

Enumeration of Autocatalytic Subsystems in Large Chemical Reaction Networks

Richard Golnik^{1*}, Thomas Gatter¹, Peter F. Stadler^{1,2,3,4,5,6,7},
Nicola Vassena¹

¹Bioinformatics Group, Department of Computer Science Leipzig
University, Härtelstraße 16–18, D-04107 Leipzig, Germany.

²Interdisciplinary Center for Bioinformatics & Zuse School for Embedded
and Composite Artificial Intelligence (SECAI) & Center for Scalable
Data Analytics and Artificial Intelligence, Leipzig University, D-04107
Leipzig, Germany.

³Max Planck Institute for Mathematics in the Sciences, Inselstraße 22,
D-04103 Leipzig, Germany.

⁴Department of Theoretical Chemistry, University of Vienna,
Währingerstraße 17, A-1090 Wien, Austria.

⁵Facultad de Ciencias, Universidad National de Colombia, Bogotá,
Colombia.

⁶Center for non-coding RNA in Technology and Health, University of
Copenhagen, Ridebanevej 9, DK-1870 Frederiksberg, Denmark.

⁷Santa Fe Institute, 1399 Hyde Park Rd., Santa Fe, NM 87501, USA.

*Corresponding author(s). E-mail(s): richard@bioinf.uni-leipzig.de;
Contributing authors: thomas@bioinf.uni-leipzig.de;
studla@bioinf.uni-leipzig.de; nicola.vassena@uni-leipzig.de;

Abstract

Autocatalysis is an important feature of metabolic networks, contributing crucially to the self-maintenance of organisms. Autocatalytic subsystems of chemical reaction networks (CRNs) are characterized in terms of algebraic conditions on submatrices of the stoichiometric matrix \mathbf{S} . Here, we derive sufficient conditions for subgraphs supporting irreducible autocatalytic systems in the bipartite König representation of the CRN. On this basis, we develop an efficient algorithm to enumerate autocatalytic subnetworks and, as a special case, autocatalytic cores, i.e., minimal autocatalytic subnetworks, in full-size metabolic networks. The same algorithmic approach can also be used to determine autocatalytic cores only. As a showcase application, we provide a complete analysis of autocatalysis in the core metabolism of *E. coli* and enumerate irreducible autocatalytic subsystems of limited size in full-fledged metabolic networks of *E. coli*, human erythrocytes, and *Methanosaerina barkeri* (Archea). The mathematical and algorithmic results are accompanied by software enabling the routine analysis of autocatalysis in large CRNs.

1 Introduction

An autocatalytic reaction is “a chemical reaction in which a product (or a reaction intermediate) also functions as a catalyst” [1]. Self-replication, i.e., the ability of multiplying instances of the self, is a special case of autocatalysis that is inherent to all living organisms. The emergence of self-replicating systems hence is a key issue in theories of the origin of life, independent of whether an RNA world, a lipid world, or a metabolism-first scenario is envisioned [2, 3, 4, 5, 6, 7, 8]. In a more general setting, autocatalysis is a property of chemical reaction networks (CRNs) that collectively implements an autocatalytic overall-reaction without any of the constituent reactions being autocatalytic. It is important to distinguish two fundamentally modeling frameworks: networks of autocatalysts such as the hypercycles of Eigen & Schuster [3] and catalytic reaction systems of Hordijk & Steele [9], which presuppose that *all* reactions are explicitly and specifically catalyzed by members of the system. Such systems thus represent interactions of complex entities such as RNAs, proteins, or other heteropolymers. In contrast, catalysis is an emergent property in networks of chemical reactions among small molecules. More precisely, catalysis in this setting is the net effect of a sequence of individual reactions. Here, we will be concerned exclusively with CRNs.

Autocatalysis in CRNs was generally considered to be scarce in non-enzymatic chemistry [10, 11]. On the other hand, it has been argued repeatedly, that metabolic networks are dominated by autocatalytic sub-systems [9, 11, 12]. Until recently, the lack of a consistent definition of autocatalysis made it difficult to discuss the prevalence of autocatalytic structures in chemical networks [13]. This situation changed when [14] proposed an algebraic definition of autocatalytic submatrices based only on structural properties of a chemical network encoded by the stoichiometric matrix. This notion of autocatalysis has become widely accepted because it does not only capture key features of collective autocatalysis but also turned out to mathematically well-behave [15] and to be suitable for constructing practical algorithms [16] identifying autocatalytic sub-networks. To the best of our knowledge, available tools enumerating

autocatalytic cycles are restricted to network sizes of approximately 300 metabolites and reactions [16, 17, 18]. Gagrani *et al.* [16] at present offer the most capable method currently accessible for networks of this scale, while the other approaches are either not yet publicly available [17] or limited to smaller networks [18]. This falls short of the capability to analyze the much larger metabolic networks in living organisms from bacteria to animals and plants [19, 20, 21, 22, 23, 24, 25].

Here we describe a graph-theoretical approach to identify irreducible autocatalytic subsystems. To this end, we first review the structure of autocatalytic matrices, expanding on the mathematical analysis of [15]. In Sec. 3 we introduce the new concept of *centralized autocatalysis* and show that four of the five classes of autocatalytic cores described in [14] are centralized. In Sec. 4 we characterize a class of subgraphs of the bipartite König representation of a CRN that we term fluffles and show that only fluffles can induce irreducible autocatalytic subsystems in the CRN. We then introduce an equivalence relation on these graphs that corresponds to equality of the autocatalytic matrices. We show that representatives of these equivalence classes can be obtained efficiently by superimposing elementary circuits in the König graph. This observation constitutes the basis for the graph-theoretical algorithm in Sec. 5 for the enumeration of irreducible autocatalytic subsystems in large CRNs. A Python implementation was applied to several metabolic networks of different sizes ranging from the *E. coli* core model to a nearly complete network of *E. coli*. The computational results in Section 6 indicate that autocatalysis is prevalent, in all domains of life, however, to varying degrees. Moreover, they establish that the graph-theoretic approach is robustly applicable to metabolic network models of practical interest. Nevertheless, some technical questions of interest remain open; they are briefly summarized in Sec. 7.

2 Chemical reaction networks

2.1 Network structure

A chemical reaction network (CRN) Γ is a pair of finite sets $\Gamma := (X, R)$, where X is a set of chemical species or metabolites and R is a set of chemical reactions. A reaction r is a directed transformation between nonnegative linear combinations of metabolites and can be described as:

$$\sum_{x \in X} s_{xr}^- \cdot x \quad \xrightarrow{r} \quad \sum_{x \in X} s_{xr}^+ \cdot x \quad (1)$$

with $s_{xr}^- \geq 0$ and $s_{xr}^+ \geq 0$ denoting the nonnegative stoichiometric coefficients of the molecular count. Typically these coefficients are integer, but our theory does not make a distinction about it. Metabolites x appearing with nonzero $s_{xr}^- > 0$ on the left-hand side of (1) are called *educts* or *reactants* of r , while metabolites x appearing with nonzero $s_{xr}^+ > 0$ on the right-hand side of (1) are called *products* of r . We may collect the stoichiometric coefficients of the reactant species in the $|X| \times |R|$ *reactant matrix* $\mathbf{S}^- : \mathbf{S}_{xr}^- := s_{xr}^-$ and the coefficients of the product species in the $|X| \times |R|$ *product matrix* $\mathbf{S}^+ : \mathbf{S}_{xr}^+ := s_{xr}^+$. The difference of the two matrices gives rise to the

stoichiometric matrix $\mathbf{S} := \mathbf{S}^+ - \mathbf{S}^-$ with entries:

$$\mathbf{S}_{xr} := s_{xr}^+ - s_{xr}^-. \quad (2)$$

The most natural representation of Γ is a directed, weighted hypergraph with vertex set X and edge set R . Each reaction $r \in R$ is represented by a hyperarrow, whose inputs correspond to the reactant species and outputs to the product species. The stoichiometric coefficients appear as weights for the hyperarrow. Building on this perspective, throughout we consider CRNs in *König representation* [26], i.e., we represent the hypergraph Γ as a directed bipartite graph

$$\mathbf{K} := \mathbf{K}(X, R) := (X \cup R, E), \quad (3)$$

with disjoint vertex sets X and R and edge set $E := E_1 \cup E_2$ where

$$\begin{cases} E_1(\mathbf{K}) := \{(x, r) \mid s_{xr}^- > 0\}; \\ E_2(\mathbf{K}) := \{(r, x) \mid s_{xr}^+ > 0\}. \end{cases} \quad (4)$$

The $(|X| + |R|) \times (|X| + |R|)$ adjacency matrix A of $\mathbf{K}(X, R)$ is therefore given as:

$$A = \begin{pmatrix} 0_{|X|} & S^+ \\ S^- & 0_{|R|} \end{pmatrix}, \quad (5)$$

where 0_n indicates the $n \times n$ zero matrix. The stoichiometric coefficient appear as edge-weights in $\mathbf{K}(X, R)$. For integer coefficients, we may also think of $\mathbf{K}(X, R)$ as a directed multigraph with arc multiplicities defined by the stoichiometric coefficients.

Subgraph notation. In the following, we will also consider subgraphs \mathbf{K}' of \mathbf{K} . We will write $V(\mathbf{K}') = X(\mathbf{K}') \cup R(\mathbf{K}')$, and $E(\mathbf{K}') = E_1(\mathbf{K}') \cup E_2(\mathbf{K}')$ for the vertex and edge set of \mathbf{K}' , respectively. Moreover, we denote the set of vertices incident with the edges in any edge set E_i by $V(E_i)$.

2.2 Parameter-rich chemical kinetics

Let $z(t) \in \mathbb{R}_{\geq 0}^{|X|}$ indicate the vector of the chemical concentration of the species at time t in a well-mixed, spatially homogeneous, reactor. The time-evolution for $z(t)$ is described by the system of Ordinary Differential Equations (ODEs):

$$\dot{z} = f(z) := \mathbf{S} \cdot \nu(z) \quad (6)$$

where \mathbf{S} is the stoichiometric matrix defined in (2), and $\nu(z) \in \mathbb{R}^{|R|}$ is the vector of the reaction rate functions (kinetics). A primary modeling issue in reaction networks is the ubiquitous lack of precise knowledge of the mathematical form of the rates $\nu(z)$. For this reason, the literature typically resorts to kinetic models: a wide class of reaction functions defined as follows.

Definition 1. A monotone kinetic model for a reaction network $\Gamma := (X, R)$ is a vector-valued function $\nu : \mathbb{R}_{\geq 0}^{|X|} \mapsto \mathbb{R}_{\geq 0}^{|R|}$, which satisfies the following conditions:

- i. $\nu_r(z) \geq 0$, for all $z \in \mathbb{R}_{\geq 0}^{|X|}$;
- ii. $\nu_r(z) > 0$ implies $z_x > 0$ for all species z with $s_{xr}^- > 0$;
- iii. $s_{xr}^- = 0$ implies $\partial\nu_r/\partial z_x \equiv 0$;
- iv. $z > 0$ and $s_{xr}^- > 0$ implies $\partial\nu_r/z_z > 0$.

Since this work always focuses on monotone kinetic models, for brevity, throughout we write simply ‘kinetic models’. Given the aforementioned uncertainty on the precise quantities involved, it is typical to consider parametric kinetic models. Whenever necessary, we will refer to this dependency by writing $\nu(z, p)$. Standard examples of such parametric kinetic models are classic [27] and generalized [28] mass-action kinetics, both polynomials, and more involved rational functions such as Michaelis–Menten kinetics [29] and the Hill model [30].

Consider now a network Γ endowed with a kinetic model ν . Fixed points $\bar{z} \geq 0$ of f in (6), i.e.,

$$0 = f(\bar{z}) = \mathbf{S} \cdot \nu(\bar{z}), \quad (7)$$

are called *steady-states* of Γ . Throughout this work, we consider only *consistent* networks [31], whose stoichiometric matrix admits a positive right kernel vector $\nu > 0 : \mathbf{S} \cdot \nu = 0$, which is a necessary condition for a network Γ to admit at least one positive steady-state. It is well-known [32] that the dynamical stability of \bar{z} can be addressed at first approximation by studying the linearization of system (6) at \bar{z} :

$$\dot{z} = \mathbf{D}_f(\bar{z})z = (\mathbf{S} \cdot \mathbf{N}(\bar{z}))z, \quad (8)$$

where $\mathbf{D}_f(\bar{z})$ is the *Jacobian matrix* evaluated at \bar{z} . The nonnegative matrix $\mathbf{N} \in \mathbb{R}^{|R| \times |X|}$ with entries

$$\mathbf{N}_{rx}(\bar{z}) := \left. \frac{\partial \nu_r(z)}{\partial z_x} \right|_{z=\bar{z}} \quad (9)$$

is called *reactivity matrix*. In particular, for hyperbolic steady-states \bar{z} , i.e., for which the Jacobian $\mathbf{D}_f(\bar{z})$ has only eigenvalues with nonzero real part, the spectrum of the Jacobian determines the dynamical stability: the steady-state \bar{z} is stable if $\mathbf{D}_f(\bar{z})$ is Hurwitz-stable, i.e. it possesses only eigenvalues with negative-real part, and \bar{z} is unstable if $\mathbf{D}_f(\bar{z})$ is Hurwitz-unstable, i.e. it possesses at least one eigenvalue with positive-real part.

Given a network Γ endowed with a parametric kinetic model $\nu(z, p)$, the relation between the network structure and the possible spectrum configurations of the Jacobian at the varying of parameters p is a classic problem that has turned out to be very challenging [33]. These relationships become much more tractable if the parametric kinetic model has sufficient internal freedom, at least as long existence results are the major concern [15].

Definition 2. A monotone kinetic rate model $\nu(z, p)$ is parameter-rich if, for every positive steady-state $\bar{z} > 0$ and every choice of an $|R| \times |X|$ matrix \mathbf{N} satisfying $\mathbf{N}_{rx} > 0$ iff $s_{xr}^- > 0$, there exists a choice of parameters $\bar{p} = p(\bar{z}, \mathbf{N})$ such that $\nu'_{rx}(\bar{z}, \bar{p}) = \mathbf{N}_{rx}$.

Far from being just a theoretical construct, widely used schemes in biochemistry – such as Michaelis-Menten, Hill, and generalized mass action – are naturally parameter-rich. Classical mass-action kinetics, however, lacks sufficient parametric freedom and is therefore not parameter-rich.

The advantage of the parameter-rich framework is that we may then consider a *symbolic reactivity matrix* \mathbf{N} , that is, any $|R| \times |X|$ matrix whose nonnegative symbolic entries satisfy $N_{rx} > 0 \iff s_{xr}^- > 0$, and study the spectrum of the associated *symbolic Jacobian matrix*

$$\mathbf{D} := \mathbf{S}\mathbf{N}, \quad (10)$$

which no longer depends explicitly on the steady-state value \bar{z} . In particular, under parameter-rich kinetics, the existence of some evaluation of \mathbf{N} such that \mathbf{D} has a given spectrum directly implies the existence of kinetic parameters for which this very \mathbf{D} is realized as the actual Jacobian at a steady state \bar{z} , and thus dynamical conclusions can be drawn. Accordingly, we say that the network Γ *admits instability* if there exists a choice \mathbf{N} of the kinetic matrix such that the symbolic Jacobian \mathbf{D} is Hurwitz-unstable.

On the other hand, since the entries of the symbolic reactivity matrix are determined as zero or positive solely from the stoichiometric matrix, this approach can be used to draw conclusions about a network's capacity for certain dynamical properties based on stoichiometry alone. The main tool to do so are *Child-Selections* (CS) [15].

Definition 3. A k -Child-Selection triple, or k -CS for short, is a triple $\kappa = (X_\kappa, R_\kappa, \kappa)$ such that $|X_\kappa| = |R_\kappa| = k$, $X_\kappa \subseteq X$, $R_\kappa \subseteq R$, and $\kappa : X_\kappa \rightarrow R_\kappa$ is a bijection satisfying $s_{x\kappa(x)}^- > 0$ for all $x \in X_\kappa$. We call κ a CS bijection.

The restriction to sub-matrices that derive from Child-Selections is motivated by the analysis of the symbolic Jacobian by means of the Cauchy–Binet decomposition, which shows that only Child-Selections contribute [15]. To any given k -CS $\kappa = (X_\kappa, R_\kappa, \kappa)$, we associate a $k \times k$ CS-matrix $\mathbf{S}[\kappa]$ defined as follows:

$$\mathbf{S}[\kappa]_{xw} := s_{x\kappa(w)}^+ - s_{x\kappa(w)}^-, \quad \text{for all } x, w \in X_\kappa. \quad (11)$$

Note that a CS-matrix may differ from a submatrix of \mathbf{S} by having a different column order. In particular, for a fixed ordering of the species $X = \{x_1, \dots, x_{|X|}\}$, the column order of the stoichiometric matrix \mathbf{S} depends on the ordering of the set R , whereas the ordering of $\mathbf{S}[\kappa]$ is independent of it, depending only on the order of X .

Let κ be any k -CS. Without loss of generality, consider now $X_\kappa = \{x_1, \dots, x_k\} \subset X$. By choosing the following rescale for the symbolic reactivity matrix \mathbf{N} ,

$$N_{rx}(\varepsilon) = \begin{cases} 1 & \text{if } x \in X_\kappa \text{ and } r = \kappa(x) \\ \varepsilon & \text{otherwise, if } s_{xr}^- > 0 \end{cases} \quad (12)$$

a straightforward computation shows that the associated symbolic Jacobian $\mathbf{D}(\varepsilon)$ reads:

$$\mathbf{D}(\varepsilon) = \begin{pmatrix} \mathbf{S}[\kappa] + O(\varepsilon) & O(\varepsilon) \\ \dots & O(\varepsilon) \end{pmatrix}, \quad (13)$$

where $O(\varepsilon)$ indicate an expression of order ε . We again refer to [15] for a detailed derivation. In particular, for ε small enough, the k eigenvalues of $\mathbf{S}[\kappa]$ approximate

the k largest (in absolute value) eigenvalues of $\mathbf{D}(\varepsilon)$. This argument shows that any k -CS-matrix can be used to approximate k dominant eigenvalues of the Jacobian. In particular, we obtain a straightforward condition for a network to admit instability:

Proposition 4 (Cor. 5.1 in [15]). *Consider a network $\Gamma := (X, R)$ with parameter-rich kinetics. If there is k -CS κ such that its associated $k \times k$ CS-matrix is Hurwitz-unstable, then the network admits instability.*

Moreover, using the same line of reasoning, it is shown in [15] that the presence of autocatalysis in the network always implies that the network admits instability. The next Sec. 2.3 briefly reviews this connection.

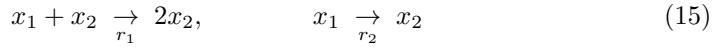
2.3 Autocatalytic matrices

Autocatalytic processes can be encoded either by explicitly autocatalytic reactions or by more involved network processes including multiple reactions and species [3, 7]. An explicitly catalytic reaction has the form

$$s_{xr}^- \cdot x + \dots \xrightarrow{r} s_{xr}^+ \cdot x + \dots, \quad (14)$$

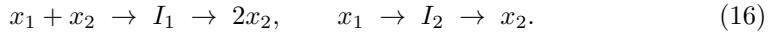
where $s_{xr}^+ \cdot s_{xr}^- \neq 0$, i.e., a species x participates in the reaction r both as a reactant and as a product. Such a reaction is autocatalytic (for a species x) if the net difference in stoichiometric coefficients is positive: $s_{xr}^+ - s_{xr}^- > 0$.

A technical issue arises when explicitly catalytic reactions are admitted in the network representation: the stoichiometric matrix \mathbf{S} no longer uniquely determines the network. For instance, both reactions

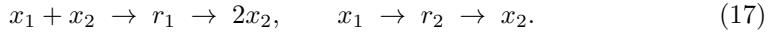


correspond to the same column in the stoichiometric matrix: $\mathbf{S} = (-1 \ 1)^T$. Yet while the first reaction is autocatalytic, the second is not. This simple example shows that in the presence of explicitly catalytic reactions the stoichiometric matrix alone cannot always distinguish between autocatalysis and its absence.

A standard resolution is to introduce intermediates, which restores uniqueness:



Adding such intermediates is in principle a straightforward preprocessing step. Moreover, the theory presented here is unaffected by this choice: (i) as shown in [15], autocatalysis in networks with explicitly catalysis (15) corresponds one-to-one with autocatalysis in their intermediate form (16); (ii) our present algorithm is based on bipartite graphs, in which the two cases are already naturally distinguished:



However, presenting the preliminary theory while accounting for explicit catalysis requires a more delicate and technically involved algebraic treatment. For ease of

presentation, we proceed under the assumption that explicit autocatalytic reactions are absent. This assumption simplifies the exposition without loss of generality.

We can then continue by speaking of matrices only. Blokhuis et al. [14] derived a matrix definition of autocatalysis from the definition of the IUPAC (International Union of Pure and Applied Chemistry) [1]:

Definition 5. A matrix $\mathbf{A} \in \mathbb{R}^{n,m}$ is autocatalytic if

- (i) there is $v \in \mathbb{R}_{>0}^m$ such that $\mathbf{A}v > 0$;
- (ii) for each $r \in \{1, \dots, m\}$ there is $x, y \in \{1, \dots, n\}$: $\mathbf{A}_{xr} < 0$ and $\mathbf{A}_{yr} > 0$.

We use $\mathbf{S}[X_\kappa, R_\kappa]$ to refer to the submatrix of \mathbf{S} with species index in $X_\kappa \subseteq X$ and reaction index in $R_\kappa \subseteq R$, and we can then define an autocatalytic network consequently.

Definition 6. A network Γ is autocatalytic if its stoichiometric matrix \mathbf{S} possesses an autocatalytic submatrix.

Recent literature on autocatalysis [14, 15, 16] put considerable emphasis on minimal autocatalytic matrices, so-called autocatalytic cores:

Definition 7. A matrix \mathbf{A} is an autocatalytic core if \mathbf{A} is autocatalytic and does not contain a proper autocatalytic submatrix.

Key features of autocatalytic cores are collected in the following proposition. For a proof see [14] and [15].

Proposition 8. Let $\tilde{\mathbf{A}}$ be an autocatalytic core. The following all hold true:

1. $\tilde{\mathbf{A}}$ is an invertible square matrix;
2. There exists a unique autocatalytic core \mathbf{A} with strictly negative diagonal obtained by reordering the columns of $\tilde{\mathbf{A}}$;
3. The off-diagonal entries of \mathbf{A} obtained at point 2 are nonnegative.

Square matrices with nonnegative off-diagonal entries are called *Metzler* in the literature, and their stability properties have been extensively studied in connection with the Frobenius-Perron Theorem [34]. Throughout this paper, we refer to autocatalytic cores \mathbf{A} always intending the Metzler representation with negative diagonal and nonnegative off-diagonal. In this case, further properties – related to dynamical stability – were shown in [15]:

Proposition 9. Let \mathbf{A} be an $n \times n$ autocatalytic core in Metzler form. The following all hold true:

1. $\mathbf{A} = \mathbf{S}[\kappa]$ for a unique Child-Selection κ ;
2. \mathbf{A} is irreducible;
3. \mathbf{A} is Hurwitz-unstable. More precisely, \mathbf{A} possesses exactly one eigenvalue with positive real part, and thus its determinant is of sign $\det \mathbf{A} = (-1)^{n-1}$.

As a key consequence these properties imply:

Corollary 10. If the network is autocatalytic, then it admits instability.

The emphasis on minimal autocatalytic subnetworks is mostly justified for qualitative and classification purposes. In contrast, the Jacobian rescaling (13) suggests that larger CS-matrices may better capture the overall dynamical impact of autocatalysis on the system, in terms of instability and growth rate, since fewer variables are ε -rescaled. We therefore aim to develop a detection algorithm that goes beyond autocatalytic cores. A natural broader class of interest is given by CS-matrices that are irreducible Metzler matrices. For this class, the link between autocatalysis and

instability is fully preserved, as in Prop. 9. It can be restated as a direct consequence of the Perron–Frobenius theorem [15, 34] as follows:

Lemma 11. *Let $\mathbf{S}[\kappa]$ be an irreducible Metzler matrix. The following are equivalent:*

1. $\mathbf{S}[\kappa]$ is Hurwitz-unstable;
2. $\mathbf{S}[\kappa]$ has a real positive eigenvalue;
3. $\mathbf{S}[\kappa]$ is autocatalytic.

For irreducible Metzler matrices, Hurwitz-instability (a spectral property in general sensitive to column ordering) is therefore equivalent to autocatalysis (a structural property independent of ordering). On the other hand, without loss of generality in network labeling, any *reducible* Metzler CS-matrix $\mathbf{S}[\kappa]$ can be represented in block form as

$$\mathbf{S}[\kappa] = \begin{pmatrix} \mathbf{S}[\kappa'] & 0 \\ \mathbf{B} & \mathbf{S}[\kappa''] \end{pmatrix}, \quad (18)$$

where $\mathbf{S}[\kappa']$ is irreducible. In the above representation, we say that $\mathbf{S}[\kappa]$ is decomposed as a *cascade originating from $\mathbf{S}[\kappa']$* . The next result shows that autocatalysis for $\mathbf{S}[\kappa]$ necessarily requires autocatalysis in $\mathbf{S}[\kappa']$.

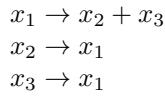
Proposition 12 (Proof: SI). *Let $\kappa = (X_\kappa, R_\kappa, \kappa)$ be a k -CS whose associated CS-matrix $\mathbf{S}[\kappa]$ is reducible, Metzler, and autocatalytic. Then there exists a k' -CS $\kappa' = (X_{\kappa'}, R_{\kappa'}, \kappa')$ with $X_{\kappa'} \subset X_\kappa$, $R_{\kappa'} \subset R_\kappa$, and $\kappa'(X_{\kappa'}) = \kappa(X_{\kappa'})$, such that its associated CS-matrix $\mathbf{S}[\kappa']$ is an irreducible autocatalytic Metzler matrix.*

In light of Prop. 12, any reducible autocatalytic CS-matrix $\mathbf{S}[\kappa]$ can always be decomposed into a cascade originating from an irreducible autocatalytic Metzler matrix $\mathbf{S}[\kappa']$. Thus, our emphasis naturally falls on irreducible Metzler autocatalytic CS-matrices.

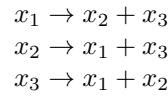
3 Centralized autocatalysis

In analyzing autocatalytic structures, it is natural to note a qualitative dichotomic difference among them. Consider the following two examples:

Example I:



Example II:



with associated CS-matrices, respectively,

$$\begin{pmatrix} -1 & 1 & 1 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{pmatrix} \quad \begin{pmatrix} -1 & 1 & 1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \end{pmatrix}. \quad (19)$$

In Example I, species x_1 produces x_2 and x_3 , and both in turn react back to x_1 . The autocatalytic process is thus *centralized* around x_1 : every reaction cycle passes through x_1 , so that autocatalysis can be interpreted as an amplification mechanism for x_1 . In contrast, in Example II no single species plays such a role, due to a stronger interconnection of the network: there is a clear symmetry among indices 1, 2, 3, and the network topology is invariant under permutations of these labels. To capture and

formalize this intuitive difference, we introduce the notion of *centralized autocatalysis*. Our first formulation is based on *permutation cycles* of the associated CS-matrix. Subsequently, we will provide an equivalent graph-theoretic characterization. Recall the standard Leibniz formula for the determinant of a $k \times k$ CS-matrix $\mathbf{S}[\kappa]$:

$$\det \mathbf{S}[\kappa] = \sum_{\pi \in P_k} \text{sgn}(\pi) \prod_{m=1}^k \mathbf{S}[\kappa]_{m,\pi(m)}, \quad (20)$$

where P_k denotes the symmetric group on k elements. Every non-identity permutation $\pi \neq \text{id}$ can be decomposed as a product of disjoint cyclic permutations of at least two elements,

$$\pi = C_1 \cdot \dots \cdot C_{n_\pi}. \quad (21)$$

We write \mathcal{C}_π for the set of permutation cycles of π . For simplicity of notation, we consider each cycle C_i as an element itself of P_k , i.e., as a permutation of its $k = |X_\kappa|$ elements. We say that a permutation cycle C *contributes* (to $\det \mathbf{S}[\kappa]$) if

$$\prod_{m=1}^k \mathbf{S}[\kappa]_{m,C(m)} \neq 0. \quad (22)$$

We denote with $\mathfrak{K}[\kappa]$ the set of non-trivial permutation cycles contributing to $\det \mathbf{S}[\kappa]$, i.e. with length ≥ 2 . In turn, permutation cycles without a contribution in the Leibniz formula (20) can be ignored in the following. We are now in the position to formalize the distinction between the two examples above:

Definition 13. Let $k \times k$ be an irreducible autocatalytic Metzler matrix $\mathbf{S}[\kappa]$ and denote by M_κ the set of all $m^* \in \{1, \dots, k\}$ such that every permutation cycle C contributing to $\det \mathbf{S}[\kappa]$ satisfies $\mathbf{S}[\kappa]_{m^*,C(m^*)} > 0$. Then $\mathbf{S}[\kappa]$ is centralized if $M_\kappa \neq \emptyset$. The elements of M_κ are the autocatalytic centers of $\mathbf{S}[\kappa]$ and we say that $\mathbf{S}[\kappa]$ is centered at M_κ provided $M_\kappa \neq \emptyset$.

