MATCH
Communications in Mathematical and in Computer Chemistry

MATCH Commun. Math. Comput. Chem. **80** (2018) 587-604

ISSN 0340 - 6253

Molecular Codes in Large Metabolic Networks

Christoph Neu, Bashar Ibrahim, Peter Dittrich

Bio Systems Analysis Group
Faculty of Mathematics and Computer Science
Friedrich Schiller University Jena

Ernst-Abbe-Platz 2, 07743 Jena, Germany

(Received May 13, 2017)

Abstract

We provide an approach for identifying molecular codes in large reaction networks. The method exploits particular algebraic properties of closed sets of species forming an algebraic lattice. In a first step the network is reduced by unconnected subnetwork removal and pair merging both preserving molecular code properties. Then connected and closed sub-networks are sampled, each being subsequently analyzed for molecular codes separately by a deterministic algorithm improved by memoization. Thus a parallel computing environment can be easily exploited. We apply our method to a large-scale metabolic network model of *Helicobacter pylori* encompassing 485 species and 554 reactions. 421 unique molecular codes have been found. The vast majority of these codes contain at least one ubiquitous species like protons or water. Filtering for molecular codes without these species in key positions like signs, meaning, or intermediate species, reduces the number of identified codes to 22 only. Whether these codes are utilized by the cell in processing "meaningful" information is yet unknown. All presented data and source code of the new algorithms is available for download.[1]

[1]Download data, source code and supplementary material at:
`https://github.com/biosystemsanalysis/match2018`

1 Introduction

The question of how life has originated from matter is of high interest [1]. This includes, in particular, the question how communication and the processing of semantic information including its encoding [2] has emerged in a chemical prebiotic world [3] and subsequently evolved [4, 2, 5].

In this context, a formal method to asses the capacity of a chemical reaction network to process "meaningful" information has been recently suggested [6] and applied [7]. The basic idea is to measure how easy it is to implement with this network a molecular code. This code is a contingent (arbitrary) mapping between species, that is, a mapping that cannot be inferred from knowing the species and the network alone. Examples for contingent mappings are the genetic code [8], the histone codes [9], and the relation of extracellular signaling molecules to second messengers within the cell [10]. In the genetic code DNA triplets are mapped to amino acids [11]. Contingency refers to the fact that this relation is not a physical consequence of the triplets and amino acids alone. The relation is established by a contextual system, the translation machinery, in an "arbitrary" way, that is, the mapping can be changed by changing the contextual system [12]. Given a reaction network, we can count the number of contingent mappings among molecular species the network can implement, which can be taken as a measure of its "semantic capacity" [6].

A preliminary computational analysis of various chemical systems revealed a quite large spectrum of different semantic capacities [6]. Basically no semantic capacity was found in a model of the atmosphere photochemistry of Mars (cf. Ref. [13]) and four models of combustion chemistries, whereas biochemical systems (like translation [8, 6], gene regulation [6], or molecular self-assembly [7, 14]) posses very high semantic capacities. From this, the hypothesis has been derived that life over the course of evolution is gaining access to (chemical) systems with increasing semantic capacity, that is, with an increasing ability to implement contingent mappings.

The algorithm used so far guarantees finding all molecular codes for a given reaction network. However, due to its high time complexity, only a limited number of networks can be analyzed. In particular, larger networks like many large-scale metabolic networks ($> 10^2$ species and reactions) cannot be processed. Thus a method applicable to large networks would be desirable.

In this work, we provide a novel heuristic approach applicable to large reaction net-

works. The first step consists of network reduction preserving molecular code properties. Then sub-networks are sampled, which are analyzed for molecular codes separately, and the result is collected and integrated. Thus a parallel computing environment can be easily exploited. To exemplify our method, we have applied it to a large-scale metabolic network model of *Helicobacter pylori*. In combustion chemistries, no molecular codes have been found. [6]. Hence, it would be interesting to check if this would be true also for metabolic networks. Furthermore, it has been hypothesized that primitive life might have used the metabolism for processing environmental signals as in chemotaxis [15]. A direct non-coded mapping, as in the droplet systems [16, 17, 18], might have later been extended to include coded (i.e. contingent) mappings as well.

2 Prerequisites

We restrict ourselves to binary sets of signs and meanings, thus binary mappings and binary molecular codes; noting that a generalization towards larger sets and mappings is straight forward.

2.1 Binary molecular code (BMC)

Reaction network A *reaction network* $N = \langle \mathcal{M}, \mathcal{R} \rangle$ is defined by a set of molecular species $\mathcal{M}$ and a set of reactions $\mathcal{R}$ occurring among the molecular species $\mathcal{M}$. For each reaction $\rho \in \mathcal{R}$, let $\mathrm{LHS}(\rho)$ and $\mathrm{RHS}(\rho)$ denote the set of reacting and produced species of reaction ρ, respectively. Note that by using LHS and RHS we abstract from more detailed stoichiometric coefficients and enzymatic control. We simply need to know which species are required and which species are produced by a reaction ρ.

