# Mining conserved neighborhood patterns in metabolic and genomic contexts

Alexandra Zaharia

université PARIS-SACLAY

UNIVERSITÉ PARIS SUD
Comprendre le monde, construire l'avenir

NNT : 2018SACLS275

Thèse de doctorat

# Identification des motifs de voisinage conservés dans des contextes métaboliques et génomiques

Thèse de doctorat de l'Université Paris-Saclay
préparée à l'Université Paris-Sud et réalisée dans le cadre du LRI

Ecole doctorale n°580 Sciences et technologies de l'information et de la communication (STIC)
Spécialité de doctorat : Informatique

Thèse présentée et soutenue à Orsay, le 28 septembre 2018, par

**ALEXANDRA ZAHARIA**

Composition du Jury :

| | |
|---|---|
| Alessandra Carbone<br>PR, Sorbonne Université (LCQB) | Présidente |
| Marie Beurton-Aimar<br>MC HDR, Université de Bordeaux (LaBRI) | Rapporteure |
| Jérémie Bourdon<br>PR, Université de Nantes (LS2N) | Rapporteur |
| Jean-Loup Faulon<br>DR, INRA (MICALIS) | Examinateur |
| Christine Froidevaux<br>PR, Université Paris-Sud (LRI) | Directrice de thèse |
| Alain Denise<br>PR, Université Paris-Sud (LRI) | Co-encadrant |



PhD thesis
Université Paris-Saclay / Université Paris-Sud

Doctoral School n°580
Sciences et Technologies de l'Information et de la Communication (STIC)

PhD specialty: Computer Science

# Mining conserved neighborhood patterns in metabolic and genomic contexts

Defended at LRI, Orsay, on September 28, 2018

by

Alexandra Zaharia

Jury:

| | | |
|---|---|---|
| Alessandra Carbone | PR<br>Sorbonne Université (LCQB) | President |
| Marie Beurton-Aimar | MC HDR<br>Université de Bordeaux (LaBRI) | Referee |
| Jérémie Bourdon | PR<br>Université de Nantes (LS2N) | Referee |
| Jean-Loup Faulon | DR<br>INRA (MICALIS) | Examiner |
| Christine Froidevaux | PR<br>Université Paris-Sud (LRI) | PhD advisor |
| Alain Denise | PR<br>Université Paris-Sud (LRI) | PhD co-advisor |

*This thesis is dedicated to the memory of my grandparents, Ana and Andrei Mălăescu.*

*My grandmother passed away during the last year of my PhD. She raised me and helped shape me into the person I am today. She was always encouraging of my inquisitiveness and always disapproving of my impulsiveness. I've learned a lot lately!*

*My grandfather passed away long before I got into bioinformatics. He gave me my first PC and taught me my first programming language, FoxPro. When we weren't coding or quarreling about open-source, we were DYI-ing everything from furniture to electronics.*

*I still miss them both dearly.*

# *Acknowledgements*

I would like to express my profound gratitude to my PhD advisors at LRI, Christine Froidevaux and Alain Denise. Without their continuous guidance and mentoring, this thesis would not have been possible. Apart from brainstorming research directions together, they showed me how to think like a scientist and how to present my work. Christine's attention to detail and Alain's top-down scientific view made for an always challenging duo of supervisors from whom I have learned so much over these past three years.

I am very grateful to Bernard Labedan for the helpful discussions and suggestions on my research, throughout my whole stay at LRI. His biological expertise was paramount for this thesis.

I thank my referees, Marie Beurton-Aimar and Jérémie Bourdon, for their insightful comments on the first draft of this manuscript. I also thank the two other members of my committee, Alessandra Carbone and Jean-Loup Faulon, for having accepted to be examiners at my defense.

I am especially grateful to Florence d'Alché-Buc and Jean-Loup Faulon for having taken the time to review my early work. At MICALIS, Jean-Loup and Thomas Duigou gave me invaluable suggestions on possible developments, while also letting me get a glimpse into their research. This encounter was decisive for my growing interest in retrosynthesis for biotechnological applications.

I would like to thank Guillaume Fertin and Irena Rusu at LINA, who contributed useful feedback on HNET.

I thank all my colleagues in the Bioinfo team at LRI for their support and friendship. My office ex-mate, Adrien Rougny, and my current office mate, Cécile Moulin, were especially subjected to my bouts of enthusiasm or desperation; I thank them for being there for me (not that they had a choice...) and for their understanding.

A special mention to my fellow doctoral students in the VALS team at LRI (Hai, Mattias, Robin, Albin, Julien), whose humor during coffee breaks was contagious.

Thanks to Jorge Cham (whom I do not know personally) for creating the PhD comic strip.

Balthazar, who favored feather-chasing play sessions at 3 a.m. during the writing of my manuscript, certainly did *not* contribute to the timely progress of the draft, but his antics did help me keep my humor.

I thank my family and friends for their encouragement. Last but not least, a heartfelt thanks to Romuald, whose unwavering friendship and encouragement during these past two years helped me to pull through.

# Contents

# Introduction

Systems biology is an ever-expanding field where new developments in molecular biology techniques yield richer biological data (a few examples being genomic, transcriptomic, proteomic, interactomic, and metabolomic data) or produce such data faster. This deluge of biological information creates the need for increasingly specialized and efficient processing and analysis algorithms. A strong emphasis is placed on integrative approaches capable of incorporating data from heterogeneous sources in order to advance our understanding when considering the wholeness of cellular systems.

In this context, numerous approaches for heterogeneous biological networks are modeled as graph problems. Broadly speaking, such approaches are directed either at the integration of heterogeneous networks, or at motif extraction. From an algorithmic point of view, the work presented in this thesis fits within the latter category. Its main goal is to explore the relationship between metabolism and the genome.

Genomic data and chemical reactions embody the dual aspect of metabolism [Muto *et al.*, 2013] that allows exploring the links between genome evolution and chemical evolution of enzyme-catalyzed reactions [Kanehisa, 2013]. It is well established that neighboring reactions corresponding to neighboring genes underline an evolutionary advantage in keeping the genes involved in succeeding reactions in close genomic proximity [Alves *et al.*, 2002; Rison *et al.*, 2002]. Finding almost identical sequences of reactions being catalyzed by products of neighboring genes in various species suggests that such sequences are made up of key enzymatic steps, best performed when their encoding genes are adjacent and co-transcribed. This type of metabolic and genomic organization strongly suggests the various species have

been under strong evolutionary pressure to optimize the expression of enzyme-coding genes involved in successive reactions [Zaslaver *et al.*, 2006; Wells *et al.*, 2016].

This thesis focuses on the identification of conserved metabolic and genomic patterns. Roughly speaking, *metabolic and genomic patterns* can be defined as corresponding neighborhoods of reactions and genes for a given species. More precisely, metabolic and genomic patterns may be described as sequences of reactions having certain features, such that the reactions are catalyzed by products of neighboring genes. *Conserved metabolic and genomic patterns* represent similar neighborhoods of reactions and genes for a variety of species. Interspecies comparisons based on conserved patterns may help to shed light onto the evolution of conserved metabolic and genomic neighborhoods.

The identification of metabolic and genomic patterns requires extraction of relevant information from metabolic and genomic contexts as well as its simultaneous integrated analysis. Knowledge extraction from biological networks has been the topic of numerous research efforts, mainly concentrated on 'omics' data integration, network alignment, and network mining. The problem addressed in this thesis involves knowledge extraction from heterogeneous (as opposed to homogeneous) biological networks, containing different types of information that describe distinct aspects of related processes for the same biological entity. The source of biological data used in this thesis is the well-known KEGG (Kyoto Encyclopedia of Genes and Genomes) knowledge base.

The main contributions of this thesis are the following:

- We propose algorithms for trail finding, corresponding to the identification of metabolic and genomic patterns. Trails of reactions are sequences in which reactions (but not the links between them) may be repeated in order to account for cycles, which are ubiquitous in metabolism.
- We describe two trail grouping methods, corresponding to the detection of conserved metabolic and genomic patterns.
- We introduce `CoMetGeNe`, a fully automated open-source pipeline for the detection of metabolic and genomic patterns and their conservation.
- We conduct an investigation into the metabolic and genomic organization of a bacterial data set.
- We provide preliminary results on the integration of a chemical similarity criterion into the trail grouping methodology, leading to the identification of metabolic and genomic patterns that perform the same types of chemical transformations.

- In addition, we report existing consistency issues in the KEGG knowledge base and outline approaches for their systematic discovery.

This document is organized as follows:

- Chapters I and II provide the necessary biological and graph-theoretical background, respectively, for the work presented in this thesis. Chapter II also reviews state of the art methods for the comparison of heterogeneous biological networks.
- Chapter III represents the transition between the background chapters and the pure contribution chapters. On the one hand, it introduces the KEGG knowledge base, an essential resource for the metabolic and genomic data used in the applications presented herein. On the other hand, this chapter contributes to the detection of consistency issues in KEGG, relevant to the bioinformatics community relying on this knowledge base.
- Chapters IV, V, and VI refer to pattern detection. Chapter IV proposes a method for trail finding, which translates to the identification of metabolic and genomic patterns. Chapter V describes how such patterns may be analyzed and grouped in order to reveal conserved metabolic and genomic contexts across multiple species. Chapter VI gives a brief overview of `CoMetGeNe`, an open-source pipeline that we designed to detect metabolic and genomic patterns, as well as their conservation.
- Chapter VII shows how the trail finding and trail grouping methodologies reveal conserved metabolic and genomic patterns in practice. A data set of 50 bacterial species chosen to represent major phyla of the bacterial tree of life is investigated using `CoMetGeNe`. The patterns thus discovered are then described and analyzed, revealing interesting aspects of the relationship between metabolic architecture and genome structure.
- Chapter VIII discusses how the definition of metabolic and genomic patterns may account for the similarity of the chemical transformations performed by reactions in `CoMetGeNe` trails. Two approaches for evaluating the chemical similarity of `CoMetGeNe` trails are proposed and illustrated using preliminary results. The developments in this chapter are not yet integrated into the `CoMetGeNe` pipeline.

This document concludes with a short chapter that summarizes our main contributions and outlines future research prospects.

# I

# Biological context

# 1 Introduction

This chapter introduces the biological context of this thesis.

In the first part of the chapter, metabolism is presented as a system where different actors interact. Enzymes play an important role in this system, as they enable chemical reactions to take place. In addition, enzymes are frequently the target of diverse mechanisms that allow metabolism to evolve.

The rest of the chapter describes the interconnection between the metabolism of an organism and its genome. Genes encode proteins and special proteins called enzymes make metabolism possible, but how can we tell what the purpose of any given gene is? The answer to this question is explored throughout the sections on homology and functional annotation.

# 2 Metabolism

*metabolism*

Several definitions for metabolism exist. **Metabolism** can be seen as the set of life-sustaining biochemical processes that allow a cell to develop, reproduce, and interact with its environment. The term "metabolism" comes from Greek and signifies "change" or "transformation".

*primary metabolism*

*secondary metabolism*

With respect to organism survival, metabolism is divided into primary metabolism and secondary metabolism. *Primary metabolism* consists in metabolic transformations that are essential for survival and is usually well conserved across the tree of life. *Secondary metabolism* consists in metabolic transformations that are not essential for survival under normal conditions. Antibiotics and toxins are examples of end products of secondary metabolism.

