Detection of Hopf Bifurcations in Chemical Reaction Networks Using Convex Coordinates

Hassan Errami[a], Markus Eiswirth[b,c], Dima Grigoriev[d], Werner M. Seiler[a],
Thomas Sturm[e], Andreas Weber[f]

[a]Institut für Mathematik, Universität Kassel, 34132 Kassel, Germany;
[b]Fritz-Haber Institut der Max-Planck-Gesellschaft, Berlin, Germany
[c] Ertl Center for Electrochemisty and Catalysis, Gwangju Institute of Science and Technology (GIST), South Korea
[d]CNRS, Mathématiques, Université de Lille, Villeneuve d'Ascq, 59655, France
[e]Max-Planck-Institut für Informatik , RG 1: Automation of Logic, Saarbrücken, Germany
[f]Institut für Informatik II, Universität Bonn, Bonn, Germany;

Abstract

We present efficient algorithmic methods to detect Hopf bifurcation fixed points in chemical reaction networks with symbolic rate constants, thereby yielding information about the oscillatory behavior of the networks. Our methods use the representations of the systems on convex coordinates that arise from stoichiometric network analysis. One of our methods then reduces the problem of determining the existence of Hopf bifurcation fixed points to a first-order formula over the ordered field of the reals that can then be solved using computational logic packages. The second method uses ideas from tropical geometry to formulate a more efficient method that is incomplete in theory but worked very well for the examples that we have attempted; we have shown it to be able to handle systems involving more than 20 species.

Keywords:
Hopf Bifurcation, Chemical Reaction Networks, Convex Coordinates, Stoichiometric Network Analysis

1. Introduction

The dynamics of (bio)-chemical systems are usually described by power-law kinetics, i.e. the reaction rates are proportional to some power of the

Preprint submitted to Journal of Computational Physics *July 28, 2013*

species concentrations involved. If it is assumed that these (bio)-chemical systems follow mass action kinetics then the dynamics of these reactions can be represented by ordinary differential equations (ODE) for systems without additional constraints or by differential algebraic equations (DAEs) for systems with constraints. In complex systems it is sometimes difficult to estimate the values of the parameters of these equations, so simulation studies involving the kinetics constitute a daunting task. Nevertheless, quite a few conclusions regarding the dynamics can be drawn from the structure of the reaction network itself. In this context, there has been a surge in the development of algebraic methods that are based on the structure of the network and the associated stoichiometry of the chemical species. These methods are aimed at understanding the qualitative behavior (e.g., steady states, stability, bifurcations, and periodic orbits) of the network. In particular, the analysis of chemical reaction networks by detecting the occurrence of Hopf bifurcations was a topic of considerable research effort in the last decade due to its relation to oscillatory behavior. A fully algebraic method for the computation of Hopf bifurcation fixed points for systems with polynomial vector fields has already been introduced by El Kahoui and Weber [1] using the powerful technique of quantifier elimination on real closed fields [2]. This technique has already been successfully applied to the mass action kinetics of few dimensions [3]. Although the method is complete in theory it fails in practice for systems of higher dimensions and for systems with constraints which occur in chemical and biochemical systems. Using ideas from so called *stoichiometric network analysis* (SNA) [4], it is possible to analyze the system dynamics in flux space instead of the concentration space and to represent the space of the steady states with a combination of subnetworks using methods from convex geometry. Methods for detecting Hopf bifurcations using similar approaches have been used in several "hand computations" in a semi-algorithmic way for parametric systems, the most elaborate of which is described in [5].

In this paper we present efficient algorithmic methods to detect Hopf bifurcation fixed points in chemical reaction networks with symbolic rate constant; our methods are based on combinations, enhancements and extensions of these previous methods. In the first algorithmic method presented in this paper we applied a combination of the known (and already demonstrated) algorithmic reduction to quantifier elimination problems over the reals and the algorithmic solutions of these problems with techniques arising from stoichiometric network analysis, such as the use of convex coordinates.

Technically this combination will yield an existentially quantified problem that consists of determining Hopf bifurcation fixed point with empty unstable manifold involving the conjunction of the following condition: an equality condition on the principal minor $\Delta_{n-1} = 0$ of the Jacobian of the vector field in conjunction with inequality conditions on $\Delta_{n-2} > 0 \wedge \cdots \wedge \Delta_1 > 0$ and positivity conditions on the variables and parameters.

Another method for the parametric detection of Hopf bifurcations that also uses techniques of stoichiometric networks analysis is presented as the second algorithm in this paper. This algorithm builds on the basic observation that the condition for existence of Hopf bifurcation fixed points when using convex coordinates is given by the single polynomial equation $\Delta_{n-1} = 0$ (together with positivity conditions on the convex coordinates) and (drop resp. delaying a test for the existence of unstable empty manifolds on already determined witness points for Hopf bifurcations). Therefore the main algorithmic problem is to determine whether a single multivariate polynomial can have a zero for positive coordinates. For this purpose we provide heuristics on the basis of the Newton polytope that ensure the existence of positive and negative values of the polynomial for positive coordinates.

We evaluate our methods on a variety of examples—some of which concern a number of dimensions even higher than 20. Considering the performance of our methods we could even analyze some networks in their unreduced forms, a task for which the only previously available approach was the analysis of quasi-steady state approximations.

2. Chemical Reaction Networks

In chemical and biochemical systems, reactions networks can be represented as a set of reactions. A chemical reaction occurs when two or more chemical species react to become new chemical species. This process is usually represented by an equation in which the *reactants* are given on the left-hand side of an arrow and the *products* on the right-hand side; the numbers next to the species, called *stoichiometric coefficients*, present the relative amounts in which the chemical species participate in a reaction; and the parameter on the arrow, called the *rate constant*, stands for an experimental constant that influences the reaction velocity. A chemical reaction is called *irreversible* if it proceeds only in one direction and is called *reversible*, if it can proceeds in either direction. In order to be compatible with thermodynamics, in reversible reactions, the difference between the kinetic exponents of the reverse and forward reactions must be equal to the stoichiometric coefficient for each species; this is referred to as mass-action kinetics.

An example of a chemical reaction, as it usually appears in the literature, is the following:

$$A + B \xrightarrow{k} 3A + C$$

In this reaction, one unit of chemical *species* A and one of B react (at reaction rate k) to form three units of A and one of C. The concentrations of these three species, denoted by x_a,x_b and x_c, will change in time as the reaction occurs. Under the assumption of *mass-action kinetics*, species A and B react at a rate proportional to the product of their concentrations, where the proportionality constant is the rate constant k. Noting that the reaction yields a net change of two units in the amount of A [6, 7, 5], we obtain the following corresponding differential equations:

$$\begin{aligned} \dot{x_a} &= 2kx_ax_b \\ \dot{x_b} &= -kx_ax_b \\ \dot{x_c} &= kx_ax_b \end{aligned} \tag{1}$$

A *chemical reaction network* can be defined as a finite set of chemical reactions. It can be presented as a finite directed graph whose vertices are labeled with complexes and whose edges are labeled with parameters (reaction rate constants). Specifically, the digraph is denoted as $G = (V, E)$, with vertex set $V = \{1, 2, ..., m\}$ and edge set $E \subseteq \{(i, j) \in V \times V : i \neq j\}$. A network is reversible if the graph G is undirected, in which case each undirected edge has two labels k_{ij} and k_{ji} [7, 6].

2.1. Flux Cone and Convex Parameters

The usual way to understand the behavior of mass-action chemical systems is to observe the time evolution of the species concentration. This can be mathematically represented by a system of coupled differential equations, where each equation represent a change in a corresponding species concentration. With this approach, the analysis of chemical systems in concentration space increases in difficulty as the number of species increases.

In 1980, Clarke introduced a new method, called stoichiometric network analysis (SNA), to analyze the stability of mass-action chemical reaction systems[4]. The idea of SNA is to observe the dynamics of the system in reaction space instead of concentration space. This leads to the expansion of the steady state into a combination of subnetworks that form a convex cone in flux-space, called a *flux cone* [8].

To analyze a chemical system one is interested in its stationary reaction behavior, which is observable in experiments, e.g., one investigates the solution set of

$$\mathcal{S}v(x,k) = 0. \tag{2}$$

where $\mathcal{S}$ represents the stoichiometric matrix and $v(x,k)$ represents the flux vector. As long as we split each reversible reaction into two irreversible reactions (corresponding to the forward and backward directions) the flux through these reactions must be greater than or equal to zero, i.e.,

$$v(x,k) \geq 0 \tag{3}$$

The set of all possible stationary solutions over the network $\mathcal{N}$ that fulfil the equation (2) and the constraint (3) defines a convex polyhedral cone, called *flux cone* [4, 9]. The minimal set of generating vectors $\mathcal{E}$, which can be geometrically interpreted as the edges of the flux cone are known in chemistry as *extreme fluxes* or *extreme currents*. Each flux vector satisfying the steady-state equations can be represented in flux space as a linear combination of the extreme currents $\mathcal{E}$ with nonnegative coefficients j_i called the *convex parameters*.

2.2. Modeling Chemical Systems with Pseudolinear Ordinary Differential Equations

The differential equations in chemical reaction networks are usually constrained reflecting various physical conservation laws. The systems with linear constraints that are often found in chemical reaction networks can easily be generalized to *pseudolinear ordinary differential equations*. The basic underlying property of the considered differential equations is captured by the following definition.

Definition 1. We call an autonomous system of ordinary differential equations $\dot{\mathbf{x}} = \boldsymbol{\phi}(\mathbf{x})$ for an unknown function $\mathbf{x} : \mathbb{R} \to \mathbb{R}^n$ *pseudolinear*, if its right hand side can be written in the form $\boldsymbol{\phi}(\mathbf{x}) = N\boldsymbol{\psi}(\mathbf{x})$ with a constant matrix $N \in \mathbb{R}^{n\times m}$ and some vector valued function $\boldsymbol{\psi} : \mathbb{R}^n \to \mathbb{R}^m$.

Obviously, any *polynomially* nonlinear system can be written in such a form, if we take as $\boldsymbol{\psi}(\mathbf{x})$ the vector of all terms appearing on the right-hand side of the system. As one can see from the following two lemmata, the pseudolinear structure is of interest only in the case that the matrix N does not possess full row rank and thus, the range of N is not the full space $\mathbb{R}^n$. In the following, we will always assume that the function $\boldsymbol{\psi}$ satisfies $m \geq n$, as this is usually the case in applications like reaction kinetics.

Lemma 1. *For a pseudolinear system* $\dot{\mathbf{x}} = N\boldsymbol{\psi}(\mathbf{x})$ *any affine subspace of the form* $\mathcal{A}_{\mathbf{y}} = \mathbf{y} + \operatorname{im} N \subseteq \mathbb{R}^n$ *for an arbitrary constant vector* $\mathbf{y} \in \mathbb{R}^n$ *defines an invariant manifold.*

Proof. Obviously, we have $\dot{\mathbf{x}}(t) \in \operatorname{im} N$ for all times t and $T_{\mathbf{x}}\mathcal{A}_{\mathbf{y}} = \operatorname{im} N$ for all points $\mathbf{x} \in \mathcal{A}_{\mathbf{y}}$ by the definition of an affine space. Thus, if $\mathbf{x}(0) \in \mathcal{A}_{\mathbf{y}}$, then the entire trajectory remain in $\mathcal{A}_{\mathbf{y}}$. $\square$

For application to reaction kinetics, the following minor strengthening of Lemma 1 is of interest. Assume that the function $\boldsymbol{\psi}$ additionally satisfies $\boldsymbol{\psi}(\mathbf{x}) \in \mathbb{R}^m_{\geq 0}$ for all $\mathbf{x} \in \mathbb{R}^n_{\geq 0}$ which is for example trivially the case when each component of $\boldsymbol{\psi}$ is a polynomial with positive coefficients. If we solve our differential equation for non-negative initial data $\mathbf{x}(0) = \mathbf{x}_0 \in \mathbb{R}^n_{\geq 0}$, then the solution always remains in the convex polyhedral cone $\mathbf{x}_0 + \bigl\{\sum_{i=1}^m \lambda_i \mathbf{n}_i \mid \forall\, i : \lambda_i \geq 0\bigr\}$ where the vectors $\mathbf{n}_i$ are the columns of the matrix N. Indeed, in this case the tangent vector $\dot{\mathbf{x}}(t)$ along the trajectory is trivially always a non-negative linear combination of the columns of N.

Lemma 2. *Let* $\mathbf{v}^T \cdot \mathbf{x} = \mathrm{Const}$ *for some vector* $\mathbf{v} \in \mathbb{R}^n$ *be a* linear *conservation law of a pseudolinear system* $\dot{\mathbf{x}} = N\boldsymbol{\psi}(\mathbf{x})$ *such that* $\operatorname{im} \boldsymbol{\psi}$ *is not contained in a hyperplane. Then* $\mathbf{v} \in \ker N^T$. *Conversely, any vector* $\mathbf{v} \in \ker N^T$ *induces a linear conservation law.*

Proof. Let us first assume that $\mathbf{v} \in \ker N^T$. Then

$$\frac{\mathrm{d}}{\mathrm{d}t}\bigl(\mathbf{v}^T \cdot \mathbf{x}\bigr) = \mathbf{v}^T N \boldsymbol{\psi}(\mathbf{x}) = \bigl(N^T \mathbf{v}\bigr)^T \boldsymbol{\psi}(\mathbf{x}) = 0\ .$$

If $\mathbf{v}^T \cdot \mathbf{x} = \mathrm{Const}$ is a conservation law, then differentiation with respect to time yields $\bigl(N^T\mathbf{v}\bigr)^T \boldsymbol{\psi}(\mathbf{x}) = 0$. Because of our assumption regarding the function $\boldsymbol{\psi}$, this implies that $N^T\mathbf{v} = 0$. $\square$

By a classical result in linear algebra (the four “fundamental spaces” of a matrix), we have the direct sum decomposition $\mathbb{R}^n = \operatorname{im} N \oplus \ker N^T$, which is an orthogonal decomposition with respect to the standard scalar product. Therefore we may consider Lemma 1 as a corollary to Lemma 2, as the above described invariant manifolds are simply defined by all the linear conservation laws produced by Lemma 2.[1]

Gatermann and Huber [10] speak of a conservation law only in the case that $v_i \geq 0$ for all components v_i of the vector $\mathbf{v}$. In mathematics, we are not aware of such a restriction and cannot see any physical reasons to impose it.