Given a set of species $A \subseteq \mathcal{M}$, we define $R_A = \{\rho \in \mathcal{R} | \mathrm{LHS}(\rho) \subseteq A\}$ as the set of reactions that can "fire" in A and we define $\mathrm{dp}(A) = \bigcup_{\rho \in R_A} \mathrm{RHS}(\rho)$ as the set of species that can be *directly produced* by the reactions that can fire in A (cf. Ref. [19] for relation to point set topology, where $\mathrm{dp}(A)$ it is denoted by $cl(A)$).

Closure A subset of molecular species $C \subseteq \mathcal{M}$ is *closed*, iff $\mathrm{dp}(C) \subseteq C$, that is, iff the application of all possible reactions from $\mathcal{R}$ on C does only produce species from C [20, 21]. For every set of species $A \subseteq \mathcal{M}$ there exists a smallest closed set $G_{CL}(A)$ containing A [22, 23]. We say that $G_{CL}(A)$ is the *closure* of A. Intuitively, the closure of a set of

species contains these species and all those species that can be reached by an arbitrary long reaction path starting with species from A [20]. From an algorithmic perspective, we can construct the closure iteratively by $G_{CL}(A) = A \cup \mathrm{dp}(A) \cup \mathrm{dp}(\mathrm{dp}(A)) \cup \ldots$, or recursively by $A_0 := A$, $A_{i+1} := A_i \cup \mathrm{dp}(A_i)$ with $G_{CL}(A) = A_\infty = \lim_{i\to\infty} A_i$ [21].

Molecular mapping Given a reaction network $N = \langle \mathcal{M}, \mathcal{R} \rangle$ and two sets of molecular species $S, M \subseteq \mathcal{M}$, we say that $f : S \to M$ is a *molecular mapping* with respect to N, iff there exist a set of species $C \subseteq \mathcal{M}$ (called *context*), such that for each pair $s, s' \in S$ with $s \neq s'$: $f(s) \in G_{CL}(C \cup \{s\})$ and $f(s') \notin G_{CL}(C \cup \{s\})$. If there exists a molecular mapping f with respect to N, we also say that N can *implement* the molecular mapping f.

Note that in a reaction network there is usually more than one molecular context $\mathbf{C}$ that implements a particular molecular mapping f. Intuitively, in order to "compute" $f(s)$ with the reaction network N, we put all molecules from the context C together with s in a reaction vessel. Then we repeatedly apply all applicable reaction rules and add the products to the reaction vessel until no novel molecular species can be added anymore. Then we check which molecular species from M is present, which must be – according to our definition – only one species from M and the result of $f(s)$.

Contingent mapping and binary molecular code (BMC) Given a reaction network $N = \langle \mathcal{M}, \mathcal{R} \rangle$ and two binary sets of molecular species $S = \{s_1, s_2\} \subseteq \mathcal{M}$ and $M = \{m_1, m_2\} \subseteq \mathcal{M}$. The mapping $f : S \to M$ with $f(s_1) = m_1, f(s_2) = m_2$ is called a *contingent mapping* and *binary molecular code* (BMC), iff the mapping f and an alternative mapping $g : S \to M$ with $g(s_1) = m_2$ $g(s_2) = m_1$ can be implemented by the reaction network N [6]. This implies that there exist (at least) two sets $C, C' \subseteq \mathcal{M}$, such that the following *BMC condition* holds:

$$
\begin{aligned}
f(s_1) &\in G_{CL}(\{s_1\} \cup C), \text{ and } f(s_2) \notin G_{CL}(\{s_1\} \cup C), \text{ and} \\
f(s_2) &\in G_{CL}(\{s_2\} \cup C), \text{ and } f(s_1) \notin G_{CL}(\{s_2\} \cup C), \text{ and} \\
f(s_2) &\in G_{CL}(\{s_1\} \cup C'), \text{ and } f(s_1) \notin G_{CL}(\{s_1\} \cup C'), \text{ and} \\
f(s_1) &\in G_{CL}(\{s_2\} \cup C'), \text{ and } f(s_2) \notin G_{CL}(\{s_2\} \cup C').
\end{aligned}
$$

The definition catches the notion of contingency as mentioned above, i.e., the elements

of the domain can be mapped to the elements of the codomain in a contingent way by changing the molecular context (cf. [8, 10]). In a semiotic interpretation we can also say domain and codomain contain the signs and meanings, respectively. The molecular context thus becomes the "codemaker" [8], i.e. it is necessary to implement the code.

In general, the definition given above allows for codes of arbitrary size. In order to keep our study tractable, we will focus on molecular codes that are binary, i.e. where S as well as M contain exactly two molecular species [6], as in the example depicted by Figure 1.

Note that for each BMC there is a second *alternative code* implementing the mapping g with $g(s_1) = f(s_2)$ and $g(s_2) = f(s_1)$. $\langle f, g \rangle$ is called a *code pair* (BMCp). In following we count the code pairs.