As a direct consequence of the definition, moreover, for centralized autocatalysis the determinant computed via the Leibniz formula (20) naturally provides the sum over the weights of the different cycles in the network, as the next proposition states.

Proposition 14 (Proof: SI). Let $\mathbf{S}[\kappa]$ be a $k \times k$ irreducible autocatalytic Metzler matrix that exhibits centralized autocatalysis. Then

$$\frac{\det \mathbf{S}[\kappa]}{\prod_{m=1}^k \mathbf{S}[\kappa]_{mm}} = 1 - \sum_C \prod_{m \in C} \frac{\mathbf{S}[\kappa]_{m,C(m)}}{|\mathbf{S}[\kappa]_{mm}|} \quad (23)$$

where the sum runs on all permutation cycles.

In essence, Prop. 14 differs from the Leibniz formula (20) because the sum runs over permutation cycles, only, instead over all permutations. Prop. 14 holds for centralized autocatalysis. Nevertheless, equality (23) does not solely apply to centralized autocatalysis as stated in Thm. 19. Hence, it does not provide a characterization of centralized autocatalysis.

Although examples of stoichiometric coefficients different from $(0, 1)$ do exist also in metabolic networks they are very rare. A well-known example is the condensation of two acetyl-CoA molecules into acetoacetyl-CoA in the synthesis of HMG-CoA during cholesterol or isopentenyl pyrophosphate (IPP) biosynthesis [35]. In the special case of unit stoichiometric coefficients, Prop. 14 simplifies to an easily interpretable statement regarding the *number* of contributing permutation cycles of the stoichiometric matrix.

Corollary 15. *Let $\mathbf{S}[\kappa]$ be a $k \times k$ irreducible autocatalytic Metzler matrix that exhibits centralized autocatalysis, and such that*

$$\mathbf{S}[\kappa]_{ij} \in \{-1, 0, 1\} \quad \text{for all } (i, j).$$

Then

$$\det \mathbf{S}[\kappa](-1)^k = 1 - \#_C,$$

where $\#_C$ is the number of permutation cycles C such that $\prod_{m=1}^k \mathbf{S}[\kappa]_{m,C(m)} \neq 0$.

Proof. It directly follows from Prop. 14. \square

For a k -CS $\kappa = (X_\kappa, R_\kappa, \kappa)$, centralized autocatalysis can be characterized in graph-theoretical terms using a correspondence between contributing permutation cycles in its CS-matrix $\mathbf{S}[\kappa]$ and directed elementary circuits in the induced subgraphs $\mathbf{K}[\kappa] := \mathbf{K}[X_\kappa \cup R_\kappa]$ of the König graph of the CRN. The key observation is that if x and y are consecutive vertices in a permutation cycle C that contributes to $\mathbf{S}[\kappa]$, then $(x, \kappa(x), y)$ is a path in $\mathbf{K}[\kappa]$ and, *vice versa*, if (x, r, y) is a path in $\mathbf{K}[\kappa]$ of an irreducible autocatalytic Child-Selection, then $r = \kappa(x)$. Denoting the elementary circuits (viewed as subgraphs of $\mathbf{K}[\kappa]$) by $\mathbf{C}(\kappa)$, we obtain the following formal statement:

Lemma 16 (Proof: SI). *Let $\mathbf{S}[\kappa]$ be an autocatalytic CS Metzler matrix and $\mathfrak{K}[\kappa]$ the set of contributing permutation cycles of length ≥ 2 . Then there is a one-to-one correspondence between $\mathfrak{K}[\kappa]$ and $\mathbf{C}(\kappa)$ such that a contributing permutation cycle (x_1, x_2, \dots, x_k) corresponds to the elementary circuit $(x_1, \kappa(x_1), x_2, \kappa(x_2), \dots, \kappa(x_{k-1}), x_k, \kappa(x_k), x_1)$ in $\mathbf{K}[\kappa]$.*

As a direct consequence of Lemma 16, we can now rephrase Def. 13 in graph-theoretical terms:

Corollary 17. *Let $\mathbf{A} := \mathbf{S}[\kappa]$ be an autocatalytic Metzler matrix. Then \mathbf{A} is centralized if and only if there is a vertex $x^* \in X_\kappa$ such that $x^* \in X(C)$ for all $C \in \mathbf{C}(\kappa)$.*

We collect all center species x^* in a set X_κ^* and refer to it as the autocatalytic center of \mathbf{A} . Note that x^* and X_κ^* correspond to m^* and M_κ above.

To simplify the notation, we introduce a normalized version of the matrix \mathbf{A} by setting $N(\mathbf{A})_{ij} := \mathbf{A}_{ij}/|\mathbf{A}_{jj}|$ for all i, j . With this notation, Prop. 14 and Lemma 16 immediately imply

Corollary 18. *Let $\mathbf{A} \in \{\mathbb{Z}\}^{k \times k}$ be centralized in X_κ^* , denote by $\mathbf{C}_{x^*}(\kappa)$ the set of cycles containing x^* , and let $s_C(y)$ be the successor of y along the elementary circuit C . Then*

$$\det(\mathbf{A}) \cdot (-1)^{k-1} = \sum_{C \in \mathbf{C}_{x^*}(\kappa)} \prod_{y \in X(C)} |N(\mathbf{A})_{y, s_C(y)}| \tag{24}$$

for all autocatalytic centers $x^* \in X_\kappa^*$. In particular if $\mathbf{A} \in \{-1, 0, 1\}^{k \times k}$ then

$$\det(\mathbf{A}) \cdot (-1)^{k-1} = |\mathbf{C}_{x^*}(\boldsymbol{\kappa})| - 1, \quad (25)$$

We conclude this section by connecting the concept of centralized autocatalysis with the classification of autocatalytic cores proposed by Blokhuis et al. [14]. Their five types in essence correspond to the following five motifs (up to different stoichiometric coefficients):

$$\text{Type I: } x_1 \rightarrow x_2 \rightarrow 2x_1 \quad \begin{pmatrix} -1 & 2 \\ 1 & -1 \end{pmatrix} \quad (26)$$

$$\text{Type II: } \begin{cases} x_1 \rightarrow x_2 + x_3 \\ x_2 \rightarrow x_3 \\ x_3 \rightarrow x_1 \end{cases} \quad \begin{pmatrix} -1 & 0 & 1 \\ 1 & -1 & 0 \\ 1 & 1 & -1 \end{pmatrix} \quad (27)$$

$$\text{Type III: } \begin{cases} x_1 \rightarrow x_2 + x_3 \\ x_2 \rightarrow x_1 \\ x_3 \rightarrow x_1 \end{cases} \quad \begin{pmatrix} -1 & 1 & 1 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{pmatrix} \quad (28)$$

$$\text{Type IV: } \begin{cases} x_1 \rightarrow x_2 + x_3 \\ x_2 \rightarrow x_1 + x_3 \\ x_3 \rightarrow x_1 \end{cases} \quad \begin{pmatrix} -1 & 1 & 1 \\ 1 & -1 & 0 \\ 1 & 1 & -1 \end{pmatrix} \quad (29)$$

$$\text{Type V: } \begin{cases} x_1 \rightarrow x_2 + x_3 \\ x_2 \rightarrow x_1 + x_3 \\ x_3 \rightarrow x_1 + x_2 \end{cases} \quad \begin{pmatrix} -1 & 1 & 1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \end{pmatrix} \quad (30)$$

Examples 1 and 2 above are of Type III and Type V, respectively. A more formal analysis shows that four of the five types are centralized, and that Eq. (23) holds nevertheless for all five cores:

Theorem 19 (Proof: SI, Thm. B). *An autocatalytic core is centralized if and only if it is of type I, II, III, or IV in the classification of [14]. Moreover, Eq. (23) holds for all five types of cores.*

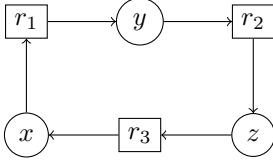
4 Autocatalytic König graphs

4.1 Child-selective subgraphs

We first turn our attention to identify substructures in the König graph that induce Child-Selections (CS). To this end, we consider subgraphs \mathbf{K}' of $\mathbf{K}(X, R)$, which are not necessarily induced, i.e. subgraphs \mathbf{K}' that not necessarily include all edges $e \in \mathbf{K}$ between two selected vertices $x, r \in \mathbf{K}'$.

Definition 20. *A subgraph \mathbf{K}' of \mathbf{K} is child-selective if there exists a map $\kappa : X(\mathbf{K}') \rightarrow R(\mathbf{K}')$ such that $\kappa = (X(\mathbf{K}'), R(\mathbf{K}'), \kappa)$ is a CS.*

Since the map κ in a CS is bijective, \mathbf{K}' can only be child-selective if $|X(\mathbf{K}')| = |R(\mathbf{K}')|$. Recall that a *matching* in a graph is a set of vertex-disjoint edges, while a *perfect matching* is one incident with every vertex.



$$\mathbf{S}[\kappa] = \begin{pmatrix} -1 & -1 & 1 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \end{pmatrix}$$

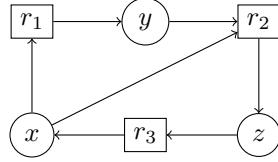


Fig. 1: In general, $\mathbf{K}(\kappa)$ is a proper subgraph of the induced subgraph $\mathbf{K}[\kappa]$. The example corresponds to the CS matrix $\mathbf{S}[\kappa]$ shown on the left with columns ordered as x, y, z .

Theorem 21. A subgraph $\mathbf{K}' \subseteq \mathbf{K}$ is child selective if and only if the subset $E_1(\mathbf{K}') := \{(x, r) \mid s_{xr}^- > 0\} \subset E(\mathbf{K}')$ of the reactant-to-reaction edges contains a perfect matching.

Proof. If \mathbf{K}' is child-selective with CS κ , we directly obtain a perfect matching by $E_\kappa := \{(x, \kappa(x)) \mid x \in X(\mathbf{K}')\} \subseteq E_1$. Conversely, let $M \subseteq E_1(\mathbf{K}') \subseteq E_1$ be a perfect matching in \mathbf{K}' , then there is $(x, y) \in M$ for all $x \in X(\mathbf{K}')$ and y is uniquely defined for every x . Thus $\kappa : X(\mathbf{K}') \rightarrow R(\mathbf{K}')$ with $\kappa(x) = y$ if $(x, y) \in M$ is uniquely defined and injective. Moreover, there is u with $(u, v) \in M$ for all $v \in R(\mathbf{K}')$. Hence κ is a bijection, and $\kappa := (X(\mathbf{K}'), R(\mathbf{K}'), \kappa)$ is a CS. \square

The proof of Thm. 21 contains an explicit recipe to construct CS. In fact, there is a 1-1 correspondence between perfect matchings in $E_1(\mathbf{K}')$ and bijections $\kappa : X(\mathbf{K}') \rightarrow R(\mathbf{K}')$. Moreover, the spanning subgraph $\mathbf{K}'' \subseteq \mathbf{K}'$ with $X(\mathbf{K}'') = X(\mathbf{K}')$, $R(\mathbf{K}'') = R(\mathbf{K}')$ and $E(\mathbf{K}'') = M \cup E_2(\mathbf{K}')$ is child selective for every perfect matching $M \subseteq E_1(\mathbf{K}')$. Conversely, for any CS $\kappa = (X_\kappa, R_\kappa, \kappa)$, let E_1^κ and E_2^κ be subsets of E_1 and E_2 , respectively, where edges have both adjacent vertices in $(X_\kappa \cup R_\kappa)$. Then we write

$$\mathbf{K}(\kappa) := (X_\kappa \cup R_\kappa, M_\kappa \cup E_2^{(\kappa)}) \quad (31)$$

defined by the perfect matching $M_\kappa \subseteq E_1^{(\kappa)}$. We note that $\mathbf{K}(\kappa)$ is a spanning subgraph of the induced subgraph $\mathbf{K}[\kappa] = \mathbf{K}[X_\kappa \cup R_\kappa, E_1^{(\kappa)} \cup E_2^{(\kappa)}]$, see Fig. 1 for an example. In the following subsections, we will exclusively investigate $\mathbf{K}(\kappa)$. We return to the induced subgraphs $\mathbf{K}[\kappa]$ in section 4.5 only.

Moreover, each CS κ' for this induced subgraph gives rise to a distinct subgraph $\mathbf{K}(\kappa')$, see Fig. 2, left. We note in passing that polynomial-delay algorithms exist for enumerating perfect matchings in bipartite graphs [36, 37, 38]. In the present work, however, we adopt a different approach to constructing the relevant child-selective subgraphs of \mathbf{K} . The following statement is a direct consequence of the fact that κ is a bijection:

Corollary 22. Let κ be a CS. Then every substrate vertex $x \in X_\kappa$ in $\mathbf{K}(\kappa)$ has out-degree 1 and every reaction vertex $r \in R_\kappa$ in $\mathbf{K}(\kappa)$ has in-degree 1.

The subsequent results provides us with a purely graph-theoretical characterization of the subgraphs of \mathbf{K} that derive from child selections.

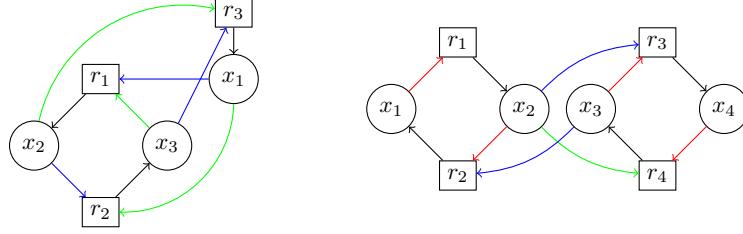


Fig. 2: Left: Multiple CS may exist in a given induced subgraph of \mathbf{K} . In the example, there are indeed two perfect matchings and thus two CS $\kappa_1 = (X_\kappa, R_\kappa, \kappa_1)$ and $\kappa_2(X_\kappa, R_\kappa, \kappa_2)$: $\kappa_1(x_1) = r_1$, $\kappa_1(x_2) = r_2$, $\kappa_1(x_3) = r_3$ and $\kappa_2(x_1) = r_2$, $\kappa_2(x_2) = r_3$, $\kappa_2(x_3) = r_1$. **Right:** A strongly connected child selective subgraph may not have a CS matrix with irreducible Metzler part. Here, choosing, $\kappa(x_1) = r_1$, $\kappa(x_2) = r_2$, $\kappa(x_3) = r_3$, and $\kappa(x_4) = r_4$ (edges depicted in red) yields a block diagonal $\mathfrak{M}(\mathbf{S}[\kappa])$ composed of two 2×2 blocks.

Lemma 23 (Proof: SI). *Let $\mathbf{K}' = (X' \cup R', E'_1 \cup E'_2)$ be a subgraph of \mathbf{K} with reactant vertices X' , reaction vertices R' , and edges $E_1 \subseteq X' \times R'$ and $E'_2 \subseteq R' \times X'$ such that*

1. $|X'| = |R'|$;
 2. *every $x \in X'$ has out-degree 1 and every $x \in R'$ has in-degree 1.*
- Then \mathbf{K}' is child-selective with $\kappa(x) = r$ for $(x, r) \in E'_1$.*

As an immediate consequence of Thm. 21 we note:

Corollary 24. *Let \mathbf{K} be an even elementary circuit graph. Then \mathbf{K} is child-selective with a unique Child-Selection κ . Moreover, $\mathbf{K}(\kappa) = \mathbf{K}$.*

Corollary 25. *If \mathbf{K} is not connected, then it is child-selective if and only if each weakly connected component is child-selective.*

Every child-selective subgraph $\mathbf{K}(\kappa)$ of the König graph of a reaction network can be associated with a matrix obtained from \mathbf{S} by removing the contributions of reactants to all reactions except those specified by the CS-bijection κ :

Definition 26. *For every child-selective subgraph of \mathbf{K}' with CS κ on $(X(\mathbf{K}'), R(\mathbf{K}'))$ we define the $k \times k$ matrix $\mathfrak{M}(\mathbf{S}[\kappa])$ with entries:*

$$\mathfrak{M}(\mathbf{S}[\kappa])_{xr} := \begin{cases} \mathbf{S}_{xr}[\kappa] & \text{if } x = r \text{ or } x \neq r \text{ and } \mathbf{S}_{xr}[\kappa] > 0 \\ 0 & \text{otherwise} \end{cases} \quad (32)$$

By construction, we have $\mathfrak{M}(\mathbf{S}[\kappa])_{xr} \neq \mathbf{S}[\kappa]_{xr}$ if and only if $r \neq \kappa(x)$ and $\mathbf{S}_{xr} < 0$, i.e., if and only if x is a reactant of a reaction r other than the one assigned to x by the CS-bijection κ . These entries correspond exactly to the negative off-diagonal elements of $\mathbf{S}[\kappa]$. Consequently, $\mathfrak{M}(\mathbf{S}[\kappa])$ contains only nonnegative off-diagonal entries and is therefore a Metzler matrix. We therefore call $\mathfrak{M}(\mathbf{S}[\kappa])$ the *Metzler part* of $\mathbf{S}[\kappa]$.

4.2 Irreducibility and Strong Connectedness

We start this section with two simple technical observations:

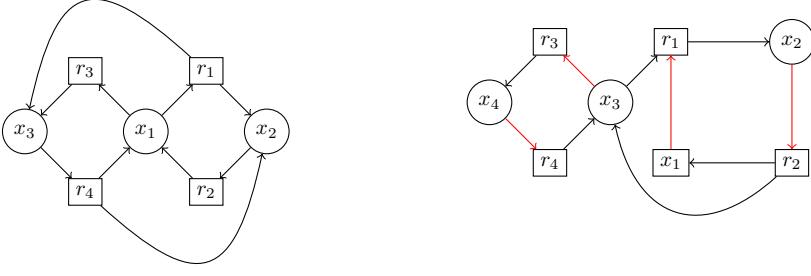


Fig. 3: **Left:** Strongly-connected bipartite graph without a cut-vertex but not child-selective, as the four reaction vertices are more than the three species vertices. **Right:** Strongly-connected bipartite digraph with a cut-vertex and child-selective (red edges).

Lemma 27 (Proof: SI). *Let $\kappa = (X_\kappa, E_\kappa, \kappa)$ be a CS. Then $\mathbf{K}(\kappa)$ is strongly connected if and only if $\mathfrak{M}(\mathbf{S}[\kappa])$ is irreducible.*

Lemma 28 (Proof: SI). *If $\mathbf{K}(\kappa)$ is strongly connected, then it does not contain a cut vertex*

Thus, if $\mathbf{K}(\kappa)$ is strongly connected, then its underlying undirected graph is also 2-connected. Such graphs are called *strong blocks* in the literature, see e.g. [39]. Combining Lemmas 27 and 28 yields the main result of this section:

Theorem 29. *Let κ be a CS. Then $\mathbf{K}(\kappa)$ is a strong block if and only if $\mathfrak{M}(\mathbf{S}[\kappa])$ is irreducible.*

The delicate interplay between the notions of child-selectiveness and strong blocks is exemplified in Fig. 3 and Fig 4. In particular, Fig. 4b and Fig. 4c depicts graphs that are strongly connected, albeit not strong blocks, but not child-selective. Fig. 4d and Fig. 4e exemplify strong-blocks that are not child-selective.

We conclude this section by revisiting the notion of autocatalysis and expressing Def. 5 in terms of the following graph-theoretic characterization of an autocatalytic CS. We recall the standard notions of *source* and *sink* vertices in a directed graph, i.e. respectively vertices with no incoming edges (zero in-degree) and with no outgoing edges (zero out-degree).

Lemma 30 (Proof: SI). *A CS $\kappa = (X_\kappa, E_\kappa, \kappa)$ is autocatalytic if and only if the following conditions both hold:*

1. *there is a positive vector $v > 0$ such that $\mathbf{S}[\kappa]v > 0$.*
2. *$\mathbf{K}(\kappa)$ does not possess source and sink vertices;*

Finally, our emphasis on Metzler parts and associated graphs $\mathbf{K}(\kappa)$ is justified by the following necessary condition.

Proposition 31. *If $\mathbf{S}[\kappa]$ is an autocatalytic CS matrix, then its Metzler part $\mathfrak{M}(\mathbf{S}[\kappa])$ is autocatalytic.*

Proof. We observe that $\mathfrak{M}(\mathbf{S}[\kappa])_{ij} \geq \mathbf{S}[\kappa]_{ij}$ for all i, j , and hence $\mathfrak{M}(\mathbf{S}[\kappa])v \geq \mathbf{S}[\kappa]v$ holds for every nonnegative vector v . \square

4.3 Fluffles and Circuitnets

Let us summarize the main arguments in the discussion so far. (1) By Prop. 31, we may restrict our attention to the Metzler parts of CS-matrices. (2) By Prop. 12, an arbitrary autocatalytic CS matrix is either irreducible or it contains one or more disjoint irreducible autocatalytic blocks, so we may restrict the attention to CS-matrices with irreducible Metzler part. (3) By Lemma 27 and Thm. 29, CS-matrices with irreducible Metzler part correspond to strong blocks $\mathbf{K}(\kappa)$. (4) Using Cor. 22, Lemma 23 and Thm. 29, we finally derive the following central observation:

Proposition 32. *Let G be a subgraph of \mathbf{K} . Then there is a CS κ with a CS-matrix that has an irreducible Metzler part $\mathfrak{M}(\mathbf{S}[\kappa])$ such that $G = \mathbf{K}(\kappa)$ if and only if G satisfies*

- (i) *G is bipartite with vertex partition $V(G) = X(G) \cup R(G)$ such that $|X(G)| = |R(G)|$,*
- (ii) *if $x \in X'$ then x has out-degree 1 and if $r \in R'$, then r has in-degree 1*
- (iii) *G is a strongly connected block.*

We will call a graph satisfying these three properties a *fluffle*¹, and we denote such subgraphs with G throughout. Prop. 32 in particular implies that the enumeration of fluffle subgraphs in the König graph \mathbf{K} encompasses all autocatalytic CS matrices with irreducible Metzler part. In the following, we describe how the fluffles can be constructed recursively.

Strongly connected blocks are precisely those graphs that admit an *open directed ear decomposition* [39], i.e., they can be constructed from an elementary circuit by iteratively adding open directed ears, which are directed paths whose endpoints are distinct vertices already present in the graph. Throughout the remainder of this contribution, we refer to open directed ear decompositions and open directed ears simply as “ear decompositions” and “ears”, respectively. After attaching an ear, the interior vertices of the ear have in-degree and out-degree 1, while its initial vertex has out-degree > 1 and its terminal vertex has in-degree > 1 . In our bipartite setting, we then have the following theorem characterizing fluffle graphs.

Theorem 33 (Proof: SI). *A graph G is a fluffle if and only if it is bipartite with vertex set $X \cup R$ and it has an ear decomposition such that every ear initiates in a reaction vertex $r \in R$ and terminates in a substrate vertex $x \in X$. In this case, all directed open ear decompositions have this property.*

In the following, it will be useful to note that strong blocks in fluffles are again fluffles themselves:

Lemma 34 (Proof: SI). *Let G be a fluffle in \mathbf{K} and G' a subgraph of G that is a strong block. Then G' is a fluffle.*

Clearly, every elementary circuit in \mathbf{K} is a fluffle. It is therefore natural to identify fluffles as unions of elementary circuits. According to Thm. 33, however, such unions must be consistent with an ear decomposition in which each ear originates at a reaction vertex and terminates at a substrate vertex. This requirement is made explicit in the following theorem. See also Fig. 4 for an overview of possible ways to combine two elementary circuits, only one of which actually constitutes a fluffle.

Theorem 35 (Proof: SI). *Let G be a fluffle with vertex partition $X \cup R$ and C an elementary circuit such that $\emptyset \subset G \cap C \subset C$. Then, the connected components of $G \cap C$*

¹Fluffle is an informal, whimsical term for a group of rabbits, motivated by the rabbit's ears in our context.

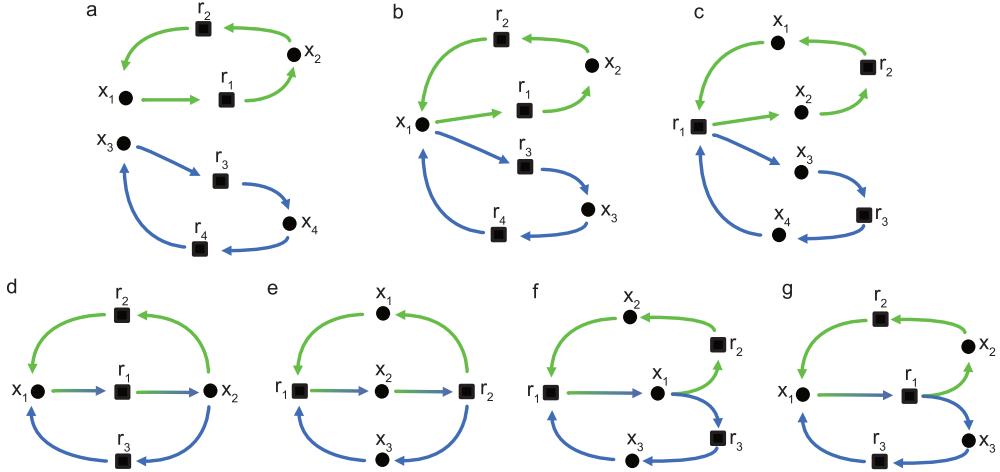


Fig. 4: Unions of two elementary circuits, depicted in green and blue, yielding different configurations: **a** is not connected and thereby violates the fluffle condition (iii) in Prop. 32, **b-e** possess more metabolites than reaction vertices or vice-versa and contradict fluffle condition (i) in Prop. 32. In addition, **b** and **c** are also not a strong block and thus violate (iii) as well. The union **f** has a substrate vertex x_1 with out-degree and a reaction vertex r_1 with in-degree two, respectively, which contradicts fluffle condition (ii) in Prop. 32. The only combination consistent with the fluffle definition is depicted in **g** where indeed the intersection of the two elementary circuits is a path from a substrate to a reaction vertex, as prescribed by Thm. 35. The two elementary circuits in **d, e, f**, moreover, also, constitute an example of circuitnets that are not associated to fluffles.

are directed paths P_i . Moreover, $G \cup C$ is a fluffle if, and only if, all such paths P_i start from a substrate vertex $x_i \in X$ and terminate with a reaction vertex $r_i \in R$.

Definition 36. A set of $\mathcal{C} = \{C_1, \dots, C_n\}$ of elementary circuits in \mathbf{K} is a circuitnet if there is an ordering π such that the union $G_k := \bigcup_{i=1}^k C_{\pi(i)}$ is a strong block for all $1 \leq k \leq h$.

We say that \mathcal{C} is a circuitnet for a graph G if G , as a graph, is the union of all the elementary circuits in \mathcal{C} , and we write $G = \bigcup(\mathcal{C})$. The next theorem guarantees that there is a circuitnet for any fluffle G .

