by computation of components by dimension. The latter (the so called *lazy* variant) was key to solving the 1-parameter MAPK problem in [5]. However, for the zero dimensional computations of this paper there is only one solution component and so no savings from lazy computations.

For a given system and sample point we apply the real triangularization (RT) on the quantifier-free formula (as described at the end of Sect. 2.1: a quantifier free conjunction of equities and inequalities) evaluated with the parameter estimates and sample point values. This produces a simplified system in several senses. First, as guaranteed by the algorithm, the output is triangular according to a variable ordering. So there is a univariate component, then a bivariate component introducing one more variable and so on. Secondly, for all the MAPK models we have studied so far, all but the final (univariate) of these equations has been linear in its main variable. This thus allows for easy back substitution. Thirdly, most of the positivity conditions are implied by the output rather than being an explicit part of it, in which case a simpler sub-system can be solved and back substitution performed instantly.

Biomod-26 For the original version of Biomod-26 the output of RT was a component consisting of 11 equations and a single inequality. The equations were in ascending main variable according to the provided ordering (same as the labelling). All but the final equation is linear in its main variable, with the final equation being univariate and degree 6 in x_1. The output of the triangularization requires that this variable be positive, $x_1 > 0$, with the positivity of the other variables implied by solutions to the system. So to proceed we must find the positive real roots of the degree 8 univariate polynomial in x_1: counting these will imply the number of real positive solutions of the parent system. We do this using the root isolation tools in the Regular Chains Library. This whole process was performed iteratively for the same sampling regime as Bertini used to produce Fig. 4.

We repeated the process on the reduced version of the system. The triangularization again reduced the problem to univariate real root isolation, this time with only one back substitution step needed. As to be expected from a fully symbolic computation, the output is identical and so again represented by Fig. 4. However, the computation was significantly quicker with this reduced system. More details are given in the comparison in Sect. 4.3.

---

[8] www.regularchains.org

Biomod-28 The same process was conducted on Biomod-28. As with Biomod-26 the system was triangular with all but the final equation linear in its main variable; this time the final equation is degree 8. However, unlike Biomod-26 two positivity conditions were returned in the output meaning we must solve a bivariate problem before we can back substitute to the full system. Rather than just perform univariate real root isolation we must build a Cylindrical Algebraic Decomposition (CAD) (see, e.g., [4] and the references within) sign invariant for the final two equations and interrogate its cells to find those where the equations are satisfied and variable positive. Counting these we find always 1 or 3 cells, with the latter indicating bistability. This is similar to the approach used in [5], although in that case the 2D CAD was for one variable and one parameter. We used the implementation of CAD in the Regular Chains Library [7, 3] with the results producing the plots in Fig. 6.

For the reduced system we proceeded similarly. A 2D CAD still needed to be produced after triangularization and so in this case there was no reduction in the number of equations to study with CAD via back substiution. However, it was still beneficial to pre-process CAD with real triangularization: the average time per sample point with pre-processing (and including time taken to pre-process) was 0.485 seconds while without it was 3.577 seconds. Further details are given in the comparison in Sect 4.3.

4.3   Comparison

Figure 2, Fig. 3, and Fig. 4 all refer to Biomod-26. The latter, produced using the symbolic techniques in Maple, is guaranteed correct. We see that computing with the reduced system rather than the original system allowed Bertini to avoid the numerical errors (rouge red points in the yellow zone and rouge green points in the blue zone) in Fig. 2. However, in the case of Biomod-28 the reduction led to catastrophic effects for Bertini: built-in heuristics quickly (and wrongly) concluded that there are no zero dimensional solutions for the system, and when switching to a positive dimensional run also no solutions could be found.

Bertini computations (v1.5.1) were carried out on a Linux 64 bit Desktop PC with Intel i7. Maple calculations (v2016 with April 2017 Regular Chains) were carried out on a Windows 7 64 bit Desktop PC with Intel i5.