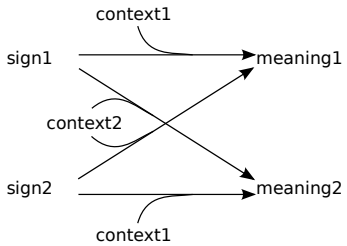

Figure 1. Example of a reaction network (with six species and four reactions) containing four binary molecular codes (BMCs) or two code pairs (BMCps). The reactions are: $\mathcal{R} = \{sign1 + context1 \rightarrow meaning1, sign1 + context2 \rightarrow meaning2, sign2 + context1 \rightarrow meaning2, sign2 + context2 \rightarrow meaning1\}$. The codes are: $\{sign1 \mapsto meaning1, sign2 \mapsto meaning2\}$ with context $C = \{context1\}$, $\{sign2 \mapsto meaning1, sign1 \mapsto meaning2\}$with context $C = \{context2\}$, $\{context1 \mapsto meaning1, context2 \mapsto meaning2\}$ with context $C = \{sign1\}$, and $\{context2 \mapsto meaning1, context1 \mapsto meaning2\}$ with context $C = \{sign2\}$. Note that in general a context can contain many molecular species and that a context cannot always be a sign like in the example. Signs and meanings are always single molecular species.

3 Method and algorithms

We developed a set of algorithms making up a pipeline that searches heuristically for molecular codes in large reaction networks. In an initial step that network is read (SBML input file format) and preprocessed by applying two model reduction rules according to Algorithm 1. In a second step, small connected subnetworks are sampled by Algorithm

2. Subsequently, the subnetworks are filtered by Algorithm 3 to avoid duplicates. In Step 4, each subnetwork is analyzed independently (e.g. in parallel) by an improved exact deterministic molecular code analysis (improved version of rea2bmc from [6]). Finally, all found molecular codes are collected and duplicates removed. With the information from Step 1, the codes lost due to network preprocessing could be reconstructed (straight forward algorithm not implemented). The following describes the algorithms in more detail.

3.1 Network preprocessing (SBML2rea)

The initial network processing step reads the input network (SBML format) and applies two network reduction rules, ensuring that we do not loose information about the network's ability to implement molecular codes.

Firstly, small subnetworks that have less than six species and that are not connected to the remaining network are removed. Obviously, we need at least two signs, two meanings, and two contexts to implement a code, and all of those have to be represented by different molecular species. Thus removing the unconnected small subnetwork does not change the codes the network can implement.

Secondly, groups of species always occurring together on the left hand side or right hand side of any reaction can be replaced by a new pseudo species representing all of them. We do so in Algorithm 1 (pair merging) by successive pairwise merging until no pairs can be merged anymore.

Reducing the network by pair merging can reduce the number of molecular codes, but will not remove a code generating mechanism. That means that codes will always remain in the resulting reduced network allowing to reconstruct the removed codes in a straight forward way. This is expressed by the following lemmata:

Lemma 1 (pair merging 1) *Given a pair of species x, y meeting the condition for pair merging of Algorithm 1, let n_x, n_y be the number of codes where x, y is a sign, respectively, let m_x, m_y be the number of codes where x, y is a meaning, respectively, then $n_x = n_y$ and $m_x = m_y$.*

Proof: Given a code with context C in which x is a sign, due to the "pair condition" (Algorithm 1, Line 11) the single molecular closure $\mathrm{dp}(\{x\})$ contains only x, because

there is no y it can react with. Furthermore y must be contained in the context C, otherwise x would not react with molecules from the context C (Algorithm 1, condition of line 11). There is another (different) code with x and y being exchanged, that is, in which y is a sign and the context $C' = C/\{y\} \cup \{x\}$. Therefore, $n_x = n_y$. Analog argument for $m_x = m_y$. ■

Lemma 2 (pair merging 2) *Reducing the network by merging the pair (x, y), the number of molecular codes is reduced by $n_x + m_x = n_y + m_y$.*

Proof: Due to the "pair condition" (Algorithm 1, Line 11), there are no codes in which x and y appear at the same time as signs or meanings. Nor is it possible that x and y are two different signs in the same code, since one of them must be in the context. Therefore, each of the dual codes (when exchanging x by y) are different and thus we remove half of the codes in which x or y act as a sign or meaning. ■

Since the other half is left and uses the same reaction mechanism to implement the code, we do not loose information about the coding abilities of our network by pair merging.

Algorithm 1 Network Preprocessing

INPUT: Reaction network $(\mathcal{M}, \mathcal{R})$ (SBML format)
OUTPUT: Reduced reaction network (internal .rea format)

Unconnected Subnetworks Removal
1: Remove all subnetworks (reactions and species) that have less than six species and that are not connected to the remaining network.

Pair Merging
2: **repeat**
3: CREATE List of pairs (x, y) of species $x \in \mathcal{M}, y \in \mathcal{M}$ occurring only together as reactants or products, that is, for each reaction $\rho \in \mathcal{R}$: if $x \in \text{LHS}(\rho)$ $(x \in \text{RHS}(\rho))$ then $y \in \text{LHS}(\rho)$ $(y \in \text{RHS}(\rho))$.
4: REMOVE trivial duplicate (y, x), if (x, y) is contained in the list.
5: **for all** Remaining pairs (x, y) of the list **do**
6: **for all** Reactions ρ **do**
7: REPLACE Species x and y in reaction ρ by a new species `x_and_y`.
8: **end for**
9: **end for**
10: **until** No pair has been found

Technically, we use the python library libSBML [24] for reading and generate an internal data format (rea format).