Theorem 37. Let G be a fluffle. Then there exists a circuitnet \mathcal{C} for G , i.e.,

$$G = \bigcup(\mathcal{C}). \quad (33)$$

Proof. The statement follows directly from the well-known connection of elementary circuits, ear decompositions, and cycles bases: The cycle space of strongly-connected digraphs has a circuit basis [40], and for strong blocks such a basis can be constructed from an ear decomposition by completing each ear P_i to a directed circuit C_i using any

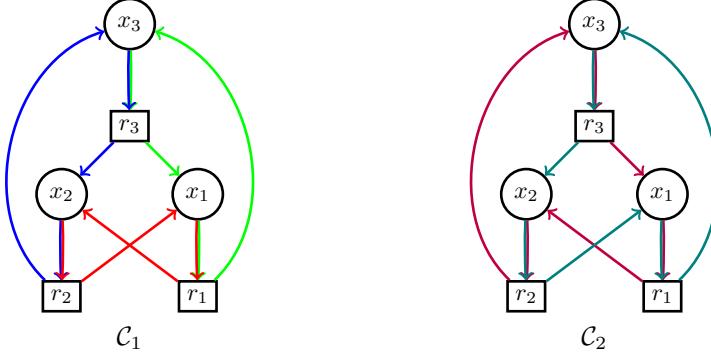


Fig. 5: The fluffle corresponding to an autocatalytic core of Type V, Eq. (30). The fluffle F is depicted as a union of the elementary circuits in two *different* circuitnets. The circuitnet $\mathcal{C}_1 = \{C_1, C_2, C_3\}$ (left) comprises the three elementary circuit $C_1 = (x_1, r_1, x_3, r_3, x_1)$ (green), $C_2 = (x_2, r_2, x_3, r_3, x_2)$ (blue), and $C_3 = (x_1, r_1, x_2, r_2, x_1)$ (red); the circuitnet $\mathcal{C}_2 = \{C_4, C_5\}$ (right) consists of the two elementary circuits $C_4 = (x_1, r_1, x_3, r_3, x_2, r_2, x_1)$ (teal) and $C_5 = (x_1, r_1, x_2, r_2, x_3, r_3, x_1)$ (purple). We have $F = \bigcup(\mathcal{C}_1) = \bigcup(\mathcal{C}_2)$.

directed path in G_{i-1} from the terminal to the initial vertex of the ear. In particular, therefore, every fluffle has a circuitnet. \square

Thm. 37 guarantees that there exists at least one circuitnet for a fluffle G . In general, however, there may exist more circuitnets for the same fluffle. An example is depicted as the autocatalytic core of type V, (30): see Fig. 5. A simple corollary follows from Thm. 37.

- Corollary 38.** *Let \mathcal{C} be a circuitnet for the fluffle G . Then the following hold true.*
1. *$\mathcal{C}' \subseteq \mathcal{C}$ is a circuitnet for a fluffle $G' \subseteq G$ if and only if G' is a strong block.*
 2. *There exist an ordering π of the circuits in \mathcal{C} such that $G_k = \bigcup_{i=1}^k C_{\pi(i)}$ is a fluffle.*

Proof. The first statement follows directly from Lemma 34 and the second statement from the first and Def. 36. \square

4.4 The set of fluffles of a CRN

Summarizing the discussion so far, we have shown that all CS-matrices in Γ with an irreducible Metzler part, i.e., the viable candidates for “interesting” autocatalytic sub-networks, are the fluffles in the associated bipartite König graph \mathbf{K} , (Prop. 32). Moreover, any fluffle can be constructed by superimposing elementary circuits such that each intermediate step is itself a fluffle, (Cor. 38). Equivalently, this boils down to enumerating the sets of circuitnets whose union is a fluffle. Elementary circuits can be enumerated efficiently in a lazy manner [41, 42, 43] with linear delay, i.e., $O(|V| + |E|)$. Moreover, algorithms exist that allow to restrict circuit length [44].

Denote by \mathfrak{F} the set of circuitnets $\mathcal{C} \in \mathfrak{F}$ whose union $\bigcup(\mathcal{C})$ is a fluffle. In SI Sec. C we summarize the properties of \mathfrak{F} that enable efficient enumeration. The example

in Fig. 5 suggests to investigate the equivalence relation on \mathfrak{F} defined by $\mathcal{C}_1 \sim \mathcal{C}_2$ iff $\bigcup(\mathcal{C}_1) = \bigcup(\mathcal{C}_2)$. These equivalence classes are specified by subsets of edges in \mathbf{K} . More precisely, we have $\mathcal{C}_1 \sim \mathcal{C}_2$ if and only if $E(\bigcup(\mathcal{C}_1)) = E(\bigcup(\mathcal{C}_2))$ because the corresponding vertex set is given implicitly by the vertices incident with these edge sets. Our primary aim, however, is not to enumerate fluffles, but to enumerate CS-matrices $\mathbf{S}[\kappa]$ with irreducible Metzler parts. To this end, we recall that the CS κ is equivalent to the substrate-reaction edges in a fluffle, hence these edges completely determine $\mathbf{S}[\kappa]$. This suggests to consider the following equivalence relation.

Definition 39. Let \mathcal{C}_1 and \mathcal{C}_2 be two circuitnets for fluffles G_1 and G_2 . We say that \mathcal{C}_1 and \mathcal{C}_2 are CS-equivalent and we write $\mathcal{C}_1 \simeq \mathcal{C}_2$ if

$$E_1\left(\bigcup(\mathcal{C}_1)\right) = E_1\left(\bigcup(\mathcal{C}_2)\right). \quad (34)$$

Let now $\kappa_1 = (X_{\kappa_1}, R_{\kappa_1}, \kappa_1)$ and $\kappa_2 = (X_{\kappa_2}, R_{\kappa_2}, \kappa_2)$ be two CS. We recall that we consider two CS-matrices $\mathbf{S}[\kappa_1]$ and $\mathbf{S}[\kappa_2]$ are the same if

$$X_{\kappa_1} = X_{\kappa_2}, \quad R_{\kappa_1} = R_{\kappa_2}, \quad \kappa_1 = \kappa_2. \quad (35)$$

In other words, we consider two CS-matrices the same if they involve the same species, reactions and bijection between species and reactions. We do not require that the ordering of the rows is the same. That is, we consider $\mathbf{S}[\kappa_1]$ and $\Pi^\top \mathbf{S}[\kappa_2] \Pi$ to be same for any permutation matrix Π on $X_\kappa \times X_\kappa$. With this notion of same-ness, we obtain the following result.

Lemma 40 (Proof: SI). Two circuitnets \mathcal{C}_1 and \mathcal{C}_2 for fluffles G_1 and G_2 yield the same CS matrix $\mathbf{S}[\kappa]$ if and only if $\mathcal{C}_1 \simeq \mathcal{C}_2$.

For example, consider the autocatalytic core of type V (30) depicted in Fig. 5. The two circuitnets $\mathcal{C}_1 = \{C_1, C_2, C_3\}$ and $\mathcal{C}_2 = \{C_4, C_5\}$ both are associated to the same fluffle and thus in particular they are CS-equivalent and give rise to the same CS-matrix:

$$\mathbf{S}[\kappa] = \begin{pmatrix} -1 & 1 & 1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \end{pmatrix}, \quad (36)$$

since $E_1(\mathcal{C}_1) = E_1(\mathcal{C}_2)$. Remarkably, 28 (!) circuitnets in the power set $\mathfrak{P}(\{C_1, C_2, C_3, C_4, C_5\})$ – all except for the empty set and the three singletons $\{C_1\}$, $\{C_2\}$, and $\{C_3\}$, which correspond to elementary circuits involving only two species – induce the same CS matrix (Eq. 36) and therefore belong to the same equivalence class. Notably, even distinct elementary circuits, such as C_4 and C_5 in this example, can be CS-equivalent. En passant, we further note that all autocatalytic cores $\mathbf{S}[\kappa]$, with the sole exception of type III, admit a single-circuit representative in the CS-equivalence class of circuitnets associated with $\mathbf{S}[\kappa]$, see SI Example 1. The next Lemma is central in exploiting the CS-equivalence to lighten the computational cost of our approach.

Lemma 41 (Proof: SI). Let \mathcal{C}_1 and \mathcal{C}_2 be circuitnets for fluffles G_1 and G_2 , respectively, and let C' and C'' be two elementary circuits. Assume $\mathcal{C}_1 \simeq \mathcal{C}_2$, $C' \simeq C''$ and

$G'_1 := \bigcup(\mathcal{C}_1 \cup \{C'\})$ is a fluffle. Then $G'_2 := \bigcup(\mathcal{C}_2 \cup \{C''\})$ is a fluffle as well with $\mathcal{C}_1 \cup \{C'\} \simeq \mathcal{C}_2 \cup \{C''\}$.

The converse of Lemma 41 does not hold. A counterexample is again the autocatalytic core of type V (30) depicted in Fig. 5. Consider indeed the two circuitnets $\mathcal{C}_2 := \{C_4, C_5\}$ and $\mathcal{C}_3 := \{C_1, C_2\}$. Note that the circuitnet \mathcal{C}_3 is obtained from $\mathcal{C}_1 = \{C_1, C_2, C_3\}$ by removing the elementary circuit C_3 , red in Fig. 5. Even if $\mathcal{C}_2 \simeq \mathcal{C}_3$ holds, the parallel removal of any elementary circuit from the two circuitnets destroy the equivalence relation as $C_i \neq C_j$ for $i = 4, 5$, and $j = 1, 2$: the elementary circuits in \mathcal{C}_2 comprise three species while the elementary circuits in \mathcal{C}_3 comprise two.

However, we may still ensure that a representative of each CS-equivalence class can be reached by adding an elementary circuit to the representative of some CS-equivalence class of a smaller fluffle. More precisely, we have the following Lemma.

Lemma 42 (Proof: SI). *For every CS-equivalence class $[\mathcal{C}]$ there is a representative $\hat{\mathcal{C}}$ such that there exists a CS-equivalence class $[\mathcal{C}']$ with representative $\hat{\mathcal{C}}'$ and an elementary circuit C^* such that $\hat{\mathcal{C}}' \cup \{C^*\} \simeq \hat{\mathcal{C}}$ and $|V(\bigcup(\mathcal{C}'))| < |V(\bigcup(\mathcal{C}))|$.*

Finally, we may naturally extend the notion of CS-equivalence to fluffles.

Definition 43. *Two fluffles G, G' are CS-equivalent if and only if $E_1(G) = E_1(G')$.*

Moreover, we observe that each \simeq -equivalence class has a natural fluffle representative, whose graph is indeed $\mathbf{K}(\kappa)$:

Proposition 44. *Let G be a fluffle and let κ be the CS defined by $E_1(G)$. Consider now the graph \bar{G} with vertex set $V(\bar{G}) = V(G)$ and edge set $E_1(G) \cup (E_2(V(G)))$, i.e., $e \in (E_2(V(G)))$ if and only if $e = (r_1, x_2)$ with $r_1, x_2 \in V(G)$. Then \bar{G} is a fluffle with $\bar{G} \simeq G$, and such that $\bar{G} = \mathbf{K}(\kappa)$.*

Proof. Since G is a fluffle of \mathbf{K} , then $E_1(G)$ is a perfect matching corresponding to the CS κ , and thus $\bar{G} = \mathbf{K}(\kappa)$ by Eq. (31). By Prop. 32, \bar{G} is a fluffle as well. As $E_1(\bar{G}) = E_1(G)$ holds by construction, we have $\bar{G} \simeq G$. \square

Taken together, we therefore can avoid the enumeration of \mathfrak{F} in favor of enumerating fluffle representatives only.

4.5 Metzler matrices and induced fluffles

Recall that by Prop 31 $\mathbf{S}[\kappa]$ corresponds to the *induced* subgraph $\mathbf{K}[\kappa]$, while $\mathfrak{M}(\mathbf{S}[\kappa])$ corresponds to the fluffle $\mathbf{K}(\kappa)$ defined on the same vertex set $X_\kappa \cup R_\kappa$. Therefore $\mathbf{K}(\kappa)$ is always a spanning subgraph of $\mathbf{K}[\kappa]$. However, in general, we have $\mathbf{K}[\kappa] \neq \mathbf{K}(\kappa)$, as shown in Fig. 1. Nevertheless, many properties of $\mathbf{K}(\kappa)$ translate to the induced subgraphs. In this section we collect some of these implications, which will be useful below in the context of autocatalytic cores.

As an immediate consequence of Lemma 23 we obtain

Corollary 45. *An induced subgraph $\mathbf{K}[X' \cup R']$ is child-selective if and only if it contains a spanning subgraph G with edge set E' such that $E' \cap (X' \times R')$ is a perfect matching in G .*

Theorem 46. *An induced subgraph $\mathbf{K}' \subseteq \mathbf{K}$ has a CS Metzler matrix $\mathbf{S}[\kappa']$ if and only if $E_1(\mathbf{K}')$ is a perfect matching in \mathbf{K}' .*

Proof. If E_1 is a perfect matching in \mathbf{K}' then each $x \in X(\mathbf{K}')$ has only one outgoing edge and each $r \in R(\mathbf{K}')$ has only one incoming edge. Hence, for each $r \in R(\mathbf{K}')$ there is a unique $x \in X(\mathbf{K}')$ with $s_{xr}^- > 0$, while for all $y \in X(\mathbf{K}')$ with $x \neq y$, $s_{yr}^- = 0$. Thus $\mathbf{S}[\kappa] = \mathfrak{M}(\mathbf{S}[\kappa])$ and hence a Metzler matrix. If, on the other hand, $\mathbf{S}[\kappa]$ is a Metzler matrix, for each $r \in R(\mathbf{K}')$ there is exactly one $x \in X(\mathbf{K}')$ with $s_{xr}^- > 0$, which implies that r has exactly one incoming edge given by $(\kappa^{-1}(r), r)$. Since κ is a bijection, κ^{-1} is in particular injective and hence these edges form perfect matching, which by construction coincides with E_1 . \square

If $\mathbf{S}[\kappa] = \mathfrak{M}(\mathbf{S}[\kappa])$ then $\mathbf{K}(\kappa)$ already contains all edges of $\mathbf{K}[\kappa]$. Conversely, if $\mathbf{K}(\kappa) = \mathbf{K}[\kappa]$, then E_1 is a perfect matching in this subgraph and hence $\mathbf{S}[\kappa]$ is Metzler.

Corollary 47. *Let κ be a CS. Then the following statements are equivalent:*

- (i) $\mathbf{S}[\kappa]$ is a Metzler matrix
- (ii) $\mathbf{K}[\kappa] = \mathbf{K}(\kappa)$.

Since $\mathbf{K}(\kappa)$ is a spanning subgraph of $\mathbf{K}[\kappa]$, the following two observations are straightforward:

Corollary 48. *If κ is a CS and $\mathfrak{M}(\mathbf{S}[\kappa])$ is irreducible then $\mathbf{K}[\kappa]$ is strongly connected.*

Corollary 49. *If κ is a CS and $\mathfrak{M}(\mathbf{S}[\kappa])$ is irreducible then $\mathbf{K}[\kappa]$ is a strong block.*

The statement of Lemma 28, however, does not hold for $\mathbf{K}[\kappa]$ (see Fig. 3, right). That is, a strongly connected induced subgraph $\mathbf{K}[\kappa]$ is not necessarily a strong block.

Moreover, the fact that $\mathbf{K}[\kappa]$ is strongly connected does not imply that $\mathfrak{M}(\mathbf{S}[\kappa])$ is irreducible. As a counterexample consider the disjoint union of two even elementary circuits, one containing a substrate vertex x_2 and the other x_3 (Fig. 2, right). This graph is child-selective, with $\kappa(z)$ defined as the successor of each substrate vertex z along its circuit. Adding the two edges $(x_2, \kappa(x_3))$ and $(x_3, \kappa(x_2))$ (blue edges in the figure) produces a strongly connected graph that remains child-selective under the same map κ . Nevertheless, $\mathbf{K}(\kappa)$ remains the disjoint union of the two circuits and thus is not strongly connected. In this example, an alternative Child-Selection exists by defining $\kappa'(z) = \kappa(z)$ for $z \notin \{x_2, x_3\}$, $\kappa'(x_2) = \kappa(x_3)$, and $\kappa'(x_3) = \kappa(x_2)$, though such a construction is not always possible. If only $(x_3, \kappa(x_2))$ of the two blue edges was present, the subgraph would still be strongly connected if there existed an additional edge (x_2, r) to a reaction $r \neq \kappa(x_3)$ in the second circuit (green edge). Suppose there is a perfect matching M that includes (x_2, r) . Then necessarily $(\kappa^{-1}(r), r) \notin M$. Since $\kappa^{-1}(r) \neq x_3$, the vertex $\kappa^{-1}(r)$ has only one successor, r , along its circuit, implying that M cannot be a perfect matching in E_1 .

We can, however, rephrase Lemma 30 in terms of the induced subgraph $\mathbf{K}[\kappa]$, making use again of the fact that $\mathbf{K}(\kappa)$ is a spanning subgraph of $\mathbf{K}[\kappa]$:

Corollary 50. *A CS $\kappa = (X_\kappa, E_\kappa, \kappa)$ is autocatalytic if and only if the following conditions both hold:*

1. *there is a positive vector $v > 0$ such that $\mathbf{S}[\kappa]v > 0$.*
2. *$\mathbf{K}[\kappa]$ does not possess source and sink vertices;*

Recall that autocatalytic cores are in particular irreducible Metzler CS matrices. Autocatalytic cores thus satisfy $\mathbf{S}[\kappa] = \mathfrak{M}(\mathbf{S}[\kappa])$, i.e., $\mathbf{K}[\kappa] = \mathbf{K}(\kappa)$. In other words, all autocatalytic cores correspond to *induced* fluffles. Prop. 44 thus implies that the

natural representative of the CS-equivalence class of an irreducible Metzler CS, and thus in particular of an autocatalytic core is an induced fluffle.

Finally, we show that irreducible Metzler CS matrices $\mathbf{S}[\kappa]$ are autocatalytic whenever they contain an autocatalytic core. Our starting point is the following technical result, which then enables us to state the main result of this subsection.

Lemma 51 (Proof: SI). *Let $\mathbf{K}(\kappa^*)$ be a fluffle with irreducible autocatalytic Metzler CS matrix $\mathbf{S}[\kappa^*]$ and let $\mathbf{K}(\kappa)$ be obtained from $\mathbf{K}(\kappa^*)$ by adding a single ear with initial vertex in $R(\mathbf{K}(\kappa^*))$, terminal vertex in $X(\mathbf{K}(\kappa^*))$, and a non-empty set of internal vertices, together with all reaction-to-metabolite edges in $R(\mathbf{K}(\kappa)) \times X(\mathbf{K}(\kappa))$. If $\mathbf{S}[\kappa^*]$ is an autocatalytic CS matrix and $\mathbf{S}[\kappa]$ is a Metzler matrix, then $\mathbf{S}[\kappa]$ is autocatalytic irreducible CS matrix.*

Theorem 52 (Proof: SI). *Let $\mathbf{S}[\kappa]$ be an irreducible Metzler CS matrix and suppose $\mathbf{S}[\kappa]$ contains an autocatalytic core $\mathbf{S}[\kappa^*]$ as a principal submatrix. Then $\mathbf{S}[\kappa]$ is autocatalytic.*

We conclude this section with a close look at autocatalytic cores. Consider a fluffle G and recall that for any fluffle H with $E_1(H) \subseteq E_1(G)$ the matrix $\mathbf{S}[E_1(H)]$ is a principal submatrix of $\mathbf{S}[E_2(G)]$. Since an autocatalytic core is irreducible and thus a superposition of elementary circuits, we immediately observe the following corollary:

Corollary 53. *Let G be an autocatalytic core in \mathbf{K} and \mathcal{C} a circuitnet for G . Then $\mathbf{S}[E_1(C)]$ is a Metzler CS matrix for every $C \in \mathcal{C}$.*

It is therefore of interest to consider the following subclass of elementary circuits:

Definition 54. *A Metzler circuit in $\mathbf{K} = (X \cup R, E)$ is an elementary circuit without a chord of the form $(x, r) \in X \times R$.*

Cor. 53 thus implies that every autocatalytic core can be constructed from the Metzler circuits in \mathbf{K} alone.

5 Algorithms

5.1 Overview

The mathematical results in the previous two sections serve as basis for a practical strategy to enumerate irreducible autocatalytic CS matrices from the König graph \mathbf{K} of a CRN. Conceptually, we can break up this task into four steps:

- (1) Enumeration the elementary circuits of \mathbf{K} and determine a single representation of each CS-equivalence class.
- (2) Construction of representatives of larger CS-equivalence classes of fluffles by iteratively adding elementary circuits.
- (3) Testing of the these candidate CS matrices $\mathbf{S}[\kappa]$ for autocatalysis.
- (4) Identification of autocatalytic cores and classification of centered autocatalytic CS Metzler matrices.

The algorithms addressing these basic tasks are described in detail in Sec. 5.2.

The direct application of this strategy to large CRNs, in particular to sized metabolic networks, requires prohibitive computational resources. We observe, however, that the König graph of metabolic networks is rather sparsely connected. Moreover, it typically contains modules with higher internal connectivity that often can be identified with functional biological submodules [45, 46, 47, 48, 49]. We use this

structure to decompose the CRN into smaller parts following a divide-and-conquer approach. Interfaces between submodules, however, may also be part of autocatalytic subsystems. We therefore consider elementary circuits that connect modules in the final stage. A major advantage of the decomposition into modules is that their analysis can be trivially parallelized, making it possible in practise to tackle large metabolic networks.

5.2 Basic Algorithms

5.2.1 Elementary circuits

Johnson's algorithm [42] enumerates all elementary circuits of a direct graph with linear delay. Since we only need an arbitrary representative of each CS-equivalence class, it suffices to record the sets $E_1(C)$. We denote the set of representatives by \mathbf{C} .

5.2.2 Recognition of fluffles

Based on Thm. 35 and Prop. 44 the key task is to expand a representative fluffle G by a representative circuit C and to test whether the union $G \cup C$ is again a fluffle. The following result greatly simplifies this task:

Lemma 55 (Proof: SI). *Let G be a fluffle and C an elementary circuit. Then $G \cup C$ is a fluffle if and only if $\emptyset \neq V(G) \cap V(C) = V(E_1(G) \cap E_1(C))$.*

As a consequence of Lemma 55 it suffices to consider only the edge sets $E_1(G)$ and $E_1(C)$ and their incident vertex sets when constructing representatives of CS-equivalence classes of fluffles. This considerably simplifies the practical implementation since we do not have to maintain graph data structures for the fluffles. By Prop. 44, we may use $\bar{G} = \mathbf{K}(\kappa)$ as canonical representative of $[G]$, where κ is the CS defined by $E_1(G)$. If desired, circuitnets for the fluffle \bar{G} can be re-constructed in linear time by means of a directed ear decomposition.

5.2.3 Representatives of CS-equivalence classes

As a consequence of Lemma 42 and Def. 43, a representative fluffle for each CS-equivalence class of \mathfrak{F} can be obtained, sparsely, by computing unions of elementary circuits with representative fluffles of CS-equivalence classes with fewer vertices. To this end, we start from the set \mathbf{C} of representatives of elementary circuits and initialize a queue Q with these elementary circuits. Q will contain an arbitrary representative of each CS-equivalence class of fluffles for further expansion. We maintain a separate set \mathcal{E} of all representatives as the output of the algorithm. The queue Q is processed in first-in-first-out order.

For each $G \in Q$, all representative elementary circuits $C \in \mathbf{C}$ are tested for $G \cup C$ constituting a fluffle by means of Lemma 55. If so, $G' = G \cup C$ serves as a representative of the CS-equivalence class $[G']$. If there is a fluffle $G'' \in \mathcal{E}$ such that $G'' \in [G']$, i.e., $E_1(G'') = E_1(G')$ then G' is discarded without changes to Q . Otherwise $[G']$ is appended to Q . If a concurrent hashset to Q is maintained with all representative edgesets as keys, then the comparison of $E_1(G')$ and $E_1(G'')$ can be performed in constant time. The algorithm terminates when Q is empty, i.e., all maximal fluffles

Algorithm 1: Assembly of equivalence classes.

```

Require:  $\mathbf{C}$  : Set of elementary circuits
Output :  $\mathcal{E}$  : Set CS-equivalence classes
 $Q \leftarrow \emptyset; \mathcal{E} \leftarrow \emptyset;$  // Init. empty queue and Output
Initialize hash map  $M$  ; // Edge → element. circuits
for  $C \in \mathbf{C}$  do
  if  $E_1(C) \notin \mathcal{E}$  then
     $\mathcal{E} \leftarrow \mathcal{E} \cup \{E_1(C)\}; Q \leftarrow Q \cup \{E_1(C)\}$  for  $e \in E_1(C)$  do
      if  $e \in M$  then
         $M[e] \leftarrow M[e] \cup \{E_1(C)\};$ 
      else
         $M[e] \leftarrow \{E_1(C)\};$ 
  while  $Q \neq \emptyset$  do
     $E_1(G) \leftarrow Q.pop();$ 
    for  $E_1(C) \in \bigcup_{e \in E_1(G)} M[e]$  do
      if  $E_1(C) \subseteq E_1(G)$  or  $E_1(G) \subseteq E_1(C)$  then
        continue
       $V(G) \leftarrow V(E_1(G)); V(C) \leftarrow V(E_1(C));$ 
      if  $\emptyset \neq V(G) \cap V(C) = V(E_1(G) \cap E_1(C))$  then
         $E_1(G') \leftarrow E_1(G) \cup E_1(C);$ 
        if  $E_1(G') \notin \mathcal{E}$  then
           $\mathcal{E} \leftarrow \mathcal{E} \cup \{E_1(G')\}; Q \leftarrow Q \cup \{E_1(G')\};$ 

```

have been found. Finding overlapping elementary circuits C for each fluffle G can be sped up by a hashmap M linking edges in E_1 to sets of circuits they are contained in. We only have to consider pairs were at least one such edge is shared. A pseudocode for this procedure is given in Alg. 1.

5.2.4 Identification of autocatalytic matrices and autocatalytic cores

Each entry in \mathcal{E} obtained by Alg. 1 constitutes a candidate $\mathbf{S}[\kappa]$ for an irreducible autocatalytic CS matrix. Since every irreducible autocatalytic CS matrix $\mathbf{S}[\kappa]$ contains an autocatalytic core, it either satisfies $\mathbf{S}[\kappa] = \mathfrak{M}(\mathbf{S}[\kappa])$ or it strictly contains a principal submatrix $\mathbf{S}[\kappa']$ with this property. In the latter case, κ' is a restriction of κ and thus $\mathbf{K}(\kappa')$ is a proper subgraph of $\mathbf{K}(\kappa)$. In fact, by Cor. 47, $\mathbf{K}(\kappa') = \mathbf{K}[\kappa']$ must also be an induced subgraph of $\mathbf{K}(\kappa)$. Moreover, we have $E_1(\mathbf{K}(\kappa')) \subseteq E_1(\mathbf{K}(\kappa))$ if and only if κ' is a restriction of κ . Thus we have the following necessary condition:

Corollary 56. *If G is a fluffle that defines an autocatalytic CS matrix, then there is an induced subgraph G' of G such that G' is also an induced subgraph of \mathbf{K} .*

On the other hand, if $\mathbf{S}[E_1(G)]$ is a Metzler matrix, i.e., if G is an induced fluffle representative, and G contains an autocatalytic core, then G is itself autocatalytic by Thm. 52.

These simple observations suggest computing the Hasse diagram with respect to set inclusion, $\text{Hasse}(\mathcal{E})$, for sets \mathcal{E} of fluffle equivalence classes. Traversing $\text{Hasse}(\mathcal{E})$ in bottom-up order, one then checks, for each \subseteq -minimal candidate G in \mathcal{E} :

- (a) whether $G = \mathbf{K}[V(E_1(G))]$, which is equivalent to G giving rise to a Metzler CS matrix $\mathbf{S}[E_1(G)]$, and, if so,
- (b) whether $\mathbf{S}[E_1(G)]$ is Hurwitz unstable.