For Biomod-26 the pairs of plots together contain 476 sample points. Table 1 shows timing data. We see that both Bertini and Maple benefited from the reduced system: Bertini took a third of the original time while the speedup for Maple was even greater: a tenth of the original. Also, perhaps surprisingly, the symbolic methods were quicker than the numerical ones here. For Biomod-28 the speed-up enjoyed by the symbolic methods was even greater (almost 100 fold). However, for this system Bertini was significantly faster. The symbolic methods used are well known for their doubly exponential computational complexity (in the number of variables) so it is not surprising that as the system size increases there so should the results of the comparison. We see some other statistical data for the timings in Maple: the standard deviation for the timings is fairly modest
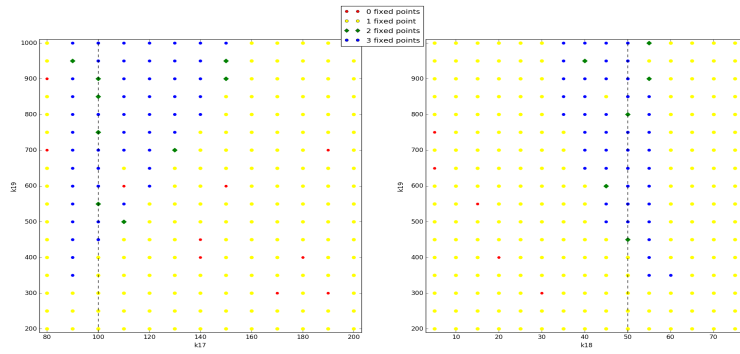
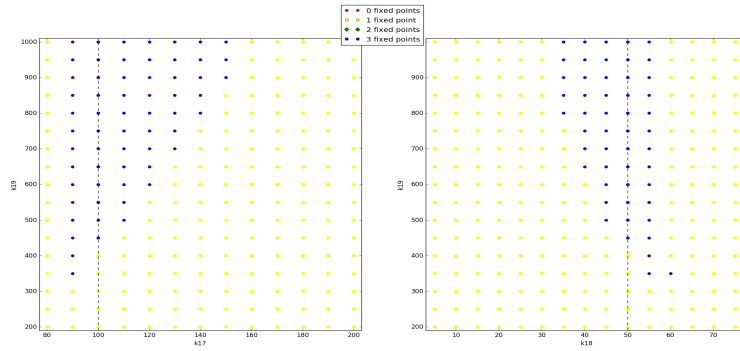**Fig. 2.** Bertini grid sampling on the original version of Biomod-26 (see Sect. 4.1)

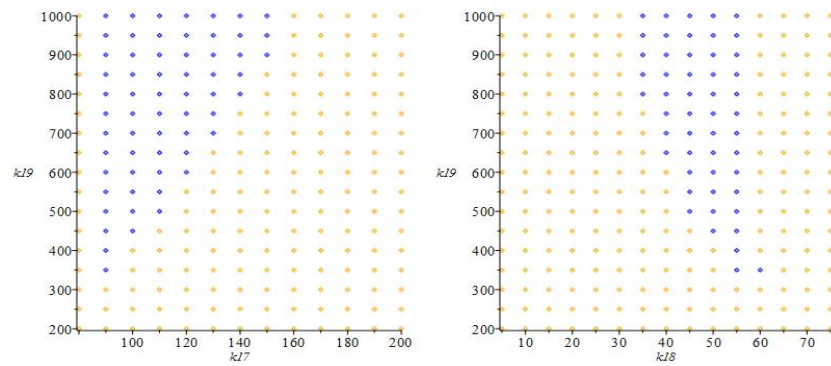**Fig. 3.** Bertini grid sampling on the reduced version of Biomod-26 (see Sect. 4.1)

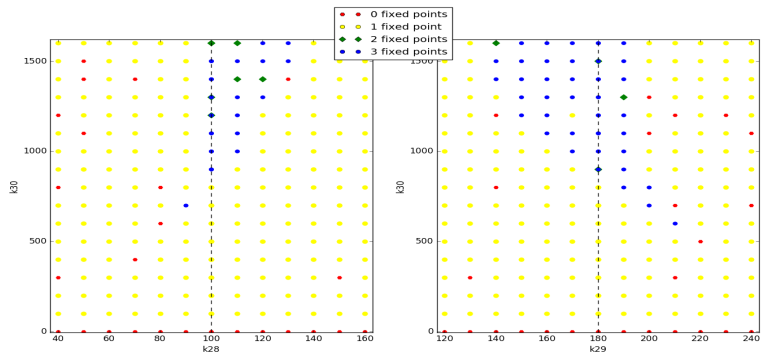**Fig. 4.** Maple grid sampling on Biomod-26 (see Sect. 4.2)