## 2.1 Main metabolic actors

*metabolic network*

A **metabolic network** can be defined as the complete set of metabolic transformations that determine the properties of a cell. From a computer science point of view, a metabolic network can be conceptualized as a collection of objects and their respective relations. Metabolic networks can be modeled intuitively with respect to objects and the relations between them through a UML diagram (see Figure I.1). This section describes the main actors involved in metabolism. An introduction to the analysis of metabolic networks can be found in Lacroix *et al.* [2008].

*compound (metabolite)*

**Chemical compounds** or **metabolites** are small molecules that are intermediate products of metabolism. They can be synthesized and/or degraded within an organism, and they can be imported and/or exported. The main atom types in the

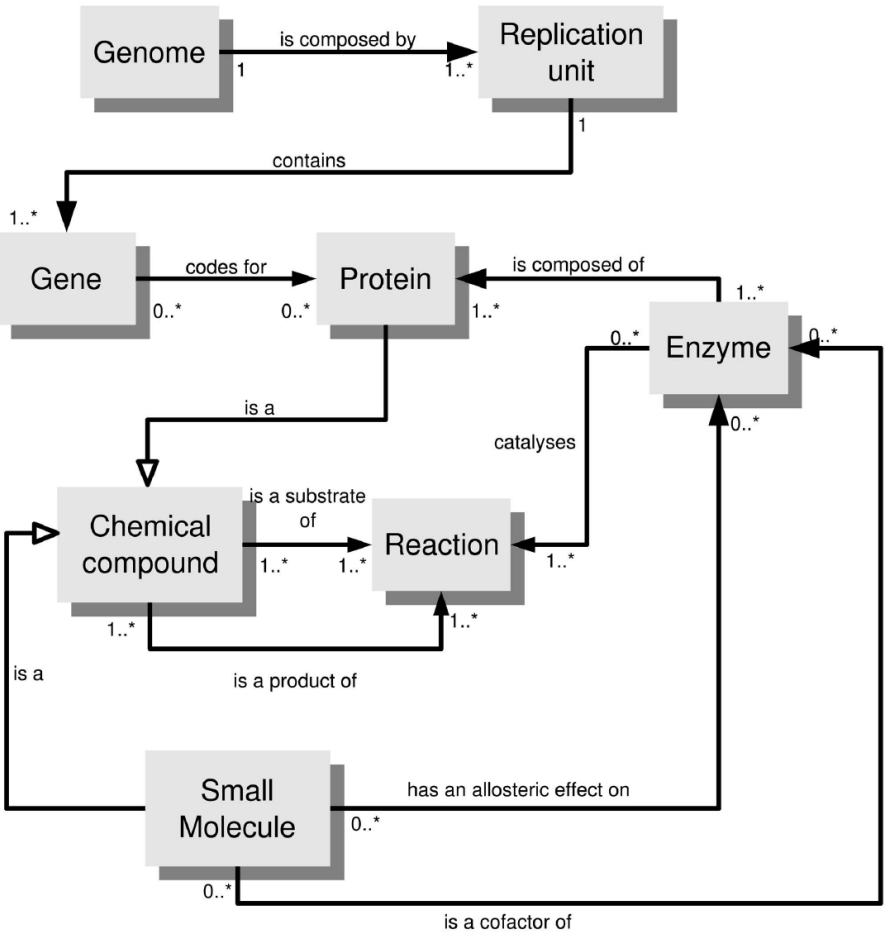



**Figure I.1 Simplified UML view of the various objects involved in a metabolic network.** A replication unit refers to a region of nucleic acid that replicates from a single origin of replication. In general, a replication unit corresponds to the entire chromosome in prokaryotes, whereas in eukaryotes there are multiple replication units per chromosome. Arrows with black arrowheads represent relations between objects. Symbols on either side of such arrows represent relation cardinality. 0..* means zero or more. For example, in the relation "codes for" between "gene" and "protein", a gene can encode one or several proteins (in the case of alternative splicing), or no protein if the gene does not code for a protein. A protein can be produced by one or more genes, or supplied by the environment (hence the 0 in the cardinality for "gene"). Reproduced with permission from Lacroix *et al.* [2008] © 2008 IEEE.

composition of metabolites are carbon (C), oxygen (O), hydrogen (H), nitrogen (N), sulfur (S), and phosphorus (P). Some compounds may contain metal atoms, such as iron (Fe), magnesium (Mg), or zinc (Zn).

*reaction*

**Biochemical reactions** consist in the transformation of a set of one or more compounds called *substrates* into a set of one or more compounds called *products*. Reactions that can occur in either direction are called *reversible*, while reactions that can

take place in only one direction are *irreversible*. The vast majority of biochemical reactions are not *spontaneous* and require *catalysis* in order to perform the chemical transformation of metabolites in a reasonable amount of time (on the time scale of cell metabolism). While reaction catalysts are generally proteins or protein complexes (see below), certain RNA molecules called *ribozymes* can also serve as catalysts [Lilley, 2003]. For example, ribosomes are ribozymes [Cech, 2000], catalyzing peptide bond formation (for linking amino acids together) through a peptidyl transferase activity.

*enzyme*

**Enzymes** are proteins or protein complexes encoded by one or several genes. A substrate binds a special region of an enzyme called *active site*, where it undergoes the biochemical reaction that the enzyme *catalyzes*. The relation between enzymes and reactions is not one-to-one, as a single reaction may be catalyzed by several enzymes, and a single enzyme may catalyze one or several reactions. Enzymes without strict specificity are called *promiscuous enzymes* and they can, for example, accept several similar substrates [Nobeli *et al.*, 2009].

*cofactor*

**Cofactors** are small molecules that bind to certain enzymes, with the effect of increasing or decreasing their activity. When binding, a cofactor generally induces a conformational change in the binded enzyme [Kern and Zuiderweg, 2003]. Cofactors with positive effects on enzyme activity are called *allosteric activators*, whereas those with negative effects are *allosteric inhibitors*. The term "allostery" signifies that the binding site for a cofactor is physically distinct from the enzyme's active site.

## 2.2 Enzymatic activities

With the first identification of an enzyme in 1833 and the introduction of the term "enzyme" in 1876, the early days of biochemistry were plagued by a systematic confusion in the naming of enzymes. It was not until the 1950s that enzymologists started addressing this problem [Tipton and Boyce, 2000].

Today, the only official enzyme nomenclature is the one established by the Nomenclature Committee of the International Union of Biochemistry and Molecular Biology (IUBMB). In this nomenclature system, Enzyme Commission numbers

*EC number*

(EC numbers) are assigned to enzymes, based on the chemical reactions that the enzymes catalyze [Webb, 1992]. It is important to note that an EC number is not equivalent to an enzyme, nor to a reaction. EC numbers simply describe enzyme-catalyzed reactions, which means that two distinct reactions can have the same EC number if they involve chemically similar transformations.

An EC number is formed by four numbers separated by periods. The first three numbers designate the enzyme class, subclass, and sub-subclass, respectively. The

fourth is a serial number uniquely identifying the activity among other activities of the same class, subclass, and sub-subclass. The serial number conveys details on substrate specificity and cofactors. For example, EC 2.7.2.4 represents an aspartate kinase (Figure I.2).

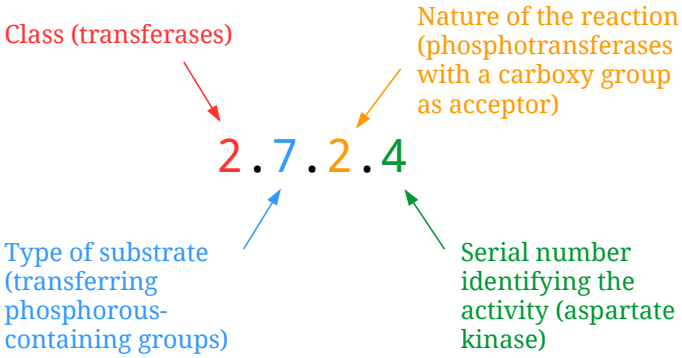



**Figure I.2** Anatomy of an EC number

EC numbers are currently organized in six major classes (Figure I.3): EC 1 (oxidoreductases), EC 2 (transferases), EC 3 (hydrolases), EC 4 (lyases), EC 5 (isomerases), and EC 6 (ligases).

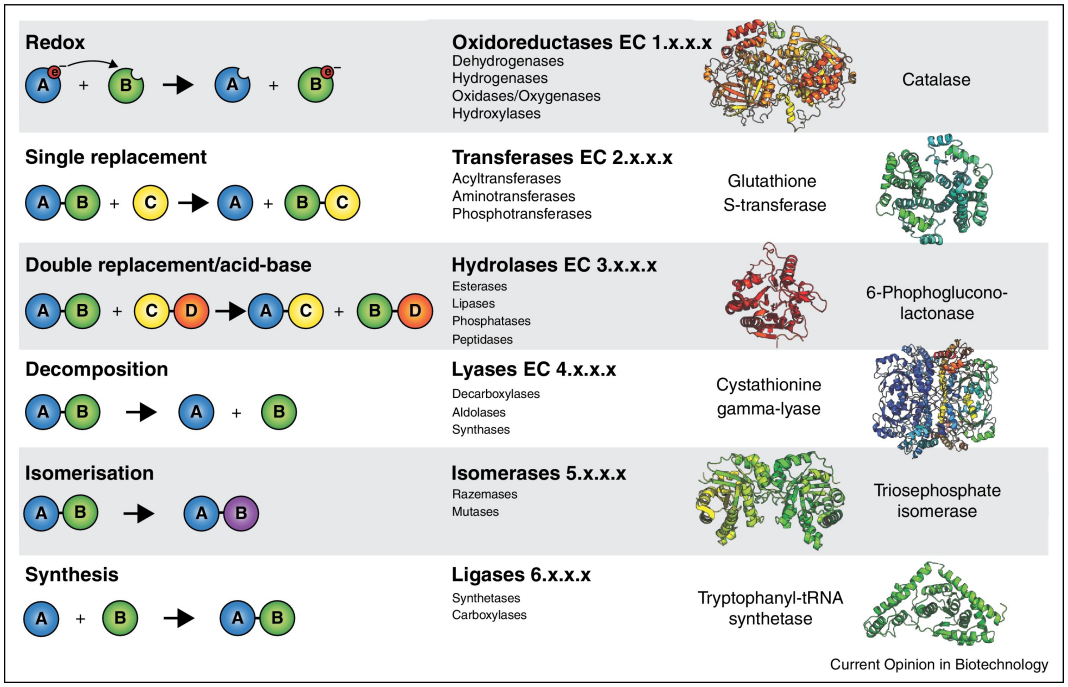



**Figure I.3 Illustration of the six major enzymatic classes.** Reproduced from Keller *et al.* [2015] (licensed under CC BY 4.0).

Despite their widespread use, EC numbers have important drawbacks when

used for functional annotation (see section 3.3) and inference of metabolic pathways (see section 2.3). Some of these drawbacks are discussed below. In addition, the reader may consult McDonald and Tipton [2014] for a survey of advances and challenges in enzyme classification.

Since the EC number classification has been in use since the 1950s when submission criteria were not as strict as nowadays[1], several older entries describe enzymatic activities with no associated experimental evidence for the reactions being catalyzed, while for other entries no candidate sequence has yet been determined. In 2004, Karp [2004] called for a collective effort combining bioinformatics and experimental approaches in order to assign at least one amino acid sequence to every biochemically characterized enzymatic activity. At the time of the study, it was found that 38% of EC numbers were lacking sequence data. The following year, Lespinet and Labedan [2005] coined the expression "orphan enzymes" to describe enzymatic activities without associated amino acid sequences. A decade after Karp's call to initiative, Sorokina *et al.* [2014] reported that the percentage of orphan enzymes had decreased from 38% to 22%.