A minimal element in (\mathcal{E}, \subseteq) that satisfies (a) and (b) is an autocatalytic core. Moreover, any Metzler matrix that contains an autocatalytic is automatically autocatalytic, and in this case test (b) can be omitted. Taken together, an explicit test for autocatalytic needs to be performed only for inclusion-minimal induced fluffe representatives that are Metzler and for non-minimal non-Metzler matrices in \mathcal{E} . If one of the conditions (a) or (b) is violated, G is removed from the Hasse(\mathcal{E}) and each parents of G is connected to each immediate descendant of G . Upon completion of the traversal, all minimal elements in Hasse(\mathcal{E}) are autocatalytic cores and thus Metzler matrices. Moreover, all descendants of a Metzler matrix in Hasse(\mathcal{E}) are again Metzler. Similarly, all parents of a non-Metzler matrix are again non-Metzler. For each of the non-Metzler matrices we explicitly test whether they are autocatalytic using an LP solver [50] to determine whether there is a vector $v > 0$ such that $\mathbf{S}[\kappa]v > 0$.

Determining the complete structure of the Hasse diagram, however, severely compromises performance and simply testing all CS-equivalence classes for their autocatalytic capacity would be more efficient. In contrast, predecessor relations are sufficient to avoid unnecessary testing and can be obtained without additional costs. By Alg 1, for each element G retrieved from Q , subset relations with all CS-equivalence classes of elementary circuits intersecting in at least one $e \in E_1(G)$ are determined. These subset relations, however, define predecessor relations in Hasse(\mathcal{E}) and can be utilized to avoid unnecessary testing for autocatalysis as suggested. By construction, only CS-equivalence of elementary circuits can be leaves. If their set of predecessors is empty, they can be excluded from testing whenever their associated CS matrix is non-Metzler. In the Metzler case, all predecessors are recursively screened for autocatalytic capacity of their associated Metzler matrix. Whenever this is the case, the search is stopped and testing can be omitted. Additional flags avoid visiting and testing an element twice.

5.2.5 Centralized Autocatalysis

To test whether an autocatalytic Metzler matrix $\mathbf{S}[\kappa]$ on (X_κ, R_κ) exhibits centralized autocatalysis, counting of elementary circuits passing each $x \in X_\kappa$ is required. Since $\kappa : X_\kappa \rightarrow R_\kappa$ is bijective, there is a 1-1 correspondence between circuits in the induced subgraph $\mathbf{K}[\kappa]$ and the graph with vertex set X_κ and edges (x, y) whenever $\mathbf{S}[\kappa]_{yx} > 0$. It therefore suffices to enumerate the set \mathcal{C} of elementary circuits in $\mathbf{K}[\kappa]$, e.g., using Johnson's algorithm and to store the vertices $V(C) \cap X_\kappa$ in a bit vector ζ_C for each $C \in \mathcal{C}$. The component-wise conjunction of these vectors

$$\zeta^* := \bigwedge_{C \in \mathcal{C}} \zeta_C \quad (37)$$

identifies the set of autocatalytic centers as $M = \{x | \zeta_x^* = 1\}$. The autocatalytic Metzler matrix $\mathbf{S}[\kappa]$ is therefore centralized if and only if there is an $x \in X_\kappa : \zeta_x^* \neq 0$.

5.3 Direct enumeration of autocatalytic cores

Since autocatalytic cores are necessarily Metzler matrices, it is possible to modify Alg. 1 to enumerate autocatalytic cores only: first, the enumeration of elementary circuits is restricted to Metzler circuits (Def. 54) since by Cor. 53 every induced fluffle, and thus every candidate for an autocatalytic core is a union of Metzler circuits. Moreover, if C_1, C_2, \dots, C_h is a circuitnet for a Metzler fluffle G and each C_i is a Metzler fluffle, then any fluffle $G_k = \bigcup_{i=1}^k C_{\pi(i)}$ (Cor. 38) leads to a corresponding Metzler matrix $\mathbf{S}[E_1(G_k)]$ since G_k is fluffle subgraph of an induced fluffle. Thereby, $\mathbf{S}[E_1(G_k)]$ is a principal submatrix of a Metzler matrix and hence itself a Metzler matrix. Therefore, all non-Metzler fluffles can be discarded immediately. Moreover, if $E_1(G)$ is an autocatalytic core, none of its extensions can be cores. Hence, $E_1(G)$ is only pushed on the queue Q if it corresponds to an induced fluffle and $\mathbf{S}[E_1(G)]$ is not autocatalytic, while only the induced autocatalytic cases are added to the output \mathcal{E} . This procedure, however, is not guaranteed to detect all predecessor relationships between autocatalytic matrices. The resulting false positive candidates can be identified in a post-processing step by checking whether there are subset relationships among the core candidates in \mathcal{E} . This inclusion testing can be parallelized to increase efficiency. However, empirical tests revealed that a different strategy performs better: pushing all elementary circuits and larger fluffles, along with their associated autocatalytic CS Metzler matrices, into the queue while restricting their processing to inclusion-relation detection only. This approach drastically reduces the number of required set-inclusion tests between candidates and therefore offers a substantial performance advantage.

5.4 Extensions for large CRNs

With increasing network size, the number of expected elementary circuits grows exponentially. Exhaustive enumeration of elementary circuits as required by Alg. 1 therefore becomes infeasible for large CRNs. A natural restriction is to limit the size of circuits, at the cost of also limiting the size of resulting fluffles in the assembly. In biochemical networks, one may expect that autocatalytic subsystems are predominantly confined to functional modules or pathways. We therefore aim to enumerate circuits first within such modules and only then extend the search for circuits to connections between modules. We proceed in two steps: the network is clustered and elementary circuits within units are enumerated exhaustively. Then circuits crossing (typically sparsely connected) borders of neighboring clusters (as identified by the cluster partition tree) are enumerated with size restrictions.

A useful decomposition of a CRN should ideally generate sub-networks of roughly equal size while preserving the cycle structure within modules as much as possible. Moreover, as mentioned, the modules should be biochemically meaningful, i.e., encapsulate specific metabolic functions. This problem has received considerable attention in applications to metabolic networks [51, 52, 49]. Here, we re-implemented the partitioning algorithm described in [49], which is based on spectral methods [53]. A detailed description and pseudocode for cluster and cycle enumeration algorithms are provided in SI Sec. F.

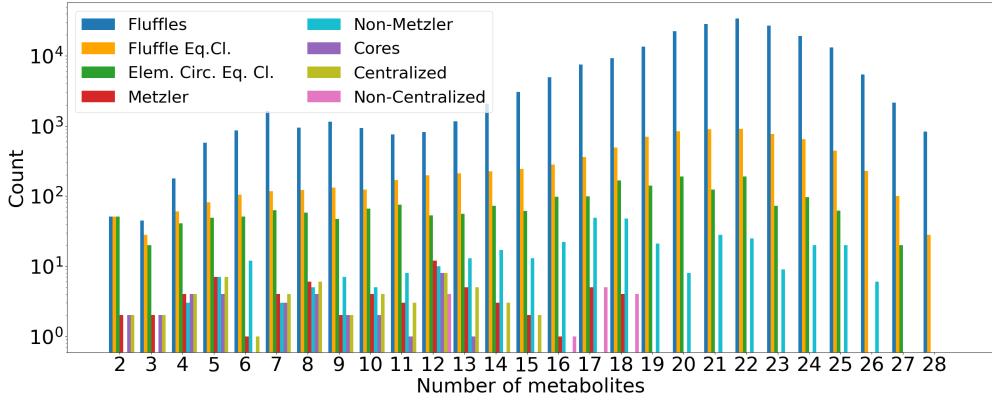


Fig. 6: Size distribution of fluffles, equivalent classes, and autocatalytic Metzler and non-Metzler matrices in the *E. coli* core metabolism. See SI Fig. 12 for additional details.

5.5 Implementation details

The algorithms detailed above are implemented as a Python package `autogato`. It is structured in different submodules. First, a metabolic model is imported in `xml`-format in `partitionNetwork.py` via `libsbml` [54] and translated into a `networkX` [55] `DiGraph`. After modularisation by means of leading eigenvector computations using `NumPy` [56] in `partitionComputations.py`, the partition tree and all relevant parameters are *pickled* and saved separately for each strongly connected component. In the second step, each strongly connected component is now analyzed separately in the module `partitionAnalysis.py`. The submatrix of the stoichiometric matrix is extracted and then the set \mathcal{E} of elementary circuits enumerated. Depending on the strategy chosen by the user, the associated CS matrices and their autocatalytic capacities are determined concurrently or downstream after assembly. During assembly, feasible combinations of CS-equivalence and CS-equivalence classes of elementary circuits are combined. Finally, autocatalytic capacity is computed using the real part of the largest eigenvalue or solving a linear programming problem with `SciPy` [50]. If feasible, partitioning, enumeration of elementary circuits, and assembly of larger equivalence classes is processed parallelized using `ConcurrentFutures ProcessPoolExecutors`. Heavy computations are processed with `Cython` [57].

6 Showcase applications

To demonstrate the performance of `autogato`, and thus the practical use of the algorithms described above, we investigated four metabolic networks: the *E. coli* core model [58], a larger model of *E. coli* DH5 α [59], a metabolic model of human erythrocytes [60], and a model of the archaeon *Methanosarcina barkeri* [61]. In all cases, we removed small, highly connected molecules (e.g., CO₂ and H₂O) as well as exchange metabolites such as ADP and NADH, since they are of little relevance for the biological interpretation of autocatalysis. A full list for each model is provided in SI Sec. D.

Our version of the *E. coli* core metabolism CRN comprises 36 metabolites and 71 reactions. Its König graph contains 2,021 elementary circuits, all belonging to distinct CS-equivalence classes. The enumeration algorithm identified 202,206 fluffles, grouped into 8,551 CS-equivalence classes. For a summary of their size distribution, we refer to Fig. 12. Even for relatively small networks, restricting to representatives of CS-equivalence classes provides a drastic reduction in computational resources: fluffle enumeration took about 12,500 seconds, while enumeration of CS-equivalence classes required only about 25 seconds and only 5 seconds when being computed parallelized. In total, `autogato` required 17.4s for completion including decomposition of the network and construction of the stoichiometric matrix from the reaction data.

Autocatalytic subsystems are common in metabolic networks. In the central carbon metabolism, represented by our *E. coli* core model, 158 of the 2,021 elementary circuits (7.9%) are autocatalytic. Of these, 42 are associated with Metzler CS matrices and 114 with non-Metzler CS matrices. Overall, approximately 5% (426 of 8,551) of CS-equivalence classes are autocatalytic; 67 of these have a Metzler $\mathbf{S}[\kappa]$, while 359 are non-Metzler. Of the 67 autocatalytic Metzler matrices, 53 were centralized and only 14 non-centralized. Interestingly, the ratio of autocatalytic CS Metzler to non-Metzler matrices roughly halves when moving from elementary circuits to all equivalence classes, from 1/3 to 1/6, which corresponds to the overall decrease in the fraction of equivalence classes associated with Metzler matrices, from 6.4% to 1.8%. Among the 67 autocatalytic CS-Metzler matrices, we identified 33 autocatalytic cores. One of these, shown in SI Fig. 14, is an autocatalytic core of Type IV. Previously, no example of this type had been reported in the literature [14, 62].

We compared our implementation with the ILP formulation of ref. [16]. To this end, we computed the stoichiometric matrix and passed these data as input to the ILP, which found 31 autocatalytic cores in 1.69s. Restricting `autogato` to enumerating autocatalytic cores exclusively, only 0.342s (averaged over 1000 iterations) were required and 33 cores identified, which included all found by [16]. The two additional cores are depicted in Fig. 13. We comment on potential reasons for these differences in the Appendix below.

The larger *E. coli* DH5 α network comprised 2779 reactions and 1951 metabolites. After removing small and highly connected metabolites (see SI Sec. D), as performed for the *E. coli* core network, 10 strongly connected components with at least 2 reactions remained. In total, we retained 1142 reactions and 622 metabolites. The largest strongly connected component comprised of 568 metabolites and 1061 reactions. Overall, 2,647,664 CS-equivalence classes with at most 10 metabolites and 10 reactions could be detected; the majority (94.8%; 2,516,295) comprised CS-equivalence classes with associated non-Metzler matrices; 161,589 (6.4%) of them autocatalytic and 2,354,706 (93.6%) non-autocatalytic. In contrast, 131,369 (5.2%) of the enumerated CS-equivalence classes were associated with a Metzler matrix; 109,391 (83.3%) autocatalytic and only 21,978 (16.7%) non-autocatalytic matrices. The majority of autocatalytic Metzler matrices (56%; 61,903) form autocatalytic cores. Centralized autocatalysis dominated with 57% non-centralized autocatalysis with 43% slightly among the autocatalytic Metzler matrices. Overall, 10% of all CS-equivalence classes were autocatalytic. The size distributions are depicted in Figure 7. In total, `autogato`

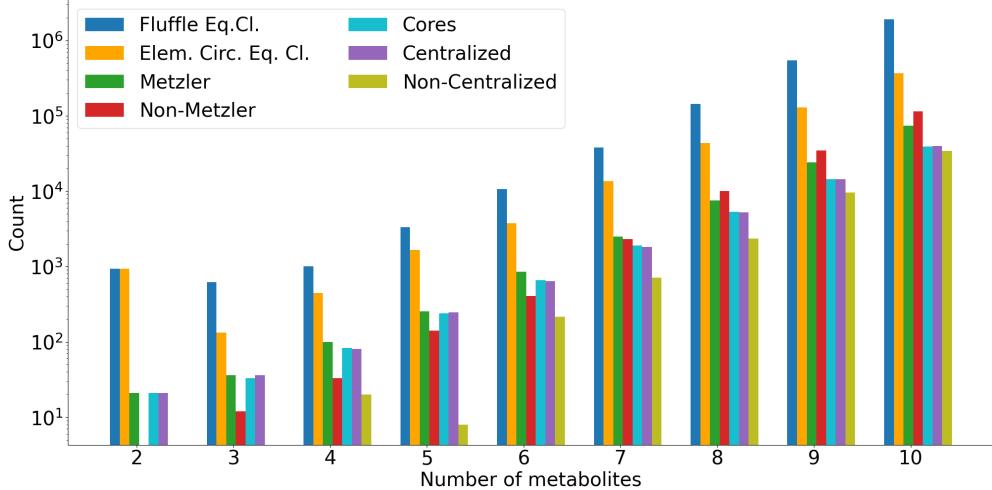


Fig. 7: Size distribution of CS-equivalent classes for elementary circuits and fluffles, as well as autocatalytic CS-Metzler and non-Metzler matrices in largest connected component of our modified *E. coli* DH5 α network.

required 1h:49min:24s with maximum consumption of 16.9 Gb internal memory. In order to compare with the ILP of [16], we used the same approach as for the smaller network and supplied the stoichiometric matrix of the largest strongly connected component as input. After 14 hours of running time, 4700 autocatalytic cores had been enumerated up to a size of 8 species and reactions. At this point, 20s were required for the computation of the next core. We, therefore, terminated the enumeration process. In contrast, the restriction of our algorithm to enumerating only cores took 4min:5s with a maximum memory consumption of 302 Mb. The majority of the time was spent on the enumeration of elementary circuits, while assembly of larger equivalence classes and post-processing finished in 26s and 17s, respectively.

To investigate, the frequency of autocatalysis in non-bacterial species, we applied the algorithm to another network, of human erythrocytes [60], which contained 342 metabolites and 469 reactions. After removal of all small metabolites, two larger strongly connected components remained: one composed of 69 metabolites and 112 reactions, respectively, covering central carbon metabolism, including glycolysis, PPP, and amino-acid metabolism, and a second component composed of 72 metabolites and 135 reactions, largely covering lipid metabolism. In summary, 1,379,913 CS-equivalence classes with a maximum size of 25 metabolites/reactions were enumerated. In contrast, to the *E. coli* networks, only 940 (0.068%) were autocatalytic. The network reflecting central carbon metabolism exhibited approximately 8% (103/1258) autocatalytic CS-equivalence classes (35/150 metzler and 113/183 non-Metzler) which is in line with the results obtained from the *E. coli* core network. However, the network reflecting lipid metabolism contained 1,378,647 CS-equivalence classes of which only 0.06% (837) were autocatalytic; of these 156 are Metzler and 681 non-Metzler matrices. Size distributions for both networks together are depicted in Fig. 8.

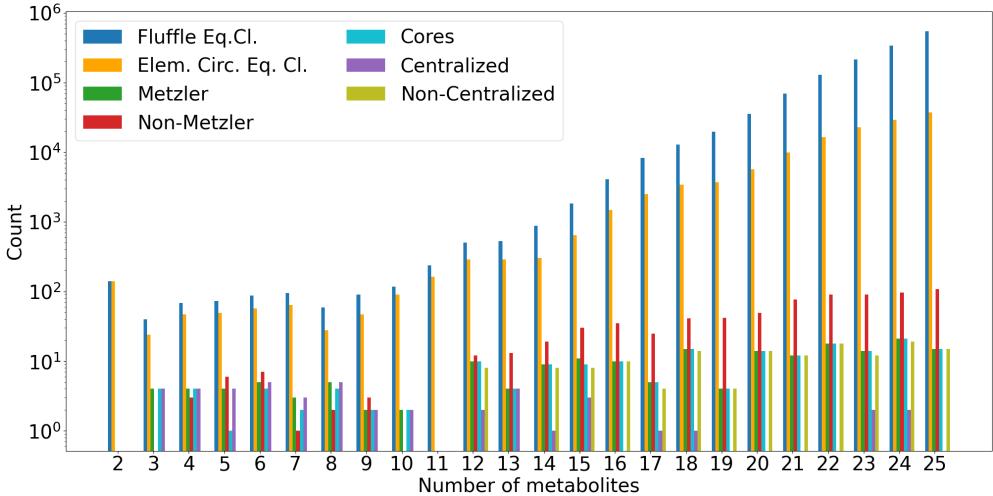


Fig. 8: Size distribution of CS-equivalent classes for elementary circuits and fluffles, as well as autocatalytic CS-Metzler and non-Metzler matrices in largest connected component of our modified *erythrocyte* network.

Finally, we investigated if autocatalysis could be found in a member of the Archaea domain. To this end, we took advantage of the metabolic model of *Methanoscincus barkeri* [61] with 690 metabolites and 692 originally, of which 249 and 402, respectively, remained in the largest strongly connected component. Within this connected component only 1.2% (20,194) of all CS-equivalence classes (1,677,604) were found to be autocatalytic; 4105 with a Metzler and 16,089 with a non-Metzler matrix. Nearly three quarters of the Metzler matrices (5483) were autocatalytic, while for the non-Metzler matrices this is the case for only 1% (16,089/1,656,158). Two thirds of the CS-equivalence classes with autocatalytic cores Metzler matrices correspond to autocatalytic cores. The size distribution is depicted in Fig. 9.

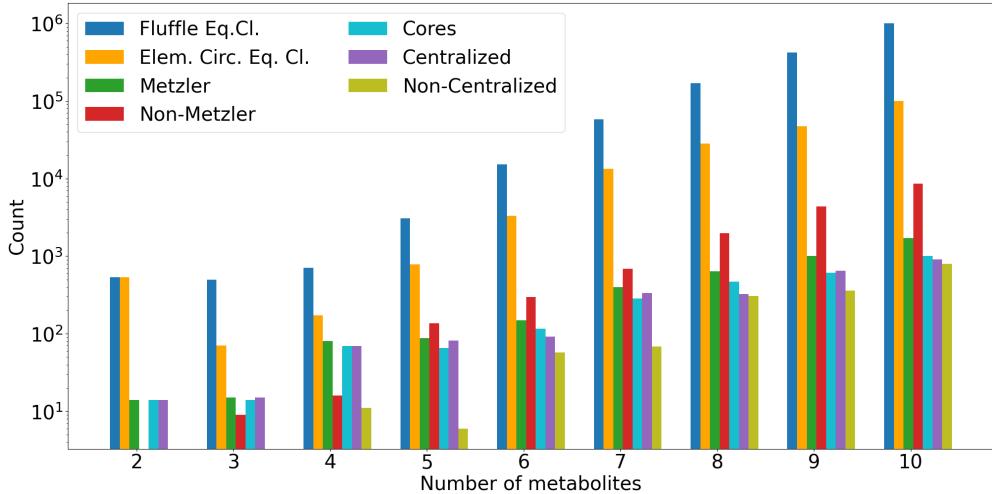


Fig. 9: Size distribution of CS-equivalent classes for elementary circuits and fluffles, as well as autocatalytic CS-Metzler and non-Metzler matrices in largest connected component of our modified *Methanosarcina barkeri* network.

7 Discussion

We have presented a detailed mathematical analysis of autocatalytic substructures in large CRNs. Starting from the stoichiometric matrix \mathbf{S} , we identify a specific class of subgraphs in the bipartite (König) representation of the CRN, called fluffles, which are necessary to support irreducible autocatalytic subnetworks. Fluffles fall into equivalence classes determined solely by the corresponding Child-Selections. These correspond to Metzler matrices that form autocatalytic cores if and only if they are induced subgraphs of the CRN, while larger irreducible autocatalytic subnetworks only need to contain a Metzler part, or equivalently, a spanning fluffle, as well as a smaller autocatalytic core.

Based on these structural insights, we developed an algorithmic approach to produce representative fluffles by superimposing elementary circuits. This purely graph-theoretical method avoids the complex ILP formulation used in [16] to detect autocatalytic cores. Furthermore, it extends to a much broader class of autocatalytic subsystems beyond the autocatalytic cores. Tests on four metabolic networks, a small model of the *E. coli* core metabolism and three much larger CRNs comprising up to more than 600 metabolites and 1100 reactions showed that our algorithmic approach is feasible in practice. For the small network, a complete analysis is obtain within about 17 seconds. For the large network, the computation had to be limited to moderate-size fluffles, with up to 10, 15, and 25 metabolites and reactions for *E. coli*, human erythrocytes, and *Methanosarcina Barker*, respectively. Clearly, this does not capture all autocatalytic cores, since cores in the smaller *E. coli* network ranged up 18 metabolites, i.e., half of the size of the CRN. Nevertheless, in the *E. coli* *DH5α* model, we identified more than 100,000 irreducible autocatalytic CS subnetworks, more than

half of which are autocatalytic cores. These results reinforce the conclusion of earlier studies (in particular those based on different definitions of autocatalysis such as [12]) on the ubiquitous nature of autocatalysis in metabolic CRNs.

A direct comparison of the restricted variant of Alg. 1 that computes autocatalytic cores only with the ILP formulation of [16] turned out favorably for our approach with respect to resource consumption. An evaluation of the *E. coli* core network shows furthermore that the ILP does not enumerate all autocatalytic cores and struggles with larger network sizes. In fact, the algorithm described in [16] focuses on enumerating only the minimal subsets of reactions that contain an autocatalytic core, without imposing any restriction on the species involved. Consequently, for reaction sets $R_1 \subset R_2$ associated with two autocatalytic cores A_1 and A_2 based on different sets of species, the ILP formulation would identify only the minimal set R_1 . We briefly elaborate on this issue in the Appendix.

Once autocatalytic subsystems have been identified, they can provide further insight into the potential behaviors of the CRN. For example, the close connection between autocatalysis and sustained oscillations has been explored in [63]. Building on this work, one can state the following sufficient condition:

Proposition 57 (Proof: SI). *Let $\mathbf{S}[\kappa]$ be a Hurwitz-stable autocatalytic CS-matrix. Then there exists a choice of parameters such that the system (6), $\dot{z} = f(z) := \mathbf{S} \cdot v(z)$, admits periodic solutions.*

This result sets the stage for identifying minimal sub-networks that are responsible for “interesting” dynamical behavior such as periodic oscillations.

For large CRNs, in particular models of complete metabolism, an exhaustive enumeration of fluffle CS-equivalence classes is probably infeasible even on an HPC system. This is certainly true in the (chemically unrealistic) worst-case scenario, since it is possible to construct CRNs in which all autocatalytic cores have size 2 but there are exponentially many autocatalytic Metzler matrices: it suffices to consider a CS matrix $\mathbf{S}[\kappa]$ such that $\mathbf{S}[\kappa]_{ii} = -1$, $\mathbf{S}[\kappa]_{ij} = 2$ for $i > j$, and $\mathbf{S}[\kappa]_{ij} = 1$ for $i < j$; in this case, only the 2x2 principal submatrices of $\mathbf{S}[\kappa]$ are autocatalytic cores while every principal submatrix of $\mathbf{S}[\kappa]$ is an irreducible, autocatalytic Metzler matrix.

By Cor. 53, all autocatalytic cores are superpositions of Metzler circuits. Since worst-case instances may also contain very large numbers of elementary circuits, this raises the question of whether it is possible to enumerate the subset of Metzler circuits without enumerating all elementary circuits. In our analysis of the large *E. coli* network, we pragmatically limited the length L of the elementary circuits. Current versions of Johnson’s algorithm allow such a cut-off. In particular, the algorithm of Gupta & Suzumura [44] for sparse graphs, with running time $O((c + |V|)L\bar{d}^L)$ where \bar{d} is the average degree, is attractive for applications to CRNs. So far, there is no comparably efficient approach to produce elementary circuits ordered by size. A related algorithm to enumerate chordless elementary circuits, optionally restricted to length L , is described in [64]. It will be interesting to see whether this approach can be adapted to enumerate Metzler circuits efficiently.

All autocatalytic cores derive from induced fluffles, and more broadly from (not necessarily maximal) induced strong blocks in \mathbf{K} . A recent linear-delay algorithm for

enumerating strongly connected induced subgraphs [65, 66] may thus serve as an alternative starting point for the efficient generation of candidate subsets for autocatalytic cores.

8 Declarations

8.1 Availability of Data and Materials

The implementation, models, and all necessary data is available at <https://github.com/hollyritch/autogato>.

8.2 Competing interests

The authors declare that they have no competing interests.

8.3 Funding

This work was support in part by the Novo Nordisk Foundation (grant no. 0066551, MATOMIC), the German Research Foundation (grant no. 234823413), and the German Federal Ministry of Education and Research (BMBF) within the German Network for Bioinformatics Infrastructure (de.NBI) under grant number w-de.NBI\303\203018.

8.4 Authors' contributions

All authors contributed to conceptualization, methodology, and writing of the manuscript.