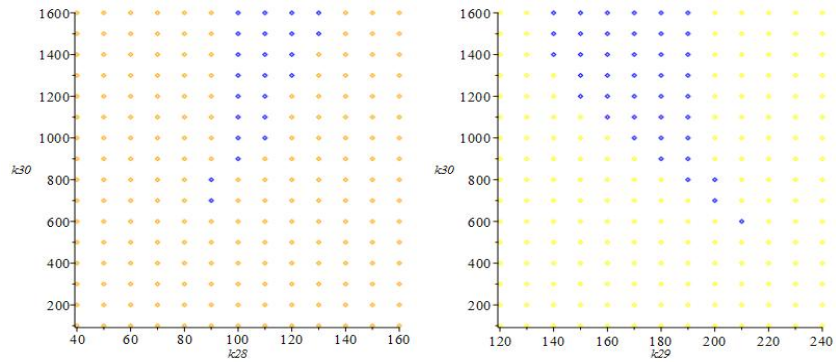**Fig. 5.** Bertini grid sampling on the original version of Biomod-28 (see Sect. 4.1).

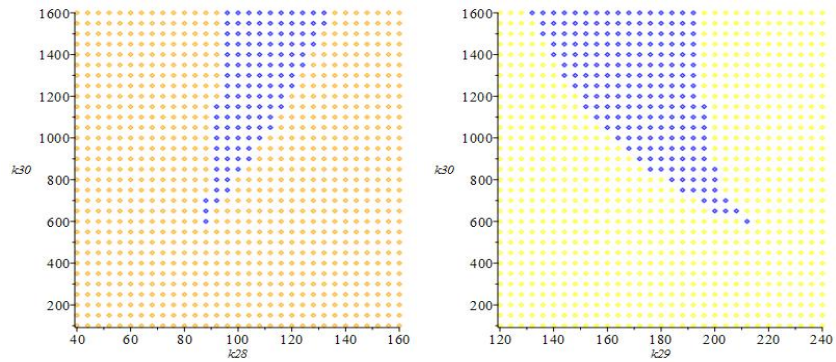**Fig. 6.** Maple grid sampling on Biomod-28 (see Sect. 4.2)

Fig. 7. As Fig. 6 but with a higher sampling rate

Table 1. Timing data (in seconds) of the grid samplings described in Sect. 4. Numerical computation is using Bertini; Symbolic computation is using Maple Regular Chains

	Numerical	Symbolic			
	Mean	Mean	Median	StdDev	Maximum
026 – Original	2.4	0.568	0.530	0.107	0.905
026 – Reduced	0.85	0.053	0.047	0.036	0.343
028 – Original	16.57	42.430	40.529	8.632	84.116
028 – Reduced	$\perp$	0.485	0.468	0.119	0.796

but in each row we see there are outliers many multiples of the mean value and so the median is always a little less than the mean average.

4.4 Going Further

Of course, we could increase the sampling density to get an improved idea of the bistability region, as in Fig. 8 and Fig. 7. However, a greater understanding comes with 3D sampling. We have performed this using the symbolic approach described above, at a linear cost proportional to the increased number of sample points. This was completed for Biomod-26: the region in question is bounded to both sides in the k_{17} and k_{18} directions but extends infinitely above in k_{19}. With the k_{19} range bound at 1000 the region is bounded by extending k_{17} to 800 and k_{18} to 600. For obtaining exact bounds (in one parameter) see [5].

Sampling in 20 seconds for k_{17} and k_{18} and 50 seconds for k_{19} produced a Maple point plot of 20400 in 18 minutes. Figure 9 shows 2D captures of the 3D bistable points and Fig. 10 the convex hull of these, produced using the `convex` package [13]. We note the lens shape seen in the orientation in the left plots is comparable with the image in the original paper of Markevich et al. [24, Fig. S7].

5 Conclusion and Future Work

We described a new graph theoretical symbolic preprocessing method to reduce problems from the MAPK network. We experimented with two systems and found the reduction offered computation savings to both numerical and symbolic approaches for the determination of multistationarity regions of parameter space. In addition, the reduction avoided instability from rounding errors in the numerical approach to one system, but uncovered major problems in that approach for the other. An interesting side result is that, at least for the smaller system, the symbolic approach can compete with and even outperform the numerical one, demonstrating how far such methods have progressed in recent years.