3.2 Subnetwork sampling (rea2reas)

Because the reduced network is usually still too large for a complete molecular code analysis by a tool like rea2bmc, we create subnetworks such that a code in a subnetwork is also a code in the full network. The subnetwork generation ensures that all species are connected and that the subnetwork is a closed set of the original one. The basic idea of Algorithm 2 (implemented by the tool *rea2reas*) is to start with a random species and then to randomly follow reactions backward and forward from the species visited. This should also lead to many reaction pathways within the set of species, which is beneficial for obtaining a molecular code [6].

Algorithm 2 Subnetwork Sampling

INPUT: Reaction network: $(\mathcal{M}, \mathcal{R})$; Number of subnetworks to generate: n; target maximal subnetwork size: s (in this study $s = 16$)
OUTPUT: Set of subnetworks

1: **repeat** n times
2: CHOOSE a random species $i := randomSpecies(\mathcal{M})$ (called "seed")
3: GENERATE a set C with that species: $C := \{i\}$.
4: **repeat**
5: CHOOSE randomly from the current set C a species $i := randomSpecies(C)$.
6: Remember C: $C' := C$ ("previous closure")
7: CHOOSE randomly a reaction ρ in which species i participates (i.e. $i \in \mathrm{LHS}(\rho)$ or $i \in \mathrm{RHS}(\rho)$) and add its reactants to the set, $C := C \cup \mathrm{LHS}(\rho)$.
8: Generate the closure of the set: $C := G_{CL}(C)$
9: **if** $|C| > s$ and previous closure $|C'| \geq 6$ **then**
10: SAVE previous closure C' as a subnetwork.
11: **else if** $|C| > s$ and previous closure $|C'| < 6$ **then**
12: CHOOSE(a) reaction ρ with respect to C' as in Step 7 that generates the smallest closure greater s and save that closure as a subnetwork $G_{CL}(C' \cup \mathrm{LHS}(\rho))$.
13: **end if**
14: **until** Subnetwork successfully generated
15: **until**

3.3 Subnetwork filtering (rea2reas)

The network sampling algorithm can produce network duplicates or a subnetwork contained in another subnetwork. Therefore we apply a network filtering according to Algorithm 3 (implemented as part of *rea2reas*), removing subnetworks that can only contain possible BMCs that are already part of a different subnetwork. Further networks that cannot contain a BMC according to the following Lemma 3 are removed. The result is a

list of individual subnetworks (*rea* files) sorted by size.

Lemma 3 (BMC requirement) *A reaction network able to implement a BMC must contain at least four reactions each with at least two reactants and one product.*

Proof: For the BMC condition, four different conditions must be distinguished, each being a conjunction of the presence of at least two different species (sign and context). A conjunction can only be implemented by a reaction with at least two molecular species.

■

.

Algorithm 3 Subnetwork Filtering

INPUT: Set of subnetworks (generated by Algorithm 2)
OUTPUT: Reduced set of subnetworks

Duplicates

1: One of two subnetworks that are identical will be removed.

Content by Species

2: A subnetwork that includes only species that are part of a bigger subnetwork is removed.

Minimal Structure

3: A subnetwork that does not contain at least 4 reactions each with at least 2 reactants and 1 product is removed.

3.4 Improved molecular code analysis (rea2bmc)

Each subnetwork is analyzed for molecular codes by an improved deterministic code analysis described below in more detail. This algorithm requires to generate all n closed sets of the subnetwork, consisting of m species, and has a time complexity of $O(n^2m^4)$. Thus, if the number of closed sets exceeds a certain limit (here, 20 000), the computation is stopped and the subnetwork discarded.

The molecular code analysis is improved by hashing and memorization techniques. In particular, we pre-calculate the lattice of closed sets, the contains-relation of all closed sets, the closure-union of a closed set with a single-molecule-closure (a closed set generated by a single molecule), and whether there is a path from a species i to a species i'. Then we check every combination of two signs, two meanings, and two contexts for the BMC condition described above. Due to memorization, the computation of unions and contains-relations during each of these checks is a fast index-set look up. Furthermore, only those

signs and meanings are checked for which there is a path from each sign to each meaning, which is again a quick look up due to memorization.

Note that, in the worst-case the time-complexity is exponential with respect to the network size, because there can be an exponential amount of closed sets.

4 Results - application to a metabolic network model of *Helicobacter pylori* 26695

We applied our algorithms to the expanded metabolic network of *Helicobacter pylori* (iIT341 GSM/GPR [2] , strain 26695) [25] from the BiGG Models Database [26]. The network contains 485 metabolites and 554 reactions.

4.1 Evaluation of original iIT341 network

We performed six identical analysis runs including network preprocessing (Algorithm 1), subnetwork sampling with $n = 20\,000$ seeds and a threshold $s = 16$ (Algorithm 2) and subnetwork filtering (Algorithm 3). The remaining subnetworks where subsequently analyzed by the improved rea2bmc algorithm.

Table 1. BMC analysis of six identical evaluations of the iIT341 network, with a seed of $n = 20\,000$ and a threshold s of 16 for the subnetwork sampling (Algorithm 2).