3. Hopf Bifurcations and Invariant Manifolds

3.1. Hopf Bifurcations

Consider a parameterized autonomous ordinary differential equation of the form $\dot{x} = f(u, x)$ with a scalar parameter u. By a classical result of Hopf, at the point (u_0, x_0), this system exhibits a Hopf bifurcation, i. e. an equilibrium transforms into a limit cycle, if $f(u_0, x_0) = 0$ and if the Jacobian $D_x f(u_0, x_0)$ has a simple pair of purely imaginary eigenvalues and no other eigenvalues with zero real parts [11, Thm. 3.4.2].[2] The proof of this result is based on the center manifold theorem. From a physical point of view, the most interesting case is that the unstable manifold of the equilibrium (u_0, x_0) is empty. However, for the mere existence of a Hopf bifurcation, this assumption is not necessary.

In [1], it is shown that for a parameterized vector field $f(u, x)$ and the autonomous ordinary differential system associated with it, there is a semi-algebraic description of the set of parameter values for which a Hopf bifurcation (with an empty unstable manifold) occurs. Specifically, this semi-algebraic description can be expressed by the following first-order formula:

$$\begin{aligned} &\exists x (f_1(u,x) = 0 \wedge f_2(u,x) = 0 \wedge \cdots \wedge f_n(u,x) = 0 \\ &\qquad \wedge a_n > 0 \wedge \Delta_{n-1}(u,x) = 0 \wedge \Delta_{n-2}(u,x) > 0 \wedge \cdots \wedge \Delta_1(u,x) > 0) \end{aligned} \tag{4}$$

---

[1] Note that in the special case most relevant for us, namely that each component of $\boldsymbol{\psi}$ is a different monomial, the assumption made in Lemma 2 is always satisfied.

[2] We ignore here the non-degeneracy condition that this pair of eigenvalues crosses the imaginary axis transversally, as it is always satisfied in realistic models.

In this formula a_n is $(-1)^n$ times the Jacobian determinant of the matrix $Df(u,x)$, and $\Delta_i(u,x)$ is the i^{th} Hurwitz determinant of the characteristic polynomial of the same matrix $Df(u,x)$.

The proof uses a formula of Orlando [12], which is also discussed in several monographs, e.g. in [13] and [14]. However, a closer inspection of the two parts of the proof of [1, Theorem 3.5] shows the following: for a fixed point (given in possibly parameterized form) the condition that there is a pair of purely imaginary eigenvalues is given by the condition $\Delta_{n-1}(u,x) = 0$ and the condition that each other eigenvalue has a negative real part is given by $\Delta_{n-2}(u,x) > 0 \wedge \cdots \wedge \Delta_1(u,x) > 0$. This statement (without referring to parameters explicitly) is also contained in [15, Theorem 2], in which a different proof technique is used.

Therefore, if we drop the condition for Hopf bifurcation points that they have empty unstable manifolds, a semi-algebraic description of the set of parameter values for which a Hopf bifurcation occurs for the system is given by the following formula:

$$\begin{aligned}\exists x(f_1(u,x) = 0 \wedge f_2(u,x) = 0 \wedge \cdots \wedge f_n(u,x) = 0\\ \wedge\, a_n > 0 \wedge \Delta_{n-1}(u,x) = 0)\end{aligned} \tag{5}$$

Notice that when the quantifier elimination procedure yields sample points for existentially quantified formulae—as is the case for the virtual-substitution based method provided by Redlog—then the condition $\Delta_{n-2}(u,x) > 0 \wedge \cdots \wedge \Delta_1(u,x) > 0)$ can be tested for the sample points later on, i.e. one can then test whether this Hopf bifurcation fixed point has an empty unstable manifold.

Example: Lorenz system. The famous “Lorenz system” [17, 11, 18] is given by the following system of ODEs:

$$\dot{x}(t) = \alpha\,(y(t) - x(t)) \tag{6}$$

$$\dot{y}(t) = r\,x(t) - y(t) - x(t)\,z(t) \tag{7}$$

$$\dot{z}(t) = x(t)\,y(t) - \beta\,z(t) \tag{8}$$

It is named after Edward Lorenz at MIT, who first investigated this system as a simple model arising in connection with fluid convection.

After imposing positivity conditions on the parameters the following answer is obtained using a combination of Redlog and formula simplification using SLFQ for the test of a Hopf bifurcation fixed point:

$$
\begin{aligned}
(-\alpha^2 - \alpha\beta + \alpha r - 3\alpha - \beta r - r = 0 \vee -\alpha\beta + \alpha r - \alpha - \beta^2 - \beta = 0)\ \wedge& \\
-\alpha^2 - \alpha\beta + \alpha r - 3\alpha - \beta r - r \leq 0\ \wedge& \\
\beta > 0\ \wedge\ \alpha > 0\ \wedge\ -\alpha\beta + \alpha r - \alpha - \beta^2 - \beta \geq 0& \quad (9)
\end{aligned}
$$

When testing for Hopf bifurcation fixed points with empty unstable manifolds, we obtain the following formulae:

$$
\begin{aligned}
&\alpha^2 + \alpha\beta - \alpha r + 3\alpha + \beta r + r = 0 \wedge \\
&\quad \alpha r - \alpha - \beta^2 - \beta \geq 0\ \wedge \\
&\quad 2\alpha - 1 \geq 0\ \wedge\ \beta > 0 \qquad (10)
\end{aligned}
$$

These two formulae are not equivalent, and therefore, for the case of the Lorenz system not all Hopf bifurcation fixed points have unstable empty manifolds.

3.2. Reduction to Invariant Manifolds

As already discussed in Sec. 2.2, chemical reaction systems with linear conservation laws can easily be generalized to pseudolinear ordinary differential equations. However the existence of these constraints makes the Jacobian matrices singular and thus leads to incorrect computations of Hopf bifurcations. We present here a method to tackle these singularities by reduction to invariant manifolds. The following material represents a slight generalization of results already well-known for systems in reaction kinetics (see, e. g. [10] and references therein).

If a dynamical system admits invariant manifolds, we may consider a system of lower dimension by reducing to such a manifold. However, in general it may not be possible to explicitly derive the reduced system. Nevertheless, for many purposes, such as stability or bifurcation analysis, one can easily reduce to smaller matrices. The following result describes such a reduction process in the linear case. It represents an elementary exercise in basic linear algebra. To avoid the inversion of matrices, we consider $\mathbb{R}^n$ here to be a Euclidean space with respect to the standard scalar product.

Lemma 3. *Let A be the matrix of a linear mapping $\mathbb{R}^n \to \mathbb{R}^n$ for the standard basis, and let $\mathcal{U} \subseteq \mathbb{R}^n$ be a k-dimensional A-invariant subspace. If the columns of the matrix $W \in \mathbb{R}^{n\times k}$ define an orthonormal basis of $\mathcal{U}$, then the restriction of the mapping to the subspace $\mathcal{U}$ with respect to the basis defined by W is given by the matrix $W^TAW \in \mathbb{R}^{k\times k}$.*

Proof. Considered as a linear map $\mathbb{R}^k \to \mathcal{U} \subseteq \mathbb{R}^n$, the matrix W defines a parametrization of $\mathcal{U}$ with inverse $W^T : \mathcal{U} \to \mathbb{R}^k$. Indeed, $W^TW = \mathbb{1}_k$, since the columns of W are orthonormal. If $\mathbf{v} \in \mathcal{U}$, then $\mathbf{v} = W\mathbf{w}$ for some vector $\mathbf{w} \in \mathbb{R}^k$ and thus $W^T\mathbf{v} = (W^TW)\mathbf{w} = \mathbf{w}$ implying that $(WW^T)\mathbf{v} = W\mathbf{w} = \mathbf{v}$, i. e. the matrix $WW^T \in \mathbb{R}^{n\times n}$ describes $\mathrm{id}_{\mathcal{U}}$. By standard linear algebra, the matrix W^TAW therefore describes the restriction of A to $\mathcal{U}$. $\square$

As a simple application, we note that in the case of a pseudolinear system $\dot{\mathbf{x}} = N\boldsymbol{\psi}(\mathbf{x})$ the stability properties of an equilibrium $\mathbf{x}_e$ of the pseudolinear system $\dot{\mathbf{x}} = N\boldsymbol{\psi}(\mathbf{x})$ are determined by the eigenstructure of the reduced Jacobian

$$J = W^TN\mathrm{Jac}\big(\boldsymbol{\psi}(\mathbf{x}_e)\big)W \in \mathbb{R}^{k\times k}$$

where the columns of W form an orthonormal basis of $\operatorname{im} N$. If parameters are present, then for a bifurcation analysis the eigenstructure of this matrix and not of the full Jacobian (which is an n-dimensional matrix), is relevant.

3.3. Stability and Bifurcations for Semi-Explicit DAEs

The considerations indicated in the previous section can be easily extended to more general situations, as they appear in the theory of DAEs. For simplicity (and because it suffices for our purposes), we assume that we are dealing with an autonomous system in the semi-explicit form

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})\,, \qquad 0 = \mathbf{g}(\mathbf{x}) \tag{11}$$

where $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$ and $\mathbf{g} : \mathbb{R}^n \to \mathbb{R}^{n-k}$. Furthermore, we assume that the above system of ordinary differential equations is involutive,[3] i. e. that it already contains all its integrability conditions. This assumption is equivalent to the existence of a matrix valued function $M(\mathbf{x})$ such that

$$\mathrm{Jac}\big(\mathbf{g}(\mathbf{x})\big) \cdot \mathbf{f}(\mathbf{x}) = M(\mathbf{x}) \cdot \mathbf{g}(\mathbf{x})\,. \tag{12}$$

---

[3] See [19] for an introduction to the theory of involutive systems.

Therefore, one may say that the components of $\mathbf{g}$ are *weak* conservation laws, as their time derivatives vanish modulo the constraint equations $\mathbf{g}(\mathbf{x}) = 0$.

Let $\mathbf{x}_e$ be an equilibrium of (11), i. e. we have $\mathbf{f}(\mathbf{x}_e) = 0$ and $\mathbf{g}(\mathbf{x}_e) = 0$. We introduce the real matrices

$$A = \mathrm{Jac}\big(\mathbf{f}(\mathbf{x}_e)\big) \in \mathbb{R}^{n\times n}\,, \quad B = \mathrm{Jac}\big(\mathbf{g}(\mathbf{x}_e)\big) \in \mathbb{R}^{(n-k)\times n}\,.$$

For simplicity, we assume in the following that the matrix B has full rank (or, in other words, that our algebraic constraints are independent) and thus that $\ker B$ is a k-dimensional subspace. The proof of the next result clearly demonstrates why the assumption that the system (11) is involutive is important, as the relation (12) is crucial for it.

Lemma 4. *The subspace* $\ker B$ *is* A*-invariant.*

PROOF. Set $\bar{M} = M(\mathbf{x}_e)$. Differentiating (12) and evaluating the result at $\mathbf{x} = \mathbf{x}_e$ yields the relation $BA = \bar{M}B$. Thus, if $\mathbf{v} \in \ker B$, then also $A\mathbf{v} \in \ker B$ because $B(A\mathbf{v}) = \bar{M}(B\mathbf{v}) = 0$. □

In the case that (11) is a linear system, i. e. we may write $\mathbf{f}(\mathbf{x}) = A\mathbf{x}$ and $\mathbf{g}(\mathbf{x}) = B\mathbf{x}$ by assuming that $\mathbf{x}_e = 0$, we can easily revert the argument in the proof of Lemma 4 and thus conclude that now (11) is involutive, if and only if $\ker B$ is A-invariant.

Proposition 5. *Let the columns of the matrix* $W \in \mathbb{R}^{n\times k}$ *define an orthonormal basis of* $\ker B$*. The linear stability of the equilibrium* $\mathbf{x}_e$ *is then decided by the eigenstructure of the matrix* W^TAW*.*

PROOF. Linearization around the equilibrium $\mathbf{x}_e$ yields the associated variational system $\dot{\mathbf{z}} = A\mathbf{z}$, $B\mathbf{z} = 0$. We complete W to an orthogonal matrix $\widehat{W}$ by adding some further columns and perform the coordinate transformation $\mathbf{z} = \widehat{W}\mathbf{y}$. This yields the system $\dot{\mathbf{y}} = \widehat{W}^TA\widehat{W}\mathbf{y}$, $B\widehat{W}\mathbf{y} = 0$. Because the columns of W span $\ker B$ by construction, the second equation implies that only the upper k components of $\mathbf{y}$ may be different from zero. Furthermore, Lemma 4 implies that the matrix $\widehat{W}^TA\widehat{W}\mathbf{y}$ is in block triangular form with the left upper $k\times k$ block given by W^TAW. If we denote the upper part of $\mathbf{y}$ by $\tilde{\mathbf{y}}$, we thereby obtain the equivalent reduced system $\dot{\tilde{\mathbf{y}}} = W^TAW\tilde{\mathbf{y}}$ which implies our claim. □

Let $\mathbf{v} \in \mathbb{R}^k$ be a (generalized) eigenvector of the reduced matrix W^TAW, i.e. we have $(W^TAW - \lambda\mathbb{1}_k)^\ell\mathbf{v} = 0$ for some $\ell > 0$ and $\lambda \in \mathbb{R}$. Because $W^TW = \mathbb{1}_k$ and WW^T defines the identity map on $\ker B$ (see the proof of Lemma 3), we obtain $W^T(A - \lambda\mathbb{1}_n)^\ell W\mathbf{v} = 0$ implying that $W\mathbf{v} \in \mathbb{R}^n$ is a (generalized) eigenvector of A for the same eigenvalue λ, since the matrix W^T defines an injective map. Therefore every eigenvalue of the reduced matrix W^TAW is also an eigenvalue of A.