References

- [1] Gold V, McNaught A, The International Union of Pure and Applied Chemistry (IUPAC) (eds) (2025) The IUPAC Compendium of Chemical Terminology: The Gold Book, 5th edn. International Union of Pure and Applied Chemistry (IUPAC), Research Triangle Park, NC, DOI 10.1351/goldbook
- [2] Eigen M (1971) Selforganization of matter and the evolution of biological macromolecules. *Die Naturwissenschaften* 58(10):465–523, DOI 10.1007/BF00623322
- [3] Eigen M, Schuster P (1977) A principle of natural self-organization: Part A: Emergence of the hypercycle. *Naturwissenschaften* 64(11):541–565, DOI 10.1007/BF00450633
- [4] Gilbert W (1986) Origin of life: The RNA world. *Nature* 319(6055):618–618, DOI 10.1038/319618a0
- [5] Joyce GF (1989) RNA evolution and the origins of life. *Nature* 338(6212):217–224, DOI 10.1038/338217a0
- [6] Joyce GF (2002) The antiquity of RNA-based evolution. *Nature* 418(6894):214–221, DOI 10.1038/418214a
- [7] Bissette AJ, Fletcher SP (2013) Mechanisms of autocatalysis. *Angewandte Chemie (International Ed in English)* 52(49):12,800–12,826, DOI 10.1002/anie.201303822

- [8] Howlett MG, Fletcher SP (2023) From autocatalysis to survival of the fittest in self-reproducing lipid systems. *Nature Reviews Chemistry* 7(10):673–691, DOI 10.1038/s41570-023-00524-8
- [9] Hordijk W, Hein J, Steel M (2010) Autocatalytic Sets and the Origin of Life. *Entropy* 12(7):1733–1742, DOI 10.3390/e12071733
- [10] Butlerow A (1861) Bildung einer zuckerartigen Substanz durch Synthese. *Justus Liebigs Annalen der Chemie* 120(3):295–298, DOI 10.1002/jlac.18611200308
- [11] Orgel LE (2008) The Implausibility of Metabolic Cycles on the Prebiotic Earth. *PLoS Biology* 6(1):e18, DOI 10.1371/journal.pbio.0060018
- [12] Barenholz U, Davidi D, Reznik E, Bar-On Y, Antonovsky N, Noor E, Milo R (2017) Design principles of autocatalytic cycles constrain enzyme kinetics and force low substrate saturation at flux branch points. *eLife* 6:e20,667, DOI 10.7554/eLife.20667
- [13] Andersen JL, Flamm C, Merkle D, Stadler PF (2021) Defining Autocatalysis in Chemical Reaction Networks. *Journal of Systems Chemistry* DOI 10.48550/arXiv.2107.03086, [2107.03086](https://doi.org/10.48550/arXiv.2107.03086)
- [14] Blokhuis A, Lacoste D, Nghe P (2020) Universal motifs and the diversity of autocatalytic systems. *Proceedings of the National Academy of Sciences* 117(41):25,230–25,236, DOI 10.1073/pnas.2013527117
- [15] Vassena N, Stadler PF (2024) Unstable cores are the source of instability in chemical reaction networks. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 480(2285):20230,694, DOI 10.1098/rspa.2023.0694
- [16] Gagrani P, Blanco V, Smith E, Baum D (2024) Polyhedral geometry and combinatorics of an autocatalytic ecosystem. *Journal of Mathematical Chemistry* 62(5):1012–1078, DOI 10.1007/s10910-024-01576-x
- [17] Honegger P, Flamm C, Fontana W (2022) Efficient Identification of Autocatalysis in Chemical and Biological Networks. In: The Astrobiology Science Conference (AbSciCon) 2022, pp 503–03
- [18] Golnik R, Gatter T, Stadler PF, Vassena N (2025) BiRNe: Symbolic bifurcation analysis of reaction networks with Python. DOI 10.48550/ARXIV.2511.14431
- [19] Norsigian CJ, Pusarla N, McConn JL, Yurkovich JT, Dräger A, Palsson BO, King Z (2020) BiGG Models 2020: Multi-strain genome-scale models and expansion across the phylogenetic tree. *Nucleic Acids Research* 48(D1):D402–D406, DOI 10.1093/nar/gkz1054
- [20] Moretti S, Martin O, Van Du Tran T, Bridge A, Morgat A, Pagni M (2016) MetaNetX/MNXref – reconciliation of metabolites and biochemical reactions to bring together genome-scale metabolic networks. *Nucleic Acids Research* 44(D1):D523–D526, DOI 10.1093/nar/gkv1117
- [21] Hawkins C, Xue B, Yasmin F, Wyatt G, Zerbe P, Rhee SY (2025) Plant Metabolic Network 16: Expansion of underrepresented plant groups and experimentally supported enzyme data. *Nucleic Acids Research* 53(D1):D1606–D1613, DOI 10.1093/nar/gkae991
- [22] De Oliveira Dal'Molin CG, Quek LE, Palfreyman RW, Brumbley SM, Nielsen LK (2010) AraGEM, a Genome-Scale Reconstruction of the Primary Metabolic

- Network in Arabidopsis. *Plant Physiology* 152(2):579–589, DOI 10.1104/pp.109.148817
- [23] Poolman MG, Kundu S, Shaw R, Fell DA (2013) Responses to Light Intensity in a Genome-Scale Model of Rice Metabolism. *Plant Physiology* 162(2):1060–1072, DOI 10.1104/pp.113.216762
- [24] Saha R, Suthers PF, Maranas CD (2011) Zea mays iRS1563: A Comprehensive Genome-Scale Metabolic Reconstruction of Maize Metabolism. *PLoS ONE* 6(7):e21,784, DOI 10.1371/journal.pone.0021784
- [25] Yuan H, Cheung CM, Poolman MG, Hilbers PAJ, Van Riel NAW (2016) A genome-scale metabolic network reconstruction of tomato (*Solanum lycopersicum* L.) and its application to photorespiratory metabolism. *The Plant Journal* 85(2):289–304, DOI 10.1111/tpj.13075
- [26] Zykov AA (1974) HYPERGRAPHS. *Russian Mathematical Surveys* 29(6):89–156, DOI 10.1070/RM1974v02n06ABEH001303
- [27] Waage P, Guldberg CM (1864) Studier over affiniteten. *Forhandlinger i Videnskabs-selskabet i Christiania* 1:35–45
- [28] Müller S, Regensburger G (2012) Generalized Mass Action Systems: Complex Balancing Equilibria and Sign Vectors of the Stoichiometric and Kinetic-Order Subspaces. *SIAM Journal on Applied Mathematics* 72(6):1926–1947, DOI 10.1137/110847056
- [29] Johnson KA, Goody RS (2011) The Original Michaelis Constant: Translation of the 1913 Michaelis–Menten Paper. *Biochemistry* 50(39):8264–8269, DOI 10.1021/bi201284u
- [30] Hill AV (1910) The possible effects of the aggregation of the molecules of haemoglobin on its dissociation curves. *The Journal of Physiology* 40:i–vii, <http://jp.physoc.org/content/40/supplement/i.full.pdf+html>
- [31] Angeli D, De Leenheer P, Sontag ED (2007) A Petri net approach to the study of persistence in chemical reaction networks. *Mathematical Biosciences* 210(2):598–618, DOI 10.1016/j.mbs.2007.07.003
- [32] Hsu SB (2013) Ordinary Differential Equations with Applications, 2nd edn. No. 21 in AIMS Series on Applied Mathematic, World Scientific, New Jersey
- [33] Aris R (1965) Prolegomena to the rational analysis of systems of chemical reactions. *Archive for Rational Mechanics and Analysis* 19(2):81–99, DOI 10.1007/BF00282276
- [34] Bullo F (2018) Lectures on Network Systems, 1st edn. CreateSpace, North Charleston, South Carolina
- [35] Lange BM, Rujan T, Martin W, Croteau R (2000) Isoprenoid biosynthesis: The evolution of two ancient and distinct pathways across genomes. *Proceedings of the National Academy of Sciences of the United States of America* 97(24):13,172–13,177, DOI 10.1073/pnas.240454797
- [36] Fukuda K, Matsui T (1994) Finding all the perfect matchings in bipartite graphs. *Applied Mathematics Letters* 7(1):15–18, DOI 10.1016/0893-9659(94)90045-0
- [37] Uno T (1997) Algorithms for enumerating all perfect, maximum and maximal matchings in bipartite graphs. In: Goos G, Hartmanis J, Van Leeuwen J, Leong HW, Imai H, Jain S (eds) *Algorithms and Computation*, vol 1350, Springer Berlin

- Heidelberg, Berlin, Heidelberg, pp 92–101, DOI 10.1007/3-540-63890-3_11
- [38] Fink J (2025) Constant time enumeration of perfect bipartite matchings. DOI 10.48550/ARXIV.2509.16135
- [39] Grötschel M (1979) On minimal strong blocks. *Journal of Graph Theory* 3(3):213–219, DOI 10.1002/jgt.3190030303
- [40] Berge C (1973) Graphs and Hypergraphs, North-Holland Mathematical Library, vol 6. North Holland, Amsterdam
- [41] Tarjan R (1973) Enumeration of the Elementary Circuits of a Directed Graph. *SIAM Journal on Computing* 2(3):211–216, DOI 10.1137/0202017
- [42] Johnson DB (1975) Finding All the Elementary Circuits of a Directed Graph. *SIAM Journal on Computing* 4(1):77–84, DOI 10.1137/0204007
- [43] Szwarcfiter JL, Lauer PE (1976) A search strategy for the elementary cycles of a directed graph. *BIT* 16(2):192–204, DOI 10.1007/BF01931370
- [44] Gupta A, Suzumura T (2021) Finding All Bounded-Length Simple Cycles in a Directed Graph. DOI 10.48550/arXiv.2105.10094, [2105.10094](#)
- [45] Ravasz E, Somera AL, Mongru DA, Oltvai ZN, Barabási AL (2002) Hierarchical organization of modularity in metabolic networks. *Science* (New York, NY) 297(5586):1551–1555, DOI 10.1126/science.1073374
- [46] Ma HW, Zeng AP (2003) The connectivity structure, giant strong component and centrality of metabolic networks. *Bioinformatics* (Oxford, England) 19(11):1423–1430, DOI 10.1093/bioinformatics/btg177
- [47] Ma HW, Zhao XM, Yuan YJ, Zeng AP (2004) Decomposition of metabolic network into functional modules based on the global connectivity structure of reaction graph. *Bioinformatics* (Oxford, England) 20(12):1870–1876, DOI 10.1093/bioinformatics/bth167
- [48] Zhao J, Yu H, Luo JH, Cao ZW, Li YX (2006) Hierarchical modularity of nested bow-ties in metabolic networks. *BMC Bioinformatics* 7(1):386, DOI 10.1186/1471-2105-7-386
- [49] Sridharan GV, Hassoun S, Lee K (2011) Identification of Biochemical Network Modules Based on Shortest Retroactive Distances. *PLoS Computational Biology* 7(11):e1002262, DOI 10.1371/journal.pcbi.1002262
- [50] Virtanen P, Gommers R, Oliphant TE, Haberland M, Reddy T, Cournapeau D, Burovski E, Peterson P, Weckesser W, Bright J, Van Der Walt SJ, Brett M, Wilson J, Millman KJ, Mayorov N, Nelson ARJ, Jones E, Kern R, Larson E, Carey CJ, Polat I, Feng Y, Moore EW, VanderPlas J, Laxalde D, Perktold J, Cimrman R, Henriksen I, Quintero EA, Harris CR, Archibald AM, Ribeiro AH, Pedregosa F, Van Mulbregt P, SciPy 10 Contributors, Vijaykumar A, Bardelli AP, Rothberg A, Hilboll A, Kloeckner A, Scopatz A, Lee A, Rokem A, Woods CN, Fulton C, Masson C, Häggström C, Fitzgerald C, Nicholson DA, Hagen DR, Pasechnik DV, Olivetti E, Martin E, Wieser E, Silva F, Lenders F, Wilhelm F, Young G, Price GA, Ingold GL, Allen GE, Lee GR, Audren H, Probst I, Dietrich JP, Silterra J, Webber JT, Slavić J, Nothman J, Buchner J, Kulick J, Schönberger JL, De Miranda Cardoso JV, Reimer J, Harrington J, Rodriguez JLC, Nunez-Iglesias J, Kuczynski J, Tritz K, Thoma M, Newville M, Kümmeler M, Bolingbroke M, Tartre M, Pak M, Smith NJ, Nowaczyk N, Shebanov N,

- Pavlyk O, Brodtkorb PA, Lee P, McGibbon RT, Feldbauer R, Lewis S, Tygier S, Sievert S, Vigna S, Peterson S, More S, Pudlik T, Oshima T, Pingel TJ, Robitaille TP, Spura T, Jones TR, Cera T, Leslie T, Zito T, Krauss T, Upadhyay U, Halchenko YO, Vázquez-Baeza Y (2020) SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods* 17(3):261–272, DOI 10.1038/s41592-019-0686-2
- [51] Holme P, Huss M, Jeong H (2003) Subnetwork hierarchies of biochemical pathways. *Bioinformatics* 19(4):532–538, DOI 10.1093/bioinformatics/btg033
 - [52] Schaeffer SE (2007) Graph clustering. *Computer Science Review* 1(1):27–64, DOI 10.1016/j.cosrev.2007.05.001
 - [53] Newman MEJ (2006) Modularity and community structure in networks. *Proceedings of the National Academy of Sciences* 103(23):8577–8582, DOI 10.1073/pnas.0601602103
 - [54] Bornstein BJ, Keating SM, Jouraku A, Hucka M (2008) LibSBML: An API Library for SBML. *Bioinformatics* 24(6):880–881, DOI 10.1093/bioinformatics/btn051
 - [55] Hagberg AA, Schult DA, Swart PJ (2008) Exploring network structure, dynamics, and function using NetworkX. In: Varoquaux G, Vaught T, Millman J (eds) *Proceedings of the 7th Python in Science Conference*, Pasadena, CA USA, pp 11–15
 - [56] Harris CR, Millman KJ, Van Der Walt SJ, Gommers R, Virtanen P, Cournapeau D, Wieser E, Taylor J, Berg S, Smith NJ, Kern R, Picus M, Hoyer S, Van Kerkwijk MH, Brett M, Haldane A, Del Río JF, Wiebe M, Peterson P, Gérard-Marchant P, Sheppard K, Reddy T, Weckesser W, Abbasi H, Gohlke C, Oliphant TE (2020) Array programming with NumPy. *Nature* 585(7825):357–362, DOI 10.1038/s41586-020-2649-2
 - [57] Behnel S, Bradshaw R, Citro C, Dalcin L, Seljebotn DS, Smith K (2011) Cython: The best of both worlds. *Computing in Science & Engineering* 13(2):31–39
 - [58] Orth JD, Fleming RMT, Palsson BØ (2010) Reconstruction and Use of Microbial Metabolic Networks: The Core *Escherichia coli* Metabolic Model as an Educational Guide. *EcoSal Plus* 4(1):10.1128/ecosalplus.10.2.1, DOI 10.1128/ecosalplus.10.2.1
 - [59] Monk JM, Koza A, Campodonico MA, Machado D, Seoane JM, Palsson BO, Herrgård MJ, Feist AM (2016) Multi-omics Quantification of Species Variation of *Escherichia coli* Links Molecular Features with Strain Phenotypes. *Cell Systems* 3(3):238–251.e12, DOI 10.1016/j.cels.2016.08.013
 - [60] Bordbar A, Jamshidi N, Palsson BO (2011) iAB-RBC-283: A proteomically derived knowledge-base of erythrocyte metabolism that can be used to simulate its physiological and patho-physiological states. *BMC systems biology* 5:110, DOI 10.1186/1752-0509-5-110
 - [61] Feist AM, Scholten JCM, Palsson BØ, Brockman FJ, Ideker T (2006) Modeling methanogenesis with a genome-scale metabolic reconstruction of Methanosarcina barkeri. *Molecular Systems Biology* 2:2006.0004, DOI 10.1038/msb4100046
 - [62] Unterberger J, Nghe P (2022) Stoichiometric and dynamical autocatalysis for diluted chemical reaction networks. DOI 10.48550/arXiv.2109.01130, [2109.01130](https://arxiv.org/abs/2109.01130)

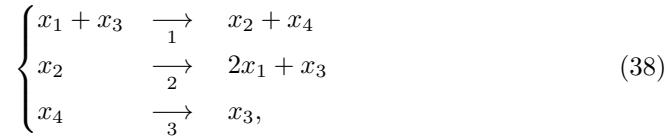
- [63] Blokhuis A, Stadler PF, Vassena N (2025) Stoichiometric recipes for periodic oscillations in reaction networks. DOI 10.48550/arXiv.2508.15273, [2508.15273](#)
- [64] Bisdorff R (2010) On detecting and enumerating chordless circuits in a digraph. Tech. rep., Univ. Luxembourg, Luxembourg
- [65] Tada T, Haraguchi K (2025) A linear delay algorithm in SD set system and its application to subgraph enumeration. Journal of Computer and System Sciences 152:103,637, DOI 10.1016/j.jcss.2025.103637
- [66] Shota K, Haraguchi K (2025) A Linear Delay Algorithm of Enumerating Strongly-Connected Induced Subgraphs Based on SSD Set System. In: Fernau H, Zhu B (eds) Combinatorial Algorithms, vol 15885, Springer Nature Switzerland, Cham, pp 89–102, DOI 10.1007/978-3-031-98740-3_7
- [67] Galluccio A, Loeb M (1996) (p,q)-odd digraphs. Journal of Graph Theory 23(2):175–184, DOI 10.1002/(SICI)1097-0118(199610)23:2<175::AID-JGT8>3.0.CO;2-Q
- [68] Loeb M, Matamala M (2001) Some remarks on cycles in graphs and digraphs. Discrete Mathematics 233(1-3):175–182, DOI 10.1016/S0012-365X(00)00236-3
- [69] Gleiss PM, Leydold J, Stadler PF (2003) Circuit bases of strongly connected digraphs. Discussiones Mathematicae Graph Theory 23(2):241, DOI 10.7151/dmgt.1200
- [70] Havet F, Nisse N (2019) Constrained ear decompositions in graphs and digraphs. Discrete Mathematics & Theoretical Computer Science vol. 21 no. 4(Graph Theory):4544, DOI 10.23638/DMTCS-21-4-3
- [71] Boley M, Horváth T, Poigné A, Wrobel S (2010) Listing closed sets of strongly accessible set systems with applications to data mining. Theoretical Computer Science 411(3):691–700, DOI 10.1016/j.tcs.2009.10.024
- [72] Conte A, Grossi R, Marino A, Versari L (2019) Listing Maximal Subgraphs Satisfying Strongly Accessible Properties. SIAM Journal on Discrete Mathematics 33(2):587–613, DOI 10.1137/17M1152206

9 Appendix

The ILP approach of [16] is based on subnetworks (X', R') of a CRN (X, R) where $R' \subseteq R$ and $X' := X(R')$ is defined as the set of species participating in the set of reactions R' either as reactant or as product (or both). From the perspective of the present paper, any Child-Selection (CS) $\kappa = (X_\kappa, R_\kappa, \kappa)$ uniquely defines a subnetwork $(X(R_\kappa), R_\kappa)$ as above, where typically $X_\kappa \subset X(R')$. The converse is, however, not true: the same subnetwork $(X(R'), R')$ may support another Child-Selection $\tilde{\kappa}$, i.e. with $R_{\tilde{\kappa}} = R_\kappa$. Thus, different child selections may be supported by the same subnetwork $(X(R'), R')$. Moreover, $\tilde{\kappa}$ need not be defined on the same set of species, i.e. we may have $X_{\tilde{\kappa}} \neq X_\kappa$ as long as $X_{\tilde{\kappa}} \subseteq X(R_{\tilde{\kappa}}) = X(R_\kappa)$. The key concept in the work of Gagrani *et al.* is introduced by Def. 3.2 in [16] (III.2 in the arXiv preprint) as follows: “*A minimal autocatalytic subnetwork (MAS) is defined to be the subnetwork with the least number of reactions containing a particular autocatalytic core.*” We rephrase this statement here as follows:

Definition 58 (Def. 3.2 in [16]). *A minimal autocatalytic subnetwork (MAS) is a subnetwork $(X(R'), R')$ containing an autocatalytic core that is induced by an inclusion-minimal set of reactions with this property.*

Inclusion minimality of the reaction set R' implies that a MAS does not contain more reactions than specified by the Child-Selection of its defining autocatalytic core, i.e. $R' = R_\kappa$ for a Child-Selection κ for which $\mathbf{S}[\kappa]$ is an autocatalytic core. Since Def. 58 imposes minimality only on the set of reactions, not all autocatalytic cores are associated with a MAS. In particular, an autocatalytic core $\mathbf{S}[\kappa_2]$ with reaction set R_{κ_2} is not associated with a MAS if R_{κ_2} strictly contains a reaction set $R_{\kappa_1} \subset R_{\kappa_2}$ that supports another autocatalytic core $\mathbf{S}[\kappa_1]$. By minimality of cores, $\mathbf{S}[\kappa_1]$ necessarily involves a set X_{κ_1} of reactants that is not contained in $X_{\kappa_2} = \kappa_2^{-1}(R_{\kappa_2})$. To see that such cases indeed exist, consider the following simple reaction network:



with stoichiometric matrix

$$\mathbf{S} = \begin{pmatrix} -1 & 2 & 0 \\ 1 & -1 & 0 \\ -1 & 1 & 1 \\ 1 & 0 & -1 \end{pmatrix}. \quad (39)$$

The first two columns and first two rows, corresponding to the CS $\kappa_1 = (\{x_1, x_2\}, \{1, 2\}, \kappa(x_1, x_2) = (1, 2))$, form the autocatalytic core

$$\mathbf{S}[\kappa_1] = \begin{pmatrix} -1 & 2 \\ 1 & -1 \end{pmatrix}. \quad (40)$$

The MAS-oriented ILP implementation of Gagrani *et al.* [16] therefore discards any reaction set strictly containing $\{1, 2\}$. However, by considering all three columns and rows 2, 3, and 4, we obtain another autocatalytic core:

$$\mathbf{S}[\kappa_2] = \begin{pmatrix} -1 & 1 & 0 \\ 1 & -1 & 1 \\ 0 & 1 & -1 \end{pmatrix}, \quad (41)$$

associated with the CS

$$\kappa_2 = (\{x_2, x_3, x_4\}, \{1, 2, 3\}, \kappa(x_2, x_3, x_4) = (2, 1, 3)),$$

where $X_{\kappa_1} = \{x_1, x_2\} \not\subseteq \{x_2, x_3, x_4\}$. Indeed, the ILP of Gagrani *et al.* does not detect the autocatalytic core $\mathbf{S}[\kappa_2]$. The two missing autocatalytic cores in the *E. coli* core metabolism can also be explained in this manner.

We note, finally, that it is a simple task to determine the set of all MAS in a given CRN if all autocatalytic cores have been computed. By considering the sets of

reactions associated to all autocatalytic cores, it suffices to remove the sets that are non-minimal with respect to the inclusion relation. The graph-theoretic approach thus can also be used to enumerate MAS in addition to autocatalytic cores.

Supplementary Information

A Background: Graphs and Matrices

The main text follows well-established textbook terminology and notation. For completeness, we briefly review the basic definitions and well-known facts about matchings and circuits in directed graphs.

A.1 Basic Notation

Graphs. We consider here directed graphs $G = (V, E)$ with vertex set V and edge set $E \subseteq V \times V$ without loops, i.e., $(v, v) \notin E$ for all $v \in V$. Where necessary, we write $V(G)$ and $E(G)$. Whenever there is a directed edge (u, v) from vertex u to v we say that u is an in-neighbor of v , and v is an out-neighbor of u . The in-degree and out-degree of a vertex is the number of its in-neighbors and out-neighbors, respectively. A H is a subgraph of G if H is graph, $V(H) \subseteq V(G)$, and $E(H) \subseteq E(G)$. The subgraph H is spanning if $V(H) = V(G)$ and induced if $u, v \in V(H)$ and $(u, v) \in E(G)$ implies $(u, v) \in E(H)$. The adjacency matrix \mathbf{A} of G is the $V \times V$ matrix with entries $\mathbf{A}_{uv} = 1$ if $(u, v) \in E(G)$, $\mathbf{A}_{uv} = 1$ if $(u, v) \in E(G)$ and $\mathbf{A}_{uv} = 0$ otherwise.

A walk of length $h \geq 0$ in G is an alternating sequence $(v_0, e_1, v_1, \dots, e_h, v_h)$ of vertices and edges such that $e_i = (v_{i-1}, v_i)$ for $1 \leq i \leq h$. A walk is closed if $v_0 = v_h$. It is a *path* if $i \neq j$ implies $v_i \neq v_j$ and thus also $e_i \neq e_j$. A closed walk is an *elementary circuit* if $i \neq j$ implies $v_i \neq v_j$ for $i, j \neq 0$, i.e., if $(v_1, e_2, \dots, e_h, v_h)$ is a path. A graph is *strongly connected* if there is a path from u to v for $u, v \in V$. The underlying undirected graph obtained from G by ignoring the direction of the edges. It is equivalent to the symmetrized graphs G_s obtained by setting $V(G_s) = V(G)$ and $(u, v), (v, u) \in E(G_s)$ whenever $(u, v) \in E(G)$. A graph G is connected if its underlying undirected graph is connected in the usual sense for undirected graphs, or equivalently, if its symmetrized graphs G_s is strongly connected.

Hypergraphs. Chemical reaction networks can be represented a directed hypergraphs with vertex X representing the chemical species and a set of directed hyperedges R denoting the reactions. A directed hyperedges (E^-, E^+) is a pair of of non-empty subsets $E^-, E^+ \subseteq V$ denoting the reactants and products, respectively. The *König representation* of a directed hypergraph (X, R) is the directed bipartite graph with vertex set $V = X \cup R$ and edge set E such that $(x, r) \in E$ if there is $r = (E^-, E^+) \in R$ such that $x \in E^-$, and $(r, x) \in E$ if there is $r = (E^-, E^+) \in R$ such that $x \in E^+$. Throughout, we denote the König graph of the CRN under consideration by \mathbf{K} .

Linear Algebra. Given an matrix A with rows and columns indexed by ordered sets N and M , respectively, we denote by $A[K, L]$ the submatrix with rows indexed by $K \subseteq N$ and columns $L \subseteq M$. When $|K| = |L|$ the determinant of $A[K, L]$ is called a *minor*. If A is a square matrix, $|N| = |M|$, a square submatrix $A[K, L]$ with $K = L$ is called *principal submatrix* and its determinant *principal minor*. Since only one set K is needed to define a principal submatrix, we refer to a principal submatrix as $A[K]$. A permutation matrix P of size n is an square matrix that has exactly one entry 1 in

each row and each column, and 0 elsewhere. A square matrix is *irreducible* if there exists no permutation matrix P of size n such that PAP^{-1} is an upper triangular matrix. Considering corresponding row and column indices as vertices and introducing an edge (i, j) if $A_{ij} \neq 0$ yields a graph Γ that is strongly connected if and only if A is irreducible. A matrix $A \in \mathbb{C}^{n \times n}$ is *Hurwitz stable* if all of its eigenvalues have negative real part. It is called *Hurwitz unstable* if it possesses an eigenvalue with positive real part.

A.2 Graph Theoretical Constructions

Matchings. A matching in $G = (V, E)$ is a subset $M \subseteq E$ of edges such that each vertex is incident to at most one edge. The notions of matchings is the same in directed and undirected graphs, i.e., the direction of the edges does not play a role. A matching M is *perfect* if every vertex is incident with an edge in M . While the existence of a perfect matching can be verified in polynomial time (using any algorithm for computing a maximum matching), counting the number of perfect matching is $\#P$ -complete. Enumeration can be achieved with constant amortized time [38]. In the applications below we consider matchings on directed bipartite graphs. Writing the vertex partition of the bipartite graph as $V =: X \cup R$, we are interested in only in matchings $M \subseteq E_1 := (X \times R) \cap E$. Clearly, this is equivalent to matchings in the subgraph $G_1 = (V, E_1)$.

Cycle Bases and Ear Decompositions in Digraphs Recall that a digraph is *strongly connected* if every vertex is reachable from every other vertex by means of a directed path. Equivalently, G is (weakly) connected if the underlying undirected graph is connected, and every vertex $x \in V(G)$ is contained in an elementary circuit. The cycle space of digraph is usually defined over \mathbb{Z} as the kernel of the directed incidence matrix \mathbf{H} with entries $H_{xe} = -1$ if x is the tail of the edge e , $H_{xe} = +1$ if x is the head of e , and 0 otherwise. The (incidence vectors of the) elementary circuits define the extremal rays of non-negative cone $\{z | \mathbf{Hz} = 0, z_e \geq 0 \forall e \in E\}$ [67, 68, 69]. Every strongly connected digraph has a basis of the cycle space that consists of elementary circuits only, see Thm.9 of [40].

A *directed ear* in a digraph G is a directed path in which all internal vertices have indegree 1 and outdegree 1, while the initial vertex has outdegree at least 2 and the terminal vertex has indegree at least 2. An ear is called open if its initial and terminal vertex are distinct, and closed otherwise. A digraph G is strongly connected if it can be obtained from a single directed cycle by successively adding (open or closed) ears [67, 68, 70]. The digraph G is a *strong block* if it is strongly connected and has no cut vertices. Equivalently, any two vertices in G lie on some elementary circuit. Moreover, G is a strong block if and only if it can be constructed by means of an open ear decomposition [39]. For the ear decompositions, an *ear basis* is obtained by completing each ear (after it has been attached) to an elementary circuit by a directed path from the terminal to the initial vertex of the ear.

In the main text, we will use the following straightforward property of ear decompositions for which we could not find a convenient reference, and thus it is proved here:

Lemma 59. Let G' be a subgraph of G and suppose both G' and G are strong blocks. Then G' can be extended to G by adding a sequence of open ears.