In future work we intend to combine the results of the present paper and our recent publication [5] to generate symbolic descriptions of the bistability region beyond the 1-parameter case. Other possible routes to achieve this is to consider the effect of the various degrees of freedom with the algorithms
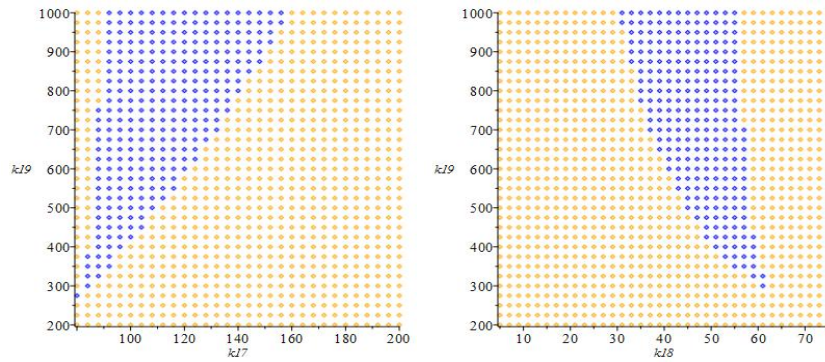
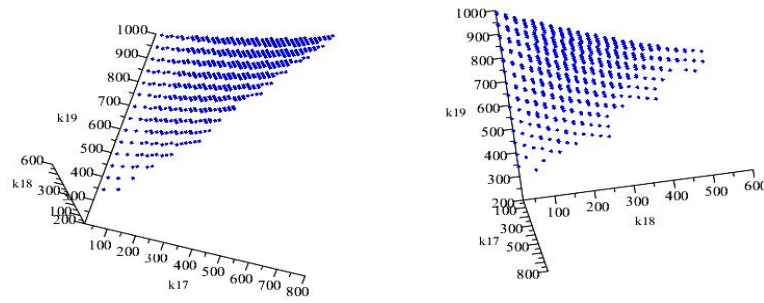**Fig. 8.** As Fig. 4 but with a higher sampling rate

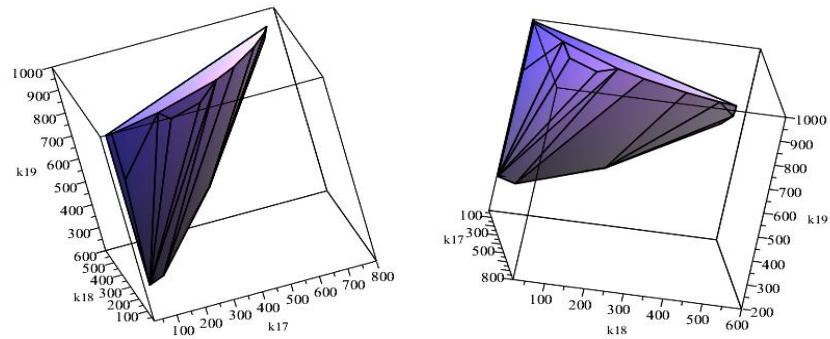**Fig. 9.** 3D Maple Point Plot produced grid sampling on Biomod-26 (see Sect. 4.4)

Fig. 10. Convex Hull of the bistable points in Fig. 9

used. For example, we have a free choice of variable ordering: Biomod-26 has 11 variables meaning $39\,916\,800$ possible orderings while Biomod-28 has 16 meaning $20\,922\,789\,888\,000$. Heuristics exist to help with this choice [8] and machine learning may be applicable [18]. Also, the MAPK problems contain many equational constraints so an approach like that described in [11] may be applicable when higher dimensional CADs are needed.