It is also not difficult to interpret the remaining (generalized) eigenvectors of A. By construction, they are transversal to the constraint manifold defined by $\mathbf{g}(\mathbf{x}) = 0$ and they describe whether this manifold is attractive or repulsive for the flow of the unconstrained system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$. While this is for example, of considerable importance to the numerical integration of (11), as it describes the drift off the constraint manifold arising from rounding and discretization errors, it has no influence on the stability of the exact flow of (11).

The irrelevance of the remaining (generalized) eigenvectors of A also becomes apparent from the following argument. Recall that the differential part of (11) defines what is often called an *underlying differential equation* for the DAE, i.e. an unconstrained differential equation which possesses for initial data satisfying the constraints the same solution as the DAE. Consider now the modified system obtained by adding to the right hand side of the differential part an arbitrary linear combination of the algebraic part. It is easy to see that the arising DAE (which simply has a different underlying equation)

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + L(\mathbf{x})\mathbf{g}(\mathbf{x})\,, \qquad 0 = \mathbf{g}(\mathbf{x})\,,$$

where $L(\mathbf{x})$ is a matrix valued function of appropriate dimensions, possesses exactly the same solutions as (11); in particular $\mathbf{x}_e$ is still an equilibrium. If we proceed as above with the linear stability analysis of $\mathbf{x}_e$, the matrix B remains unchanged, whereas A is transformed into the modified matrix $\tilde{A} = A + \bar{L}B$ with $\bar{L} = L(\mathbf{x}_e)$. Obviously, $\ker B$ is also $\tilde{A}$-invariant, and furthermore $W^T\tilde{A}W = W^TAW$, if the columns of W form a basis of $\ker B$ as in Proposition 5.

Therefore, all (generalised) eigenvectors lying in $\ker B$ are equal for A and $\tilde{A}$, so the stability of $\mathbf{x}_e$ is not affected by this transformation. However, the remaining (generalised) eigenvectors may change arbitrarily. One can for example show that by a suitable choice of the matrix L one may always achieve that the constraint manifold becomes attractive.

4. ***HoCoQ*: An Algorithm for Computing Hopf Bifurcations using Convex Coordinates and Quantifier Elimination**

In this section we present an algorithmic approach for computing the Hopf bifurcations in chemical systems using convex coordinates instead of concentration coordinates. It is based on two methods already presented in this paper: stoichiometric network analysis and manifold reduction for systems with conservation laws. It also makes fundamental use of real quantifier elimination on a real closed field. Figure 1 elucidates the workflow of the algorithm, which is explained in detail in the following subsections and in the pseudo-code presented in Algo. 4.5

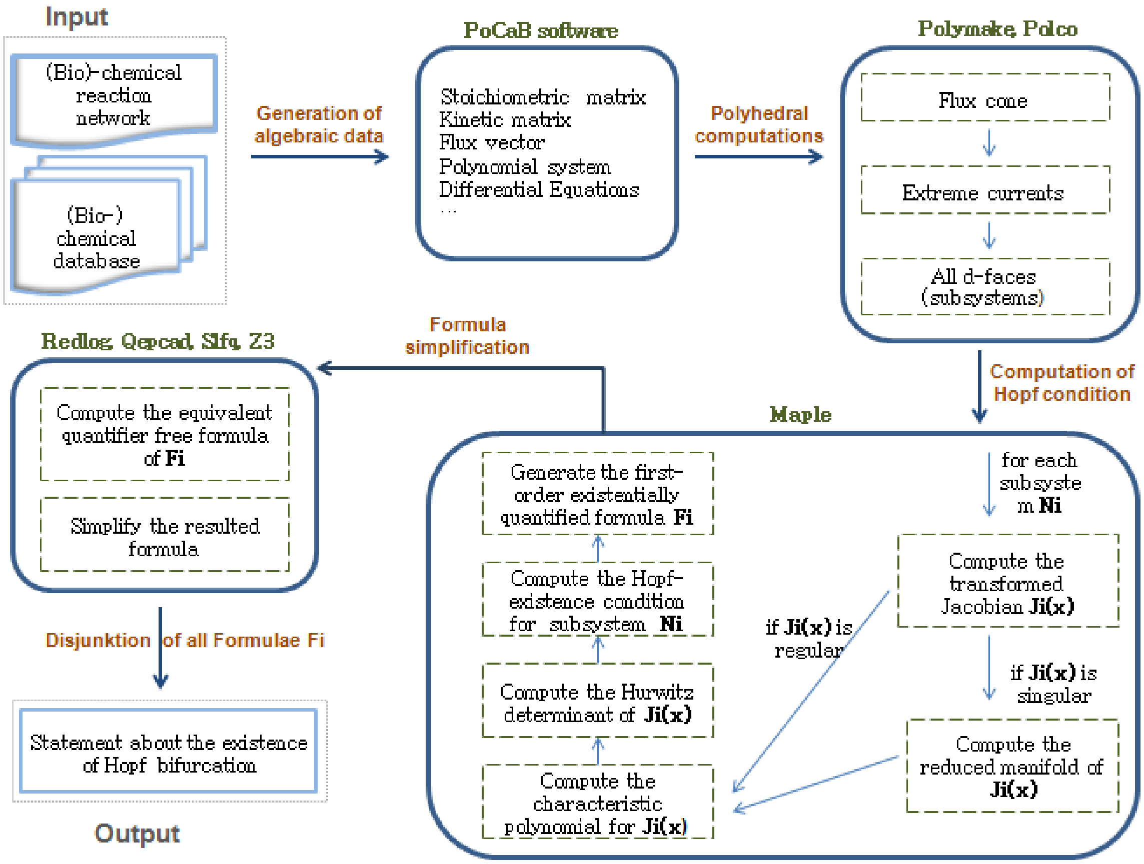

Figure 1:

4.1. Pre-processing

To begin the analysis of a chemical network we need two significant pieces of information to describe all reaction laws. The first piece of information describes the occurrence of the species in each reaction. This can be presented by a stoichiometric matrix $\mathcal{S}$, in which the species form the rows and the reactions form the columns. Each entry of the matrix presents the difference in the number of produced and consumed molecules of the corresponding species in the corresponding reaction. The second piece of information describes the velocities of the reactions. This can be presented by a flux vector $v(x, k)$ or by a kinetic matrix $\mathcal{K}$. The entries of this matrix indicate whether the species is a reactant and therefore effects the velocity of the reaction (entry = stoichiometric coefficient of species) or not (entry = 0). To enable the computational analysis of a chemical network the reactions should be presented in a format that enables the accurate representation of the network and allows the computational extraction of required data. For our computations we use the XML-based format SBML [20], which is widely used in biological research. As pre-processing step we parse the SBML file that presents the chemical network and generate the necessary algebraic data using our *PoCaB* platform [21]. *PoCaB* is a software infrastructure and data base that is used to explore algebraic methods for bio-chemical reaction networks. It provides tools to extract relevant algebraic entities from the network description such as stoichiometric matrices and their factorizations, kinetic matrices, polynomial systems, deficiencies and differential equations.

4.2. Polyhedral Computations

The advantage of stoichiometric network analysis is the ability to analyze subnetworks separately instead of analyzing the whole complex network. The first step in the analysis is the computation of extreme currents. We must therefore include algorithms that are capable of dealing with polyhedral computations. There are several software packages for such computations and in computational geometryin particular, there are two efficient tools that we use in our current implementation, namely, the Java tool POLCO[4] and the program POLYMAKE[5], which was written in Perl and C++ and designed for the algorithmic treatment of polytopes and polyhedra [22].

---

[4] `http://www.csb.ethz.ch/tools/polco`

[5] `http://www.polymake.org/doku.php`

Enumerating extreme currents $\mathcal{E}$ is the basis for simplifying the analysis of chemical networks by decomposing the network into minimal steady-state generating subnetworks. The influence of a subnetwork on the full network dynamics (i.e., how much the given subnetwork plays a part in creating a certain steady state) depends on the convex parameters j_i [4, 23]. From a chemical perspective, Hopf bifurcations occur mostly in the spaces formed by two or three adjacent extreme currents, i.e detecting Hopf bifurcations in subsystems can be restricted to the subsystems that are formed by combining 2-faces or 3-faces of the flux cone. As step 3 of our algorithms, we compute all subsystems generated by the 2- and 3-faces using POLYMAKE. Our algorithm can also handle d-faces for $d > 3$ yielding a complete method in theory, but the restricted case of $d = 2, 3, 4$ will be of the greatest practical interest.

4.3. Computation of the Hopf Condition

The central task of this approach is to formulate a condition for the existence of Hopf bifurcations using convex coordinates and based on the Routh-Hurwitz criterion for each computed subsystem . We first compute the Jacobian in reaction space using convex parameters, if the Jacobian is singular, we reduce the subsystem to the invariant manifold, we compute a semi-algebraic formula expressing the condition for the occurrence of Hopf bifurcations, and finally, we generate the first-order existentially quantified formula.

4.3.1. Computation of the Jacobian in Reaction Space

Gatermann et al. [5] proved that the Jacobian of the reaction coordinates z can be transformed into the following form:

$$\widehat{\mathrm{Jac}}(z) = \mathcal{S}\mathrm{diag}(z)\mathcal{K}^t \tag{13}$$

If x is a steady state we transform into convex coordinates j_i with $z = \sum_i^d j_i\mathcal{E}_i$ where d is the dimensionality of the face. When we replace $\widehat{\mathrm{Jac}}(z)$ in equation (13) we obtain the new Jacobian in reaction space:

$$\mathrm{Jac}(x) = \mathcal{S}\widehat{\mathrm{Jac}}(j)\mathrm{diag}(\sum_i^d j_i\mathcal{E}_i)\mathcal{K}^t\mathrm{diag}(1/x_1, ..., 1/x_m) \tag{14}$$

.

4.3.2. Jacobian on the Reduced Manifold

Chemical reaction networks with conservation laws give rise to a singularity of the Jacobian of the entire polynomial system that presents the network and also of some Jacobian matrices of the computed subsystems. To compute the Hopf condition the Jacobian matrices should be transformed into nonsingular matrices. Therefore, we reduce them by computing the Jacobian Jac_i on the reduced manifolds using the method presented in sect. 2.2.

4.3.3. Semi-Algebraic Description of Hopf Bifurcations

We compute the Hopf condition based on the Hurwitz-Hopf criterion. Therefore, we compute the Hurwitz matrix and the Hurwitz determinants Δ_i. The Hopf condition of a subsystem can be expressed in reaction space using the semi-algebraic description shown in [1] by the following first-order formula:

$$\exists x(a_n > 0 \wedge \Delta_{n-1}(j,x) = 0 \wedge \Delta_{n-2}(j,x) > 0 \wedge \cdots \wedge \Delta_1(j,x) > 0) \quad (15)$$

where n denotes the number of species in the reaction network.

Our method then involves the solution of these existentially quantified formulae, which can be computed using general packages for quantifier elimination on real closed fields yielding an answer of true or false, or packages to test for the satisfiability of the existentially quantified formulae yielding an answer of *satisfiable* (sat) or *unsatisfiable* (unsat).

Notice that real quantifier elimination and formula simplification are known to be computationally hard problem [24, 25]; there has been considerable and quite successful research on efficient implementations of these problems during the past decades.

4.4. Integration of Computational Logic Tools

We integrated into our computations the systems listed below, which are all capable of solving formula (15). Because of the modular structure of our approach, we will be able to integrate other packages—either elements of commercial systems or novel developments—easily.

REDLOG[6] [26, 27], which was originally motivated by the efficient implementation of quantifier elimination based on virtual substitution methods [24, 28, 29]. REDLOG also includes CAD and Hermitian quantifier elimination

---

[6] `http://www.redlog.eu/`

[30, 31, 32] for the reals as well as quantifier elimination for various other domains [33] including the integers [34, 35]. The development of REDLOG was initiated in 1992 by one of the authors (T. Sturm) of this paper and continues until today. REDLOG is included in the computer algebra system REDUCE, which is open source.[7] In addition to regular quantifier elimination methods for the reals, REDLOG includes several variants of quantifier elimination. In particular, these variants include *extended quantifier elimination* [36], which additionally yields sample solutions for existential quantifiers, and *positive quantifier elimination* [37, 3], which includes powerful simplification techniques based on the knowledge that all considered variables are restricted to positive values. In chemical systems, the region of interest is the positive cone of the state variables, and the parameters of interest are known also to be positive, positive quantifier elimination is therefore of special importance and will be used for our computations.