Proof. Consider a vertex $x \in V(G) \setminus V(G')$. Since G is a strong block, there is a vertex $y \in V(G')$ and an elementary circuit C that contains both x and y . Let $u \in V(G')$ and $v \in V(G')$ be the first predecessor of x on C and v the first successor on C in G' . Then the path $P = (u, \dots, x, \dots, v)$ is an ear. Clearly $G'' = G' \cup P$ is again a strong block. Thus all vertices in $V(G) \setminus V(G')$ can be added to $V(G')$ by sequentially attaching ears. The resulting graph G^* is a spanning subgraph of G . Any missing edges have both endpoints in $V(G^*) = V(G)$ and thus are ears. As an immediate consequence, any circuit basis of G' can be extended to a circuit basis of G by adding elementary circuits composed of an ear as described above and a directed path connecting its attachment vertices in the previously constructed subgraph. \square

The same argument works for strongly connected graphs if open and closed ears are allowed.

B Proofs of Statements in the Main Text

Proposition M12. Let $\kappa = (X_\kappa, R_\kappa, \kappa)$ be a k -CS whose associated CS-matrix $\mathbf{S}[\kappa]$ is reducible, Metzler, and autocatalytic. Then there exists a k' -CS $\kappa' = (X_{\kappa'}, R_{\kappa'}, \kappa')$ with $X_{\kappa'} \subset X_\kappa$, $R_{\kappa'} \subset R_\kappa$, and $\kappa'(X_{\kappa'}) = \kappa(X_{\kappa'})$, such that its associated CS-matrix $\mathbf{S}[\kappa']$ is an irreducible autocatalytic Metzler matrix.

Proof. Since $\mathbf{S}[\kappa]$ is reducible there exists a permutation matrix such that

$$P\mathbf{S}[\kappa]P^{-1} = \begin{pmatrix} A & 0 \\ B & C \end{pmatrix} \quad (42)$$

with irreducible A . Let now $X_{\kappa'} \subset X_\kappa$ represent the species corresponding to the rows of A . Then the triple $\kappa' := (X_{\kappa'}, R_{\kappa'} := \kappa(X_{\kappa'}), \kappa' := \kappa|_{X_{\kappa'}})$ is a CS satisfying $\kappa(X_{\kappa'}) = \kappa'(X_{\kappa'})$. Moreover, $\mathbf{S}[\kappa'] := A$ has negative diagonal entries since $\mathbf{S}[\kappa]$ has negative diagonal entries, which is not changed upon rearrangement of rows and columns. $\mathbf{S}[\kappa]$ being a Metzler matrix implies that $\mathbf{S}[\kappa']$ has only non-negative off-diagonal entries, thus $\mathbf{S}[\kappa']$ is Metzler. An analogous argument can be made for $\mathbf{S}[\kappa''] := C$.

In addition, $\mathbf{S}[\kappa]$ autocatalytic implies that there exists $v > 0 : \mathbf{S}[\kappa]v > 0 \Rightarrow P\mathbf{S}[\kappa]v > 0$. We let $w := Pv$, then $w > 0$:

$$0 < P\mathbf{S}[\kappa]v = P\mathbf{S}[\kappa]P^{-1}Pv = \begin{pmatrix} \mathbf{S}[\kappa'] & 0 \\ B & \mathbf{S}[\kappa''] \end{pmatrix} w = \begin{pmatrix} \mathbf{S}[\kappa']w_1 \\ Bw_1 + \mathbf{S}[\kappa'']w_2 \end{pmatrix} \quad (43)$$

Thus $\mathbf{S}[\kappa']w_1 > 0$. Assume now there is a column in $\mathbf{S}[\kappa']$ without a positive entry. Then one reaction of $\mathbf{S}[\kappa']$ has no product. Hence, there exists a permutation matrix P' such that:

$$P'\mathbf{S}[\kappa']P'^{-1} = \begin{pmatrix} A' & \overset{\rightarrow}{0} \\ C' & x \end{pmatrix} \quad (44)$$

with $x < 0$ and $\vec{0} \in 0^{(m-1) \times 1}$. However, this implies that $\mathbf{S}[\kappa']$ is reducible which is a contradiction. Thereby, $\mathbf{S}[\kappa']$ is autocatalytic. \square

Proposition M14. Let $\mathbf{S}[\kappa]$ be a $k \times k$ irreducible autocatalytic Metzler matrix that exhibits *centralized autocatalysis*. Then

$$\frac{\det \mathbf{S}[\kappa]}{\prod_{m=1}^k \mathbf{S}[\kappa]_{mm}} = 1 - \sum_C \prod_{m \in C} \frac{\mathbf{S}[\kappa]_{m,C(m)}}{|\mathbf{S}[\kappa]_{mm}|} \quad (45)$$

where the sum runs on all permutation cycles.

Proof. We recall the notation P_k for the permutation group on k elements. The first step is noting that if $\mathbf{S}[\kappa]$ is centralized with center m^* , then for each permutation $\pi \in P_k$ with nonzero contribution, i.e. such that $\prod_{m=1}^k \mathbf{S}[\kappa]_{m,\pi(m)} \neq 0$, we get that there exists exactly one permutation cycle C_π with $\pi = C_\pi$, i.e., any permutation with nonzero contribution is a single-cycle permutation. To confirm this, assume indirectly that there exists a permutation π with nonzero contribution and such that $\pi = C_1 \cdot \dots \cdot C_i$, with $i \geq 2$. In particular, C_1 and C_2 have disjoint support and thus $C(m^*) \neq m^*$ cannot hold for both C_1 and C_2 , which leads to contradiction with the definition of centralized autocatalysis. The second step is just computing

$$\frac{\det \mathbf{S}[\kappa]}{\prod_{m=1}^k \mathbf{S}[\kappa]_{mm}}, \quad (46)$$

where the numerator is expanded via the Leibniz formula.

$$\begin{aligned} \frac{\det \mathbf{S}[\kappa]}{\prod_{m=1}^k \mathbf{S}[\kappa]_{mm}} &= \frac{\sum_{\pi \in P_k} \text{sgn}(\pi) \prod_{m=1}^k \mathbf{S}[\kappa]_{m,\pi(m)}}{\prod_{m=1}^k \mathbf{S}[\kappa]_{mm}} \\ &= \frac{\prod_{m=1}^k \mathbf{S}[\kappa]_{mm}}{\prod_{m=1}^k \mathbf{S}[\kappa]_{mm}} + \frac{\sum_C \text{sgn}(C) \prod_{m \in C} \mathbf{S}[\kappa]_{m,C(m)} \prod_{m \notin C} \mathbf{S}[\kappa]_{mm}}{\prod_{m=1}^k \mathbf{S}[\kappa]_{mm}} \\ &= 1 + \sum_C (-1)^{|C|-1} \prod_{m \in C} \frac{\mathbf{S}[\kappa]_{m,C(m)}}{\mathbf{S}[\kappa]_{mm}} \\ &= 1 - \sum_C \prod_{m \in C} \frac{\mathbf{S}[\kappa]_{m,C(m)}}{|\mathbf{S}[\kappa]_{mm}|} \end{aligned} \quad (47)$$

\square

For the following result we recall that the S-graph or metabolite graph of $\mathbf{K}(X, R)$ is defined as $M(\mathbf{K}) := (X(\mathbf{K}), E)$, with

$$E := \{(u, v) \in X \times X \mid \exists r \in R : (u, r), (r, v) \in E(\mathbf{K})\}. \quad (48)$$

Lemma M16. Let $\mathbf{S}[\kappa]$ be an autocatalytic CS Metzler matrix and $\mathfrak{K}[\kappa]$ the set of permutation cycles with non-zero contribution of length ≥ 2 (i.e., nontrivial cycles). Then there is a one-to-one correspondence between $\mathfrak{K}[\kappa]$ and the elementary circuits of the induced subgraph $\mathbf{K}[\kappa]$ such that a permutation cycle (x_1, x_2, \dots, x_k) corresponds to the elementary circuit $(x_1, \kappa(x_1), x_2, \kappa(x_2), \dots, \kappa(x_{k-1}), x_k, \kappa(x_k), x_1)$ in $\mathbf{K}[\kappa]$.

Proof. The fact that $\mathbf{S}[\kappa]$ is a Metzler matrix has two important implications. First, it holds that $\mathbf{A}_{M(\kappa)} = \mathbf{S}[\kappa] - D_k$ with $D_k = \text{diag}(\mathbf{S}[\kappa])$. Thus, there is a one-to-one correspondence α between the set of elementary circuits in the S-graph $M[\kappa]$ and $\mathfrak{K}[\kappa]$. Second, there is a bijection between the elementary circuits in $\mathbf{K}[\kappa]$ and the elementary circuits in $M[\kappa]$ by virtue of the map $\beta : (x, \kappa(x), y) \mapsto (x, y)$ between paths $(x, \kappa(x), y)$ in $\mathbf{K}[\kappa]$ and edges (x, y) in M . Indeed, if (x, y) is an edge in M , then there is a reaction $r \in R$ such that (x, r, y) is a path in $\mathbf{K}[\kappa]$. Since $\mathbf{S}[\kappa]$ is a Metzler matrix, the only such reaction is $r = \kappa(x)$, i.e., the path (x, r, y) is unique, and thus β is indeed a bijection. Thus the concatenation α and β yields the required bijection. \square

Theorem M19. An autocatalytic core is centralized if and only if it is of type I, II, III, or IV in the classification of [14]. Moreover, Eq. (23), i.e.

$$\frac{\det \mathbf{S}[\kappa]}{\prod_{m=1}^k \mathbf{S}[\kappa]_{mm}} = 1 - \sum_C \prod_{m \in C} \frac{\mathbf{S}[\kappa]_{m,C(m)}}{|\mathbf{S}[\kappa]_{mm}|}, \quad (49)$$

holds for all five types of cores.

Proof. We prove the theorem identifying the five autocatalytic cores exactly with the following five motifs and associated CS-matrices, respectively,

$$\text{Type I:} \quad x_1 \xrightarrow[1]{} x_2 \xrightarrow[2]{} 2x_1 \quad \begin{pmatrix} -1 & 2 \\ 1 & -1 \end{pmatrix} \quad (50)$$

$$\text{Type II:} \quad \begin{cases} x_1 \xrightarrow[1]{} x_2 + x_3 \\ x_2 \xrightarrow[2]{} x_3 \\ x_3 \xrightarrow[3]{} x_1 \end{cases} \quad \begin{pmatrix} -1 & 0 & 1 \\ 1 & -1 & 0 \\ 1 & 1 & -1 \end{pmatrix} \quad (51)$$

$$\text{Type III:} \quad \begin{cases} x_1 \xrightarrow[1]{} x_2 + x_3 \\ x_2 \xrightarrow[2]{} x_1 \\ x_3 \xrightarrow[3]{} x_1 \end{cases} \quad \begin{pmatrix} -1 & 1 & 1 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{pmatrix} \quad (52)$$

$$\text{Type IV:} \quad \begin{cases} x_1 \xrightarrow[1]{} x_2 + x_3 \\ x_2 \xrightarrow[2]{} x_1 + x_3 \\ x_3 \xrightarrow[3]{} x_1 \end{cases} \quad \begin{pmatrix} -1 & 1 & 1 \\ 1 & -1 & 0 \\ 1 & 1 & -1 \end{pmatrix} \quad (53)$$

$$\text{Type V: } \begin{cases} x_1 \xrightarrow{1} x_2 + x_3 \\ x_2 \xrightarrow{2} x_1 + x_3 \\ x_3 \xrightarrow{3} x_1 + x_2 \end{cases} \quad \begin{pmatrix} -1 & 1 & 1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \end{pmatrix}. \quad (54)$$

For a better visualization, we prove the theorem using the correspondence of elementary cycles in $\mathbf{K}[\kappa]$ and the permutation cycles as cycles established in the proof of Lemma 16 above.

For type I, there is only one (permutation) cycle,

$$x_1 \rightarrow r_1 \rightarrow x_2 \rightarrow r_2 \rightarrow 2x_2, \quad (55)$$

and thus the autocatalytic core is centralized with centers both $\{x_1, x_2\}$.

For type II, there are two (permutation) cycles:

$$\begin{cases} x_1 \rightarrow r_1 \rightarrow x_2 \rightarrow r_2 \rightarrow x_3 \rightarrow r_3 \rightarrow x_1; \\ x_1 \rightarrow r_1 \rightarrow x_3 \rightarrow r_3 \rightarrow x_1, \end{cases} \quad (56)$$

with $\{x_1, x_3\}$ being both centers.

For type III, there are two (permutation) cycles:

$$\begin{cases} x_1 \rightarrow r_1 \rightarrow x_2 \rightarrow r_2 \rightarrow x_1; \\ x_1 \rightarrow r_1 \rightarrow x_3 \rightarrow r_3 \rightarrow x_1, \end{cases} \quad (57)$$

with $\{x_1\}$ being a center.

For type IV, there are three (permutation) cycles:

$$\begin{cases} x_1 \rightarrow r_1 \rightarrow x_2 \rightarrow r_2 \rightarrow x_1; \\ x_1 \rightarrow r_1 \rightarrow x_3 \rightarrow r_3 \rightarrow x_1; \\ x_1 \rightarrow r_1 \rightarrow x_2 \rightarrow r_2 \rightarrow x_3 \rightarrow r_3 \rightarrow x_1, \end{cases} \quad (58)$$

with $\{x_1\}$ being a center.

For type V, there are five permutation cycles:

$$\begin{cases} x_1 \rightarrow r_1 \rightarrow x_2 \rightarrow r_2 \rightarrow x_1; \\ x_1 \rightarrow r_1 \rightarrow x_3 \rightarrow r_3 \rightarrow x_1; \\ x_2 \rightarrow r_2 \rightarrow x_3 \rightarrow r_3 \rightarrow x_2; \\ x_1 \rightarrow r_1 \rightarrow x_2 \rightarrow r_2 \rightarrow x_3 \rightarrow r_3 \rightarrow x_1; \\ x_1 \rightarrow r_1 \rightarrow x_3 \rightarrow r_3 \rightarrow x_2 \rightarrow r_2 \rightarrow x_1, \end{cases} \quad (59)$$

with no species being a center, see Fig. 10. Finally, an explicit straightforward computation shows the validity of Eq. (49) for all five types. The straightforward generalization with different stoichiometric coefficients and by addition of monomolecular

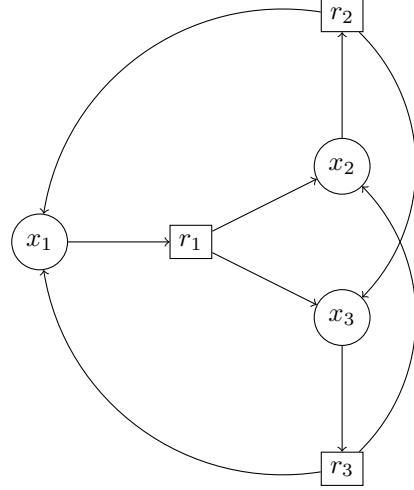


Fig. 10: Depiction of a type V autocatalytic core.

intermediates, e.g., by substituting $x_1 \rightarrow x_2$ with $x_1 \rightarrow I_1 \rightarrow \dots \rightarrow I_n \rightarrow x_2$, is omitted for simplicity of presentation. For the latter, it suffices to say that any argument based on the number of (permutation) cycles are indeed insensitive to the addition of intermediates. \square

Lemma M23. Let $\mathbf{K}' = (X' \cup R', E'_1 \cup E'_2)$ be a subgraph of \mathbf{K} with reactant vertices X' , reaction vertices R' , and edges $E_1' \subseteq X' \times R'$ and $E_2' \subseteq R' \times X'$ such that

1. $|X'| = |R'|$;
2. every $x \in X'$ has out-degree 1 and every $r \in R'$ has in-degree 1.

Then \mathbf{K}' is child-selective with $\kappa(x) = r$ for $(x, r) \in E'_1$.

Proof. Since $x \in X'$ has out-degree 1 and \mathbf{K}' is bipartite, there is a unique $\kappa(x) \in R'$. Analogously, for every $r \in R'$ there is a unique $\mu(r) \in X'$ and we have $(x, \kappa(x)) \in E'_1$ for all $x \in X'$ as well as $(\mu(r), r) \in E'_1$ for all $r \in R'$. Thus we have $\kappa(\mu(r)) = r$ and $\mu(\kappa(x)) = x$, i.e., $\mu(\kappa)$ is the identity of X' and $\kappa(\mu)$ is the identity on R' . Hence κ is a bijection and $\kappa = (X', R', \kappa)$. By construction we have $\mathbf{K}' = \mathbf{K}(\kappa)$. \square

Lemma M27. Let $\kappa = (X_\kappa, E_\kappa, \kappa)$ be a CS. Then $\mathbf{K}(\kappa)$ is strongly connected if and only if $\mathfrak{M}(\mathbf{S}[\kappa])$ is irreducible.

Proof. There is a path (x, r, y) in $\mathbf{K}(\kappa)$ if and only if $r = \kappa(x)$ and y is a product of r in X_κ , which is the case if and only if $\mathbf{S}_{y\kappa(x)} = \mathbf{S}[\kappa]_{xy} > 0$, and hence if and only if $\mathbf{M}_{xy} = \mathfrak{M}(\mathbf{S}[\kappa])_{xy} > 0$.

Firstly suppose $\mathbf{K}(\kappa)$ is strongly connected. Then there is a path from x to y for all $x, y \in X_\kappa$, say $(x = z_0, r_1, z_1, \dots, r_k, y = z_k)$ with $r_i = \kappa(z_{i-1})$. Thus $\mathbf{M}_{z_{i-1}z_i} > 0$ and hence \mathbf{M} is irreducible. Conversely, suppose \mathbf{M} is irreducible. Then for every pair $x, y \in X_\kappa$ there is a sequence of vertices $z_i \in X_\kappa$ with $x = z_0$ and $y = z_k$ such that

$\mathbf{M}_{z_{i-1}z_i} > 0$ and hence $(x = z_0, \kappa(z_0), z_1, \kappa(z_1), \dots, \kappa(z_{k-1}), y = z_k)$ is a path in $\mathbf{K}(\kappa)$. Moreover, for every reaction vertex r there is an edge (x, r) with $x = \kappa^{-1}(r)$ and an edge (r, y) since there is $y \in X_\kappa$ with $\mathbf{M}_{\kappa^{-1}(r), y} > 0$. Hence all vertices of $\mathbf{K}(\kappa)$ are reachable from each other. \square

Lemma M28. If $\mathbf{K}(\kappa)$ is strongly connected, then it does not contain a cut vertex.

Proof. Indirectly assume that v is a cut vertex and $\mathbf{K}(\kappa)$ is strongly connected: this assumption implies that v has at least two in-edges and two out-edges. If v is reaction vertex, $v = r$, its in-edges in $\mathbf{K}(\kappa)$ are of the form $(\kappa^{-1}(r), r)$. Since κ is a bijection, there is at most one such edge, and thus we reach a contradiction. If z is in turn a substrate vertex, i.e. $z = x$, then all its out-edges are of the form $(x, \kappa(x))$, i.e., again there is exactly one such edge, and thus we reach analogously a contradiction. Thus $\mathbf{K}(\kappa)$ cannot contain a cut vertex. \square

Lemma M30. A CS $\kappa = (X_\kappa, E_\kappa, \kappa)$ is autocatalytic if and only if the following conditions both hold:

1. there is a positive vector $v > 0$ such that $\mathbf{S}[\kappa]v > 0$.
2. $\mathbf{K}(\kappa)$ does not possess source and sink vertices;

Proof. Property 1 is identical to property (i) of Def. 5.

First, suppose κ is autocatalytic. Property (i) of Def. 5 further implies that no substrate vertex is a source in $\mathbf{K}(\kappa)$ because the row $\mathbf{S}[\kappa]_x$, corresponding to substrate x , satisfies $\mathbf{S}[\kappa]_x v > 0$ and thus x is product in at least one reaction. By Eq. (31) in the main text, this implies that x is not a source in $\mathbf{K}(\kappa)$. Property (ii) in Def. 5, on the other hand, implies that no reaction vertex is a sink. By Thm. 21, the CS bijection κ explicitly guarantees the existence of a perfect matching in the set of reactant-to-reaction edges in $\mathbf{K}(\kappa)$. Thus, no substrate vertex is a sink and no reaction vertex is a source in $\mathbf{K}(\kappa)$. In summary, statement 2 of the lemma is satisfied.

Conversely, suppose conditions 1 and 2 hold. Since κ is a CS and there is no source or sink vertex in $\mathbf{K}(\kappa)$ then for every reaction vertex r there is an edge (x, r) and an edge (r, y) and thus for every reaction r (that is, for every column) we have $\mathbf{S}[\kappa]_{xr} < 0$ and $\mathbf{S}_{ry} > 0$, i.e., condition (ii) in Def. 5 is satisfied. Together with (i), $\mathbf{S}[\kappa]$ is autocatalytic. \square

Theorem M33. A graph G is a fluffle if and only if it is bipartite with vertex set $X \cup R$ and it has an ear decomposition such that every ear initiates in a reaction vertex $r \in R$ and terminates in a substrate vertex $x \in X$. In this case, all directed open ear decompositions have this property.

Proof. Let G be a fluffle. Then G is bipartite and a strong block. Thus, in particular, it has a directed open ear decomposition [39]. Now consider *any* decomposition $(P_1 = C, P_2, \dots, P_h)$, where $h \geq 1$ and P_1 is an elementary circuit and $P_h, h \neq 1$, is a path. As any ear decomposition starts from an elementary circuit C , it follows that for any substrate vertex $x \in V(C) \cap X$ of C , an out-neighbor of x is contained as well in $V(C)$. By condition (ii) in Prop. 32, such out-neighbor must be unique in the fluffle, and thus any substrate vertex $x \in V(C) \cap X$ cannot be an initial vertex of any ear. Respectively

and in total analogy, a reaction vertex $r \in V(C) \cap R$ of the elementary circuit C cannot be a terminal vertex of an ear, since its only in-neighbor is also located along $V(C)$. Now let G_2 be the graph obtained by attaching the ear P_2 to C . Since G_2 is 2-connected, every substrate vertex $x \in V(G_2) \cap X$ already has a unique out-neighbor in $V(G_2)$ and every reaction vertex $r \in V(G_2) \cap R$ has a unique in-neighbor in $V(G_2)$. Thus, the next ear can only initiate at a vertex $r' \in V(G_2) \cap R$ and terminate at a vertex $x' \in V(G_2) \cap X$. Inductively, this argument holds true for all subsequent ears. Moreover, all this form.

Conversely assume that G is bipartite with vertex partition $X \cup R$, let (C, P_2, \dots, P_h) be an ear decomposition of G such that each P_i initiates in a reaction vertex $r \in V(G) \cap R$ and terminates in a substrate vertex $x \in V(G) \cap X$. Since G has an ear decomposition, then G is a strong block, i.e., G satisfies (iii) in Prop. 32. Moreover, as G is bipartite, the vertices along each ear P_i alternate between reaction vertices in R and substrate vertices in X , and since the first and last vertex of P_i belong to different sets we have $|V(P_i) \cap X| = |V(P_i) \cap R| = |V(P_i)|/2$. Writing $G_1 = C$ and G_i for the graph obtained by attaching the ear P_i to G_{i-1} , we have $|V(G_1) \cap X| = |V(G_1) \cap R| = |C|/2$ and $|V(G_i) \cap R| = |V(G_{i-1}) \cap R| + (|P_i|/2 - 1)$ as well as $|V(G_i) \cap X| = |V(G_{i-1}) \cap X| + (|P_i|/2 - 1)$, where the -1 accounts for the fact that initial and terminal vertices of P_i are already present in G_{i-1} . By induction, it follows immediately that $|V(G_i) \cap R| = |V(G_i) \cap X|$ for all i , and thus $|R(G)| = |X(G)|$, i.e., G satisfies (i). By construction, every substrate vertex $x \in X$ has a single out-neighbor r_x . We have $r_x \in V(C)$ if $x \in V(C)$ and $r_x \in V(P_i)$ if $x \in V(P_i)$. Similarly, every reaction vertex $r \in R$ has a single in-neighbor x_r satisfying $x_r \in V(C)$ if $r \in V(C)$ and $x_r \in V(P_i)$ if $r \in V(P_i)$. Thus G satisfies property (ii). Taking (iii), (i), (ii) together, G is a fluffle. \square

Lemma M34. Let G be a fluffle in \mathbf{K} and G' a subgraph of G that is a strong block. Then G' is a fluffle.

Proof. Trivially, G' is bipartite, each substrate vertex $x \in X(G')$ has out-degree at most 1 and each reaction vertex $r \in R(G')$ has in-degree at most 1. Since G' is a strong block by assumption, it has no vertices with in-degree or out-degree 0, i.e., every $x \in X(G')$ has out-degree 1 and every $r \in R(G')$ has in-degree 1. Thus G' satisfied condition (ii). Moreover, $|X(G')| = |R(G')|$ is satisfied because indirectly $|X(G')| < |R(G')|$ would imply that there is reaction vertex $r \in R(G')$ without in-edge and $|X(G')| > |R(G')|$ would imply that there is a substrate vertex $x \in X(G')$ without out-edge, both leading to contradiction to property (ii), which we just proved. Hence G' also satisfies (i) and is a fluffle. \square

Theorem M35. Let G be a fluffle with vertex partition $X \cup R$ and C an elementary circuit such that $\emptyset \subset G \cap C \subset C$. Then, the connected components of $G \cap C$ are directed paths P_i . Moreover, $G \cup C$ is a fluffle if, and only if, all such paths P_i start from a substrate vertex $x_i \in X$ and terminate with a reaction vertex $r_i \in R$.

Proof. Trivially, any connected component of proper subsets of an elementary circuit are a path whenever they start and terminate with a vertex. In particular, then, the connected components P_i of the intersection $\emptyset \subset G \cap C \subset C$ are paths. Without loss

of generality, we can arrange such paths P_i in circular order along C . Let then Q_i identify the paths in C that start with the terminal vertex of P_{i-1} and terminates with the starting vertex of P_i (here the index i is to be intended in a cyclic group). Note that the Q_i are in 1-to-1 relation with the connected components of the complement $C \setminus G$ of $C \cap G$ in C : they are obtained by adding to each connected component of $C \setminus G$ the starting and terminal vertices to obtain a path. Clearly, by construction, the paths Q_i are ears for G . Since G is a fluffle then it admits itself an ear decomposition. Because the paths Q_i are vertex-disjoint, any arbitrary ear decomposition for G can then be extended to an ear decomposition for $G \cup C$ by adding in arbitrary order the paths Q_i . Theorem 33 therefore implies that $G \cup C$ is a fluffle if and only if each ear Q_i initiates in a reaction vertex $r \in R$ and terminates in a substrate vertex $x \in X$. By complementary construction, this is equivalent to each path P_i initiating in a substrate vertex $x \in X$ and terminating in a reaction vertex $r \in R$. \square

Lemma M40. Two circuitnets \mathcal{C}_1 and \mathcal{C}_2 for fluffles G_1 and G_2 yield the same CS matrix $\mathbf{S}[\kappa]$ if and only if $\mathcal{C}_1 \simeq \mathcal{C}_2$.

Proof. Two circuitnets yield the same CS-matrix if and only if eq. (M35) holds. That is, for the graphs $\bigcup(\mathcal{C}_1) = (X^{(1)} \cup R^{(1)}, E_1^{(1)} \cup E_2^{(1)})$, $\bigcup(\mathcal{C}_2) = (X^{(2)} \cup R^{(2)}, E_1^{(2)} \cup E_2^{(2)})$ it holds that $X^{(1)} = X^{(2)}$, $R^{(1)} = R^{(2)}$, $E_1^{(1)} = E_1^{(2)}$, and in particular $\mathcal{C}_1 \simeq \mathcal{C}_2$. In turn, since \mathcal{C}_1 and \mathcal{C}_2 are circuitnets for fluffles G_1 and G_2 , then $E_1 = (x_i, r_j)$ is a perfect matching (Prop. 32 and Eq. (31)) and thus fully specifies (X, R) . \square

Lemma M41. Let \mathcal{C}_1 and \mathcal{C}_2 be circuitnets for fluffles G_1 and G_2 , respectively, and let C' and C'' be two elementary circuits. Assume $\mathcal{C}_1 \simeq \mathcal{C}_2$, $C' \simeq C''$ and $G'_1 := \bigcup(\mathcal{C}_1 \cup \{C'\})$ is a fluffle. Then $G'_2 := \bigcup(\mathcal{C}_2 \cup \{C''\})$ is a fluffle as well with $\mathcal{C}_1 \cup \{C'\} \simeq \mathcal{C}_2 \cup \{C''\}$.