Subnetwork #	1	2	3	4	5	6	1-6
Total Number of Subnetworks	7981	8088	7966	8000	8070	7985	48090
Subnetworks without a BMC	6720	6820	6728	6613	6774	6675	40330
Runs canceled due to closure size	1152	1163	1154	1272	1212	1220	7173
Subnetworks with BMCs	109	105	84	115	84	90	587
Total Number of found BMCps	340	297	239	362	266	332	1836
Total Number of BMCps After Duplication-removal	174	155	137	169	126	165	421

[2]GSM refers to genome-scale model, and GPR refers to gene-protein-reaction associations where the letter i stands for an *in silico* strain, IT is the initial of the principal author of the reconstruction, and 341 is the number of genes included in the reconstruction.

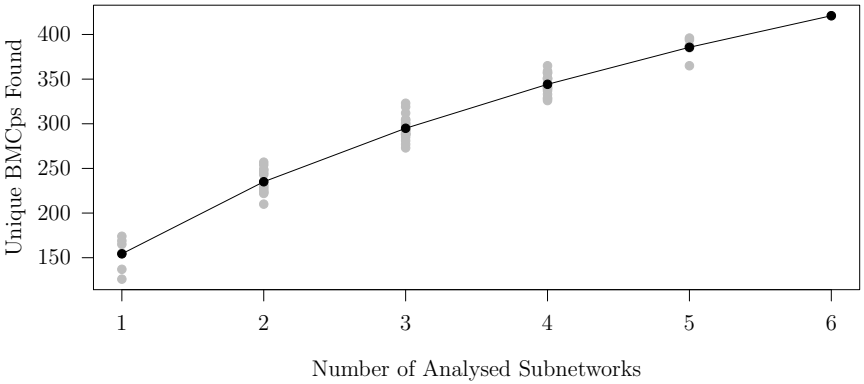

Figure 2. Number of unique BMCps versus the number of analyzed iIT341 metabolic subnetworks. Every combination of six subnetworks have been analyzed, each generated with a seed of $n = 20\,000$ and a threshold s of 16 for the subnetwork sampling (Algorithm 2).

In each run between 126 to 174 unique BMCps have been found (see Table 1). Joing the results of all six runs and removing duplicates, 421 unique BMCps could be identified. Figure 2 shows how the number of unique BMCps found increases and slowly converges with each run, and that we can expect to find more unique BMCps when conducting more runs.

A typical molecular code found can be seen in Figure 3. Observing this BMC in greater detail one can see, that the code is implemented via important reactions of the tyrosin and phenylalanine biosynthesis and tyrosin degradation. Infect, if the mapping from tyrosin to 3-(4-Hydroxyphenyl)pyruvate is deficient it can cause tyrosinemia type II in humans [27].

What is also apparent is that the mapping from prephenate and L-tyrosin to carbonic acid is implemented by H_2O and H^+ respectively. Infect, the overwhelming amount of BMCs contain ubiquitous species like water or protons as key components. Although this does not invalidate the mapping from the sign to the meaning, it is questionable whether this code can be used to transduct information. This can be seen in the shown example. There, a given context will be sufficient to produce carbonic acid whether prephenate is present or not, since water, as the intermediate, is ubiquitous.

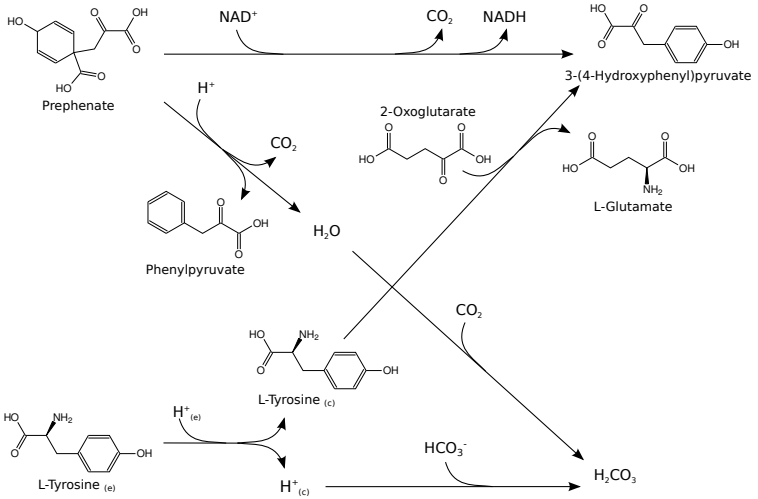

Figure 3. Representative example for a common BMC found in the unmodified metabolic network iIT341. Prephenate and extracellular L-tyrosine are the signs and 3-(4-hydroxyphenyl)pyruvate and carbonic acid are the meaning. All species react in the cytosol except for species indexed with (e) which are present in the extracellular space. All shown molecules of all figures are protonated as represented in the KEGG Database [28, 29, 30]. Abbreviations: Nicotinamide adenine dinucleotide (NAD^+). Parameter: $s = 16$.

4.2 Evaluation of iIT341 network with ubiquitous species as inflow

To skip molecular codes with ubiquitous species, we have added to the reaction network a reaction with no reactants and the ubiquitous species H_20, H^+,CO_2, H_2CO_3 and HCO_3^- as its products. This spontaneous inflow ensures that these species are contained in any closed set and thus can neither be a sign nor a meaning nor an intermediate reactant necessary for implementing a code.