QEPCAD [38] implements partial cylindrical algebraic decomposition (CAD). The development of QEPCAD started with the early work of Collins and his collaborators on CAD circa 1973 and continues to this today. QEPCAD is supplemented by another software package called SLFQ for simplifying quantifier-free formulas using CAD. Both QEPCAD and SLFQ are freely available.[8]

The SLFQ system[9] uses QEPCAD as a black box for simplifying quantifier-free formulas. QEPCAD is able to simplify formulae, but its time and space requirements become prohibitive when input formulae are large. SLFQ essentially breaks large input formulae into small pieces, uses QEPCAD to simplify the pieces, and starts a process of combining simplified subformulae and applying QEPCAD to simplify the combined subformulae. Eventually this process produces a simplification of the entire initial formula.

The commercial computer algebra system Mathematica includes an efficient implementation of CAD-based real quantifier elimination by Strzebonski [39, 40], the development of which began circa 2000.

Z3 is a new and efficient SMT solver that is freely available from Microsoft

---

[7] `http://reduce-algebra.sourceforge.net/`

[8] `http://www.usna.edu/Users/cs/qepcad/B/QEPCAD.html`

[9] Available at `http://www.cs.usna.edu/~qepcad/SLFQ/Home.html`

Research[10]. It uses novel algorithms for quantifier instantiation and theory combination [41]. The first external release of Z3 was in 2007.

RSolver[11] is a program for solving quantified inequality constraints. Problems like projecting the solution set of a set of inequality constraints to two dimensions, or the parametric robust stability of linear differential equations can be directly formulated as such constraints.

In our software we integrated all the tools listed above. However, in this paper, we present only results obtained with the freely available tools REDLOG and Z3, which provided the best computation time. REDLOG returns *true* and Z3 returns *sat* if the condition for the occurrence of Hopf bifurcation is satisfied. If the condition is not satisfied, they return *false* and *unsat*, respectively.

4.5. Pseudo-Code of the HoCoQ *algorithm*

Alg. 1 summarizes the steps discussed above and outlines our method *HoCoQ* in an algorithmic fashion.

---

[10] `http://z3.codeplex.com/`

[11] `http://rsolver.sourceforge.net/`

Algorithm 1: *HoCoQ* Method for Computing Hopf Bifurcations in Reaction Space.

```
Input: A chemical reaction network $\mathcal{N}$ with $\dim(\mathcal{N}) = n$.
Output: The algorithm returns a statement concerning the existence
        of a Hopf bifurcation
begin
    R:= false;
    generate the stoichiometric matrix $\mathcal{S}$ and kinetic matrix $\mathcal{K}$ from
    the reaction network
    compute the minimal set $\mathcal{E}$ of the vectors generating the flux cone
    for $d = 1 \dots n$ do
        compute all $d$-faces (subsystems) $\{\mathcal{N}_i\}_i$ of the flux cone
    for each subsystem $\mathcal{N}_i$ do
        compute from $\mathcal{K}$, $\mathcal{S}$ the transformed Jacobian $\mathrm{Jac}_i$ of $\mathcal{N}_i$ in
        terms of convex coordinates $j_i$;
        if $\mathrm{Jac}_i$ is singular then
            compute the reduced manifold of $\mathrm{Jac}_i$ calling the result also
            $\mathrm{Jac}_i$
        compute the characteristic polynomial $\chi_i$ of $\mathrm{Jac}_i$;
        compute the Hurwitz determinants of $\chi_i$;
        compute the Hopf existence condition for $\mathcal{N}_i$;
        generate the first-order existentially quantified formula $\mathcal{F}_i$
        expressing the Hopf existence condition, the constraints on the
        concentrations and the constraints on the cone coordinates;
        reduce and simplify the generated formula $\mathcal{F}_i$
        R:= $R \vee \mathcal{F}_i$
    return R
```

4.6. Computation of Examples using HoCoQ *Method*

We have applied our algorithm *HoCoQ* on various chemical reaction networks that have been discussed in various monographs and for which the existing algorithms for the symbolic computations approach fails. We were able to detect the existence of Hopf bifurcations in some of them, which are listed below. We thereby demonstrate the results provided by REDLOG and Z3.

4.6.1. Example1: Phosphofructokinase reaction

As a first example, we consider the main example used in the hand computation presented in [5]—the phosphofructokinase reaction. There are 3 chemical species and 7 reactions. S_1 denotes the product Fructose-1,6-biphosphate, S_2 denotes the reactant Fructose-6-phosphate, and the extension S_3 stands for another intermediate that is in equilibrium with Fructose-1,6-biphosphate. The network (16) represents the phosphofructokinase reaction.

$$\begin{gathered} 2\mathrm{S}_1 + \mathrm{S}_2 \xrightarrow{k_1} 3\mathrm{S}_1 \\ \mathrm{S}_2 \underset{k_4}{\overset{k_5}{\rightleftharpoons}} 0 \underset{k_3}{\overset{k_2}{\rightleftharpoons}} \mathrm{S}_1 \underset{k_7}{\overset{k_6}{\rightleftharpoons}} \mathrm{S}_3. \end{gathered} \tag{16}$$

This chemical reaction system yields the following stoichiometric matrix $\mathcal{S}_1$ and kinetic matrix $\mathcal{K}_1$:

$$\begin{aligned} \mathcal{S}_1 &= \begin{pmatrix} 1 & 1 & -1 & 0 & 0 & -1 & 1 \\ -1 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{pmatrix} \\ \mathcal{K}_1 &= \begin{pmatrix} 2 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \end{aligned}$$

The flux cone is spanned by the following four vectors (extreme currents):

$$\begin{aligned} \mathcal{E}_1 &= \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}, \\ \mathcal{E}_2 &= \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{pmatrix}, \\ \mathcal{E}_3 &= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}, \\ \mathcal{E}_4 &= \begin{pmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}. \end{aligned}$$

This problem has previously been investigated using its formulation in reaction coordinates in [3]. Using currently available quantifier elimination packages, the problem could not be solved in its parametric form. Only when using existential closure on the parameters could it be shown by successful quantifier eliminations performed in REDLOG that there exist positive parameters for which there exists a Hopf bifurcation fixed point in the positive orthant. When replicating the experiments we found that the situation described in [3] still applies.

The results on the subsystems involving 1-faces, 2-faces, 3-faces, and 4-faces are summarized in Table 1. A Hopf bifurcation can be found using the 1-face $\mathcal{E}_4$ and most of the subsystems extending it in less than one second. While Z3 provides no results for the 4-face $\mathcal{E}_1\mathcal{E}_2\mathcal{E}_3\mathcal{E}_4$ after 10000 seconds computation time, REDLOG requires only a few seconds of computation time to find a Hopf bifurcation fixed point.

Table 1: Computation of Hopf bifurcations in the phosphofructokinase reaction using *HoCoQ* algorithm

Subsystem	Redlog		Z3	
	Result	Time(s)	Result	Time(s)
$\mathcal{E}_1$	false	< 1	unsat	< 1
$\mathcal{E}_2$	false	< 1	unsat	< 1
$\mathcal{E}_3$	false	< 1	unsat	< 1
$\mathcal{E}_4$	true	< 1	sat	< 1
$\mathcal{E}_1\mathcal{E}_2$	false	< 1	unsat	< 1
$\mathcal{E}_1\mathcal{E}_3$	false	< 1	unsat	< 1
$\mathcal{E}_1\mathcal{E}_4$	true	< 1	sat	< 1
$\mathcal{E}_2\mathcal{E}_3$	false	< 1	unsat	< 1
$\mathcal{E}_2\mathcal{E}_4$	true	< 1	sat	< 1
$\mathcal{E}_3\mathcal{E}_4$	true	< 1	sat	< 1
$\mathcal{E}_1\mathcal{E}_2\mathcal{E}_3$	false	< 1	unsat	< 1
$\mathcal{E}_1\mathcal{E}_2\mathcal{E}_4$	true	< 1	sat	< 1
$\mathcal{E}_1\mathcal{E}_3\mathcal{E}_4$	true	1	sat	< 1
$\mathcal{E}_2\mathcal{E}_3\mathcal{E}_4$	true	2.5	sat	< 1
$\mathcal{E}_1\mathcal{E}_2\mathcal{E}_3\mathcal{E}_4$	true	6	no result	> 10000

4.6.2. Example 2: Enzymatic transfer of calcium ions

Our second example is a biochemical model that was investigated in [5]—the enzymatic transfer of calcium ions, Ca^{++}, across cellmembranes. It includes as shown in network (17) six reactions and four species, where S_1

stands for cytosolic Ca^{++}, S$_2$ stands for Ca^{++} in the endoplasmic reticulum, S$_3$ denotes the enzyme catalyzing the transport of Ca^{++} into the endoplasmic reticulum, and S$_4$ denotes the enzyme-substrate complex. This system is autocatalytic insofar as the concentration of cytosolic Ca^{++} stimulates the release of stored Ca^{++} from the endoplasmic reticulum [5].

$$\begin{aligned}
0 &\underset{k_{21}}{\overset{k_{12}}{\rightleftharpoons}} \mathrm{S}_1 \\
\mathrm{S}_1 + \mathrm{S}_2 &\xrightarrow{k_{43}} 2\mathrm{S}_1 \\
\mathrm{S}_1 + \mathrm{S}_3 \underset{k_{65}}{\overset{k_{56}}{\rightleftharpoons}} \mathrm{S}_4 &\xrightarrow{k_{76}} \mathrm{S}_2 + \mathrm{S}_3
\end{aligned} \tag{17}$$

The following stoichiometric matrix $\mathcal{S}_2$ and kinetic matrix $\mathcal{K}_2$ represent the kinetic description of the network (17).

$$\begin{aligned}
\mathcal{S}_2 &= \begin{pmatrix} -1 & 1 & 1 & 1 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & -1 & 1 \\ 0 & 0 & 0 & -1 & 1 & -1 \end{pmatrix} \\
\mathcal{K}_2 &= \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{pmatrix}
\end{aligned}$$

$$\begin{aligned}
\mathcal{E}_1 &= \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}, \\
\mathcal{E}_2 &= \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}, \\
\mathcal{E}_3 &= \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}.
\end{aligned}$$

For this system the Jacobian matrix is singular—therefore, in the classical sense there are no Hopf bifurcations. However, in the reduced system we find that there are Hopf bifurcations—and we can compute them in concentration space as well as using convex coordinates. The results and computation times are summarized in Table 2.

Table 2: Enzymatic transfer of calcium ions: Computation of Hopf bifurcations using *HoCoQ* algorithm

Subsystem	Redlog		Z3	
	Result	Time(s)	Result	Time(s)
$\mathcal{E}_1$	false	< 1	unsat	< 1
$\mathcal{E}_2$	false	< 1	unsat	< 1
$\mathcal{E}_3$	false	< 1	unsat	< 1
$\mathcal{E}_1\mathcal{E}_2$	true	< 1	sat	< 1
$\mathcal{E}_1\mathcal{E}_3$	false	< 1	unsat	< 1
$\mathcal{E}_2\mathcal{E}_3$	false	< 1	unsat	< 1
$\mathcal{E}_1\mathcal{E}_2\mathcal{E}_3$	true	11	no result	> 10000

4.6.3. Example 3: Model of calcium oscillations in the cilia of olfactory sensory neurons

As the next example, we consider the model for calcium oscillations in the cilia of olfactory sensory neurons discussed in [42]. The underlying mechanism of this model is based on direct negative regulation of cyclic nucleotide-gated channels by calcium/calmodulin and does not require any autocatalysis such as calcium-induced calcium release. Reidl et al. presented a mathematical model for this example in [42] and gave predictions for the parameter ranges in which oscillations should be observable. This model contains a fractional exponent ε, as shown in the following differential equations.

$$\begin{aligned} \dot{x} &= k_1 - k_5 x z \\ \dot{y} &= k_2 x - 4k_3 y^2 + 4k_4 z - k_6 y^{\varepsilon} \\ \dot{z} &= k_3 y^2 - k_4 z \end{aligned}$$

The model yields the following stoichiometric matrix $\mathcal{S}_3$ and kinetic matrix $\mathcal{K}_3$:

$$\mathcal{S}_3 = \begin{pmatrix} 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & -4 & 4 & 0 & -1 \\ 0 & 0 & 1 & -1 & 0 & 0 \end{pmatrix}$$

$$\mathcal{K}_3 = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 2 & 0 & 0 & \varepsilon \\ 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

The representative vectors of the flux cone of this model are:

$$\begin{aligned}\mathcal{E}_1 &= \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix},\\ \mathcal{E}_2 &= \begin{pmatrix} 0 & 0 & 1 & 1 & 0 & 0 \end{pmatrix},\\ \mathcal{E}_3 &= \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}.\end{aligned}$$

Table 3: Model of Calcium Oscillations: Computation of Hopf bifurcations using *HoCoQ* algorithm

Subsystem	Redlog		Z3	
	Result	Time(s)	Result	Time(s)
$\mathcal{E}_1$	false	< 1	unsat	< 1
$\mathcal{E}_2$	false	< 1	unsat	< 1
$\mathcal{E}_3$	false	< 1	unsat	< 1
$\mathcal{E}_1\mathcal{E}_2$	false	< 1	unsat	< 1
$\mathcal{E}_1\mathcal{E}_3$	false	< 1	unsat	< 1
$\mathcal{E}_2\mathcal{E}_3$	false	< 1	unsat	< 1
$\mathcal{E}_1\mathcal{E}_2\mathcal{E}_3$	true	< 1	sat	< 1

In concentration space the solution of a quantifier elimination problem is valid only for integer values of the parameter ε; this is because ε appears in the exponent, and the techniques of quantifier elimination over the ordered field of the reals is restricted to polynomials (or rational functions).