EC numbers are not appropriate for the inference of metabolic pathways from complete genomes because of inherent differences between various types of metabolism. For example, in the KEGG knowledge base (see Chapter III), the metabolic network is a reference map representing the set of all known metabolic variations for all sequenced organisms. This is precisely the reason for which KEGG is used as a source of metabolic information throughout this thesis. Unlike MetaCyc, KEGG has a top-down approach to representing metabolism, with less pathway maps encompassing more reactions than pathways in MetaCyc, on average [Altman *et al.*, 2013]. In KEGG, the general metabolism of a given species is a subset of the reference metabolic map. Using solely EC numbers to infer metabolic pathways for a newly sequenced organism is error-prone, as half of the reactions in KEGG pathway maps did not have an associated EC number in 2013 [Kanehisa, 2013] and not all EC numbers have associated sequence data (see above).

Promiscuous enzymes can catalyze more than a single reaction, in which case they might be assigned different EC numbers. The use of a rigid hierarchy is impractical in this case, as it does not allow easy identification of an enzyme based on the EC number that is assigned to it. For example, Bastard *et al.* [2014] described a strategy for exploring the functional diversity of a previously uncharacterized enzyme family. They found that 20% of the enzymes in this family displayed im-

[1] Enzyme nomenclature (2018): recommendations of the Nomenclature Committee of the International Union of Biochemistry and Molecular Biology on the nomenclature and classification of enzymes by the reactions they catalyze: http://www.sbcs.qmul.ac.uk/iubmb/enzyme

portant substrate promiscuity, acting on at least five different substrates.

Finally, non-enzymatic reactions, for which the EC hierarchy is not applicable, are known to play important roles in metabolic networks [Keller *et al.*, 2015].

## 2.3 Metabolic pathway

While a metabolic network can be seen as the complete set of metabolic transformations, a metabolic pathway is a subjective interpretation of the manner in which a metabolic network is partitioned. Although there is no consensus on a formal definition for metabolic pathways [Faust *et al.*, 2011; Lacroix *et al.*, 2008], several definitions that have been proposed over time are summarized below (from Faust *et al.* [2011]).

**Source–target definition** A metabolic pathway is a "sequence of enzyme-catalyzed reactions by which a living organism transforms an initial source compound into a final target compound" [Nelson and Cox, 2005]. This definition does not take into account branched pathways, nor spontaneous reactions.

**Topological definition** "A *metabolic network* is a directed reaction graph with substrates as vertices and directed, labeled edges denoting reactions between substrates catalyzed by enzymes (labels). A *metabolic pathway* is a special case of a metabolic network with distinct start and end points, initial and terminal vertices, respectively, and a unique path between them" [Forst and Schulten, 1999]. In other words, a metabolic pathway is seen as a subnetwork of a metabolic network. This definition accounts for branched pathways and spontaneous reactions, but it does not distinguish among biochemically valid and invalid pathways.

**Atom flow definition** A metabolic pathway "from metabolite $X$ to $Y$ is defined as a sequence of biochemical reactions through which at least one carbon atom in $X$ reaches $Y$. Only carbon atoms are considered [...]. A metabolite $Y$ is called reachable from $X$ if there is a pathway from $X$ to $Y$" [Arita, 2004]. This definition does not take into account transformations on molecules without carbon atoms.

**Functional definition** A metabolic pathway is "a set of interconnected reactions that can be activated coordinately to ensure a particular cellular function" [Faust *et al.*, 2011]. As noted by its authors, this definition cannot be effectively exploited unless an exact definition of cellular function has been provided.

*metabolic pathway map*

In the context of this thesis, the term *metabolic pathway* is used interchangeably with the concept of *metabolic pathway map* from KEGG (see Chapter III). Since KEGG provides a global, top-down view of metabolism, a metabolic pathway map may represent a collection of metabolic pathways (according to the above definitions), grouped around a central metabolic process.

## 2.4 Representation of metabolic networks

Metabolism can be modeled through either graph representations or constraint-based approaches [Lacroix *et al.*, 2008]. The latter approach is not used throughout this thesis. Briefly, constraint-based modeling consists in representing the metabolic network as a stoichiometric matrix. The distribution of mass fluxes is analyzed under steady state and thermodynamic constraints.

When modeling metabolism by means of graphs, it is natural to consider the directed case (see definition II.1), as reactions can be reversible (see section 2.1). However, applications exist where undirected graphs have been employed. The most commonly used graph models are listed below.

*compound network*

**Compound network** The *compound network* is a directed graph in which vertices are compounds. An arc from a compound $A$ to a compound $B$ represents the fact that $A$ and $B$ are the substrate and product, respectively, of a metabolic reaction.

*reaction network*

**Reaction network** The *reaction network* is a directed graph in which vertices are reactions. An arc from a reaction $r_i$ to a reaction $r_j$ signifies that $r_i$ produces a compound that is also a substrate for the reaction $r_j$. This is the modeling that we choose for the method proposed in Chapter IV.

**Bipartite graph** If both compounds and reactions need to be accounted for, a bipartite graph may be used[2]. A bipartite graph has two types of vertices and each of its arcs has endpoints in both types of vertices (see also definition II.12). The two types of vertices in the bipartite representation are reactions and compounds, respectively.

A problem that arises in practice is obtaining overly-connected graphs because of hub compounds (i.e., highly connected compounds such as water, ATP, or cofactors). Three possible strategies for dealing with this problem are discussed below.

[2] Equivalently, a hypergraph may be used instead of a bipartite graph.

**Removing ubiquitous compounds** Frequent metabolites can be removed from the graph representation by deleting the corresponding compound vertices. This approach has the disadvantage of removing legit metabolites in certain situations. For example, ATP is a frequent compound that would get removed using this strategy. However, ATP is also a main compound in the reaction leading to its synthesis from ADP.

**Distinguishing main from side compounds** This strategy keeps all the vertices but removes arcs from the graph model, for example between a compound $C$ and a reaction $r$ if $C$ is a side compound or a cofactor involved in $r$ in the case of a bipartite graph representation. The KEGG knowledge base (see Chapter III) used to contain this information in the RPAIR database [Kotera *et al.*, 2004; Faust *et al.*, 2009; Muto *et al.*, 2013], allowing to distinguish main and side compounds, or cofactors, for example. The distinction was made based on the atom flow within and between molecules, but the assignment was made manually. Due to the effort required to manually create and maintain these assignments, the RPAIR database was discontinued in 2016[3].

**Relabeling vertices** If compounds and reactions have unique labels, reactions and the compounds participating in reactions can be clearly distinguished from repeated occurrences of the same reactions and/or compounds. Consequently, this strategy avoids the topological hub problem and is the strategy used in this thesis (see sections III.2.3 and IV.2 for more details).

## 2.5 Metabolic evolution

This section summarizes the most important views on the origins of metabolic pathways and briefly describes the forces at play in metabolic evolution.

### 2.5.1 Main hypotheses

Several hypotheses have been proposed to explain the origin and evolution of metabolic pathways, three of which are presented here (see also Fani and Fondi [2009]). Rather than viewing these hypotheses in opposition, they can be seen as models of putative metabolic evolution. Any given hypothesis cannot realistically explain every particular detail of current-day metabolic pathways. Instead, the

[3]The announcement on KEGG RPAIR being discontinued dates from May 18, 2016 and can be found at https://www.kegg.jp/kegg/docs/announce.html?past. KEGG RPAIR was effectively discontinued on October 1, 2016.

following hypotheses propose complementary models that may be used jointly to gain insight into metabolic evolution. It has already been suggested that a network perspective may serve to reconcile the different hypotheses on the origin and evolution of metabolism [Díaz-Mejía *et al.*, 2007].

*retrograde hypothesis*

**Retrograde hypothesis** The *retrograde hypothesis* [Horowitz, 1945], also known as the *stepwise hypothesis*, proposes that sequential enzymes may have been acquired in reverse order with respect to their order in extant (current-day) pathways. The underlying assumption is that preexisting chemical compounds were already available in the "primordial soup", and that they could be synthesized via chemical reactions when depleted.

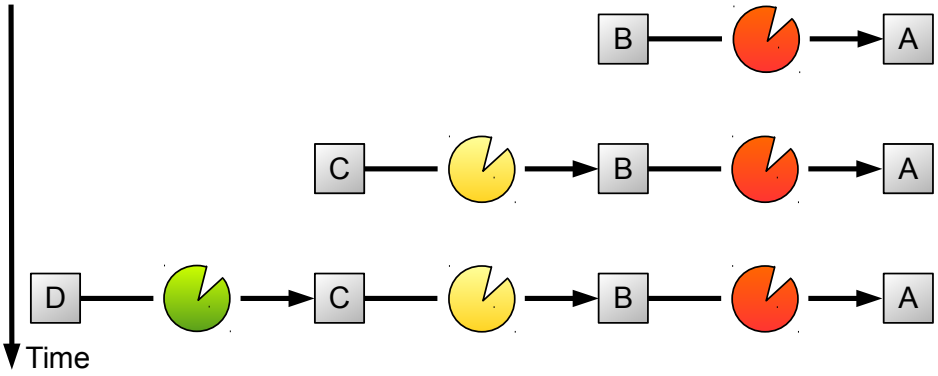



**Figure I.4 Schematic representation of the retrograde hypothesis.** Enzymes are colored circles. *A*, *B*, *C*, and *D* are chemical compounds.

Figure I.4 illustrates the retrograde hypothesis. If compound *A* was essential for survival, its depletion would put the organism under selective pressure. Organisms capable of producing the red enzyme to obtain *A* from a preexisting precursor *B* would survive. Then, as *B* became depleted, in some organisms the gene encoding the red enzyme might get duplicated. In turn, some of the copies might be mutated versions that would encode the yellow enzymes instead of the red one, becoming thus capable of synthesizing compound *B* from its precursor *C*. Finally, the same process would take place when *C* became depleted, with some organisms being able to obtain it from compound *D* using the green enzyme.

In support of the retrograde hypothesis, Alves *et al.* [2002] have found that homologous enzymes (i.e., with a common origin; see section 3.2) are less than three steps away from each other with a significantly higher frequency than non homologous enzymes.

**Patchwork hypothesis** The *patchwork hypothesis* [Jensen, 1976] proposes that metabolic pathways may have evolved by recruiting enzymes with low specificity, i.e. multifunctional enzymes that can react with a broad range of substrates. Following gene duplication events, recruited enzymes would increase their substrate specificity, becoming more effective at catalyzing a narrower range of substrates. The ancestral enzymes could thus be recruited for other pathways.