6   Acknowledgements

References

1. Bates, D.J., Hauenstein, J.D., Sommese, A.J., Wampler, C.W.: Bertini: Software for numerical algebraic geometry. doi:10.7274/R0H41PB5
2. Bhalla, U.S., Iyengar, R.: Emergent properties of networks of biological signaling pathways. Science **283**(5400) (1999) 381–387
3. Bradford, R., Chen, C., Davenport, J., England, M., Moreno Maza, M., Wilson, D.: Truth table invariant cylindrical algebraic decomposition by regular chains. In: Proceedings of the CASC 2014. Volume 8660 of LNCS. Springer (2014) 44–58
4. Bradford, R., Davenport, J., England, M., McCallum, S., Wilson, D.: Truth table invariant cylindrical algebraic decomposition. J. Symb. Comput. **76** (2016) 1–35
5. Bradford, R., Davenport, J.H., England, M., Errami, H., Gerdt, V., Grigoriev, D., Hoyt, C., Kosta, M., Radulescu, O., Sturm, T., Weber, A.: A case study on the parametric occurrence of multiple steady states. In: Proceedings of the ISSAC 2017. ACM (2017) Accepted for publication.
6. Chen, C., Davenport, J., May, J., Moreno Maza, M., Xia, B., Xiao, R.: Triangular decomposition of semi-algebraic systems. J. Symb. Comput. **49** (2013) 3–26
7. Chen, C., Moreno Maza, M., Xia, B., Yang, L.: Computing cylindrical algebraic decomposition via triangular decomposition. In: Proceedings of the ISSAC 2009, ACM (2009) 95–102
8. Dolzmann, A., Seidl, A., Sturm, T.: Efficient projection orders for CAD. In: Proceedings of the ISSAC 2004. ACM (2004) 111–118
9. Dolzmann, A., Sturm, T.: Redlog: Computer algebra meets computer logic. ACM SIGSAM Bulletin **31**(2) (1997) 2–9
10. Dolzmann, A., Sturm, T.: Simplification of quantifier-free formulae over ordered fields. J. Symb. Comput. **24**(2) (1997) 209–231
11. England, M., Bradford, R., Davenport, J.: Improving the use of equational constraints in cylindrical algebraic decomposition. In: Proceedings of the ISSAC 2015, ACM (2015) 165–172
12. Famili, I., Palsson, B.Ø.: The convex basis of the left null space of the stoichiometric matrix leads to the definition of metabolically meaningful pools. Biophys. J. **85**(1) (2003) 16–26

13. Franz, M.: Convex—a Maple package for convex geometry, version 1.2. Available at http://www.math.uwo.ca/~mfranz/convex/ (2016)
14. Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. Proc. Natl. Acad. Sci. USA **99**(12) (2002) 7821–7826
15. Grandoni, F., Könemann, J., Panconesi, A.: Distributed weighted vertex cover via maximal matchings. ACM Trans. Algorithms **5**(1) (2008) 1–12
16. Grigoriev, D., Samal, S.S., Vakulenko, S., Weber, A.: Algorithms to study large metabolic network dynamics. Math. Model. Nat. Phenom. **10**(5) (2015) 100–118
17. Gross, E., Davis, B., Ho, K.L., Bates, D.J., Harrington, H.A.: Numerical algebraic geometry for model selection and its application to the life sciences. Journal of The Royal Society Interface **13**(123) (2016)
18. Huang, Z., England, M., Wilson, D., Davenport, J., Paulson, L., Bridge, J.: Applying machine learning to the problem of choosing a heuristic to select the variable ordering for cylindrical algebraic decomposition. In: Intelligent Computer Mathematics. Volume 8543 of LNAI. Springer (2014) 92–107
19. Karp, R.M.: Reducibility among combinatorial problems. In: Complexity of Computer Computations. Plenum Press, New York (1972) 85–103
20. Košta, M.: New Concepts for Real Quantifier Elimination by Virtual Substitution. Doctoral dissertation, Saarland University, Germany (December 2016)
21. Legewie, S., Schoeberl, B., Blüthgen, N., Herzel, H.: Competing docking interactions can bring about bistability in the mapk cascade. Biophys. J. **93**(7) (2007) 2279–2288
22. Li, C., Donizelli, M., Rodriguez, N., Dharuri, H., Endler, L., Chelliah, V., Li, L., He, E., Henry, A., Stefan, M.I., Snoep, J.L., Hucka, M., Le Novère, N., Laibe, C.: BioModels database: An enhanced, curated and annotated resource for published quantitative kinetic models. BMC Systems Biology **4** (2010) 92
23. Loos, R., Weispfenning, V.: Applying linear quantifier elimination. THE Computer Journal **36**(5) (1993) 450–462
24. Markevich, N.I., Hoek, J.B., Kholodenko, B.N.: Signaling switches and bistability arising from multisite phosphorylation in protein kinase cascades. J. Cell Biol. **164**(3) (2004) 353–359
25. Rashevsky, N.: Mathematical Biophysics: Physico-Mathematical Foundations of Biology. Dover New York (1960)
26. Wang, D.: Elimination Methods. Springer (2000)
27. Zumsande, M., Gross, T.: Bifurcations and chaos in the mapk signaling cascade. J. Theor. Biol. **265**(3) (2010) 481–491