However, in the formulation in reaction coordinates the parameter ε appears as a variable with values in the real closed field used in the computations.

Therefore for a given subsystem we cannot ask only whether a Hopf bifurcation fixed point exists, but we can formulate the question with a free parameter ε.

The answer—a quantifier free formula involving ε—gives the condition for ε for which a Hopf bifurcation occurs for the subsystem. When investigating subsystems resulting from 2-faces we found no Hopf bifurcations, but for the parametric question on 3-faces we obtained the following answer in less than 10sec of computation time using a combination of Redlog and qepcad:

$$\varepsilon + 2 > 0 \wedge 4\varepsilon - 1 < 0$$

Thus for $\varepsilon \in (-2, 0.25)$ we have shown that Hopf bifurcation fixed points exist (for suitable reaction constants). Using numerical simulations of this model Reidl et al. [42] could not find Hopf bifurcations for values of the parameter ε larger than approximately 0.05.

5. ***HoCaT*: Algorithm for Computing Hopf Bifurcations using Convex Coordinates and Tropical Geometry**

The algorithmic method *HoCoQ* discussed in Sect 4 enabled us to determine the existence of Hopf bifurcations in various (bio-)chemical reaction networks even for those with conservation laws. For some chemical networks with complex dynamics, however, it remained difficult to process the final obtained quantified formulae with the currently available quantifier elimination packages.

In this section we present an efficient algorithmic approach, called *HoCaT*, which is sketched in Fig. 2. This algorithm uses the basic ideas of the previous algorithm *HoCoQ*, namely stoichiometric network analysis and manifold reduction method for systems with conservation laws. However, when the discussion provided in Sect. 3.1 for a criterion for the occurrence of Hopf bifurcations without requiring empty unstable manifolds is carried over to convex coordinates, the new condition for the existence of Hopf bifurcations is given by $\Delta_{n-1}(j, x) = 0$ only. Solving such single equations enables us to refrain from utilizing quantifier elimination techniques. Instead, the main algorithmic problem is to determine whether a single multivariate polynomial has a zero for positive coordinates.

For this purpose, in Sect. 5.1, we provide heuristics on the basis of the Newton polytope that ensure the existence of positive and negative values of the polynomial for positive coordinates, in Sect. 5.2 we present a summary of the *HoCaT* Algorithm, and in Sect. 5.3 we apply our method to several (bio)chemical reaction networks.

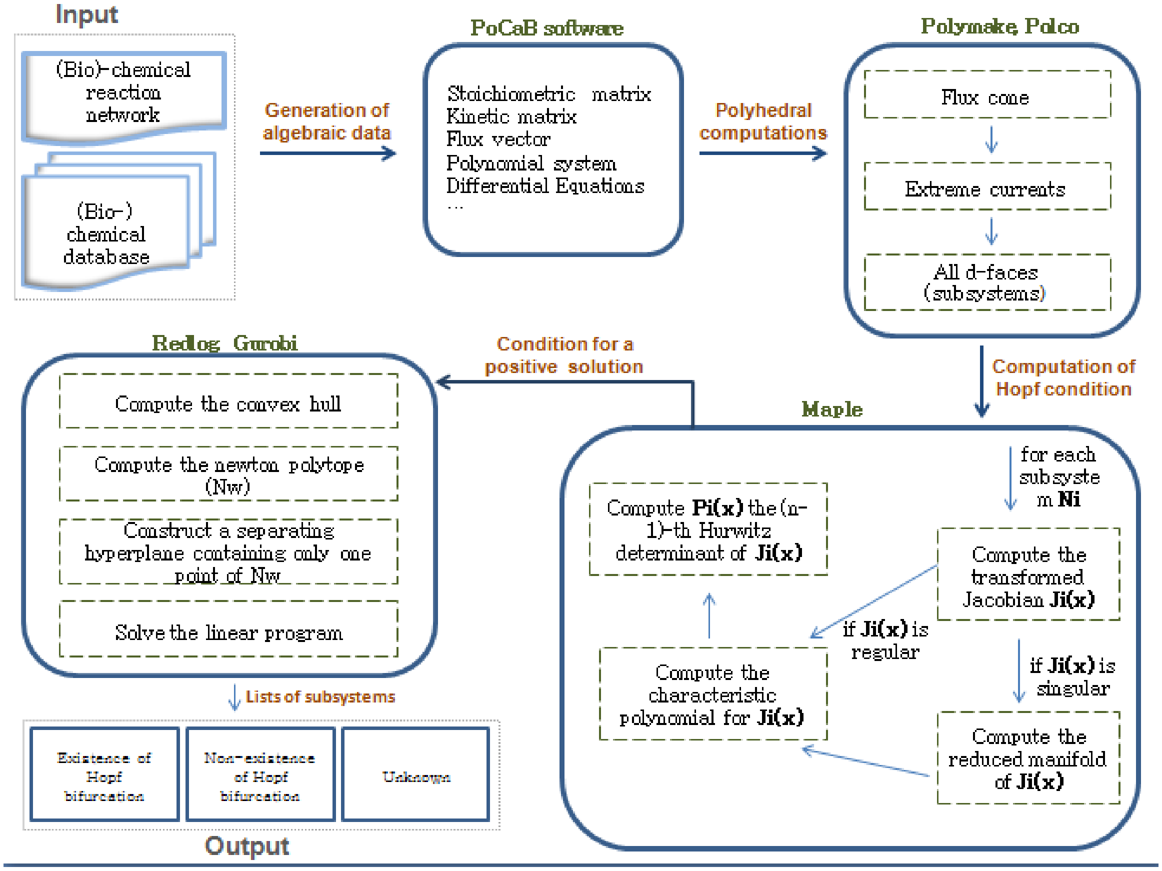

Figure 2:

5.1. *Sufficient Conditions for a Positive Solution of a Single Multivariate Polynomial Equation*

The method discussed in this section is summarized in an algorithmic way in Alg. 2, which uses Alg. 3 as a subalgorithm.

Given $f \in \mathbb{Z}[x_1, \ldots, x_m]$, our goal is to heuristically certify the existence of at least one zero $(z_1, \ldots, z_m) \in \,]0, \infty[^m$ for which all coordinates are strictly positive. To start with, we evaluate $f(1, \ldots, 1) = f_1 \in \mathbb{R}$. If $f_1 = 0$, then we are done. If $f_1 < 0$, then by the intermediate value theorem, it is sufficient to find $p \in \,]0, \infty[^m$ such that $f(p) > 0$. Similarly, if $f_1 > 0$ it is sufficient to find $p \in \,]0, \infty[^m$ such that $(-f)(p) > 0$. This algorithmically reduces our original problem to finding, for given $g \in \mathbb{Z}[x_1, \ldots, x_m]$, at least one $p \in \,]0, \infty[^m$ such that $g(p) = f_2 > 0$.

Algorithm 2: pzerop

Input: $f \in \mathbb{Z}[x_1, \dots, x_m]$

Output: One of the following:

(A) 1, which means that $f(1, \dots, 1) = 0$.

(B) (π, ν), where $\nu = (p, f(p))$ and $\pi = (q, f(q))$ for p, $q \in \,]0, \infty[^m$, which means that $f(p) < 0 < f(q)$. Then there is a zero on $]0, \infty[^m$ by the intermediate value theorem.

(C) $+$, which means that f has been identified as positive definite on $]0, \infty[^m$. Then there is no zero on $]0, \infty[^m$.

(D) $-$, which means that f has been identified as negative definite on $]0, \infty[^m$. Then there is no zero on $]0, \infty[^m$.

(E) $\perp$, which means that this incomplete procedure failed.

```
begin
    $f_1 := f(1, \dots, 1)$
    if $f_1 = 0$ then
        return 1
    else if $f_1 < 0$ then
        $\pi :=$ pzerop$_1(f)$
        $\nu := ((1, \dots, 1), f_1)$
        if $\pi \in \{\perp, -\}$ then
            return $\pi$
        else
            return $(\nu, \pi)$
    else
        $\pi := ((1, \dots, 1), f_1)$
        $\nu' :=$ pzerop$_1(-f)$
        if $\nu' = \perp$ then
            return $\perp$
        else if $\nu' = -$ then
            return $+$
        else
            $(p, f(p)) := \nu'$
            $\nu := (p, -f(p))$
            return $(\nu, \pi)$
```

Algorithm 3: pzerop_1

Input: $f \in \mathbb{Z}[x_1, \dots, x_m]$

Output: One of the following:

(A) $\pi = (q, f(q))$, where $q \in]0, \infty[^m$ with $0 < f(q)$.

(B) $-$, which means that f has been identified as negative definite on $]0, \infty[^m$. Then there is no zero on $]0, \infty[^m$.

(C) $\perp$, which means that this incomplete procedure failed.

```
begin
    $F^+ := \{\, d \in \mathrm{frame}(f) \mid \mathrm{sgn}(d) = 1 \,\}$
    if $F^+ = \emptyset$ then
        return $-$
    foreach $(d_1, \dots, d_m) \in F^+$ do
        $L := \{d_1 n_1 + \dots + d_m n_m - c = 0\}$
        foreach $(e_1, \dots, e_m) \in \mathrm{frame}(f) \setminus F^+$ do
            $L := L \cup \{e_1 n_1 + \dots + e_m n_m - c \leq -1\}$
        if $L$ is feasible with solution $(n_1, \dots, n_m, c) \in \mathbb{Q}^{m+1}$ then
            $g :=$ the principal denominator of $n_1, \dots, n_m$
            $(N_1, \dots, N_m) := (g n_1, \dots, g n_m) \in \mathbb{Z}^m$
            $\bar{f} := f[x_1 \leftarrow \omega^{N_1}, \dots, x_m \leftarrow \omega^{N_m}] \in \mathbb{Z}(\omega)$
            assert $\mathrm{lc}(\bar{f}) > 0$ when using non-exact arithmetic in the LP solver
            $k := \min\{\, k \in \mathbb{N} \mid \bar{f}(2^k) > 0 \,\}$
            return $((2^{kN1}, \dots, 2^{kN_m}), \bar{f}(2^k))$
    return $\perp$
```

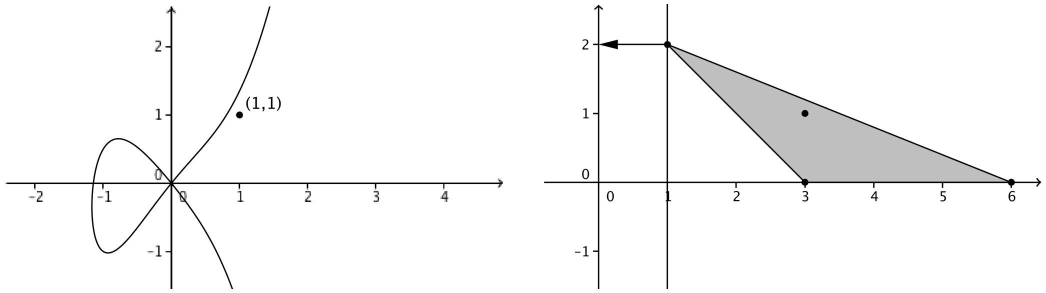

Figure 3: We consider $g_0 = -2x_1^6 + x_1^3x_2 - 3x_1^3 + 2x_1x_2^2$. The left hand shows the variety $g_0 = 0$. The right hand side shows the frame, the Newton polytope, and a separating hyperplane for the positive monomial $2x_1x_2^2$ with its normal vector.

We will accompany the description of our method with the example $g_0 = -2x_1^6 + x_1^3x_2 - 3x_1^3 + 2x_1x_2^2 \in \mathbb{Z}[x_1, x_2]$. Fig. 3 shows an implicit plot of this polynomial. In addition to its variety, g_0 has three sign invariant regions, one bounded one and two unbounded ones. One of the unbounded regions contains our initial test point $(1, 1)$, for which we find that $g_0(1, 1) = -2 < 0$. Therefore our goal is to find one point $p \in \left]0, \infty\right[^2$ such that $g_0(p) > 0$.

In the spirit of tropical geometry—and we refer to [43] as a standard reference with respect to its application for polynomial system solving—we take an abstract view of

$$g = \sum_{d \in D} a_d x^d := \sum_{(d_1,\dots,d_m) \in D} a_{d_1,\dots,d_m} x_1^{d_1} \cdots x_m^{d_m}$$

as the set $\operatorname{frame}(g) = D \subseteq \mathbb{N}^m$ of all exponent vectors of the contained monomials. For each $d \in \operatorname{frame}(g)$, we are able to determine $\operatorname{sgn}(d) := \operatorname{sgn}(a_d) \in \{-1, 1\}$. The set of vertices of the convex hull of the frame is called the *Newton polytope* $\operatorname{newton}(g) \subseteq \operatorname{frame}(g)$. In fact, the existence of at least one point $d^* \in \operatorname{newton}(g)$ with $\operatorname{sgn}(d^*) = 1$ is sufficient for the existence of $p \in \left]0, \infty\right[^m$ with $g(p) > 0$.

In our example, we have $\operatorname{frame}(g_0) = \{(6, 0), (3, 1), (3, 0), (1, 2)\}$ and $\operatorname{newton}(g_0) = \{(6, 0), (3, 0), (1, 2)\} \subseteq \operatorname{frame}(g_0)$. We are particularly interested in $d^* = (d_1^*, d_2^*) = (1, 2)$, which is the only point that has a positive sign as it corresponds to the monomial $2x_1x_2^2$.