*patchwork hypothesis*

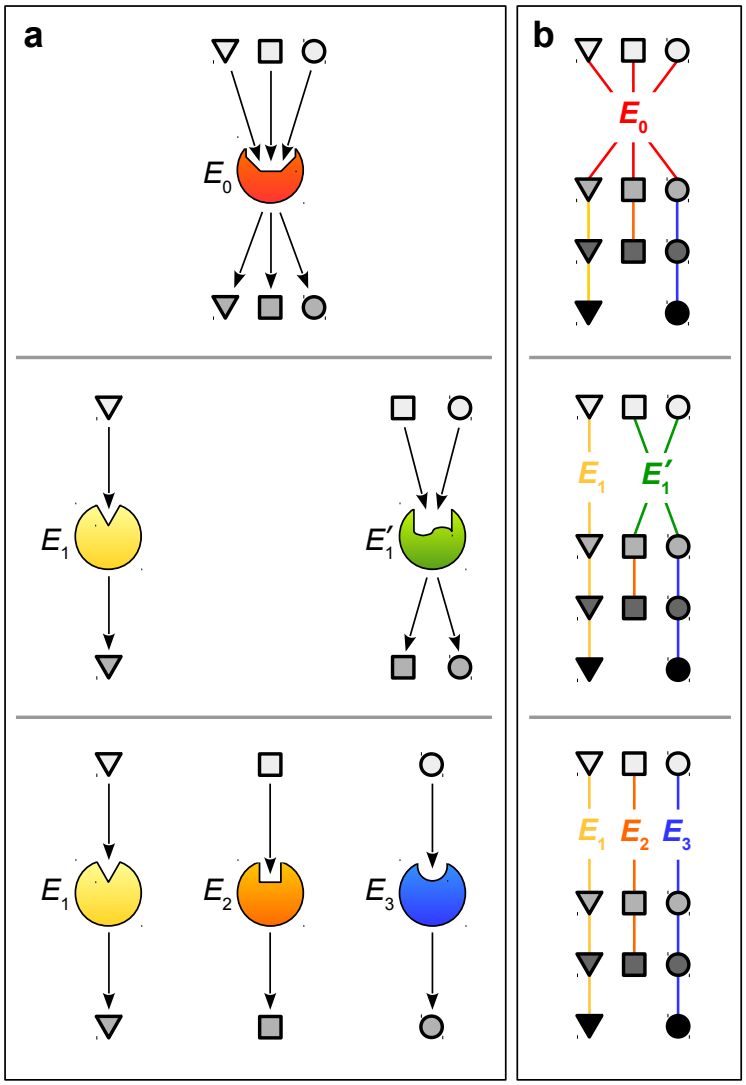



**Figure I.5 Schematic representation of the patchwork hypothesis.** Inspired by Fani and Fondi [2009]. In both panels, the two horizontal lines signify gene duplication events followed by evolutionary divergence. **(a)** Progressive specialization of a multifunctional enzyme $E_0$. **(b)** Hypothetical structure of the metabolic pathways involving the enzymes from panel (a).

Figure I.5 illustrates the patchwork hypothesis. Panel (a) shows that, follow-

ing a gene duplication event, the multifunctional enzyme $E_0$ narrows down its substrate specificity from three to two substrates ($E'_1$). A specialized enzyme is also formed ($E_1$). Following a second gene duplication event, $E'_1$ gives rise to two specialized enzymes, $E_2$ and $E_3$. In parallel, panel (b) shows that how the three metabolic pathways involving these enzymes might have formed. Whenever more specialized enzymes replace multifunctional enzymes (for instance, when $E_0$ is replaced by $E_1$ and $E'_1$), the more primitive multifunctional enzyme can be recruited in other pathways.

In support of the patchwork hypothesis, Teichmann *et al.* [2001] have found that homologous enzymes in *Escherichia coli* belong to distinct pathways twice as often than they appear in the same pathway.

*semienzymatic origin*

**Semienzymatic origin of metabolic pathways** In order to explain the early origins of metabolic pathways, Lazcano and Miller [1999] have proposed that non-specific enzymes might have operated slight changes in the chemical environment of the "primordial soup", thus enabling certain reactions to occur spontaneously.

### 2.5.2 Mechanisms

**Gene duplication** Gene duplication is a powerful mechanism for evolution in general. Duplicated genes may conserve their function if the functional redundancy is beneficial. They can also specialize further (a process known as *subfunctionalization*), acquire novel functions (*neofunctionalization*), or become inactivated (*pseudogenization*) [Zhang, 2003]. Both the retrograde and the patchwork hypotheses assume gene duplication events (see section 2.5.1 above).

It was proposed that, in plants, the presence of duplicated genes can be selected or counter-selected, according to the required level of genetic variation [Kliebenstein, 2008]. According to this model, gene duplication (hence variation) is beneficial in secondary metabolism, but detrimental in primary metabolism. Other studies have focused on the role that gene duplication plays in yeast [Kuepfer *et al.*, 2005] and bacterial [Marri *et al.*, 2006] metabolism. In addition, an *in silico* network perspective approach was used to analyze the impact of gene duplication on the evolution of metabolism in *E. coli* [Díaz-Mejía *et al.*, 2007].

**Pathway duplication** Conceptually, pathway duplication complements the patchwork hypothesis (see section 2.5.1 above), which suggests that new pathways may have emerged through the reuse of existing pathways and the recruitment of new enzymes. For example, Gerlee *et al.* [2009] studied the phenomenon of pathway

duplication in computer-simulated organisms as well as in the yeast metabolic network and suggested that pathway duplication is an important mechanism in the emergence of novel metabolic function.

**Horizontal gene transfer** *Horizontal gene transfer* is the process by which genetic material gets transferred between different species (as opposed to "vertical" transmission from parent to offspring, which takes place within the same species). Horizontal gene transfer occurs frequently in bacteria, being the main mechanism for acquiring antibiotic resistance. It has been shown that horizontal gene transfer is equally involved in the evolution of prokaryotic metabolic pathways [Pál *et al.*, 2005; Iwasaki and Takagi, 2009].

**Enzyme promiscuity** The concept of *enzyme promiscuity*, referring to the ability of an enzyme to catalyze a side reaction in addition to its main reaction, is closely linked to the patchwork hypothesis on the origins and evolution of metabolic pathways (see section 2.5.1 above) [Nobeli *et al.*, 2009; Khersonsky and Tawfik, 2010]. While usually taking place in an enzyme's active site, promiscuous enzymatic activity where the active site is not involved has also been reported [Taglieber *et al.*, 2007].

Depending on how promiscuous enzymatic activities are classified, several levels or types of promiscuity can be defined. Without going into details, a clear distinction can be made between catalytic promiscuity (when referring to an enzyme that performs different chemical transformations) and substrate promiscuity (when referring to an enzyme that uses similar substrates to perform a given reaction). Braakman and Smith [2012] observed that substrate promiscuity is the main type of promiscuity leading to the diversification of protein families.

A method for the quantification of enzyme promiscuity [Carbonell and Faulon, 2010] based on molecular signatures [Faulon *et al.*, 2003] (see also Chapter VIII) has led to the finding that promiscuous enzymes are mainly involved in amino acid and lipid metabolism [Carbonell *et al.*, 2011a]. The authors advanced the explanation that reactions from amino acid and lipid metabolism, being probably the earliest form of biochemical reactions, were and still are performed by multifunctional enzymes.

**Cofactors** It has been proposed that cofactors play an important role in shaping metabolic evolution [Braakman and Smith, 2012]. As topological hubs in metabolic pathways, cofactors are found in key positions to exert control over metabolism. The authors equally note that cofactors, occupying an intermediary position

between small molecules and more complex metabolites, may have provided the support for transitioning from mineral-based to organic chemistry.

**Ribozymes** In support of the RNA world hypothesis, it has been proposed that ribozymes may have played a critical part in the origins of life through their double role as support for genetic information and in their crude ability to serve as chemical catalysts [Lilley, 2003; Cech, 2012].

**Chemical selection** Meléndez-Hevia *et al.* [2008] propose that *chemical selection* preceded natural selection in protocellular entities. In effect, as natural selection requires genetic information, metabolism, and membranes in order to operate, the authors hypothesized that, in the absence of these prerequisites, the emergence of life was governed by a chemical pre-enzymatic selection process that relied on stoichiometry and thermodynamic strategies.

## 3 Relationship between metabolism and the genome

This section aims to clarify the link between metabolism (see section 2) and the genome. First, the molecular mechanisms leading to proteins synthesis are presented. Next, the notion of homology is introduced. Homology, a central concept in phylogenetics and evolutionary biology, is an important indicator to the function of biological sequences. Finally, the section concludes with an overview of the approaches used to predict protein function, collectively referred to as functional annotation.

### 3.1 From genes to proteins

The flow of genetic information is explained by the so-called central dogma of molecular biology, formulated in 1958 by Francis Crick and revised in 1970 [Crick, 1970]. An updated view of the central dogma is given in Figure I.6. The rest of this section briefly describes the entities involved in the central dogma, namely DNA, RNA, and proteins, as well as the molecular processes connecting DNA to RNA (transcription) and RNA to proteins (translation). Transcription and translation enable *gene expression*, the mechanism through which genetic information is used to obtain a functional gene product. DNA replication, as well as the two infrequent types of information flow (reverse transcription and RNA replication) are beyond the scope of this introduction. An in-depth explanation on these topics is provided in Alberts *et al.* [2008]. Note that this section only presents a simplified version of

*gene expression*

transcription and translation in prokaryotes. The same processes in eukaryotes are much more complex [Alberts *et al.*, 2008].

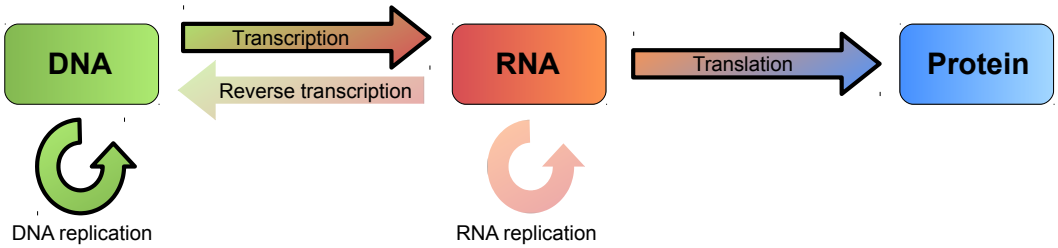



**Figure I.6 The central dogma of molecular biology.** Arrows with, respectively without contours represent general (frequent), respectively special (infrequent) types of information flow.

The support for genetic information is typically a macromolecule called *deoxyribonucleic acid* (DNA). The other nucleic acid is *ribonucleic acid* (RNA), but it only serves as support for genetic information in RNA-based viruses. A consensus has still not been reached regarding the inclusion of viruses in the tree of life [Koonin and Starokadomskyy, 2016; Moreira and López-García, 2009].

**DNA** DNA (deoxyribonucleic acid) is a double-stranded helix made up of four types of building blocks called nucleotides. Each nucleotide is composed of a five-carbon sugar molecule called deoxyribose, a phosphate group, and a nitrogenous base. The four types of nitrogenous bases are adenine (A), cytosine (C), guanine (G), and thymine (T). The nitrogenous bases on opposite DNA strands form *base pairs* by establishing hydrogen bonds between A and T, and between C and G (see Figure I.7).

*base pair*

In its double-stranded form, DNA is the main constituent of **chromosomes**. Most prokaryotes (bacteria and archaea) have a single circular chromosome containing most of the organism's genetic information. Certain organisms (with bacteria being the most frequent) may also exhibit *plasmids*, meaning small DNA molecules that are found outside of the chromosome and that can replicate independently. The term **genome** designates the physical support for genetic information in a given organism. Prokaryotic genomes, for example, typically consist of a chromosome and sometimes one or several plasmids. A **gene** is a portion of a DNA molecule that can be transcribed into RNA (see below).