To reduce computational time, all species contained in the smallest closed set (i.e., the closure of the empty set) are removed from the reaction network. This deletion does not change the molecular codes, because the smallest closed set is contained in any other closed set. Note that a resulting code with context C is also a code of the unmodified network, for example by using a context $C \cup \{$ H_20, H^+,CO_2, H_2CO_3 and $HCO_3^-\}$.

Again we analyze the "modified"-iIT341 six times like described previously. As

expected a smaller amount of BMCps were found, namely 22 only (see Table 2 and Figure 4 for how these are distributed over the six runs). Typical examples are shown by Figures 5 and 6).

Table 2. BMC analysis of six identical evaluations of the "modified"-iIT341 metabolic network (ubiquitous species removed), with a seed of $n = 20\,000$ and a threshold $s = 16$ for the subnetwork sampling (Algorithm 2).

Subnetwork #	1	2	3	4	5	6	1-6
Total Number of Subnetworks	5803	5705	5752	5771	5845	5837	34713
Subnetworks without a BMC	5705	5599	5660	5682	5750	5732	34128
Runs canceled due to closure size	97	98	87	84	91	103	560
Subnetworks with BMCs	1	8	5	5	4	2	25
Total Number of found BMCps	2	14	8	11	8	7	50
Total Number of BMCps After Duplication-removal	1	10	7	9	7	4	22

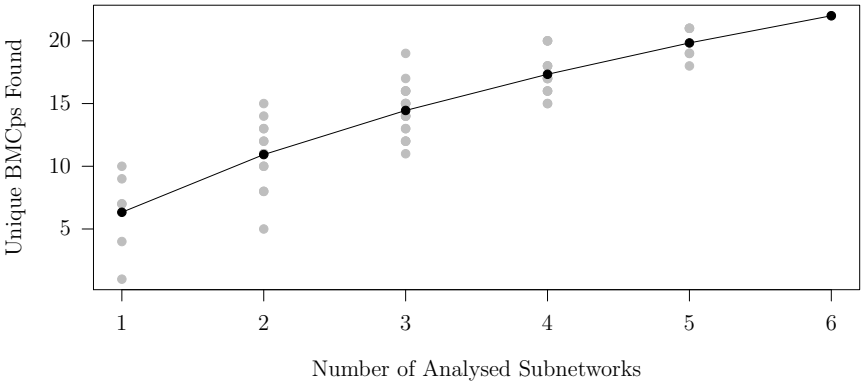

Figure 4. Number of unique BMCps versus the number of analyzed "modified" iIT341 metabolic subnetworks (ubiquitous species removed). Every combination of six subnetworks have been analyzed, each generated with a seed of $n = 20\,000$ and a threshold $s = 16$ for subnetwork sampling (Algorithm 2).

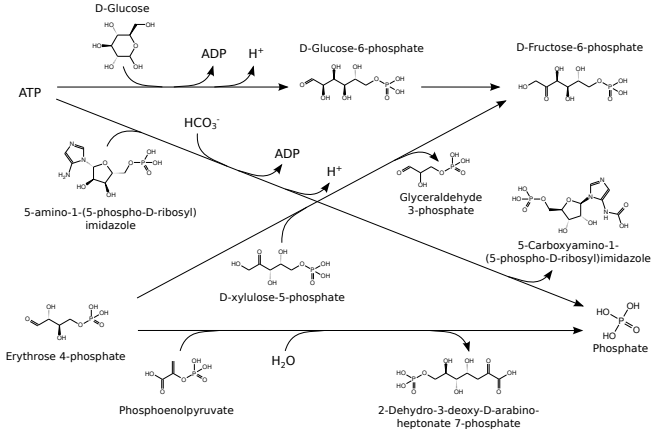

Figure 5. A schematic representation of one BMCp found in the "modified"-iIT341 metabolic network. ATP and Erythrose 4-phosphate are the signs and D-Fructose-6-phosphate and Phosphate are the meanings. Abbreviations: Adenosine diphosphate (ADP). Parameter: $s = 16$.

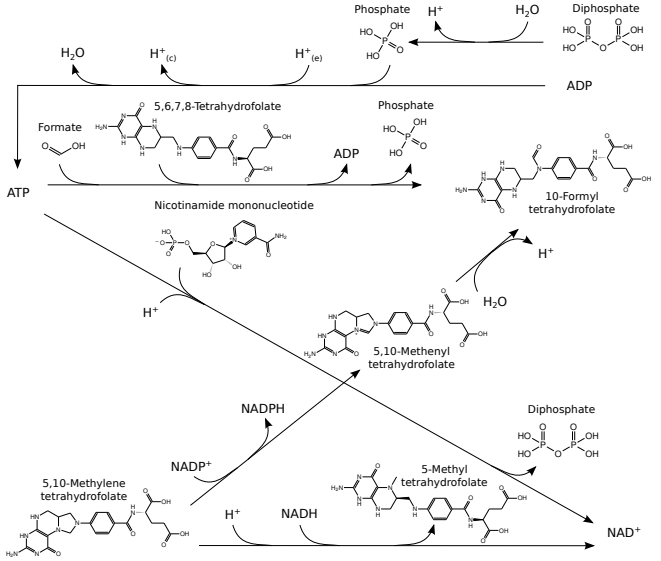

Figure 6. A schematic representation of three BMCps found in the "modified"-iIT341 metabolic network. All three BMCps consist of 5,10-Methylenetetrahydrofolate as a sign and 10-Formyltetrahydrofolate as well as NAD^+ as the meanings. The second sign could either be ATP, ADP or Diphosphate. Parameter: $s = 16$.