To understand this sufficient condition, we are now going to compute from d^* and g a suitable point p. We construct a hyperplane $H : n^Tx = c$ containing d^* such that all other points of $\operatorname{newton}(g)$ are not contained in H

and lie on the same side of H. We choose the normal vector $n \in \mathbb{R}^m$ such that it points into the halfspace that does not contain the Newton polytope. The vector $c \in \mathbb{R}^m$ is such that $\frac{c}{|n|}$ is the offset of H from the origin in the direction of n.

In our example H is the line $x = 1$ given by $n = (-1, 0)$ and $c = -1$. Fig. 3 depicts the situation.

Considering the standard scalar product $\langle \cdot | \cdot \rangle$, it turns out that generally $\langle n | d^* \rangle = \max\{\, \langle n | d \rangle \mid d \in \operatorname{newton}(g)\,\}$, and that this maximum is strict. For the monomials of the original polynomial $g = \sum_{d \in D} a_d x^d$ and a new variable ω this observation translates via the following identity:

$$\bar{g} = g[x \leftarrow \omega^n] = \sum_{d \in D} a_d \omega^{\langle n | d \rangle} \in \mathbb{Z}(\omega).$$

Therefore, plugging a number $\beta \in \mathbb{R}$ into $\bar{g}$ corresponds to plugging the point $\beta^n \in \mathbb{R}^m$ into g and from our identity, we see that in $\bar{g}$ the exponent $\langle n | d^* \rangle$ corresponding to our chosen point $d^* \in \operatorname{newton}(g)$ dominates all other exponents, so for large β, the sign of $\bar{g}(\beta) = g(\beta^n)$ equals the positive sign of the coefficient a_{d^*} of the corresponding monomial. To find a suitable β, we successively compute $\bar{g}(2^k)$ for increasing $k \in \mathbb{N}$.

In our example we obtain $\bar{g} = 2\omega^{-1} - 2\omega^{-3} - 2\omega^{-6}$, and we obtain $\bar{g}(1) = -2$, but already $\bar{g}(2) = \frac{23}{32} > 0$. In terms of the original g this corresponds to plugging in the point $p = 2^{(-1,0)} = \left(\frac{1}{2}, 1\right) \in \,]0, \infty[^2$.

It remains to be clarified how to construct the hyperplane H. Consider $\operatorname{frame}(g) = \{\, (d_{i1}, \dots, d_{im}) \in \mathbb{N}^m \mid i \in \{1, \dots, k\}\,\}$. If $\operatorname{sgn}(d) = -1$ for all $d \in \operatorname{frame}(g)$, then we know that g is negative definite on $]0, \infty[^m$. Otherwise, assume, without loss of generality, that $\operatorname{sgn}(d_{11}, \dots, d_{1m}) = 1$. We write down the following linear program:

$$\begin{pmatrix} d_{11} & \dots & d_{1m} & -1 \end{pmatrix} \cdot \begin{pmatrix} n_1 \\ \vdots \\ n_m \\ c \end{pmatrix} = 0, \qquad \begin{pmatrix} d_{21} & \dots & d_{2m} & -1 \\ \vdots & \ddots & \vdots & \vdots \\ d_{k1} & \dots & d_{km} & -1 \end{pmatrix} \cdot \begin{pmatrix} n_1 \\ \vdots \\ n_m \\ c \end{pmatrix} \leq -1.$$

This is feasible if and only if $(d_{11}, \dots, d_{1m}) \in \operatorname{newton}(g)$. In the negative case, we know that $(d_{11}, \dots, d_{1m}) \in \operatorname{frame}(g) \setminus \operatorname{newton}(g)$, and we iterate with another $d \in \operatorname{frame}(g)$ with $\operatorname{sgn}(d) = 1$. If we finally fail on all such d, then our incomplete algorithm has failed. In the positive case, the solution provides a normal vector $n = (n_1, \dots, n_m)$ and the offset c for a suitable hyperplane H. Our linear program can be solved with any standard LP solver. For our purposes here, we have used Gurobi[12]; the dual simplex of GLPSOL[13] also performs quite similarly on the input considered here.

For our example $g_0 = -2x_1^6 + x_1^3 x_2 - 3x_1^3 + 2x_1 x_2^2$, we generate the linear program

$$\begin{aligned} n_1 + 2n_2 - c &= 0 \\ 6n_1 - c &\leq -1 \\ 3n_1 + n_2 - c &\leq -1 \\ 3n_1 - c &\leq -1, \end{aligned}$$

for which Gurobi computes the solution $n = (n_1, n_2) = (-0.5, 0)$, $c = -0.5$. Notice that the solutions obtained from the LP solvers are typically floats, which we lift to integer vectors by suitable rounding and GCD computations.

Note that we do not explicitly construct the convex hull newton(g) of the frame(g) although there are advanced algorithms and implementations like QuickHull[14] available for this purpose. Instead we favour a linear programming approach for several reasons. Firstly, we do not require that comprehensive information, instead, it is sufficient to find one vertex of the covex hull that has a positive sign. Secondly, for the application dicussed here, it turns out that there typically exist only a few (approximately 10%) such candidate points. Finally, it is known that for high dimensions, the subset of frame(g) establishing vertices of the convex hull gets comparatively large. Practical experiments using QuickHull on our data support these theoretical considerations.

5.2. Summarizing the HoCaT *Algorithm*

The steps involving the pre-precessing procedure, polyhedral computation, and computation of the reduced Jacobian that we previously used for the *HoCoQ* method and discussed in Sect. 4 remain the same. After computing the characteristic polynomial of the Jacobian matrix of each subsystem, we compute the $(n-1)^{\text{th}}$ Hurwitz determinant of the characteristic polynomial, and we apply Alg. 2 to check for positive solutions of the respective

---

[12] `www.gurobi.com`

[13] `www.gnu.org/software/glpk`

[14] `www.qhull.org`

polynomial equations $\Delta_{n-1}(j, x) = 0$. Alg. 4 outlines our efficient approach in an algorithmic fashion.

Algorithm 4: *HoCaT* Method for Computing Hopf Bifurcations in Reaction Space.

Input: A chemical reaction network $\mathcal{N}$ with $\dim(\mathcal{N}) = n$.

Output: (L_t, L_f, L_u), which are defined as follows: L_t is a list of subsystems containing a Hopf bifurcation, L_f is a list of subsystems in which the occurrence of Hopf bifurcations is excluded, and L_u is a list of subsystems for which the incomplete sub-procedure pzerop fails.

```
begin
    $L_t = \emptyset$
    $L_f = \emptyset$
    $L_u = \emptyset$
    generate the stoichiometric matrix $\mathcal{S}$ and the kinetic matrix $\mathcal{K}$ of $\mathcal{N}$
    compute the minimal set $\mathcal{E}$ of the vectors generating the flux cone
    for $d = 1 \ldots n$ do
        compute all $d$-faces (subsystems) $\{\mathcal{N}_i\}_i$ of the flux cone
    for each subsystem $\mathcal{N}_i$ do
        compute from $\mathcal{K}$, $\mathcal{S}$ the transformed Jacobian $\mathrm{Jac}_i$ of $\mathcal{N}_i$ in terms of convex coordinates $j_i$
        if $\mathrm{Jac}_i$ is singular then
            compute the reduced manifold of $\mathrm{Jac}_i$ calling the result also $\mathrm{Jac}_i$
        compute the characteristic polynomial $\chi_i$ of $\mathrm{Jac}_i$
        compute the $(n-1)^{\text{th}}$ Hurwitz determinant $\Delta_{n-1}$ of $\chi_i$
        compute $\mathcal{F}_i :=$ pzerop$(\Delta_{n-1}(j, x))$ using Algorithm 2
        if $\mathcal{F}_i = 1$ or $\mathcal{F}_i$ is of the form $(\pi, \nu)$ then
            $L_t := L_t \cup \{\mathcal{N}_i\}$
        else if $\mathcal{F}_i = +$ or $\mathcal{F}_i = -$ then
            $L_f := L_f \cup \{\mathcal{N}_i\}$
        else if $\mathcal{F}_i = \perp$ then
            $L_u := L_u \cup \{\mathcal{N}_i\}$
    return $(L_t, L_f, L_u)$
```

5.3. Computation of Examples using the HoCaT *Method*

In this section, we will demonstrate the efficiency of our novel approach *HoCaT* by analyzing several chemical networks with different dimensions. We will first compute Hopf bifurcations in the reaction networks already discussed in 4.6 using the *HoCaT* method. We will also wish to discuss and detect the occurrence of Hopf bifurcations in higher dimensional networks. We will therefore apply our new method to the 5-dimensional system of electro-oxidation of methanol presented in [44], to the well-known 9-dimensional example *MAPK* discussed in [45] and in other papers and to the 22-dimensional network modeling the control of DNA replication in fission yeast [46]. We will also compute Hopf bifurcations in the family of original models that describe a gene regulated by a polymer of its own protein, which are well-studied using the quasi-steady state approximation method in [47].

5.3.1. Example1: Phosphofructokinase reaction

As the first example we consider the phosphofructokinase reaction discussed in 4.6.1.

Table 4: Computation of Hopf bifurcations in the phosphofructokinase reaction using *HoCaT* algorithm

Subsystem	Result	Time
$\mathcal{E}_1$	unsat	< 1
$\mathcal{E}_2$	unsat	< 1
$\mathcal{E}_3$	unsat	< 1
$\mathcal{E}_4$	sat	< 1
$\mathcal{E}_1\mathcal{E}_2$	unsat	< 1
$\mathcal{E}_1\mathcal{E}_3$	unsat	< 1
$\mathcal{E}_1\mathcal{E}_4$	sat	< 1
$\mathcal{E}_2\mathcal{E}_3$	unsat	< 1
$\mathcal{E}_2\mathcal{E}_4$	sat	< 1
$\mathcal{E}_3\mathcal{E}_4$	sat	< 1
$\mathcal{E}_1\mathcal{E}_2\mathcal{E}_3$	unsat	< 1
$\mathcal{E}_1\mathcal{E}_2\mathcal{E}_4$	sat	< 1
$\mathcal{E}_1\mathcal{E}_3\mathcal{E}_4$	sat	< 1
$\mathcal{E}_2\mathcal{E}_3\mathcal{E}_4$	sat	< 1
$\mathcal{E}_1\mathcal{E}_2\mathcal{E}_3\mathcal{E}_4$	sat	< 1

As shown in Table 4, using the *HoCaT* algorithm, we were able to detect the occurrence of Hopf bifurcations in less than 1 second for all computed faces. For comparison, in the case of 4-faces the *HoCoQ* method requires 6 seconds.

5.3.2. Example 2: Enzymatic transfer of calcium ions

The computation of Hopf bifurcations in the model of the enzymatic transfer of calcium ions discussed in Sect. 4.6.2 using the *HoCaT* method yields the results presented in Table 5.

Table 5: Computation of Hopf bifurcations in the model 'Enzymatic transfer of calcium ions' using *HoCaT* algorithm

Subsystem	Result	Time(s)
$\mathcal{E}_1$	unsat	< 1
$\mathcal{E}_2$	unsat	< 1
$\mathcal{E}_3$	unsat	< 1
$\mathcal{E}_1\mathcal{E}_2$	sat	< 1
$\mathcal{E}_1\mathcal{E}_3$	unsat	< 1
$\mathcal{E}_2\mathcal{E}_3$	unsat	< 1
$\mathcal{E}_1\mathcal{E}_2\mathcal{E}_3$	sat	< 1

While the *HoCoQ* method requires 11 seconds of computation time for the 3-faces, the *HoCaT* method needs less than 1 second.

5.3.3. Example 3: Model of calcium oscillations in the cilia of olfactory sensory neurons

Table 6 shows the results of computing Hopf bifurcations in the model calcium oscillations in the cilia of olfactory sensory neurons discussed in Sect. 4.6.3.

Table 6: Model for Calcium Oscillations in the cilia of olfactory sensory neurons: Computation of Hopf bifurcations using *HoCaT* algorithm

Subsystem	Result	Time
$\mathcal{E}_1$	unsat	< 1
$\mathcal{E}_2$	unsat	< 1
$\mathcal{E}_3$	unsat	< 1
$\mathcal{E}_1\mathcal{E}_2$	unsat	< 1
$\mathcal{E}_1\mathcal{E}_3$	unsat	< 1
$\mathcal{E}_2\mathcal{E}_3$	unsat	< 1
$\mathcal{E}_1\mathcal{E}_2\mathcal{E}_3$	sat	< 1

5.3.4. Example 4: Electro-oxidation of methanol

Sauerbrei et al. [44] developed a model for a mechanism for the kinetic instabilities observed in the galvanostatic electro-oxidation of methanol. To keep the model simple, they neglected the side reactions and assumed that the whole process runs through HCO and CO. They then proposed the reaction network (18), which involves five essential species (nonessential species are enclosed in square brackets).

$$\begin{aligned}
[\mathrm{MeOH_b}] + 3* &\xrightarrow{k_1,\Phi} \mathrm{HCO} + \left[3\mathrm{H}^+\right] + 3\mathrm{e}^- \\
\mathrm{HCO} &\xrightarrow{k_2} \mathrm{CO} + 2* + \left[\mathrm{H}^+\right] + (\mathrm{e}^-) \\
[\mathrm{H_2O}] + * &\xrightarrow{k_3,\Phi} \mathrm{O} + \left[2\mathrm{H}^+\right] + (2\mathrm{e}^-) \\
\mathrm{CO} + \mathrm{O} &\xrightarrow{k_4} 2* + [\mathrm{CO_2}] \\
\left[2\mathrm{H}^+\right] + (2\mathrm{e}^-) + \mathrm{O} &\xrightarrow{k_5,-\Phi} * + [\mathrm{H_2O}].
\end{aligned} \tag{18}$$

Electrochemical reactions depend exponentially on the double layer potential Φ, so there is no power law kinetics initially. The system can, however, be transformed into power laws forms by using $x_3 = e^{k_6\Phi}$ as a variable. By performing certain substitutions as shown in [44] the model yields the following differential equations and matrices. Note that this model has a negative exponent.

$$
\begin{aligned}
\dot{x}_1 &= -3k_1x_1^2x_3 + 2k_2x_4 - k_3x_1x_3 + 2k_4x_2x_5 + k_5x_2x_3^{-1} \\
\dot{x}_2 &= k_3x_1x_3 - k_4x_2x_5 - k_5x_2x_3^{-1} \\
\dot{x}_3 &= k_6k_7x_3 - k_1k_6x_1^2x_3^2 \\
\dot{x}_4 &= k_1x_1^2x_3 - k_2x_4 \\
\dot{x}_5 &= k_2x_4 - k_4k_2x_5
\end{aligned} \tag{19}
$$

$$
\mathcal{S}_4 = \begin{pmatrix} -3 & 2 & -1 & 2 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & 0 \end{pmatrix}
$$

$$
\mathcal{K}_4 = \begin{pmatrix} 2 & 0 & 1 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & -1 & 2 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}
$$

The stoichiometric matrix $\mathcal{S}_4$ yields the following extreme currents:

$$
\begin{aligned}
\mathcal{E}_1 &= \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 & 0 \end{pmatrix}, \\
\mathcal{E}_2 &= \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}, \\
\mathcal{E}_3 &= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}.
\end{aligned}
$$

We applied the *HoCaT* algorithm to all possible faces and we were able to find the occurrence of Hopf bifurcations in the 2-faces $\mathcal{E}_2\mathcal{E}_3$ and the 3-faces $\mathcal{E}_1\mathcal{E}_2\mathcal{E}_3$ as shown in Table 8.