**RNA** RNA (ribonucleic acid) is similar to DNA in its composition. It is made up of ribonucleotides in which the five-carbon sugar molecule is ribose (instead of deoxyribose). The thymine nitrogenous base in DNA is replaced with uracil (U)

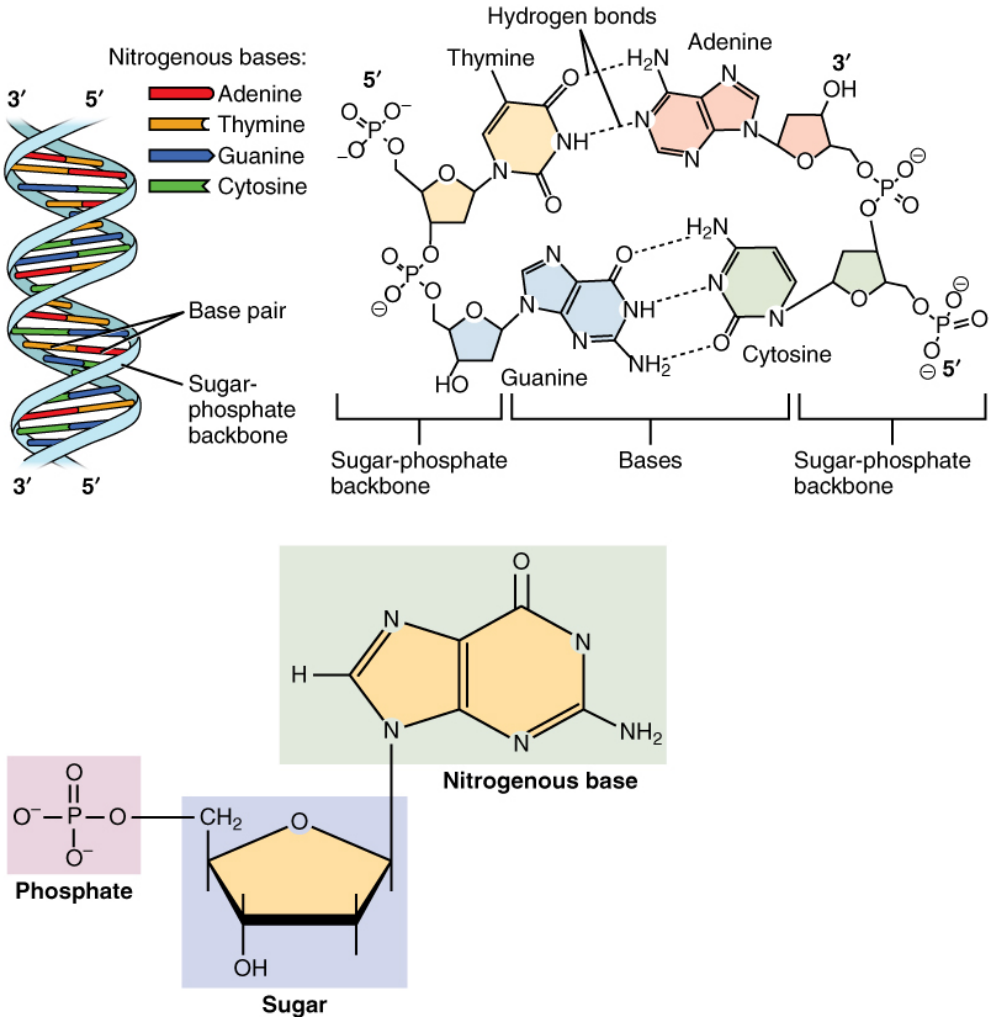



**Figure I.7 DNA structure.** Bottom panel: a nucleotide, with its sugar, phosphate, and nitrogenous base. Upper panel (left): structure of the DNA double helix, with its sugar and phosphate backbone and hydrogen bonds between base pairs. Upper panel (right): double hydrogen bonds form between thymine and adenine nitrogenous bases on opposite DNA strands, and triple hydrogen bonds form between guanine and cytosine. Source: OpenStax [CC BY 4.0], via Wikimedia Commons.

in RNA. There are several types of RNA, the three most common being *messenger RNA* (mRNA), *transfer RNA* (tRNA) and *ribosomal RNA* (rRNA).

DNA molecules are very large in comparison to RNA. The human genome, for instance, has over 3 billion base pairs organized in 23 pairs of chromosomes. Because of its size, DNA needs to be packed into a highly compact form within a cell. RNA, however, is a short molecule with respect to DNA. Often existing as a single strand, RNA is thus free to fold onto itself and adopt three-dimensional

conformations that serve various functional roles.

**Proteins** Proteins are macromolecules formed by one or several amino acid chains. They are synthesized from mRNA during translation (see below). A special class of proteins are enzymes, which catalyze biochemical reactions as explained in section 2.1.

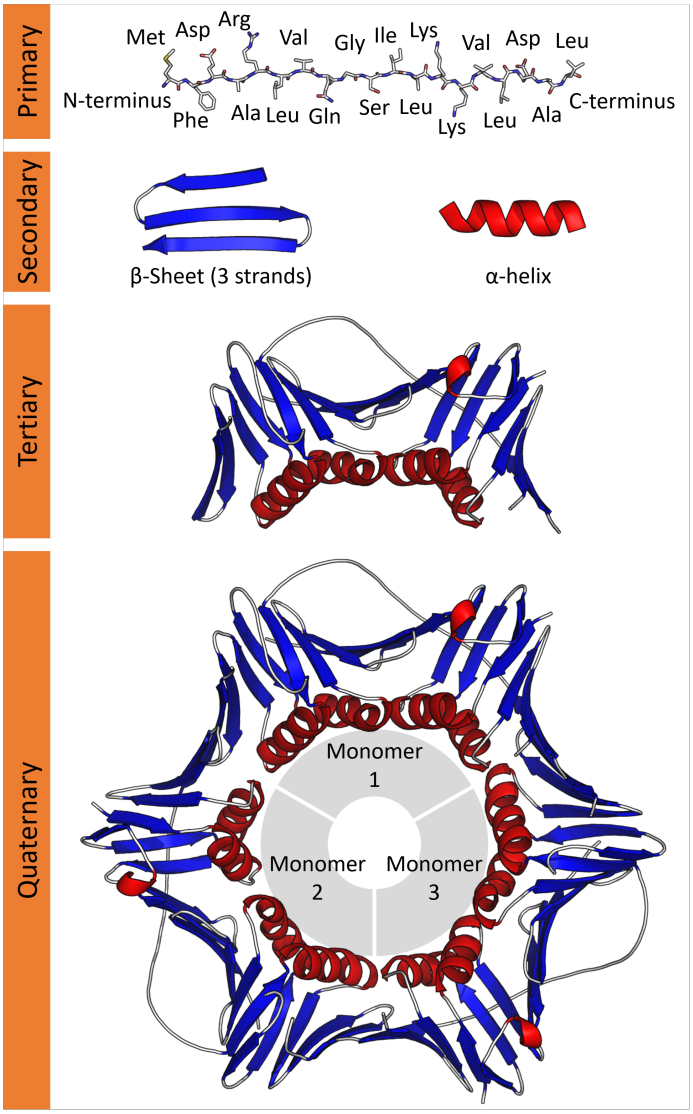



**Figure I.8 The four levels of protein structure.** Source: Thomas Shafee [CC BY 4.0], via Wikimedia Commons.

Similarly to RNAs, proteins adopt three-dimensional conformations that are linked to their function. There are four levels of protein structure (Figure I.8): the *primary structure* is the amino acid sequence; the *secondary structure* refers to local

segments of the protein, the most common being $\alpha$-helices and $\beta$-sheets; the *tertiary structure* is the three-dimensional form of the protein; finally, the *quaternary structure* refers to proteins that are made up of several monomers (subunits).

**Transcription** Transcription is the process through which a portion of DNA gets copied into a RNA molecule by an enzyme complex named RNA polymerase. This task is rendered possible by the identification of the start and end points of a gene, named promoter and terminator, respectively. In prokaryotes, several genes can share the same promoter and terminator. In such cases, they are transcribed jointly into a single RNA molecule. It is therefore more appropriate to use the term **transcription unit** to refer to what gets transcribed.

If genes in a transcription unit encode proteins, they are transcribed into mRNA. Such genes are referred to as protein-coding genes. Alternatively, genes may not code for proteins, but for non-coding RNAs, including tRNA and rRNA (see translation below).

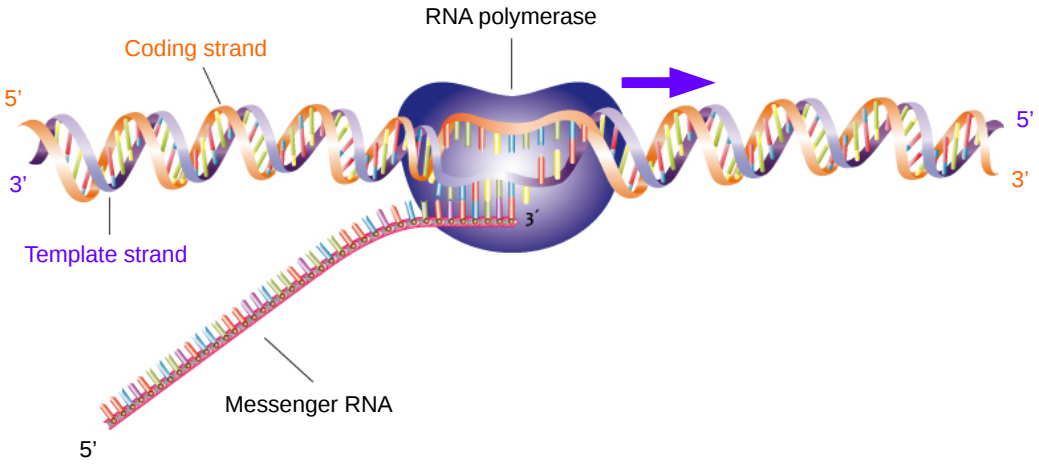



**Figure I.9 Simplified view of bacterial transcription.** The direction of the RNA polymerase is indicated by the purple arrow. The template strand (in purple) must be traversed in the $3' \to 5'$ direction. The messenger RNA, complementary to the template strand and an exact copy of the coding strand (with thymine replaced by uracil), is thus elongated in the $5' \to 3'$ direction. Adapted from: Genomics Education Programme [CC BY 2.0], via Wikimedia Commons.

The two DNA strands are complementary and *antiparallel*, which is represented through the notations $5' \to 3'$ and $3' \to 5'$ (with respect to the DNA sugar-phosphate backbone), as shown in Figure I.7 (upper left). In order for a transcription unit to get transcribed, the RNA polymerase traverses the strand containing the transcription unit (named *coding strand*) in the $5' \to 3'$ direction. It then synthesizes the RNA

transcript using the other strand, named *template strand*, as a template. The mRNA transcript is complementary to the template strand and an exact copy of the coding strand in which thymine is replaced with uracil. Figure I.9 shows a simplified model of transcription in bacteria.

**Translation** Translation is the process through which the genetic information in mRNA molecules is decoded in order to synthesize proteins. Translation takes place in small complexes named *ribosomes*, made up of proteins and rRNA. The ribosome moves along the mRNA molecule and, for every group of three ribonucleotides, adds an amino acid according to the genetic code (see Shu [2017]) to the growing polypeptide chain. Amino acids are provided to ribosomes by bounded tRNA molecules.

## 3.2 Homology of biological sequences

Homology is an important, albeit often misused, concept with important evolutionary and functional connotations [Koonin, 2005].