Proof. First consider any fluffle $G = \bigcup(\mathcal{C})$ with circuitnet \mathcal{C} and let C be an elementary circuit such that $G \cup C$ is a fluffle. By construction, we have $E_1(G \cup C) = E_1(G) \cup E_1(C)$. Moreover, $E(G) \cap E(C) \neq \emptyset$ since $G \cup C$ is fluffle, and hence a strong block. In particular, via Thm. 35, G and C share at least one directed path P initiating at a substrate-vertex and terminating at a reaction-vertex, i.e., $E_1(G) \cap E_1(C) \neq \emptyset$. Hence we have $E_1(G_1) \cup E_1(C') = E_1(G_1) \cup E_1(C'') = E_1(G_2) \cup E_1(C') = E_1(G_2) \cup E_1(C'')$ and $\emptyset \neq E_1(G_1) \cap E_1(C') = E_1(G_2) \cap E_1(C'')$, and thus Thm. 35 applies for G'_2 , concluding that G'_2 is a fluffle. Def. 39 implies $\mathcal{C}_1 \cup \{C'\} \simeq \mathcal{C}_2 \cup \{C''\}$. \square

Lemma M42. For every CS-equivalence class $[\mathcal{C}]$ there is a representative $\hat{\mathcal{C}}$ such that there exists a CS-equivalence class $[\mathcal{C}']$ with representative $\hat{\mathcal{C}'}$ and an elementary circuit C^* such that $\hat{\mathcal{C}'} \cup \{C^*\} \simeq \hat{\mathcal{C}}$ and $|V(\bigcup(\mathcal{C}'))| < |V(\bigcup(\mathcal{C}))|$.

Proof. The statement trivially holds for circuitnets that are single elementary circuits. Let G be the fluffle associated to the circuitnet $\mathcal{C} = \{C_1, \dots, C_h\}$, $h > 1$, listed according to the ordering in Def. 36. Pick now the first circuitnet $\hat{\mathcal{C}} \subseteq \mathcal{C}$ for G such that any strictly subset of $\hat{\mathcal{C}}$ is *not* anymore a circuitnet for G . Clearly, we can always find such suited candidate $\hat{\mathcal{C}}$ from any circuitnet \mathcal{C} for G by iteratively removing the single elementary circuit C_i^* with highest index i and checking whether the remaining set is a circuitnet for the very same fluffle G . Once $\hat{\mathcal{C}}$ is found, a further removal of

the elementary circuit \hat{C}_i^* with highest index i identifies a circuitnet $\hat{\mathcal{C}}' = \hat{\mathcal{C}} \setminus \hat{C}_i^*$ for $G' \subset G$ which, by Cor. 38, is itself a fluffle. Moreover, since $G' \subsetneq G$ and the removal of \hat{C}_i^* removes a ear, $|V(G')| < |V(G)|$. \square

Lemma M51. Let $\mathbf{K}(\kappa^*)$ be a fluffle with irreducible autocatalytic Metzler CS matrix $\mathbf{S}[\kappa^*]$ and let $\mathbf{K}(\kappa)$ be obtained from $\mathbf{K}(\kappa^*)$ by adding a single ear with initial vertex in $R(\mathbf{K}(\kappa^*))$, terminal vertex in $X(\mathbf{K}(\kappa^*))$, and a non-empty set of internal vertices, together with all reaction-to-metabolite edges in $R(\mathbf{K}(\kappa)) \times X(\mathbf{K}(\kappa))$. If $\mathbf{S}[\kappa^*]$ is an autocatalytic CS matrix and $\mathbf{S}[\kappa]$ is a Metzler matrix, then $\mathbf{S}[\kappa]$ is autocatalytic irreducible CS matrix.

Proof. Since $\mathbf{K}(\kappa^*) = \mathbf{K}[\kappa^*]$ is in particular a strong block, the addition of an ear makes the resulting graph G is also a strong block, and thus a fluffle by Thm. 35, since the ear can be extended to an elementary cycle by any directed path in $\mathbf{K}(\kappa^*)$ from its terminal to its initial vertex. Inserting the additional R -to- X edges do not affect the fluffle property, completing it to the corresponding representative CS-equivalence class, i.e., $\mathbf{K}(\kappa)$. Since $\mathbf{S}[\kappa^*]$ is irreducible by assumption, $\mathfrak{M}(\mathbf{S}[\kappa])$ is also irreducible.

Now suppose $\mathbf{S}[\mathbf{K}]$ is a Metzler matrix and let \mathbf{A} be the matrix obtained by renumbering the vertices such that the initial vertex of the ear is $k = |X(\mathbf{K}[\kappa^*])|$, its terminal vertex is 1, and the substrate vertices are ordered consecutively along the directed ear from $k+1$ to l . By construction \mathbf{A} has the form

$$\begin{pmatrix} \mathbf{A}^* & \vec{f} & & & \\ -a_{k+1} & f_{k+2} & \dots & & \\ & -a_{k+2} & f_{k+3} & \dots & \\ & & \ddots & & \\ & & \dots & -a_{l-2} & f_{l-1} \\ & & \dots & -a_{l-1} & f_l \\ \vec{g}_l^\top & \dots & & & -a_l \end{pmatrix}$$

where the vector $\vec{f} \geq 0$ has a strictly positive last (k -th) entry, the vector $\vec{g} \geq 0$ has a strictly positive first entry, and all a_i and f_i are strictly positive. Moreover, all entries that are left blank are non-negative, since \mathbf{A} , like $\mathbf{S}[\kappa]$, is a Metzler matrix. Multiplying \mathbf{A} with a strictly positive vector $\vec{u} = (\vec{u}^*, u_{k+1}, u_{k+2}, \dots, u_l)^\top$ yields $\mathbf{A}\vec{u} = \vec{z} = (\vec{z}^*, z_{k+1}, z_{k+2}, \dots, z_l)^\top$. Taking into account that the blank entries as well as products of the components of \vec{f} with u_{k+1} yields only non-negative contributions, we obtain component-wise inequalities for \vec{z} from the terms that are shown explicitly:

$$\begin{aligned} \vec{z}^* &\geq \mathbf{A}^* \vec{u}^* \\ z_j &\geq f_{j+1} u_{j+1} - a_j u_j \quad \text{for } k+1 \leq j < l \\ z_l &\geq \langle \vec{g}, \vec{u} \rangle - a_l u_l \end{aligned}$$

Hence $\mathbf{A}^* \vec{u}^* > 0$ implies $\vec{z}^* > 0$. Requiring $z_l > 0$ in the third line, we obtain $u_l < \langle \vec{g}, \vec{u}^* \rangle / a_l$, where $\langle \vec{g}, \vec{u}^* \rangle > 0$ since the first entry in \vec{g} and all entries of \vec{u}^* are strictly positive. Thus there is always a positive choice u_l that ensures $z_l > 0$.

The second set of equations implies $z_j > 0$ whenever $0 < u_j < (f_{j+1}/a_j)u_{j+1}$, for $k+1 \leq j \leq l-1$. By induction from $j = l-1$ down to $j = k+1$, therefore there is always a positive choice of u_j that yields a positive entry z_j . In summary, therefore, if $\mathbf{A}^* \vec{u}^* > 0$, i.e., $\mathbf{S}[\kappa^*]$ is autocatalytic, there is $\vec{u} > 0$ such that $\mathbf{A}\vec{u} > 0$, i.e., such that $\mathbf{S}[\kappa]$ is autocatalytic. \square

Theorem M52. Let $\mathbf{S}[\kappa]$ be an irreducible Metzler CS matrix and suppose $\mathbf{S}[\kappa]$ contains an autocatalytic core $\mathbf{S}[\kappa^*]$ as a principal submatrix. Then $\mathbf{S}[\kappa]$ is autocatalytic.

Proof. It suffices to recall that any fluffle can be obtained from a sub-fluffle by adding ears. Going from a fluffle to the canonical representative of its CS-equivalence class amount to adding edges of the form (r, x) , i.e., ears without internal vertices. Thus if $\mathbf{S}[\mathbf{K}^*]$ is an autocatalytic core that is a principal submatrix of an $\mathbf{S}[\mathbf{K}]$ there is a sequence of ears, and thus of corresponding sequence of child selections $\kappa^* = \kappa_0, \kappa_1, \dots, \kappa_h = \kappa$, such that $\mathbf{K}(\kappa_i)$ is obtained from $\mathbf{K}(\kappa_{i-1})$ by adding an ear with a non-empty set of interior vertices. Since $\mathbf{S}[\kappa_h]$ is Metzler and all $\mathbf{S}[\kappa_i]$, $0 \leq i \leq h$ are principal submatrices of $\mathbf{S}[\kappa_h]$, each of the $\mathbf{S}[\kappa_i]$ is an irreducible Metzler CS matrix. Applying Lemma 51 to each of the steps from $\mathbf{S}[\kappa_{i-1}]$ to $\mathbf{S}[\kappa_i]$ for $1 \leq i \leq h$ now implies that $\mathbf{S}[\kappa_i]$ is autocatalytic whenever $\mathbf{S}[\kappa_{i-1}]$ is autocatalytic. \square

Lemma M55. Let G be a fluffle and C an elementary circuit. Then $G \cup C$ is a fluffle if and only if $\emptyset \neq V(G) \cap V(C) = V(E_1(G) \cap E_1(C))$.

Proof. Since fluffles are connected by definition, we may assume that $V(G) \cap V(C) \neq \emptyset$. We observe $B := V(E_1(G) \cap E_1(C)) \subseteq V(E_1(G)) \cap V(E_1(C)) = V(G) \cap V(C) =: A$. First assume $A = B$. Thus $V(G) \cap V(C) \neq \emptyset$ implies that $E_1(G) \cap E_1(C) \neq \emptyset$ and thus $G \cup C$ is a strong block. Moreover, every edge in $E_1(C)$ is either contained in G or disjoint from G , and thus every maximal path in the intersection $G \cap C$ initiates with a metabolite $x \in X$ and terminates with a reaction $r \in R$. Thm. 35 now implies that $G \cup C$ is a fluffle. For the converse, assume that there is $z \in A \setminus B$. If $z \in X$, then there is a unique $y_1 \in V(G)$ with $(z, y_1) \in E_1(G)$ and $y_2 \in V(C)$ with $(z, y_2) \in E_1(C)$. We have $y_1 \neq y_2$ since otherwise $(z, y_1) = (z, y_2) \in E_1(G) \cap E_1(C)$. Thus $z \in X$ has out-degree 2 in $G \cup C$ and hence is not a fluffle. Similarly, if $z \in R$, there is $(y_1, z) \in E_1(G)$ and $(y_2, z) \in E_1(C)$ with $y_1 \neq y_2$ and thus z has indegree 2 in $G \cup C$, which therefore is not a fluffle. \square

Proposition M57. Let $\mathbf{S}[\kappa]$ be a Hurwitz-stable autocatalytic CS-matrix. Then there exists a choice of parameters such that the system $z = f(z) := \mathbf{S} \cdot v(z)$ admits periodic solutions.

Proof. In [63], it is shown that any CS-matrix that is Hurwitz-stable but D -unstable admits a parameter choice for which (6) has periodic solutions. Here, D -unstable means that there exists a positive diagonal matrix D such that $\mathbf{S}[\kappa]D$ is Hurwitz-unstable. Autocatalyticity of $\mathbf{S}[\kappa]$ implies D -instability: indeed, $\mathbf{S}[\kappa]$ contains an autocatalytic core $\mathbf{A}[\kappa']$ as a principal submatrix, which is Hurwitz-unstable by Prop. 11. Without loss of generality, let $\mathbf{A}[\kappa']$ be the leading k' -dimensional principal

submatrix of $\mathbf{S}[\kappa]$, and define $D(\varepsilon) = \text{diag } Po(1_1, \dots, 1_{k'}, \varepsilon_{k'+1}, \dots, \varepsilon_k)$. For $\varepsilon = 0$, $\mathbf{S}[\kappa]D(0)$ is Hurwitz-unstable, as is $\mathbf{A}[\kappa']$. By continuity of eigenvalues, $\mathbf{S}[\kappa]D(\varepsilon)$ remains Hurwitz-unstable for ε small enough, so $\mathbf{S}[\kappa]$ is D -unstable. Stability together with D -instability implies the claim. \square

C The set system of circuitnets of fluffles

Here we collect some properties of set system $\mathfrak{F} \subseteq 2^{\mathcal{C}}$ of circuitnets whose union from fluffles. This is of interest because certain simple properties guarantee simple enumeration or the existence of efficient algorithms to find maximal elements. For our purposes, the following properties of set system (U, \mathfrak{A}) with basis set A and $\mathfrak{A} \in 2^U$ are of interest:

- (i) (X, \mathfrak{A}) is *accessible* if for all $A \in \mathfrak{A}, A \neq \emptyset$: there is $a \in A$ such that $A \setminus \{a\} \in \mathfrak{A}$.
- (ii) (X, \mathfrak{A}) is *strongly accessible* [71] if it is accessible and in addition for any $A, B' \in \mathfrak{A}$ with $A \subsetneq B$ there is $b \in B \setminus A'$ such that $A \cup \{b\} \in \mathfrak{A}$.
- (iii) A strongly accessible set system (X, \mathfrak{A}) is called *commutable* [72] if for any nonempty $A, B \in \mathfrak{A}$ and $a, b \in X : A \cup \{a\} \in \mathfrak{A}, A \cup \{b\} \in \mathfrak{A}$ and $\mathfrak{A} \cup \{a, b\} \subseteq B$ implies $A \cup \{a, b\} \in \mathfrak{A}$.
- (iv) A commutable set system (X, \mathfrak{A}) is called *confluent* if for all $A, B, C \in \mathfrak{A}$ with $B \neq \emptyset$ that $B \subseteq A, B \subseteq C \Rightarrow A \cup C \in \mathfrak{A}$.
- (v) (X, \mathfrak{A}) is an *independence system* or hereditary, if $A \in \mathfrak{A}$ and $\emptyset \neq B \subseteq A$ implies $B \in \mathfrak{A}$.

Note that confluence is not comparable to the other properties.

Theorem 60. \mathfrak{F} is a commutable set system.

Proof. If $\mathcal{C} \in \mathfrak{F}$ then there is a $C \in \mathcal{C}$ such that $\mathcal{C}' := \mathcal{C} \setminus \{C\}$ is again a circuitnet. By Lemma 34, $\bigcup(\mathcal{C}')$ is again a fluffle, i.e., $\mathcal{C}' \in \mathfrak{F}$. That is, \mathfrak{F} is accessible.

Now suppose $\mathcal{C}', \mathcal{C} \in \mathfrak{F}$ and $\mathcal{C}' \subsetneq \mathcal{C}$. Then there is a circuit $C_1 \in \mathcal{C}$ that shares an edge with a cycle $C_2 \in \mathcal{C} \setminus \mathcal{C}'$, since otherwise \mathcal{C} can not be a strong block. Thus $\mathcal{C} \cup \{C_2\}$ is a strong block and Lemma 34 implies that $\mathcal{C} \cup \{C_2\}$ is fluffle. Since \mathfrak{F} is accessible, it is also strongly accessible.

Let $\mathcal{C}, \mathcal{D} \in \mathfrak{F}, \mathcal{C} \subset \mathcal{D}, C_1, C_2 \in \mathcal{D} \setminus \mathcal{C}, \mathcal{C} \cup \{C_1\} \in \mathfrak{F}, \mathcal{C} \cup \{C_2\} \in \mathfrak{F}$. Since $\mathcal{C} \neq \emptyset$, the union of the circuits in $\mathcal{C} \cup \{C_1\}$ and $\mathcal{C} \cup \{C_2\}$ are two strong blocks that share a strong block, namely the union of the circuits in \mathcal{C} . Thus $\mathcal{C}' := \mathcal{C} \cup \{C_1, C_2\}$ is also a strong block, and hence \mathcal{C}' is a circuitnet. Since $\mathcal{C}' \subseteq \mathcal{D} \in \mathfrak{F}$, the union of the circuits is a fluffle by Lemma 34, and thus $\mathcal{C}' \in \mathfrak{F}$. Together with strong accessibility, this implies that \mathfrak{F} is a commutable set system \square

The example in Fig. 11 shows that the set system of fluffles \mathfrak{F} is not confluent.

Instead of circuitnets, we use superpositions of the representatives of fluffle CS classes with the the representatives of the CS classes of elementary circuits. The following statement follows immediately from Lemmas 40, 42, and Proposition 44:

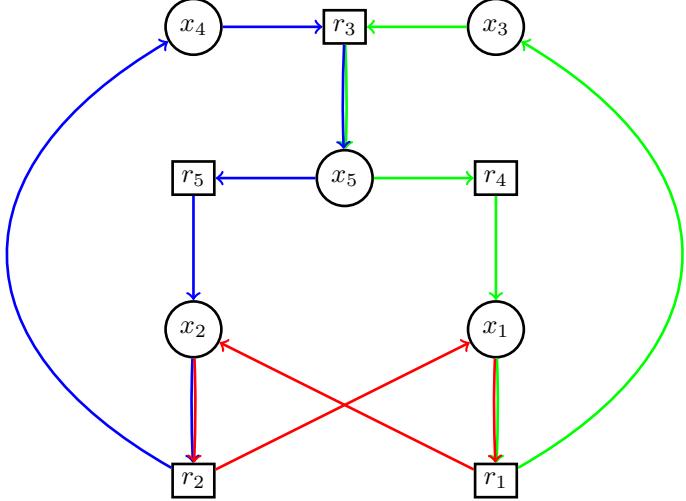


Fig. 11: Counter-example to the hypothesis that \mathfrak{F} is confluent. Consider the two child-selective elementary circuit $C_1 = (r_1, x_3, r_3, x_5, r_4, x_1, r_1)$ (green) and $C_2 = (r_2, x_4, r_3, x_5, r_5, x_2, r_2)$ (blue). Then $\{C_1, C_2\}$ is a circuitnet, but $\{C_1, C_2\} \notin \mathfrak{F}$ since $d_{in}(r_3) = 2$ or $d_{out}(x_5) = 2$. However, the circuit $C_3 = (x_1, r_1, x_2, r_2, x_1)$ (red) is a fluffle. In addition, $\{C_1, C_3\}, \{C_2, C_3\} \in \mathfrak{F}$, but $\{C_1, C_2, C_3\} \notin \mathfrak{F}$ since $\{C_1, C_2\} \notin \mathfrak{F}$.

Corollary 61. Let $\mathcal{C} = \{C_1, \dots, C_h\}$ be a circuitnet for a fluffle $G = \bigcup(\mathcal{C})$. Then the representative $[G]_\simeq$ of its CS-equivalence class is

$$\left[\bigcup(\mathcal{C}) \right]_\simeq = \bigcup_{i=1} [C_i]_\simeq \quad (60)$$

For every circuitnet $\mathcal{C} \in \mathfrak{F}$ we therefore can define a corresponding set of representatives $[\mathcal{C}]_\simeq := \{[C] \mid C \in \mathcal{C}\}$. Note that some of the representatives in \mathcal{C} may be redundant. Now we consider the corresponding set system $[\mathfrak{F}] := \{[\mathcal{C}] \mid \mathcal{C} \in \mathfrak{F}\}$. As a immediate consequence of equ.(60), the arguments in the proof of Thm. 60 carry over to $[\mathfrak{F}]$, and thus we may conclude that

Corollary 62. The set system of circuitnet representatives $[\mathfrak{F}]$ is commutable.

This observation provides a formal basis for the stepwise enumeration on the system of fluffle representatives.

In principle one could also consider the set system (E, \mathfrak{X}) with $\mathfrak{X} := \{E_1(G) \mid G \text{ is fluffle in } \mathbf{K}\}$ on the edge set E of \mathbf{K} . It is easy to see, however, that (E, \mathfrak{X}) is not an accessible set system, since it is possible that the deletion of each of the (r, x) -edge leads to subgraph that is not a strong block and thus also not a fluffle. Thus fluffles cannot be generated efficiently by exploring individual edge-additions.

D Additional computational data

E. coli core model

This CRN comprises in 72 metabolites and 95 reactions [58]. We excluded the following set of small, highly connected molecules which are of minor interest for autocatalysis: cytosolic NAD, NADH, NADP, NADPH, AMP, ADP, H⁺, H₂O, CO₂, coenzyme A, phosphate, oxygen, ubiquinone, and ubiquinol. Since this model describes the central carbon metabolism of *E. coli*, we did not remove ATP as the major energy carrier.

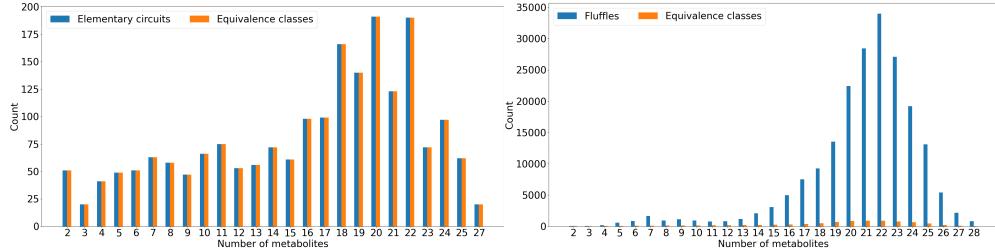


Fig. 12: Length distribution of elementary circuits (left) and size distribution of fluffles and their CS-equivalence classes (right) for the *E. coli* core model.

Fig. 12 in the main text summarizes the distribution of elementary circuits in fluffles for the *E. coli* core network on a log scale. Here, we include the same data on a linear scale

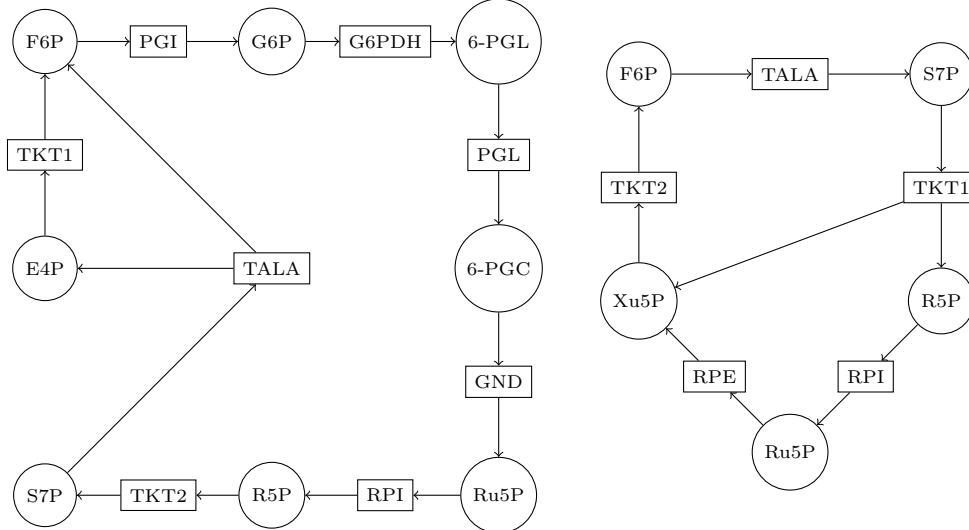


Fig. 13: Two autocatalytic cores in the *E. coli* core model were identified by our graph-theoretical algorithm but not by the ILP formulation of [16].

The comparison of our results with the ILP-formulation of [16] revealed two autocatalytic cores that were detected by our graph-theoretical algorithm only, see Fig. 13. Both are localized in the Pentose-Phosphate-Pathway.

Large E. coli DH5 α 5 model

This CRN consists of 1,951 metabolites and 2,779 [59]. Compared of the core mode, the list of molecules that were removed was augmented to allow for computational feasibility and meaningful results. For brevity, we only provide the species identifier. The assignment to the full names can be found at http://bigg.ucsd.edu/models/iEC1368_DH5a/metabolites. The list contains the metabolites from cytosol (c), periplasm (p), and extracellular (e):

M_23camp_p, M_23ccmp_p, M_23cgmp_p, M_23cump_p, M_2fe1s_c, M_2fe2s_c, M_35cgmp_c, M_3amp_p, M_3cmp_p, M_3fe4s_c, M_3gmp_p, M_3ump_p, M_4fe4s_c, M_ACP_c, M_adp_c, M_alpp_p, M_amp_c, M_amp_p, M_apoACP_c, M_arbtn_e, M_arbtn_fe3_e, M_atp_c, M_btn_c, M_btnso_c, M_ca2_c, M_ca2_p, M_camp_c, M_cdp_c, M_cl_c, M_cmp_c, M_cmp_p, M_co2_c, M_co2_p, M_coa_c, M_colipa_e, M_cpgn_e, M_cpgn_un_e, M_ctp_c, M_cu2_c, M_cu2_p, M_cu_p, M_dadp_c, M_damp_c, M_damp_p, M_datp_c, M_dcAMP_c, M_dcdp_c, M_dcMP_c, M_dcMP_p, M_dctp_c, M_dgdp_c, M_dgMP_c, M_dgMP_p, M_dgTP_c, M_didp_c, M_dimp_c, M_dimp_p, M_ditp_c, M_dnad_c, M_dsbaox_p, M_dsbard_p, M_dsbcox_p, M_dsbcrd_p, M_dsbdox_c, M_dsbdrd_c, M_dsbgox_p, M_dsbgrd_p, M_dtDP_c, M_dtMP_c, M_dtMP_p, M_dtTP_c, M_dudP_c, M_dump_c, M_dump_p, M_dutP_c, M_enter_e, M_fad_c, M_fadh2_c', M_fe2_c, M_fe2_p, M_fe3_e, M_fe3hox_e, M_fe3hox_un_e, M_fecrm_e, M_fecrm_un_e, M_feenter_e, M_feoxam_e, M_feoxam_un_e, M_fmn_c, M_fmnH2_c, M_gdp_c, M_gdp_p, M_gMP_c, M_gMP_p, M_gTP_c, M_gTP_p, M_h2_c, M_h2_p, M_h2O2_c, M_h2O2_p, M_h2O_c, M_h2O_e, M_h2O_p, M_h2S_c, M_h_c, M_h_e, M_h_p, M_hacolipa_e, M_halipa_e, M_hCO3_c, M_hdca_e, M_hqn_c, M_idp_c, M_imP_c, M_imP_p, M_itp_c, M_lipa_e, M_lipidA_e, M_lipidAp_e, M_metsox_R_L_e, M_metsox_S_L_e, M_mq18_c, M_mqn8_c, M_n2O_c, M_na1_c, M_na1_p, M_nad_c, M_nadh_c, M_nadP_c, M_nadph_c, M_nh4_c, M_nh4_p, M_nmN_c, M_nmN_p, M_no2_c, M_no2_p, M_no3_c, M_no3_p, M_no_c, M_o2_c, M_o2_p, M_o2s_c, M_o2s_p, M_pi_c, M_pi_p, M_ppi_c, M_pppi_c, M_q8_c, M_q8h2_c, M_rbflvrd_c, M_ribflv_c, M_s_c, M_sel_c, M_seLn_c, M_slnt_c, M_so2_c, M_so3_c, M_so3_p, M_so4_c, M_thm_c, M_thmmp_c, M_thmpp_c, M_trnaala_c, M_trnaarg_c, M_trnaasn_c, M_trnaasp_c, M_trnacyc_s, M_trnagln_c, M_trnagli_c, M_trnagly_c, M_trnahis_c, M_trnaile_c, M_trnaleu_c, M_trnaly_c, M_trnamet_c, M_trnaphe_c, M_trnapro_c, M_trnasecys_c, M_trnaser_c, M_trnathr_c, M_trnatrp_c, M_trnatyr_c, M_trnaval_c, M_tsul_c, M_tsul_p, M_udp_c, M_ump_c, M_ump_p, M_utp_c.