Table 7: Computation of Hopf bifurcations in ‘electro-oxidation of methanol’ using *HoCaT* algorithm

Subsystem	Result	Time(s)
$\mathcal{E}_1$	unsat	< 1
$\mathcal{E}_2$	unsat	< 1
$\mathcal{E}_3$	unsat	< 1
$\mathcal{E}_1\mathcal{E}_2$	unsat	< 1
$\mathcal{E}_1\mathcal{E}_3$	unsat	< 1
$\mathcal{E}_2\mathcal{E}_3$	sat	< 1
$\mathcal{E}_1\mathcal{E}_2\mathcal{E}_3$	sat	< 1

5.3.5. Example 5: Methylene Blue Oscillator System

As the next example we apply the *HoCaT* method on the well-known complex autocatalytic *methylen blue oscillator (MBO)* system. We attempted to compute Hopf bifurcations in all subsystems of this model that involve 2-faces and 3-faces using our original *HoCoQ* approach, but the generated quantified formulae could not be solved by quantifier elimination, even with main memory of up to 500 GB and computation times of up to one week. The *MBO* model is described by the reaction network (20):

$$
\begin{aligned}
\mathrm{MB^+ + HS^-} &\longrightarrow \mathrm{MB + HS}\\
\mathrm{H_2O + MB + HS^-} &\longrightarrow \mathrm{MBH + HS + OH^-}\\
\mathrm{HS + OH^- + MB^+} &\longrightarrow \mathrm{MB + S + H_2O}\\
\mathrm{H_2O + 2MB} &\longrightarrow \mathrm{MB^+ + MBH + OH^-}\\
\mathrm{HS^- + O_2} &\longrightarrow \mathrm{HS + O_2^-}\\
\mathrm{HS + O_2 + OH^-} &\longrightarrow \mathrm{O_2^- + S + H_2O}\\
\mathrm{2H_2O + HS^- + O_2^-} &\longrightarrow \mathrm{H_2O_2 + HS + 2OH^-}\\
\mathrm{O_2^- + HS + H_2O} &\longrightarrow \mathrm{H_2O_2 + S + H_2O}\\
\mathrm{H_2O_2 + 2HS^-} &\longrightarrow \mathrm{2HS + 2OH^-}\\
\mathrm{MB + O_2} &\longrightarrow \mathrm{MB^+ + O_2^-}\\
\mathrm{HS^- + MB + H_2O_2} &\longrightarrow \mathrm{MB^+ + HS + 2OH^-}\\
\mathrm{OH^- + 2HS} &\longrightarrow \mathrm{HS^- + S + H_2O}\\
\mathrm{MB + HS} &\longrightarrow \mathrm{MBH + S}\\
\mathrm{H_2O + MBH + O_2^-} &\longrightarrow \mathrm{MB + H_2O_2 + OH^-}\\
&\longrightarrow \mathrm{O_2} \qquad (20)
\end{aligned}
$$

The MBO reaction system contains 15 reactions and 11 species O_2, O_2^-, HS, MB^+, MB, MBH, HS^-, OH^-, S, and H_2O_2. It may be reduced to a six dimensional system by considering only the essential species MB, MB^+, HS, MBH, O_2, and O_2^-. The pre-processing step of our algorithm yields the following two matrices describing the reaction laws: stoichiometric matrix $\mathcal{S}$

and kinetic matrix $\mathcal{K}$.

$$\mathcal{S}_5 = \begin{pmatrix} 1 & -1 & 1 & -2 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & -1 & 1 & 0 \\ -1 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & -1 & 0 & 1 & -1 & 1 & -1 & 2 & 0 & 1 & -2 & -1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \end{pmatrix}$$

$$\mathcal{K}_5 = \begin{pmatrix} 0 & 1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}.$$

The flux cone of this model is spanned by 28 extreme currents. There are 187 subsystems of 2-faces and 549 subsystems of 3-faces. Using our new approach *HoCaT* we were able to detect Hopf bifurcations in the extreme current $\mathcal{E} = (0 \;\; 0 \;\; 1 \;\; 0 \;\; 0 \;\; 0 \;\; 0 \;\; 0 \;\; 1 \;\; 1 \;\; 0 \;\; 0 \;\; 1 \;\; 1 \;\; 1)$ and in 105 cases of 2-faces. The following table summarize the results.

Table 8: Results of the computation of Hopf bifurcations in 1-face and 2- faces using *HoCaT*

Subsystems	Number of cases	Satisfied	Unsatisfied	Unknown
1-face	28	1	27	0
2-faces	187	105	66	15

All computations on a single instance required at most 350 milliseconds of CPU time.

Recall that a positive answer for at least one of the cases guarantees the existence of a Hopf bifurcation for the original system in spite of the fact that there are cases without a definite answer.

*5.3.6. Example 6: mitogen-activated protein kinase (*MAPK*)*

We next consider a well-studied model in cell biology that describes the activity of mitogen-activated protein kinase (*MAPK*). This model is known to exhibit bistability, namely it has up to two stable equilibria, if the parameter vector is located in an appropriate region of parameter space [48, 49].

Conradi et al. also studied this model in [45] and mentioned that finding these regions, for example by using numerical tools like bifurcation analysis, is a non-trivial task as it amounts to searching the entire parameter space. They show that for a model of a single layer of a *MAPK* cascade it is possible to derive analytical descriptions of these regions through the use of mass action kinetics. As an example, we compute Hopf bifurcations in the extensively studied 9-dimensional network (21) that belongs to a family of network structures that has been postulated as a model for a single layer of a *MAPK* cascade. We use here the same notations as in [45]. We use A as a placeholder for either *MAPKK* or a *MAPK*, E_1 for mono-phosphorylated *MAPKKK* or double-phosphorylated *MAPKK*, and E_2 for *MAPKK*'ase or *MAPK*'ase.

$$\begin{gathered}
\mathrm{A}+\mathrm{E}_1 \underset{k_2}{\overset{k_1}{\rightleftharpoons}} \mathrm{AE}_1 \xrightarrow{k_3} \mathrm{A_p}+\mathrm{E}_1 \underset{k_5}{\overset{k_4}{\rightleftharpoons}} \mathrm{A_pE}_1 \xrightarrow{k_6} \mathrm{A_{pp}}+\mathrm{E}_1, \\
\mathrm{A_{pp}}+\mathrm{E2} \underset{k_8}{\overset{k_7}{\rightleftharpoons}} \mathrm{A_{pp}E}_2 \xrightarrow{k_9} \mathrm{A_p}+\mathrm{E}_2 \underset{k_{11}}{\overset{k_{10}}{\rightleftharpoons}} \mathrm{A_pE}_2 \xrightarrow{k_{12}} \mathrm{A}+\mathrm{E}_2.
\end{gathered} \tag{21}$$

The *MAPK* network (21) involves twelve reactions and nine species, A, E_1, AE_1, A_p, A_pE_1, A_{pp}, $E2$, $A_{pp}E_2$, and A_pE_2. The appropriate stoichiometric matrix $\mathcal{S}_6$ and kinetic matrix $\mathcal{K}_6$ are as follows:

$$\mathcal{S}_6 = \begin{pmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ -1 & 1 & 1 & -1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 1 & 0 & 0 & 0 & 1 & -1 & 1 & 0 \\ 0 & 0 & 0 & 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & -1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 \end{pmatrix}$$

$$\mathcal{K}_6 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}.$$

The flux cone of the *MAPK* network is spanned by the following six vectors of extreme currents:

$$\begin{aligned} \mathcal{E}_1 &= \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \\ \mathcal{E}_2 &= \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \\ \mathcal{E}_3 &= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}, \\ \mathcal{E}_4 &= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}, \\ \mathcal{E}_5 &= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{pmatrix}, \\ \mathcal{E}_6 &= \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}. \end{aligned}$$

Although it is difficult to compute Hopf bifurcations in the *MAPK* networks, we we were able to detect the occurrence of a Hopf bifurcation using our algorithm in the subsystem generated by the 3-face of $\mathcal{E}_1$, $\mathcal{E}_5$, and $\mathcal{E}_6$ in 34 seconds of computation time. For all the subsystems generated by 1-faces or by 2-faces we could exclude them.

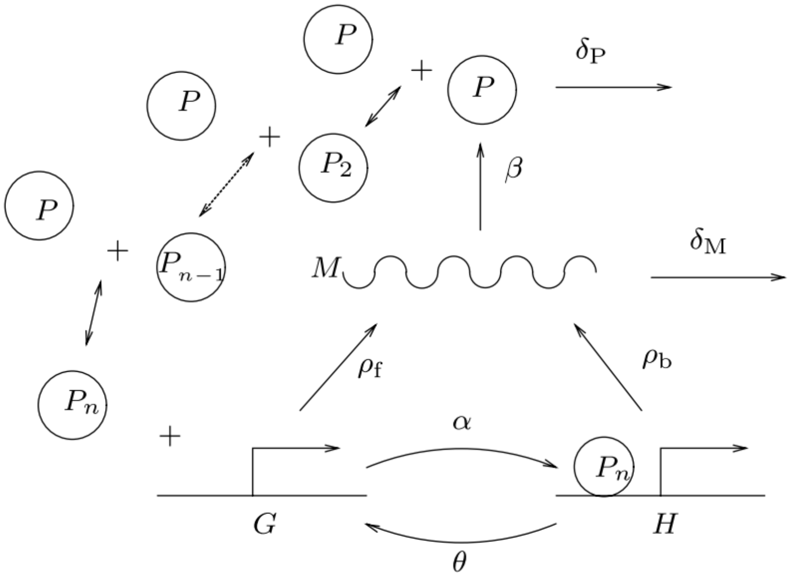

Figure 4: A gene regulated by a polymer of its protein [47]

5.3.7. Example 7: Models of Genetic Circuits

Boulier et al. [47] studied the use of a rigorous quasi-steady state approximation method to determine the existence of Hopf bifurcations in a family of models describing a gene regulated by a polymer of its own protein. This family of models is dependent on an integer parameter n that expresses the number of polymerizations and on featuring a negative feedback loop. The model sketched in Fig. 4 describes a single gene regulated by a polymer that is obtained by combining a protein n times. The variables G and H represent the state of the gene. The mRNA concentration and the concentration of the protein translated from the mRNA are represented by M and P, respectively. The n types of polymers of P are denoted by $\mathrm{G} = \mathrm{P}_1, \mathrm{P}_2, \ldots, \mathrm{P}_\mathrm{n}$. Greek letters represent parameters [47].

The family of models yields the following reaction laws.

$$\mathrm{G} + \mathrm{P_n} \underset{\theta}{\overset{\alpha}{\rightleftharpoons}} \mathrm{H}, \quad \mathrm{G} \xrightarrow{\rho_f} \mathrm{G} + \mathrm{M}, \quad \mathrm{H} \xrightarrow{\rho_b} \mathrm{H} + \mathrm{M},$$

$$\mathrm{M} \xrightarrow{\beta} \mathrm{M} + \mathrm{P}, \quad \mathrm{M} \xrightarrow{\delta_M} \emptyset, \quad \mathrm{P} \xrightarrow{\delta_P} \emptyset, \quad \mathrm{P_i} + \mathrm{P} \underset{k_i^-}{\overset{k_i^+}{\rightleftharpoons}} \mathrm{P_{i+1}} \quad (1 \le i \le n-1). \tag{22}$$

Applying a rigorous quasi-steady state approximation and several rescalings of the variables and parameters yields the following family of ordinary differential equations [47]:

$$\begin{aligned}
\dot{G(t)} &= \theta(\gamma_0 - G(t) - G(t)P(t)^n), \\
\dot{P(t)} &= n\alpha(\gamma_0 - G(t) - G(t)P(t)^n) + \delta(M(t) - P(t)), \\
\dot{M(t)} &= \lambda_1 G(t) + \gamma_0 \mu - M(t), \qquad (23)
\end{aligned}$$

where n is a natural number.