*homology*

Two genes are said to be **homologous** if they are derived from a common ancestral gene sequence. If two genes that evolved separately have a similar function, they are called *analogous*. An example of analogy is the case of non-homologous isofunctional enzymes [Omelchenko *et al.*, 2010], which catalyze the same reaction without sharing a common evolutionary history. In general, homologous sequences present high sequence similarity. Figure I.10 shows an evolutionary scenario of five homologous genes.

*analogy*

*orthology*

Two genes are said to be **orthologous** if they are derived from a common ancestral sequence through a *speciation* event. In Figure I.10, gene colors represent species. Genes in any gene pair involving the species in green are orthologous, due to the speciation event $S_1$: $(x_1, z_1)$, $(y_1, z_1)$, $(x_2, z_1)$, $(y_2, z_1)$. With respect to the speciation event $S_2$, the pairs $(x_1, y_1)$ and $(x_2, y_2)$ are orthologs.

*paralogy*

Two genes are said to be **paralogous** if they are derived from a common ancestral sequence following a gene *duplication* event. In Figure I.10, all pairs genes issued from the duplication event indicated by a red star are paralogous: $(x_1, x_2)$, $(y_1, y_2)$, $(x_1, y_2)$, and $(x_2, y_1)$. Two special cases of orthology can be defined with respect to a speciation event of reference:

- Paralogous genes are called **in-paralogs** if they were duplicated *after* the speciation event of reference. For example, genes $(x_1, y_2)$ in Figure I.10 are in-paralogs with respect to the speciation event $S_1$.

*in-paralogy*

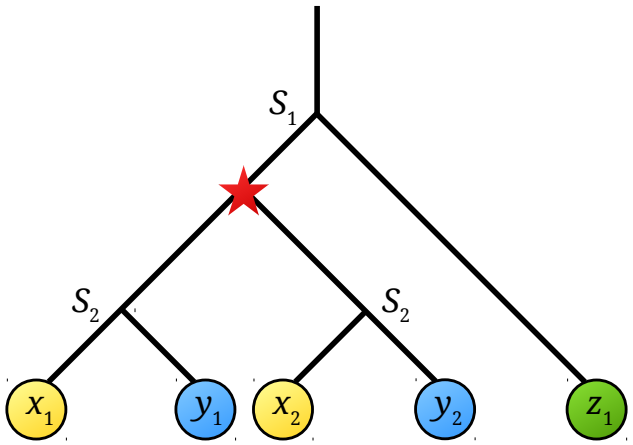



**Figure I.10 Evolutionary scenario of a gene family.** In this scenario, two speciation events ($S_1$ and $S_2$) and a duplication event (red star) took place, leading to the gene family $\{x_1, y_1, x_2, y_2, z_1\}$. Each gene color represents a species. All genes are homologous to each other. Pairs $(x_1, y_1)$, $(x_2, y_2)$, $(x_i, z_1)$ and $(y_i, z_1)$ are orthologous (with $i \in \{1, 2\}$). Pairs $(x_1, x_2)$, $(y_1, y_2)$, $(x_1, y_2)$, and $(x_2, y_1)$ are paralogous. Inspired by Altenhoff and Dessimoz [2012].

*out-paralogy*

- Paralogous genes are called **out-paralogs** if they were duplicated *before* the speciation event of reference. For example, genes $(x_1, y_2)$ in Figure I.10 are out-paralogs with respect to the speciation event $S_2$.

## 3.3 Functional annotation

As of the writing of this thesis, nearly 200,000 genomes are available at NCBI in various degrees of assembly completion, of which almost 30,000 are completely assembled[4]. In spite of these impressive figures, sequencing and assembling genomes are only the first steps to deciphering them. The next step is to understand what makes living things tick, and in order to do so knowledge of gene function is required. The process of associating functions to biological sequences is known as *functional annotation*. It relies on several strategies that this section attempts to summarize. It should be noted that the various methods used to predict protein function can and should be used in combination, as this leads to an overall increased prediction accuracy. Furthermore, integrating several approaches and types of data may enable the discovery of novel protein functions [Lobb and Doxey, 2016].

*functional annotation*

A large-scale community initiative named CAFA (Critical Assessment of Functional Annotation) evaluated existing functional annotation methods in 2010–2011

[4]The full list of NCBI genomes is available at the following address: https://www.ncbi.nlm.nih.gov/genome/browse. All degrees of assembly completion are shown by default. Filters in the web interface allow to display only completely assembled genomes.

[Radivojac *et al.*, 2013] and 2013–2014 [Jiang *et al.*, 2016], showing significant improvements from the first to the second assessment. An alternative and more epistemological view is given by Galperin and Koonin [2010] on the current understanding of genomes based on the current state of functional characterization.

### 3.3.1 Sequence similarity

As explained in section 3.2, homology shows whether two biological sequences share a common evolutionary history. It is widely assumed that sequence similarity correlates with functional similarity. Thus, historically, functional annotation has been performed by sequence comparison and transfer of functional characterization if a pre-determined threshold of similarity is reached. The most commonly used programs are FASTA [Pearson and Lipman, 1988] and BLAST (with its version PSI-BLAST for finding distantly-related protein sequences) [Altschul *et al.*, 1997].

However, this strategy has the drawback of overpredicting protein function [Schnoes *et al.*, 2009; Moreno-Hagelsieb and Hudy-Yuffa, 2014]. It is therefore necessary to complement sequence similarity search with other methods.

### 3.3.2 Orthology

There is proof to support the idea that orthologous sequences share similar functions [Rogozin *et al.*, 2014]. This phenomenon should however be considered a statistical trend rather than a rule or an implication [Gabaldón and Koonin, 2013].

For a newly sequenced genome of a species $A$, identifying genes of $A$ that are orthologs of functionally characterized genes in another organism $B$ allows the transfer of functional annotation for these orthologous genes from $B$ to $A$. For example, suppose $A$ and $B$ are the species in yellow and blue, respectively, in Figure I.10. Then the functional annotation of genes $y_1$ and $y_2$ in $B$ can be transferred to the genes $x_1$ and $x_2$, respectively, in species $A$.

Different orthology prediction methods are compared in Kristensen *et al.* [2011]; Altenhoff and Dessimoz [2012].

### 3.3.3 Genomic context

Genomic context can provide important clues to functional associations [Moreno-Hagelsieb and Santoyo, 2015], especially in prokaryotes.

A particularly useful resource for the exploration of genomic context is the STRING database [Szklarczyk *et al.*, 2014] (used in Chapter VII), as it integrates

not only protein–protein interaction data, but also genomic context and domain information, along with relevant literature references.

*synteny blocks*

**Synteny** *Synteny* represents the physical co-localization of genes on the same chromosome for a given species. In genomics, *conserved synteny blocks* can lead to evolutionary insights by indicating that particular genome regions in several species originate from an ancestral genomic region. For example, conserved synteny blocks have been used to reconstruct the architecture of the ancestral chromosome in the yeast genus *Lachancea* [Vakirlis *et al.*, 2016].

Conserved synteny blocks are also interesting for functional predictions [Overbeek *et al.*, 1999; Rogozin *et al.*, 2002]. Several detection and visualization tools for synteny detection have been proposed [Gehrmann and Reinders, 2015; Drillon *et al.*, 2014; Lemoine *et al.*, 2008; Sinha and Meller, 2007]. In addition, graph-theoretical approaches based on the extraction of maximal common connected components allow for gaps [Boyer *et al.*, 2005], may process multiple input genomes [Deniélou *et al.*, 2009], and allow for partial correspondence between the aligned networks [Deniélou *et al.*, 2011].

*operon*

**Operons** An *operon* is a group of co-localized genes that are co-regulated and co-transcribed (see also transcription in section 3.1). Genes in operons tend to be related to a given biological function [Overbeek *et al.*, 1999]. It was estimated that approximately 60% of genes in *E. coli* are organized in operons [Moreno-Hagelsieb, 2015]. In general, operons are well conserved among species, although some genes may be rearranged, gained/lost, or duplicated [Ream *et al.*, 2015].

**Gene fusion events** A gene fusion event is a physical coupling of genes that are likely to be functionally coupled as well [Enright and Ouzounis, 2001; Yanai *et al.*, 2001]. This type of information should therefore be considered when inferring protein function. An example of functional association through gene fusion will be presented in section VII.4.

**Phylogenetic profiles** *Phylogenetic profiles* describe the presence or absence of a gene or protein family across a given group of organisms [Pellegrini *et al.*, 1999]. Although primarily used to reveal coevolution, phylogenetic profiles are also useful to infer functional associations [Wu *et al.*, 2003] as well as to to predict protein–protein interactions [Sun *et al.*, 2005] and candidate genes for orphan enzymes [Chen and Vitkup, 2006] (see section 2.2). The quality of prediction, however, is dependent on the choice of genomes [Jothi *et al.*, 2007].

### 3.3.4 Protein structure

Since the end of the last century, the scientific community expected that, as protein structures became available, they would help explain protein function, especially in the absence of functionally characterized homologues [Hegyi and Gerstein, 1999].

Methods of functional prediction from protein sequence and structure have been reviewed over the years [Watson *et al.*, 2005; Lee *et al.*, 2007; Mills *et al.*, 2015; Lobb and Doxey, 2016]. As noted in the introduction, sequence-based and structure-based approaches are not mutually exclusive and are often used jointly.

*protein domain*

A protein sequence (its primary structure) may contain *domain* information. Functional and structural domains are portions of a protein's secondary and tertiary structure that are highly conserved and can therefore be found almost unaltered in several species. Several online resources including Pfam [Finn *et al.*, 2015] and InterPro [Mitchell *et al.*, 2014] may be used to detect protein domains in an input amino acid sequence.

Other approaches involve the analysis of local characteristics in the secondary and tertiary structure of a protein by comparison against large collections of known motifs. Examples include elements of secondary structure ($\alpha$-helices and $\beta$-sheets), active sites, or ligand binding sites.

In addition, docking approaches have been used successfully to predict protein function. For example, Zhao *et al.* [2013] performed metabolite docking against multiple proteins in a metabolic pathway, which allowed them to predict the function of a previously uncharacterized enzyme. The integration of genomic context information enabled to equally determine the role of the enzyme in the pathway. The functional prediction was subsequently validated experimentally.

### 3.3.5 Rule-based systems

Another avenue that can be explored in order to predict the function of biological sequences is to use rule-based systems.

One possibility is to map elements of a functional hierarchy (such as MIPS FunCat [Ruepp *et al.*, 2004]) or ontology (such as Gene Ontology [GO Consortium, 2001]) onto target sequences (i.e. the sequence to annotate), as in Azé *et al.* [2008] and Rance *et al.* [2009].

In addition, there exist description schemes capable of enriching functional annotation in Gene Ontology. For example, BioΨ is a four-level biological process description scheme, partly overlapping with Gene Ontology descriptions [Mazière *et al.*, 2004]. Although not primarily aimed at functional prediction, BioΨ was used

to annotate the tricarboxylic acid cycle, revealing information that was not readily available for automated analysis tasks [Mazière *et al.*, 2007].

Another possibility is to automate the reasoning process of a human annotator by integrating knowledge on the target sequence from several sources, including BLAST results (see section 3.3.1 above), orthology information, and known domains [Xavier *et al.*, 2015].

Methods aimed at assisting biocurators by evaluating annotation consistency have also been proposed. One such example is GROOLS [Mercier *et al.*, 2018], an expert system using paraconsistent logic.