Although the reactions of all BMCps of the "modified"-iIT341 participate in various pathways, 50% of the found BMCps include Adenosine triphosphate (ATP) as a sign. Furthermore in the BMC of Figure 6 ATP enables 3 different BMCs with one code making mechanism. Note that ATP has been identified to have a pivotal role in the formation of autocatalytic sets in metabolic networks [31, 32]. However, if this is related to ATP's role in the molecular codes found here remains unclear.

When we compare the BMCps of the modified network with the ones found in the original, we do only see one duplication (see last entries in supplementary Table 2 and 3). This indicates that our procedure allows to identify more complex codes in a given subnetwork. This complexity could additionally be changed by adjusting the threshold for the size of a subnetwork s. While a reduction of s would lead to many missed BMCs, an increase would cause more subnetworks to exceed the limit of closed sets and thus the complete subnetwork would be discarded.

5 Conclusion

The new approach introduced here can be applied to large-scale reaction networks to identify molecular codes. In the metabolic network model of *Helicobacter pylori*, consisting of 485 molecules and 554 reactions, 421 molecular codes have been identified using the full metabolic network. Although the total number of molecular codes remains unknown, Figure 2 indicates that a considerable portion of BMCs requiring small subnetworks have been found.

Basically all molecular codes found when analyzing the full network by sampling small subnetworks contained an ubiquitous species like H_20, H^+,CO_2, H_2CO_3 and HCO_3^- as a sign, meaning, or within the reaction path from sign to meaning. Thus they appear to be unlikely candidates for a coded mapping actually being used [33] in processing "meaningful" information.

When adding an inflow of the ubiquitous species, which effectively eliminates codes with ubiquitous species, more complex codes were found. Note that by removing the closure of the ubiquitous species from the network allowed us to use the same targeted maximal subnetwork size s as for the unmodified network.

However even in those codes there is often one sign and one meaning molecule having a similar chemical structure, for example, ATP and phosphate (as seen in Fig. 5), where

phosphate is a substructure of ATP. This opposes the demand that sign and meaning should come from "two different worlds" [8]. Nevertheless, the context is not always ubiquitous, e.g., the amount of D-glucose can vary. In other words, different mappings can be active at different points in time, which reflects the contingent character of the relation between sign and meaning molecules.

Analogous to removing ubiquitous species is the sampling of larger subnetworks, which should allow the discovery of even more complex codes. These codes obtained from further increasing the sampling size would be interesting insofar the signs and meanings could be chemically less similar, due to a more complex reaction path.

Searching for molecular codes in large networks is computationally expensive. The computation of the twelve runs yielding 442 unique codes required roughly seven months CPU time. Noting that the computation can be easily executed in parallel on a compute cluster running a conventional batch system, one run has been executed in 14 hours using 30 cores. However, further work is needed to target the more complex molecular codes using a large number of species as a context. For this, a different strategy for finding molecular codes that uses subgraph isomorphism [34] applied to the lattice of closed sets (not to the reaction network) might be beneficial.

We do not dare to interpret our preliminary findings further with respect to the origin of life, the origin of biological semantic information, or the processing of information by a cells metabolism. A deeper study with significantly increased computational resources and more network data especially across species would be demanded.

Acknowledgment: The authors thank the reviewers leading to a significantly improved manuscript and correction of the algorithms. This work has been brought forward through the "Mathematics in Chemistry Meeting" organized by J. Jost, P. Stadler, and G. Restrepo at the MPI for Mathematics, October 2016, Leipzig, which we gratefully acknowledge. The work has been financially supported in part by the European Union through funding under FP7-ICT- 2011-8 project HIERATIC; Contract grant number: 316705. The authors thank Richard Henze and Jakob Fischer for careful reading of the manuscript.

References

[1] S. I. Walker, P. C. Davies, G. F. Ellis, *From Matter to Life: Information and Causality*, Cambridge Univ. Press, Cambridge, 2017.

[2] M. Barbieri, *Code Biology*, Springer, Cham, 2015.

[3] B.-O. Küppers, The nucleation of semantic information in prebiotic matter, in: E. Domingo, P. Schuster (Eds.), *Quasispecies: From Theory to Experimental Systems*, Springer, 2015, pp. 23–42.

[4] G. Hernández, R. Jagus (Eds.), *Evolution of the Protein Synthesis Machinery and Its Regulation*, Springer, 2016.

[5] M. Bossert, *Information-and Communication Theory in Molecular Biology*, Springer, 2018.

[6] D. Görlich, P. Dittrich, Molecular codes in biological and chemical reaction networks, *PLoS ONE* **8** (2013) #e54694.

[7] D. Görlich, G. Escuela, G. Gruenert, P. Dittrich, B. Ibrahim, Molecular codes in the human inner-kinetochore model: Relating cenps to function, *Biosem. Neth.* **7** (2014) 223–247.

[8] M. Barbieri, Biosemiotics: a new understanding of life., *Naturwissenschaften* **95** (2008) 577–599.