Sturm et al. [37, 3] also analyzed the existence of Hopf bifurcations in the 3-dimensional steady-state approximation of the models shown in (23). They computed its occurrence in concentration space up to $n = 10$ and they found the absence of Hopf bifurcations in the family of models for $n \leq 8$ and its existence for $n \geq 9$.

We investigated the existence of Hopf bifurcations in the original family of models for $n = 2, \ldots, 10$, wherein we also considered the fast reactions. Each model thus involved then $3 + n$ species and yields corresponding to the stoichiometric matrix and kinetic matrix. The number of the vectors that span the flux cone is dependent on the parameter n, which expresses the number of polymerizations and effect that for increasing n. We applied our *HoCaT* method to all 9 models and in contrast to the results of the quasi-steady state method, we were able to detect the existence of Hopf bifurcations for $n \geq 3$ and its absence for $n = 2$.

To elucidate the cause of the occurrence of Hopf bifurcations for $n \geq 3$ in the original state of the systems, we carefully analyzed the results of the system with $n = 3$ polymerizations. The system yields the following stoichiometric and kinetic matrices:

$$
\mathcal{S}_7 = \begin{pmatrix}
-1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & -1 & -2 & 2 & -1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 \\
-1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1
\end{pmatrix}
$$

$$
\mathcal{K}_7 = \begin{pmatrix}
1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}.
$$

The following six extreme currents represent the flux cone:

$$
\begin{aligned}
\mathcal{E}_1 &= \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}, \\
\mathcal{E}_2 &= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}, \\
\mathcal{E}_3 &= \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \\
\mathcal{E}_4 &= \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \\
\mathcal{E}_5 &= \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \\
\mathcal{E}_6 &= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{pmatrix}.
\end{aligned}
$$

We observed the absence of Hopf bifurcations in the 1-faces and 2-faces and its presence in one 3-face $\mathcal{E}_1\mathcal{E}_2\mathcal{E}_5$ generated by the vectors $\mathcal{E}_1$,$\mathcal{E}_2$, and $\mathcal{E}_5$, where $\mathcal{E}_5$ represents a reversible fast reaction. We also detected its existence in the trivial cases of 4-faces that contain the subsystem $\mathcal{E}_1\mathcal{E}_2\mathcal{E}_5$ and the subsystem $\mathcal{E}_1\mathcal{E}_3\mathcal{E}_5\mathcal{E}_6$, where $\mathcal{E}_6$ also represent a reversible fast reaction. We conclude that eliminating fast reactions in the system for quasi-steady state approximation causes the disappearance of Hopf bifurcations for $n \geq 3$.

5.3.8. Example 8: Control of DNA replication in Fission Yeast

As another high-dimensional example, we consider the 22 dimensional model that describes the control of DNA replication in fission yeast. It is described in [46] and stored as a curated model in the BioModels database [50] with the ID BIOMOD0007. The stoichiometric matrix, the kinetic matrix, the set of extreme currents, and other algebraic data for this example can be obtained from our database PoCaB ("platform to explore algebraic methods for bio-chemical reaction networks")[15]. The flux cone of this model is spanned by 22 extreme currents and yields 230 2-faces and 1539 3-faces.

Using the *HoCaT* method we were able to detect the existence of Hopf bifurcations in 69 cases of the 3-faces and its absence in the 2-faces. The computation of this example also demonstrates the efficiency of our method, as it enables even the analysis of a 22-dimensional system.

Acknowledgements

This research was supported in part by *Deutsche Forschungsgemeinschaft* under the auspices of the SPP 1489 program and by the German Transregional Collaborative Research Center SFB/TR 14 AVACS. Thomas Sturm would like to thank B. Barber for his support with QuickHull and convex hull computation and W. Hagemann and M. Košta for helpful discussions on aspects of linear programming.

References

[1] M. El Kahoui, A. Weber, Journal of Symbolic Computation 30 (2000) 161–179. URL: `http://dx.doi.org/10.1006/jsco.1999.0353`. doi:`doi:10.1006/jsco.1999.0335`.

[2] A. Tarski, A Decision Method for Elementary Algebra and Geometry, second ed., University of California Press, Berkeley, 1951.

[3] T. Sturm, A. Weber, E. O. Abdel-Rahman, M. El Kahoui, Mathematics in Computer Science 2 (2009) 493–515.

[4] B. L. Clarke, Stability of Complex Reaction Networks, volume XLIII of *Advances in Chemical Physics*, Wiley Online Library, 1980. URL: `http://onlinelibrary.wiley.com/doi/10.1002/9780470142622.ch1/summary`. doi:`10.1002/9780470142622.ch1`.

---

[15] `http://pocab.cg.cs.uni-bonn.de/`

[5] K. Gatermann, M. Eiswirth, A. Sensse, Journal of Symbolic Computation 40 (2005) 1361–1382. doi:`http://dx.doi.org/10.1016/j.jsc.2005.07.002`.

[6] A. J. Shiu, Algebraic methods for biochemical reaction network theory, Phd thesis, University of California, Berkeley, 2010.

[7] M. Pérez Millán, A. Dickenstein, A. Shiu, C. Conradi, Bulletin of Mathematical Biology (2011) 1–29. URL: `http://www.ncbi.nlm.nih.gov/pubmed/21989565`. doi:`10.1007/s11538-011-9685-x`.

[8] C. Wagner, R. Urbanczik, Biophysical Journal 89 (2005) 3837–3845. URL: `http://dx.doi.org/10.1529/biophysj.104.055129`. doi:`10.1529/biophysj.104.055129`.

[9] A. Larhlimi, New Concepts and Tools in Constraint-based Analysis of Metabolic Networks, Ph.D. thesis, Berlin,Germany, 2008.

[10] K. Gatermann, B. Huber, Journal of Symbolic Computation 33 (2002) 275–305.

[11] J. Guckenheimer, P. Holmes, Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields, volume 42 of *Applied Mathematical Sciences*, Springer-Verlag, 1990.

[12] L. Orlando, Mathematische Annalen 71 (1911) 233–245.

[13] F. R. Gantmacher, Application of the Theory of Matrices, Interscience Publishers, New York, 1959.

[14] B. Porter, Stability Criteria for Linear Dynamical Systems, Academic Press, New York, 1967.

[15] P. Yu, International Journal of Bifurcation and Chaos 15 (2005) 1467–1483. URL: `http://www.worldscientific.com/doi/abs/10.1142/S0218127405012582`. doi:`10.1142/S0218127405012582`.

[16] W.-M. Liu, Journal of Mathematical Analysis and Applications 182 (1994) 250–256.

[17] E. N. Lorenz, Journal of the Atmospheric Sciences 20 (1963) 130–141.

[18] R. H. Rand, D. Armbruster, Perturbation Methods, Bifurcation Theory and Computer Algebra, volume 65 of *Applied Mathematical Sciences*, Springer-Verlag, 1987.

[19] W. M. Seiler, Involution — The Formal Theory of Differential Equations and its Applications in Computer Algebra, volume 24 of *Algorithms and Computation in Mathematics*, Springer, 2010.

[20] M. Hucka, A. Finney, H. M. Sauro, H. Bolouri, J. C. Doyle, H. Kitano, A. P. Arkin, B. J. Bornstein, D. Bray, A. Cornish-Bowden, et al., Bioinformatics 19 (2003) 524–531.

[21] S. Samal, H. Errami, A. Weber, in: Computer Algebra in Scientific Computing - 14th International Workshop (CASC 2012), Lecture Notes in Computer Science, Springer, 2012, pp. 294–307.

[22] E. Gawrilow, M. Joswig, in: G. Kalai, G. M. Ziegler (Eds.), Polytopes—Combinatorics and Computation, volume 29 of *Oberwolfach Seminars*, Birkhäuser Basel, 2000, pp. 43–73. URL: `http://dx.doi.org/10.1007/978-3-0348-8438-9\_2`, 10.1007/978-3-0348-8438-9_2.

[23] M. Domijan, M. Kirkilionis, Journal of Mathematical Biology 59 (2009) 467–501. URL: `http://www.ncbi.nlm.nih.gov/pubmed/19023573`. doi:`10.1007/s00285-008-0234-7`.

[24] V. Weispfenning, Journal of Symbolic Computation 5 (1988) 3–27.

[25] J. H. Davenport, J. Heintz, Journal of Symbolic Computation 5 (1988) 29–35.

[26] A. Dolzmann, T. Sturm, ACM SIGSAM Bulletin 31 (1997) 2–9.

[27] T. Sturm, Acta Academiae Aboensis, Ser. B 67 (2007) 177–191.

[28] V. Weispfenning, Applicable Algebra in Engineering Communication and Computing 8 (1997) 85–101.

[29] A. Dolzmann, T. Sturm, Journal of Symbolic Computation 24 (1997) 209–231.

[30] V. Weispfenning, in: B. Caviness, J. Johnson (Eds.), Quantifier Elimination and Cylindrical Algebraic Decomposition, Texts and Monographs in Symbolic Computation, Springer, Wien, New York, 1998, pp. 376–392.

[31] L. A. Gilch, Effiziente Hermitesche Quantorenelimination, Diploma thesis, Universität Passau, D-94030 Passau, Germany, 2003.

[32] A. Dolzmann, L. A. Gilch, in: J. A. C. Bruno Buchberger (Ed.), Artificial Intelligence and Symbolic Computation: 7th International Conference, AISC 2004, Linz, Austria, volume 3249 of *Lecture Notes in Computer Science*, Springer-Verlag, Berlin, Heidelberg, 2004, pp. 80–93.

[33] T. Sturm, in: V. G. Ganzha, E. W. Mayr, E. V. Vorozhtsov (Eds.), Computer Algebra in Scientific Computing: 9th International Workshop, volume 4194 of *Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg, 2006, pp. 29–35.

[34] A. Lasaruk, T. Sturm, Applicable Algebra in Engineering, Communication and Computing 18 (2007) 545–574. doi:`DOI:10.1007/s00200-007-0053-x`.

[35] A. Lasaruk, T. Sturm, in: V. G. Ganzha, E. W. Mayr, E. V. Vorozhtsov (Eds.), Computer Algebra in Scientific Computing. Proceedings of the CASC 2007, volume 4770 of *Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg, 2007, pp. 275–294.

[36] V. Weispfenning, Applicable Algebra in Engineering Communication and Computing 8 (1997) 85–101.

[37] T. Sturm, A. Weber, in: K. Horimoto, G. Regensburger, M. Rosenkranz, H. Yoshida (Eds.), Algebraic Biology – Third International Conference (AB 2008), volume 5147 of *Lecture Notes in Computer Science*, Springer-Verlag, Castle of Hagenberg, Austria, 2008. doi:`10.1007/978-3-540-85101-1_15`.

[38] C. W. Brown, ACM SIGSAM Bulletin 38 (2004) 23–24. doi:`http://doi.acm.org/10.1145/980175.980185`.

[39] A. Strzebonski, Journal of Symbolic Computation 29 (2000) 471–480.

[40] A. W. Strzebonski, J. Symb. Comput. 41 (2006) 1021–1038.

[41] L. De Moura, N. Bjørner, in: Tools and Algorithms for the Construction and Analysis of Systems, Springer, 2008, pp. 337–340.

[42] J. Reidl, P. Borowski, A. Sensse, J. Starke, M. Zapotocky, M. Eiswirth, Biophysical journal 90 (2006) 1147–55. URL: `http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1367266\&tool=pmcentrez\&rendertype=abstract`. doi:`10.1529/biophysj.104.058545`.

[43] B. Sturmfels, Solving Systems of Polynomial Equations, American Mathematical Society, Providence, RI, 2002. URL: `http://books.google.com/books?hl=en&lr=&id=N9c8bWxkz9gC&oi=fnd&pg=PR7&dq=Solving+Systems+of+Polynomial+Equations&ots=N8rdzvmSUv&sig=S82e2SQA8_hyJJeYL-rbMej0atwhttp://books.google.com/books?hl=en&lr=&id=N9c8bWxkz9gC&oi=fnd&pg=PA1&dq=Solving+systems+of+polynomial+equations&ots=N8u9wtiWTx&sig=wOIL8voiOVsgw0SvadkX2ntWkOU`.

[44] S. Sauerbrei, M. Nascimento, M. Eiswirth, H. Varela, The Journal of Chemical Physics 132 (2010) 154901–154901.

[45] C. Conradi, D. Flockerzi, J. Raisch, Mathematical Biosciences 211 (2008) 105–131.

[46] B. Novak, J. J. Tyson, Proceedings of the National Academy of Sciences 94 (1997) 9147–9152.

[47] F. Boulier, M. Lefranc, F. Lemaire, P. Morant, A. Ürgüplü, in: H. Anai, H. Horimoto, T. Kutsia (Eds.), Algebraic Biology (AB 2007), volume 4545 of *Lecture Notes in Computer Science*, Springer-Verlag, 2007, pp. 66–80. doi:`http://dx.doi.org/10.1007/978-3-540-73433-8_6`.

[48] M. Domijan (2008).

[49] N. I. Markevich, J. B. Hoek, B. N. Kholodenko, Science Signaling 164 (2004) 353.

[50] C. Li, M. Donizelli, N. Rodriguez, H. Dharuri, L. Endler, V. Chelliah, L. Li, E. He, A. Henry, M. I. Stefan, J. L. Snoep, M. Hucka, N. Le Novère, C. Laibe, BMC Systems Biology 4 (2010) 92. URL: `http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2909940\&tool=pmcentrez\&rendertype=abstract`.