## 4 Concluding remarks

The biological context of the thesis was detailed throughout this chapter.

Metabolism was described from a functional perspective and the main mechanisms of metabolic evolution were examined in a brief survey. What stands out from this survey is that metabolism evolves with the emergence of new function. Protein function, however, cannot be properly understood without exploring its connection to the genome. The aim of all genome sequencing projects is to be able to ultimately decipher the blueprint of living beings. This means going from knowing what makes up a genome to understanding how the different cogwheels work together to give rise to biological function.

In the absence of experimental characterizations of proteins, the scientific community makes great efforts at predicting their function. Many of these efforts are based on sequence data, in which the role of homology is cornerstone. Different approaches focus on exploiting the relation between protein structure and function, while others integrate genomic context information or employ rule-based systems. These approaches are collectively used for functional annotation.

The next chapter reviews current graph-theoretical approaches used in systems biology, focusing in particular on graph-theoretical approaches for heterogeneous biological networks.

# II

# State of the art

# 1 Introduction

Numerous real-life systems and processes can be modeled as graphs. One may think of public transportation and social networks, for instance. In fact, graphs and graph algorithms are ubiquitous. For example, Google search results are obtained from a knowledge graph [Sullivan, 2012], computer-aided navigation (GPS) finds shortest routes, and pathfinding in video games (consisting in finding an optimal route while avoiding obstacles) relies on graph theory algorithms [Algfoor *et al.*, 2015].

Biological networks are equally represented as graphs. This chapter describes topological, alignment, and mining approaches used in systems biology. Since the graph-theoretical context of this thesis is that of heterogeneous networks, existing methods for aligning and mining heterogeneous networks are discussed.

# 2 Elements of graph theory

This section presents basic graph theory notions that are used throughout this thesis. For more details on this topic, the reader may consult Balakrishnan and Ranganathan [2012], West [2001], and Bang-Jensen and Gutin [2008].

*graph* *edge* *arc*

**Definition II.1.** A *graph* is an ordered pair $G = (V, E)$, where $V$ is the vertex set of $G$ and $E \subseteq V \times V$ is the set of edges of $G$. An *undirected graph* is a graph in which edges have no orientation, whereas in a *directed graph* edges have orientation and are called *arcs* for convenience.

*vertex set*

*Remark.* The notation $V(G)$ is often used to denote the vertex set of a graph $G$.

*digraph*

*Remark.* A directed graph is often called a *digraph*.

*Example.* In Figure II.1, $G$ is an undirected graph (panel a) and $D$ is a directed graph (panel b). Both $G$ and $D$ have the same vertex set $V(G) = V(D) = \{1, 2, 3, 4, 5, 6\}$.

*induced subgraph*

**Definition II.2.** Let $G = (V, E)$ be a graph and $X \subseteq V$ a subset of vertices of $G$. The *subgraph of $G$ induced by $X$*, denoted $G[X]$, is the graph $G' = (X, E')$ where $E' = \{(u, v) \mid u, v \in X \text{ and } (u, v) \in E\}$.

*Example.* If the subset of vertices of $G$ is $X = \{1, 2, 3, 5\}$ for the graph $G$ in Figure II.1a, then $G[X]$ is the graph $G'$ in Figure II.1c.

*connected graph*

**Definition II.3.** An undirected graph is *connected* if every vertex is reachable from any other vertex.

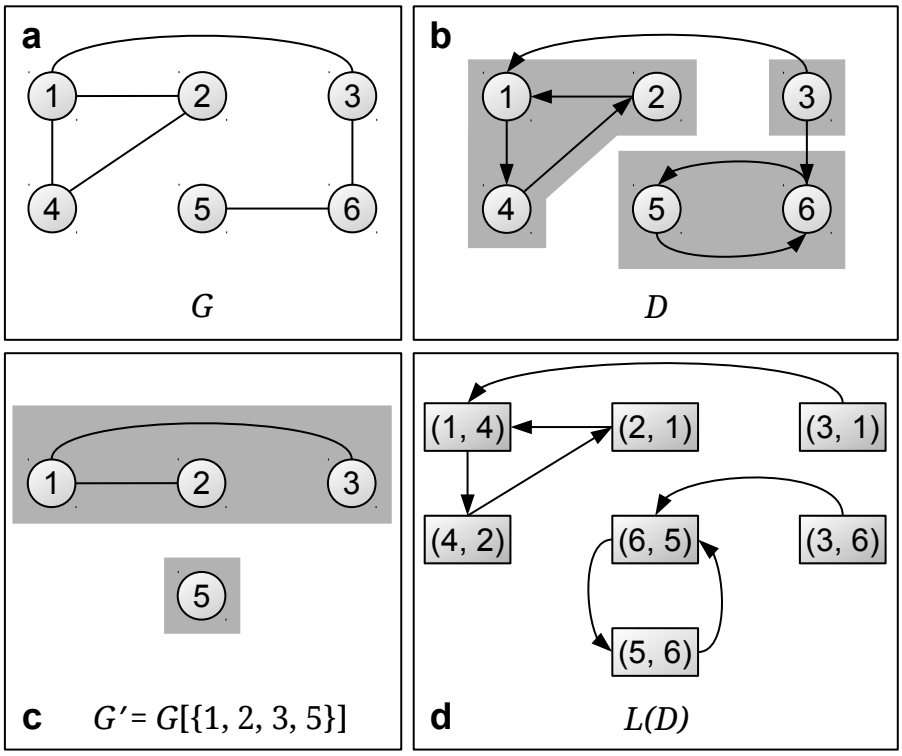



**Figure II.1 Examples of graphs. (a)** $G$ is an undirected graph. **(b)** $D$ is a directed graph. Strongly connected components in $D$ are highlighted in gray (see definition II.7). **(c)** $G'$ is the subgraph of $G$ induced by the vertex subset $X = \{1, 2, 3, 5\}$. Connected components in $G'$ are highlighted in gray (see definition II.4). **(d)** $L(D)$ is the line graph of the directed graph $D$ in (b).

*Example.* Graph $G$ in Figure II.1a is connected, as each of its vertices is reachable from any vertex in $G$. The graph $G'$ (Figure II.1c), however, is not connected, because vertex 5 is unreachable.

**Definition II.4.** A *connected component* of an undirected graph $G$ is a maximal connected subgraph of $G$.

*connected component*

*Remark.* In other words, a connected component is connected internally, but has no edges linking it to remaining vertices in $G$ [Dasgupta *et al.*, 2006].

*Remark.* If $G$ only has one connected component, then $G$ is *connected*.

*Example.* The unique connected component of graph $G$ in Figure II.1a is the graph itself, as $G$ is connected. The graph $G'$ in Figure II.1c has two connected components, with vertex sets $\{1, 2, 3\}$ and $\{5\}$, respectively (highlighted in gray in the figure).

**Definition II.5.** Given an undirected graph $G = (V, E)$, a *clique* is a subset $V' \subseteq V$ of vertices such that $(u, v) \in E$ for any $u, v \in V'$.

*clique*

*Example.* The maximum clique in the graph $G$ in Figure II.1a is $\{1, 2, 4\}$. The other cliques correspond to the remaining edges in $G$: $\{1, 3\}$, $\{3, 6\}$, and $\{5, 6\}$.

*strongly connected digraph*

**Definition II.6.** A directed graph is *strongly connected* if any two vertices in $D$ are mutually reachable.

*Example.* The graph $D$ in Figure II.1b is not strongly connected because it is not possible to reach vertex 3 from vertex 2, for instance.

*strongly connected component*

**Definition II.7.** A *strongly connected component* of a directed graph $D$ is a maximal strongly connected subgraph of $D$.

*Example.* The graph $D$ in Figure II.1b has three strongly connected components (highlighted in gray), with vertex sets $\{1, 2, 4\}$, $\{3\}$, and $\{5, 6\}$, respectively.

*line graph*

**Definition II.8.** Let $D = (V, A)$ be a directed graph. The *line graph* of $D$ is the directed graph $L(D) = (A, A')$ in which:

(i) The set of vertices of $L(D)$, $A$, represents the arcs of graph $D$, and
(ii) The set of arcs of $L(D)$, $A'$, represents adjacencies between arcs of $D$, i.e. $(x, y) \in A'$ if and only if $x = (r, s)$ and $y = (s, t)$, with $r, s, t \in A$.

*Example.* The graph $L(D)$ in Figure II.1d is the line graph of the directed graph $D$ in Figure II.1b.

*walk*

**Definition II.9** (Balakrishnan and Ranganathan [2012])**.** A *walk* in a directed graph $D$ is an ordered sequence of vertices $(v_1, v_2, \ldots, v_k)$ such that $v_i \in V(D)$ for every $i \in \{1, \ldots, k\}$ and $(v_i, v_{i+1})$ is an arc of $D$ for every $i \in \{1, \ldots, k-1\}$.

*Remark.* An equivalent definition can be formulated for an undirected graph by replacing arcs with edges.

*Example.* The sequence $(3, 1, 4, 2, 1, 4)$ is a walk in the directed graph $D$ in Figure II.1b. Vertices 1 and 4 are repeated. The arc $(1, 4)$ is also repeated.

*path*

**Definition II.10** (Balakrishnan and Ranganathan [2012])**.** A *path* is a walk without repeated vertices.

*Example.* The sequence $(3, 1, 4, 2)$ is a path in the directed graph $D$ in Figure II.1b. No vertex is repeated.

*trail*

**Definition II.11** (Balakrishnan and Ranganathan [2012])**.** A *trail* is a walk without repeated arcs (or edges, in the undirected case).

*Example.* The sequence $(3, 1, 4, 2, 1)$ is a trail in the directed graph $D$ in Figure II.1b. Vertex 1 is repeated. No arcs are repeated.

**Definition II.12.** A graph $G = (V, E)$ is called *bipartite* if its vertex set can be divided in two disjoint subsets $X$ and $Y$ such that every edge (or arc in the directed case) in $G$ has one of its endpoints in $X$ and the other in $Y$. *bipartite graph*

*Example.* The directed graph in Figure II.2 is bipartite. It has two types of vertices (green squares and gray circles) and none of its arcs have both endpoints in the same type of vertex.

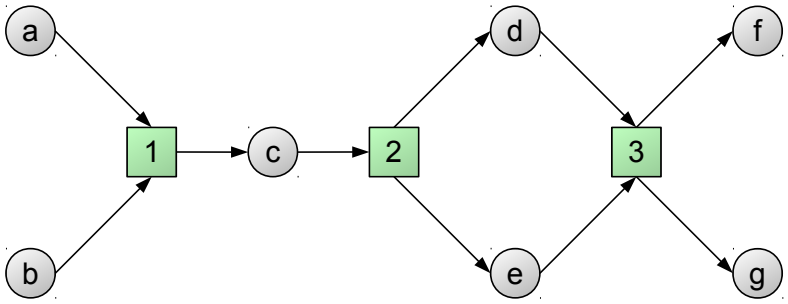



**Figure II.2 Bipartite digraph.** The two disjoint sets of vertices are $\{1, 2, 3\}$ (green squares) and $\{a, b, c, d, e, f, g\}$ (gray circles).