[9] S. Kühn, J.-H. S. Hofmeyr, Is the "histone code" an organic code? *Biosem. Neth.* **7** (2014) 203–222.

[10] J. Monod, *Zufall und Notwendigkeit*, München: dtv, 9 edition, 1971, english title: "Chance and Necessity".

[11] T. Tlusty, A colorful origin for the genetic code: Information theory, statistical mechanics and the emergence of molecular codes, *Phys. Life Rev.* **7** (2010) 362–376.

[12] C. G. Acevedo-Rocha, N. Budisa, Xenomicrobiology: a roadmap for genetic code engineering, *Microb. Biotechnol.* **9** (2016) 666–676.

[13] F. Centler, P. Dittrich, Chemical organizations in atmospheric photochemistriesa new method to analyze chemical reaction networks, *Planet. Space Sci.* **55** (2007) 413–428.

[14] J. Wang, G. Hu, M. Ji, Almost parallel strong trace model of self–assembly polypeptide nanostructure, *MATCH Commun. Math. Comput. Chem.* **77** (2017) 783–798.

[15] M. D. Egbert, X. E. Barandiaran, E. A. Di Paolo, A minimal model of metabolism-based chemotaxis, *PLoS Comput. Biol.* **6** (2010) #e1001004.

[16] M. M. Hanczyc, S. M. Fujikawa, J. W. Szostak, Experimental models of primitive cellular compartments: encapsulation, growth, and division, *Science* **302** (2003) 618–622.

[17] T. Froese, N. Virgo, T. Ikegami, Motility at the origin of life: Its characterization and a model, *Artif. Life* **20** (2014) 55–76.

[18] J. Čejková, S. Holler, T. Q. Nguyenová, C. Kerrigan, F. Štěpánek, M. M. Hanczyc, Chemotaxis and chemokinesis of living and non-living objects, in: A. Adamatzky (Ed.), *Advances in Unconventional Computing*, Springer, 2017, pp. 245–260.

[19] G. Benkö, F. Centler, P. Dittrich, C. Flamm, B. M. Stadler, P. F. Stadler, A topological approach to chemical organizations, *Artif. Life* **15** (2009) 71–88.

[20] W. Fontana, Algorithmic chemistry, in: C. G. Langton, C. Taylor, J. D. Farmer, S. Rasmussen (Eds.), *Artificial Life II*, Addison–Wesley, Redwood, 1992 pp. 159–210.

[21] W. Fontana, L. Buss, "The arrival of the fittest": Toward a theory of biological organization, *Bull. Math. Biol.* **56** (1994) 1–64.

[22] P. S. di Fenizio, P. Dittrich, W. Banzhaf, J. Ziegler, Towards a theory of organizations, in: *German Workshop on Artificial Life (GWAL 2000), Bayreuth*, 2000 pp. 1–14.

[23] P. Dittrich, P. Speroni di Fenizio, Chemical organization theory, *B. Math. Biol.* **69** (2007) 1199–1231.

[24] B. J. Bornstein, S. M. Keating, A. Jouraku, M. Hucka, Libsbml: an api library for sbml, *Bioinf.* **24** (2008) 880–881.

[25] I. Thiele, T. D. Vo, N. D. Price, B. O. Palsson, Expanded metabolic reconstruction of helicobacter pylori (iIT341 GSM/GPR): an in silico genome-scale characterization of single- and double-deletion mutants, *J. Bacteriol.* **187** (2005) 5818–5830.

[26] Z. A. King, J. Lu, A. Dräger, P. Miller, S. Federowicz, J. A. Lerman, A. Ebrahim, B. O. Palsson, N. E. Lewis, BiGG models: A platform for integrating, standardizing and sharing genome-scale models, *Nucleic Acids Res.* **44** (2015) D515–D522.

[27] R. Rettenmeier, E. Natt, H. Zentgraf, G. Scherer, Isolation and characterization of the human tyrosine aminotransferase gene, *Nucleic Acids Res.* **18** (1990) 3853–3861.

[28] M. Kanehisa, S. Goto, KEGG: Kyoto encyclopedia of genes and genomes, *Nucleic Acids Res.* **28** (2000) 27–30.

[29] M. Kanehisa, Y. Sato, M. Kawashima, M. Furumichi, M. Tanabe, KEGG as a reference resource for gene and protein annotation, *Nucleic Acids Res.* **44** (2015) D457–D462.

[30] M. Kanehisa, M. Furumichi, M. Tanabe, Y. Sato, K. Morishima, KEGG: new perspectives on genomes, pathways, diseases and drugs, *Nucleic Acids Res.* **45** (2016) D353–D361.

[31] Á. Kun, B. Papp, E. Szathmáry, Computational identification of obligatorily autocatalytic replicators embedded in metabolic networks, *Genome Biol.* **9** (2008) #R51.

[32] F. Centler, C. Kaleta, P. S. di Fenizio, P. Dittrich, Computing chemical organizations in biological networks, *Bioinf.* **24** (2008) 1611–1618.

[33] S. Artmann, Basic semiosis as code-based control, *Biosem. Neth.* **2** (2009) 31–38.

[34] E. Duesbury, J. D. Holliday, P. Willett, Maximum common subgraph isomorphism algorithms, *MATCH Commun. Math. Comput. Chem.* **77** (2017) 213–232.