Homo sapiens erythrocyte model

This CRN consists of 342 metabolites and 469 [61]. Again, we only provide the species identifier. The assignment to the full names can be found at http://bigg.ucsd.edu/models/iAB_RBC_283/metabolites. The list contains the metabolites from cytosol (c) and extracellular

M_gdp_c, M_thmtp_c, M_nad_c, M_ump_c, M_arg_L_e, M_pi_c, M_3moxtyr_e, M_normete_L_e, M_cl_c, M_mal_L_e, M_spmd_e, M_gluala_e, M_thmpp_c, M_thm_e, M_imp_c, M_cdp_c, M_o2_c, M_band_c, M_utp_c, M_cl_e, M_dnad_c, M_35cgmp_c, M_hco3_c, M_dopa_e, M_adp_c, M_na1_e, M_h_c, M_coa_c, M_ptrc_e, M_cmp_c, M_ala_L_e, M_nadp_c, M_nadh_c, M_k_c, M_ppi_c, M_gmp_c, M_nh4_c, M_co_c, M_ctp_c, M_k_e, M_bandmt_c, M_na1_c, M_acnam_e, M_gtp_c, M_nmn_c, M_camp_c, M_udp_c, M_h2o_c, M_4pyrdx_e, M_mepi_e, M_h_e, M_ribflv_c, M_nrpphr_e, M_h2o2_c, M_nadph_c, M_ca2_c, M_fad_c, M_ncam_e, M_ca2_e, M_thmmp_c, M_thm_c, M_atp_c, M_amp_c, M_co2_c, M_pi_e, M_fmn_c, M_gly_e, M_fe2_c

Methanoscina Barkeri model

This CRN consists of 628 metabolites and 690 [60]. Again, we only provide the species identifier. The assignment to the full names can be found at <http://bigg.ucsd.edu/models/iAF692/metabolites>. The list contains the metabolites from cytosol (c) and extracellular

M_f420_2_c, M_trnathr_c, M_h_c, M_h_e, M_f430p2_c, M_f420_3_c, M_imp_c, M_s_c, M_cu2_c, M_dctp_c, M_dtdp_c, M_trnaser_c, M_trnaarg_c, M_so3_e, M_trnaile_c, M_pppi_c, M_cobya_c, M_mma_e, M_ctp_c, M_ni2_c, M_dma_e, M_f420_1_c, M_ala_L_e, M_trnagly_c, M_dcdp_c, M_tma_e, M_nmn_c, M_itp_c, M_h2_c, M_cd2_e, M_btn_c, M_dcmp_c, M_dudp_c, M_no2_c, M_cmp_c, M_tsul_c, M_dgtp_c, M_ch4_e, M_cobalt2_c, M_cbi_e, M_nh4_c, M_adp_c, M_n2_e, M_nad_c, M_f420_0_c, M_cd2_c, M_co2_c, M_dtimp_c, M_trnatrp_c, M_trnalys_c, M_camp_c, M_trnagln_c, M_ca2_c, M_k_e, M_h2s_c, M_f420_5_c, M_trnaasp_c, M_mg2_c, M_co_c, M_f420_4_c, M_pi_c, M_dttp_c, M_f390a_c, M_o2_c, M_f420_6_c, M_nadh_c, M_trnaala_c, M_ind3ac_e, M_dgdp_c, M_h2o_c, M_cdp_c, M_f390g_c, M_fe2_c, M_meoh_e, M_com_c, M_dms_e, M_o2s_c, M_f420_2h2_c, M_datp_c, M_cu2_e, M_cl_e, M_na1_e, M_hco3_c, M_so3_c, M_trnamet_c, M_pac_e, M_alac_S_e, M_co1dam_c, M_dadp_c, M_gtp_c, M_trnaval_c, M_coa_c, M_nadp_c, M_thmpp_c, M_ppi_c, M_f430p1_c, M_glyald_e, M_trnacys_c, M_fmn_c, M_thm_c, M_thmmp_c, M_f420_7_c, M_dcamp_c, M_na1_c, M_nadph_c, M_actn_R_e, M_atp_c, M_dutp_c, M_cob_c, M_co2dam_c, M_dnad_c, M_ump_c, M_cl_c, M_f430p3_c, M_damp_c, M_gmp_c, M_trnaglu_c, M_k_c, M_gdp_c, M_idp_c, M_s_e, M_trnaphe_c, M_f430_c, M_cbl1hbi_e, M_btn_e, M_ribflv_c, M_h2o2_c, M_udp_c, M_trnaleu_c, M_trnatyr_c, M_h2_e, M_trnahis_c, M_unknown_rbfdeg_e, M_amp_c, M_ca2_e, M_dump_c, M_unknown_cbl1deg_e, M_ch4s_e, M_utp_c, M_trnapro_c

E Examples

Example 1 (Autocatalytic Core of Type III does not admit an elementary-circuit CS-representative). The CS-equivalence class of the autocatalytic cores of Types I, II, IV, and V contains a single circuit circuitnet; see the proof of Thm. 19 for a direct verification. In each of these cases, the list of elementary circuits includes at least one (two for Type V) circuit that passes through all species and reaction vertices. The

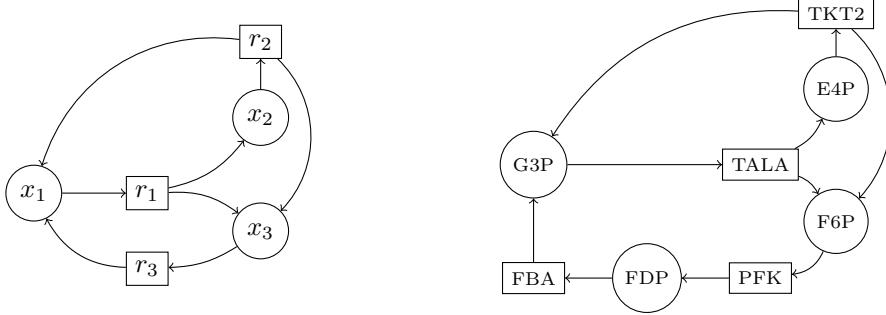


Fig. 14: Example of an autocatalytic core of Type IV according to the classification of [14] (left) and a topologically equivalent autocatalytic core detected in the pentose-phosphate-pathway (PPP) of the *E. coli* core network (right). Abbreviations of metabolites: G3P glyceraldehyde 3-phosphate; E4P erythrose 4-phosphate; F6P fructose 6-phosphate; FDP fructose 1,6-bisphosphatase; PFK phosphofructokinase; FBA fructose-bisphosphate aldolase. Reactions are catalyzed by the enzymes that catalyse them: TALA transaldolase A; TKT2 transketolase 2; PFK phosphofructokinase; FBA fructose-bisphosphate aldolase.

only exception is Type III, shown in Figure 4g. Here, the CS-matrix is

$$\mathbf{S}[\kappa] = \begin{pmatrix} -1 & 1 & 1 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{pmatrix}, \quad (61)$$

and it admits exactly two elementary circuits:

$$\begin{cases} x_1 \rightarrow r_1 \rightarrow x_2 \rightarrow r_2 \rightarrow x_1, \\ x_1 \rightarrow r_1 \rightarrow x_3 \rightarrow r_3 \rightarrow x_1, \end{cases} \quad (62)$$

neither of which traverses all vertices (both have length 2).

Example 2 (Autocatalytic core Type IV in the *E. Coli* core network). Most autocatalytic cores described in the literature are of types I, II, or III. In the *E. coli* core network we found a single example of a Type IV core in the pentose-phosphate-pathway (PPP). This example served as motivation for introducing the concept of centralized autocatalysis.

The Type IV autocatalytic core in Figure 14 introduces one unit of G3P, which yields one unit of E4P and F6P each. E4P then generates one unit of G3P and one unit of F6P. Each of the two units of F6P finally produces a G3P, resulting in a net yield of three G3P. There are indeed three elementary circuits in this network that contain G3P, while all other species are located on at most two elementary circuits. G3P therefore differs from the residual species and, since all elementary circuits coalesce in G3P it forms the autocatalytic center.

F Algorithmic Overview

The algorithm constitutes five main parts, computing the set of all autocatalytic cycles and their properties from the Königs graph of a CRN:

- preparing the network
- decomposition of the network to biochemically functional units
- enumeration of elementary circuits
- enumeration of equivalence classes of fluffles
- classification of enumerated fluffles

The individual steps will be described in the following sections.

Algorithm 2: Algorithm overview

```

Require:  $\mathbf{K}(X, R)$ , set of small molecules to remove  $S$ 
Output :  $\mathcal{A}$  - Set of all autocatalytic Metzler matrices
 $\mathcal{A} \leftarrow \emptyset;$ 
 $\mathcal{A}_Z \leftarrow \emptyset;$ 
 $\mathbf{K}(X, R) \leftarrow \text{RemoveSmallMolecules}(S, \mathbf{K}(X, R))$ ; // see Sec. F.1
for  $SCC$  of  $\mathbf{K}(X, R)$  do
     $\mathcal{T} \leftarrow \text{PartitionNetwork}(\mathbf{K}[SCC], (V, \leq))$ ; // see Sec. F.2
     $\mathbb{E} \leftarrow \text{EnumerateElementaryCircuits}(\mathcal{T})$ ; // see Sec. F.3
     $\mathcal{E} \leftarrow \text{EquivalenceClassAssembly}(\mathbb{E})$ ; // see Alg. 1
     $\mathcal{A} \leftarrow \mathcal{A} \cup \text{AutocatalyticActivity}(\mathcal{E})$ ; // see Sec. F.4
     $\mathcal{A}_Z \leftarrow \text{CheckCentrality}(\mathcal{A})$ ; // see Sec. 5.2.5

```

An overview of all components is given in Alg. 2. The enumeration of equivalence classes of fluffles is described in the main text in Alg. 1 and `CheckCentrality` follows Sec. 5.2.5. Remaining components will be covered in the following sections.

F.1 Network Preparation

Highly interconnected metabolites that do not constitute a focal compound of a reaction, i.e. co-factors like ATP, NADH, etc. and small molecules such as CO_2 and H_2O , do not contribute to the generation of chemically meaningful autocatalytic cycles. They are therefore removed to reduce complexity. We provide a manually curated list of small molecules for this purpose. Otherwise the König graph of the input CRN is not modified and follows standard definitions.

F.2 Partitioning

The partition algorithm takes the König graph of a CRN (without small molecules) as input and computes a partition tree \mathbb{T} whose nodes are labeled by subnetworks in König representation. The resulting partition tree \mathbb{T} is then used to determine interfaces between modules.

To ensure that all elements of \mathcal{E} can be generated via linear combination of elementary circuits, we treat strongly connected components S independently. For

Algorithm 3: PartitionNetwork

```

Require:  $\mathbf{K}(X, R)$ 
Output : Partitioning tree  $\mathcal{T}$ 
 $\mathcal{T} \leftarrow (\mathbf{K}(X, R), \emptyset);$ 
PartitionNetworkRecursion $\mathbf{K}(X, R), \mathcal{T};$ 

def PartitionNetworkRecursion( $\mathbf{K}(X, R)$ , shared  $\mathcal{T}$ ):
     $\mathcal{R} \leftarrow \text{GenerateReactionNetwork}(\mathbf{K}(X, R));$ 
     $ShReD \leftarrow \text{ComputeActualShReDMatrix}(\mathcal{R});$ 
     $P \leftarrow \text{ComputeExpectedShReDMatrix}(\mathcal{R});$ 
     $G := P - ShReD;$ 
     $v \leftarrow \text{ComputeLeadingEigenvector}(G);$ 
     $s \in \{-1, 1\}^n, s_i := \begin{cases} -1 & \text{if } v_i \leq 0 \\ 1 & \text{else} \end{cases};$ 
    if  $v = 0$  then
         $s \in \{-1, 1\}^n, s_i := \begin{cases} -1 & \text{if } v_i < 0 \\ 1 & \text{else} \end{cases};$ 
    if  $\sum_i \sum_j Q_{ij} s_i s_j \leq 0$  then
        return
     $\mathbf{K}(X_1, R_1), \mathbf{K}(X_2, R_2) \leftarrow \text{SplitNetwork}(v, \mathbf{K}(X, R));$ 
    if  $\mathbf{K}(X_1, R_1)$  or  $\mathbf{K}(X_2, R_2)$  is a DAG then
        return
     $V(\mathcal{T}) \leftarrow V(\mathcal{T}) \cup \mathbf{K}(X_1, R_1) \cup \mathbf{K}(X_2, R_2);$ 
     $E(\mathcal{T}) \leftarrow E(\mathcal{T}) \cup (\mathbf{K}(X, R), \mathbf{K}(X_1, R_1));$ 
     $E(\mathcal{T}) \leftarrow E(\mathcal{T}) \cup (\mathbf{K}(X, R), \mathbf{K}(X_2, R_2));$ 
    PartitionNetwork( $\mathbf{K}(X_1, R_1), \mathcal{T}$ );
    PartitionNetwork( $\mathbf{K}(X_2, R_2), \mathcal{T}$ );

```

every $\mathbf{K}[S]$, a network $\mathcal{R} := (R, E)$ with the reactions as vertices and $E := \{(r_1, r_2) | \exists x \in X(S), s_{xr_1}^- > 0, s_{xr_2}^+ > 0\}$ is generated. The next step partitions \mathcal{R} based on a round trip distance metric, called *Shortest Retroactive Distance* (ShReD) [49]. The difference between the expected and actual ShReD matrices, i.e. $G := P - ShReD$, is employed to solve an integer linear programming (ILP) problem: $\max Q := \sum_{i=1}^n \sum_{j=1}^n G_{ij} \cdot s_i \cdot s_j$, s.t. $s \in \{-1, 1\}^n$. By construction, G is symmetric. Thus all eigenvalues are real and for the sake reduced runtimes we take advantage of the fact that the leading eigenvector of G , i.e. the eigenvector to the largest eigenvalue, approximates the solution vector for the given optimization problem as proposed in [53]. More details can be found in [49]. Importantly, each partitioning step yields two submodules $\mathbf{K}(X_1, R_1), \mathbf{K}(X_2, R_2)$, where $R = R_1 \cup R_2, R_1 \cap R_2 = \emptyset$ and $X_i := \{x \in X | \exists r \in R_i : s_{xr}^- > 0 \text{ or } s_{xr}^+ > 0\}$. A partitioning tree T is constructed such that $(\mathbf{K}(X, R), \mathbf{K}(X_1, R_1)), (\mathbf{K}(X, R), \mathbf{K}(X_2, R_2)) \in E(T)$. The upper steps are repeated recursively until either $Q = 0$ or one of the submodules contained is a DAG.

For more details on the functions `GenerateReactionNetwork()`, `ComputeActualShReDMatrix()`, `ComputeExpectedShReDMatrix()`, and

`SplitNetwork()` we refer to the original publication of the implemented partition algorithm [49].

F.3 Enumeration of Elementary circuits

The enumeration of elementary circuits follows the partition tree \mathbb{T} of the last step from bottom to top. First, Johnson's algorithm [42] is applied to all leave nodes, which ensures the detection of all elementary circuits within biochemical functional modules. Upon merging two modules, i.e., for interior nodes of \mathbb{T} , we restrict Johnson's algorithm to metabolites in the intersection that lie along directed paths from one child module into the another, see Figure 15. It should be noted that while we expect that exhaustive enumeration of all circuits will always be possible, enumeration for joined partitions may become infeasible for larger networks, and is therefore size limited in practice.

Algorithm 4: OrientedNetwork

```

Require: Root, OutNetwork, InNetwork, L
Output : G
G  $\leftarrow$  Root;
for  $u \in L$  do
     $V(G) \leftarrow V(G) \cup \{u_{in}, u_{out}\};$ 
    for  $v \in V_{in}^G(u)$  do
        if  $v \in V(\text{InNetwork})$  then
            |  $E(G) \leftarrow E(G) \cup \{(v, u_{in})\};$ 
        else if  $v \in V(\text{OutNetwork})$  then
            |  $E(G) \leftarrow E(G) \cup \{(v, u_{out})\};$ 
    for  $v \in V_{out}^G(u)$  do
        if  $v \in V(\text{InNetwork})$  then
            |  $E(G) \leftarrow E(G) \cup \{(u_{in}, v)\};$ 
        else if  $v \in V(\text{OutNetwork})$  then
            |  $E(G) \leftarrow E(G) \cup \{(u_{out}, v)\};$ 
     $V(G) \leftarrow V(G) \setminus \{u\};$ 
return G

```

Theorem 63. Let (X, R) be a CRN and \mathcal{C} the set of elementary circuits generated via Algorithm 5. Then $\mathbb{E}(\mathbf{K}) = \mathcal{C}$.

Proof. We consider each strongly connected component independently since there are no elementary circuits connecting two strongly connected components by definition. By construction, \mathbb{T} is a strict binary tree, leading to a simple bottom enumeration scheme where a node is visited only after the full sub-trees of both children have been visited. Note that from a purely algorithmic perspective, nodes may be visited in arbitrary order, as circuit sets for each node are independent. However, we require an order of closed sets of nodes for proof by induction. Leave nodes serve as the base case, representing the minimal sub-networks of the partition. Here, enumeration of

Algorithm 5: EnumerateElementaryCircuits

Require: Partitioning tree \mathcal{T}
Output : Elementary Circuits \mathbb{E}

$$\mathbb{E} \leftarrow \emptyset;$$

$$t \leftarrow \text{root}(\mathcal{T});$$

$$\text{EnumElemCircuitsRec}(t, \mathbb{E});$$

def $\text{EnumElemCircuitsRec}(\text{tree node } t \in V(\mathcal{T}), \text{ shared } \mathbb{E})$:

if $t == \text{Leaf}$ **then**

$$| \quad | \quad \mathbb{E} \leftarrow \mathbb{E} \cup \text{Johnsons}(t, \{\})$$
else

$$| \quad | \quad t_1, t_2 = \text{children}(t);$$

$$| \quad | \quad \text{EnumElemCircuitsRec}(t_1);$$

$$| \quad | \quad \text{EnumElemCircuitsRec}(t_2);$$

$$| \quad | \quad Y \leftarrow X(t_1) \cap X(t_2);$$

$$| \quad | \quad L \leftarrow \{\};$$
if $Y \neq \emptyset$ **then**

$$| \quad | \quad | \quad | \quad \text{for } y \in Y \text{ do}$$

$$| \quad | \quad | \quad | \quad | \quad \text{LIRO} \leftarrow \text{OrientedNetwork}(t, t_1, t_2, L); \text{ // see Alg. 4, proof Thm. 63}$$

$$| \quad | \quad | \quad | \quad | \quad \text{RILO} \leftarrow \text{OrientedNetwork}(t, t_2, t_1, L);$$

$$| \quad | \quad | \quad | \quad | \quad \mathbb{E}'_1 \leftarrow \text{Johnsons(LIRO, } \{y\}\text{)};$$

$$| \quad | \quad | \quad | \quad | \quad \mathbb{E}'_2 \leftarrow \text{Johnsons(RILO, } \{y\}\text{)};$$

$$| \quad | \quad | \quad | \quad | \quad \mathbb{E}_1 \leftarrow \text{Backtranslation}(\mathbb{E}'_1); \text{ // see proof Thm. 63}$$

$$| \quad | \quad | \quad | \quad | \quad \mathbb{E}_2 \leftarrow \text{Backtranslation}(\mathbb{E}'_2);$$

$$| \quad | \quad | \quad | \quad | \quad \mathbb{E} \leftarrow \mathbb{E} \cup \mathbb{E}_1 \cup \mathbb{E}_2;$$

$$| \quad | \quad | \quad | \quad | \quad L \leftarrow L \cup \{y\};$$

elementary circuits is achieved by Johnsons' Algorithm [42, 44], which has shown to be complete. In the inductive step, we only consider inner tree nodes, therefore, as the partition tree is strict, parent nodes $\mathbf{K}(X_\kappa, R_\kappa)$ (node) with non-empty children, $\mathbf{K}(X_1, R_1)$ (left) and $\mathbf{K}(X_2, R_2)$ (right).

We first note that by construction $R_\kappa = R_1 \cup R_2$ and $X_\kappa = X_1 \cup X_2$. While $R_1 \cap R_2 = \emptyset$, $X_1 \cap X_2$ is not necessarily empty. Thus, for fusing two children, we consider only elementary circuits containing at least one intersecting metabolite. If $X_1 \cap X_2 = \emptyset$, there is nothing to do since there are not edges connecting $\mathbf{K}(X_1, R_1)$ and $\mathbf{K}(X_2, R_2)$. In any other case, we enumerate elementary circuits containing at least one compound of $X_1 \cap X_2$. Several algorithms have been proposed to enumerate circuits containing a fixed node, such as modifications of Johnsons' Algorithm [42], with available pre-existing implementations [55]. Simply applying Johnson on each node $y_i \in Y = X_1 \cap X_2$ yields a superset of desired circuits, as we also enumerate subsets of elementary circuits of $\mathbf{K}(X_1, R_1)$ and $\mathbf{K}(X_2, R_2)$, which have already been enumerated by induction. This is unproblematic from a purely mathematical standpoint, but would drastically increase runtime complexity. To avoid this duplicate enumeration, we need to additionally enforce the inclusion of reactions from both R_1 and R_2 in each

circuit. We say a circuit crosses child borders, implying that an intersecting metabolite is a product for a reaction from the left child and a reactant for a reaction from the right child or vice versa. More formally, a border node $y_i \in Y \cap C$ of circuit C is product of a unique reaction $r_i^+ \in C$, connected by an in-going edge $r_i^+ y_i \in E(C)$, and educt of unique reaction $r_i^- \in C$, connected by an out-going edge $y_i r_i^- \in E(C)$. If $r_i^+, r_i^- \in R_1$ or $r_i^+, r_i^- \in R_2$ for all $y_i \in Y \cap X(C)$, the circuit C includes no crossing and was therefore already enumerated by induction. Otherwise there is at least one $y_i \in Y \cap X(C)$ s.t. $r_i^+ \in R_1$ and $r_i^- \in R_2$ or vice versa. This leads to the following procedure (Figure 15): For each vertex $y_i \in Y$ we consider two antidromic networks left-in-right-out (LIRO) and right-in-left-out (RILO), derived by removing all out-going edges (respectively in-going) in R_1 and all in-going (respectively out-going) in R_2 adjacent to y_i . Calling Johnson on y_i on both networks now yields exactly the set of desired circuits containing y_i . Therefore, in the i -th iteration all elementary circuits containing y_i are enumerated crossing y_i from the left to the right child or from the right to the left child, avoiding exactly all circuits not crossing at y_i . Again, completeness of the Johnsons algorithm guarantees finding all circuits. However, completeness also implies that circuits crossing metabolites on multiple points $Y' \subseteq Y \cap X(C)$ will be enumerated exactly $|Y'|$ times without further algorithmic restrictions. Given a fixed node order in Y , on i -th iteration any circuit with $y_i \in Y'$ was already enumerated iff there exists a $y_k \in Y' : k < i$. To restrict re-enumeration, we split nodes $y_k \in Y' : k < i$ into novel nodes $y_{k(in)}$ and $y_{k(out)}$ s.t. $y_{k(in)}$ inherits edges in $\mathbf{K}(X_1, R_1)$ and $y_{k(out)}$ edges in $\mathbf{K}(X_2, R_2)$, thereby exactly disabling crossing, but not inclusion of previous metabolites. Backtranslation reverses this operation on the cycle node and edge sets. We note that elementary circuits containing both versions of an intersecting metabolite, $y_{i(in)}$ and $y_{i(out)}$, such as the blue circuit depicted in Figure 15 (y_3), are removed by backtranslation. It follows that all elementary circuits for the parent, $\mathbf{K}(X_\kappa, R_\kappa)$, are exactly enumerated without duplication.

By induction conclusion we obtain that all elementary circuits for $\mathbf{K}(X, R)$ are enumerated. \square

F.4 Autocatalytic capacity

The autocatalytic capacity of a CS matrix can be determined for Metzler and non-Metzler matrices utilizing different methods. In the Metzler case, spectral properties are of great value while for non-Metzler matrices an optimization problem needs to be solved. Consider an $n \times n$ irreducible, invertible Metzler matrix, if $\det(A) = 0$ and there exists an eigenvalue with a positive real part, then A is autocatalytic. For non-Metzler matrices, autocatalytic capacity is determined by the existence of a positive vector $v > 0$, s.t. $Av > 0$. This allows for a relatively straightforward implementation as it is shown in the pseudocode presented in Algorithm 6.

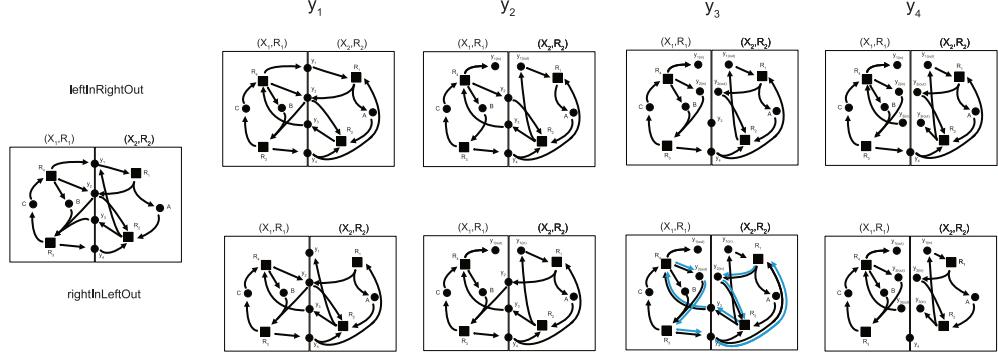


Fig. 15: Depiction of an example for the fusion of two vertices of the partitioning tree \mathcal{T} with four intersecting metabolites, i.e. $Y := X_1 \cap X_2 = \{y_1, y_2, y_3, y_4\}$. Iterative steps for intersecting metabolites are depicted in the four right panels. The upper and lower panel are illustrating the two antidromic oriented networks, left-in-right-out (LIRO) and right-in-left-out (RILO), for each iteration of intersecting metabolites. In detail, in the i -th iteration and LIRO orientation (i -th upper panel), y_i exhibits incoming and outgoing edges only from the left and right child, respectively, while for RILO orientation, y_i receives only incoming and outgoing edges from the right and the left child, respectively. In the $i + 1$ -th iteration for both orientations, LIRO and RILO, y_i is split into $y_{i(in)}$ and $y_{i(out)}$. Subsequently, $y_{i(in)}$ is added to the left and $y_{i(out)}$ to the right network (LIRO) or vice-versa (RILO). In addition, edges point into y_i are strictly retained to appurtenant oriented network, i.e. $y_{i(in)}$ is only incident to edges with origin or target in the in-oriented network. In the lower third panel, an elementary circuit is depicted in blue which contains both versions of the original vertex. Nevertheless, after re-translation into the original vertex sequence, it is no elementary circuit anymore. Thus, it is not considered to be checked for its autocatalytic capacity or for the assembly of larger cycles.

Algorithm 6: AutocatalyticCapacity

```

Require:  $\mathcal{E}$ , set of equivalence classes
Output :  $\mathcal{A}$ , set of autocatalytic matrices
 $\mathcal{A} \leftarrow \emptyset$ ;
for  $E_1(C) \in \mathcal{E}$  do
     $X(C), R(C) \leftarrow \text{SplitVertices}(E_1(C))$ ;
     $n \leftarrow |X(C)|$ ;
     $A \leftarrow \mathbf{S}[X(C), \kappa_C(R(C))]$ ;                                // Compute Matrix from Graph
    if  $A == \mathfrak{M}(A)$  then                                            // Case: Is Metzler Matrix
         $r_{\max} \leftarrow \max\{\text{Real}(\lambda) \mid \lambda \in \text{spectrum}(A)\}$ ;
        if  $r_{\max} > 0$  then                                              // If Hurwitz unstable
             $\mathcal{A} \leftarrow \mathcal{A} \cup \{A\}$ ;
    else                                                               // Case: Is non-Metzler Matrix
        if  $\exists v \in \mathbb{R}_{>0}^{|n|} : Av > 0$  then
             $\mathcal{A} \leftarrow \mathcal{A} \cup \{A\}$ ;

```