## 3 Graph-theoretical approaches in systems biology

From molecular through cellular to ecosystem level, biological systems can be modeled using graphs. At the molecular level, metabolic networks (see Chapter I), genomic context, gene regulatory networks, protein–protein interaction networks, and so on, are all examples of biological systems that may be modeled through graph-based representations. Since graph-theoretical approaches are ubiquitous in biology, this section only concentrates on the usage of such approaches in *systems* biology. They have been grouped into three broad categories, namely graph topology, graph alignment, and graph mining approaches.

Additionally, two other classes of graph theory problems with applications in systems biology are being actively researched: graph coloring and covering problems. Graph coloring is often a subproblem of graph mining [Lacroix, 2007; Sikora, 2011] and graph decomposition [Mohamed-Babou, 2012] problems. Covering problems have a variety of applications, one example being the use of vertex covering to find maximal cliques, where a maximal clique is a clique (see definition II.5) in a graph $G$ such that no other vertex of $G$ can be added to it [Chesler and Langston, 2007]. These two classes of problems, however, will not be discussed further.

## 3.1 Network topology

Network topology is an extremely vast field. An in-depth review on the structure (and dynamics, which are not discussed here) of complex networks is available in Boccaletti *et al.* [2006].

### 3.1.1 Common measures

The most basic level of network analysis involves its topological study. The numerous topological measures that can be computed allow to effectively capture the organization of a biological network, providing insights into its function and stability. Instead of summarizing frequently employed topological measures (see Steuer and López [2008]; Koschützki [2008]; Pavlopoulos *et al.* [2011] for this purpose), this section only lists a few very common ones with the aim of showing how they can lead to new findings in systems biology.

Note that, although the following definitions are given for the undirected case, equivalents for directed graphs exist.

*characteristic path length*

**Definition II.13.** Given a connected undirected graph $G = (V, E)$, the *characteristic path length* $L$ of $G$ is the average shortest distance between any two vertices in $G$. Formally, let $d_{ij}$ be the shortest distance between vertices $i$ and $j$ in $G$. If $i$ and $j$ belong to different connected components, then $d_{ij} = \infty$. Then:

$$L = \sum_{i,j \in V} \frac{d_{ij}}{n(n-1)}$$

*Example.* The graph $G$ in Figure II.1a has a characteristic path length of 2.07.

*degree*

**Definition II.14.** In an undirected graph, the *degree* of a vertex $u$ is the number of edges incident to it and is denoted as $deg(u)$.

*Example.* For graph $G$ in Figure II.1a, $deg(1) = 3$, $deg(2) = deg(3) = deg(4) = deg(6) = 2$, and $deg(5) = 1$.

*clustering coefficient*

**Definition II.15** (Rubinov and Sporns [2010])**.** The *clustering coefficient* $C$ of an undirected graph $G = (V, E)$ is a measure of the degree to which vertices in $G$ tend to cluster together. Formally, let $a_{ij}$ denote whether an edge between vertices $i$ and $j$ in $G$ exists, with $a_{ij} = 1$ if $(i, j) \in E$ and $a_{ij} = 0$ if $(i, j) \notin E$. Then:

$$C = \frac{1}{|V|} \sum_{i \in V} \frac{\sum_{j,k \in V} a_{ij} a_{ik} a_{jk}}{deg(i)(deg(i) - 1)},$$

where $deg(i)$ is the degree of vertex $i$ (see definition II.14).

**Definition II.16** (Asensio *et al.* [2017]). Given an undirected graph $G = (V, E)$, the *betweenness centrality* of a vertex $k$ in $G$ is the value

*betweenness centrality*

$$c_b(k) = \sum_{\substack{i,j \in V \\ i \neq j \neq k}} \frac{\sigma_{ij}(k)}{\sigma_{ij}},$$

where $\sigma_{ij}$ is the number of shortest paths between vertices $i$ and $j$, and $\sigma_{ij}(k)$ is the number of shortest paths between $i$ and $j$ passing through $k$.

*Example.* In Figure II.3, vertex $k$ has high betweenness centrality as all shortest paths between vertices in the blue and red subnetworks pass through $k$.

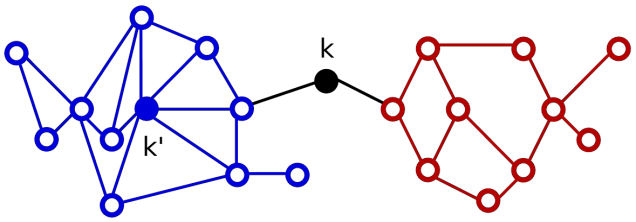



**Figure II.3 Betweenness centrality.** The degree of a vertex does not necessarily reflect its importance in the network. Although vertex $k'$ has a high degree, its removal would not affect communication within the network. However, as vertex $k$ has high betweenness centrality, its removal would impact communication. Adapted with permission from Steuer and López [2008] © 2008 John Wiley and Sons.

Below are just three examples of research in systems biology relying on topological measures.

Durek and Walther [2008] compared the topologies of protein–protein interaction (PPI) networks and metabolic networks. It was found that enzymes with high flux rates tend to be highly connected and occupy central positions in PPI networks. This was established on the basis of strong correlations between flux rate in metabolic networks on the one hand and the clustering coefficient (see definition II.15) and betweenness centrality (see definition II.16) in PPI networks the other hand.

Sorokina *et al.* [2015] proposed a novel representation of metabolism based on reaction similarity. The EC number nomenclature (see section I.2.2) is a rigid classification of enzymatic activities in which similarities are crudely assigned within a predefined hierarchy. In contrast, the authors of this paper advanced a method of grouping together reactions that perform similar chemical transformations in terms of atom and bond changes. In this new representation, vertices in the metabolic network are no longer reactions, but groups of similar reactions. The authors defined three topological measures for weighing nodes in the network according

to different biological meanings. These weights were subsequently used in scoring functions that allowed the identification of reaction modules in the new network representation.

Asensio *et al.* [2017] analyzed the role of protein–protein interactions in infectious diseases, more specifically within the pathogen and human–pathogen interactomes. Starting from the observation that highly connected nodes in protein networks tend to be essential [He and Zhang, 2006], Asensio *et al.* [2017] tackled the case of pathogenic bacteria that need to keep their host alive for their own survival. While the outcome of targeting nodes with high betweenness centrality (see definition II.16) in the host network would result in lethal effects for both the host and the pathogen, it was found that pathogens target the host network without disrupting it. It was also shown that the outcome of infection is proportional to the pathogen's ability to reorganize the host interactome. These findings open the perspective of designing drugs that target strategic interactions within the host–pathogen interactome, in addition to traditional drugs that only target essential proteins for pathogen survival.

### 3.1.2 Network models

No discussion of topological approaches in systems biology would be complete without examining the topology of biological networks themselves. Several network models have been proposed over the years, linking topology to function and network evolution. For a review on the topic, see Yamada and Bork [2009]. This section does not address network evolution.

**Random model** The earliest network model is the Erdős-Rényi random graph model [Erdős and Rényi, 1959] (Figure II.4a), in which the probability that two vertices are connected by an edge is distributed uniformly at random. Among the several variations of this model, the most common is $G_{n,p}$ describing a graph with $n$ vertices in which any given edge is present with probability $p$. These graphs do not represent biological data well [Milenković and Pržulj, 2012].

A further development is the generalized random graph model, in which edges are chosen at random as in the Erdős-Rényi model, but the degree distribution is predetermined [Newman *et al.*, 2001]. Although these graphs preserve the degree distribution of a protein–protein interaction network, for example, the clustering coefficient differs [Milenković and Pržulj, 2012].

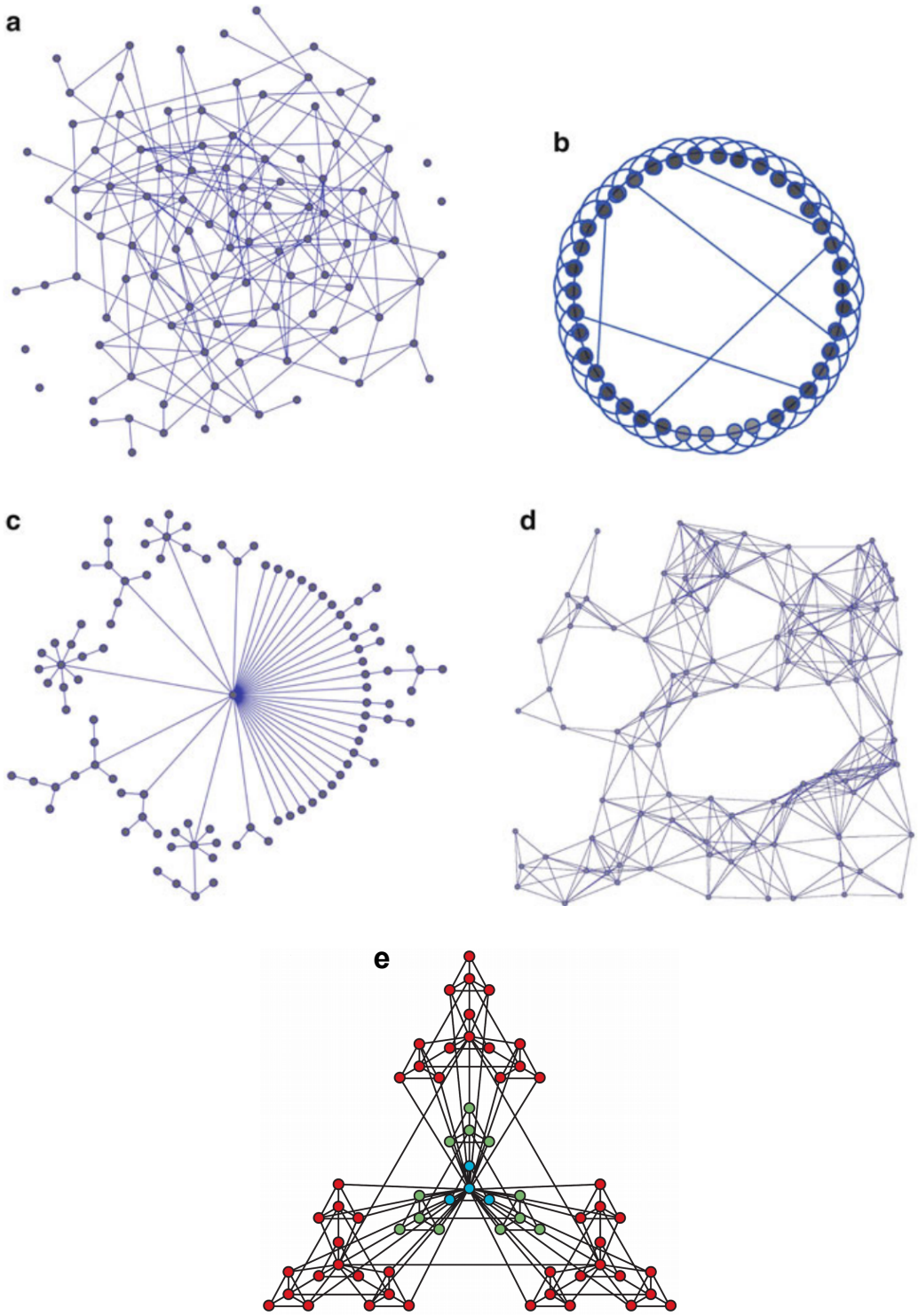



**Figure II.4 Topological models.** Reprinted by permission from Springer Nature: (a)-(d) Milenković and Pržulj [2012] © 2012, (e) Barabási and Oltvai [2004] © 2004. **(a)** An Erdős-Rényi random graph. **(b)** A small-world network. **(c)** A scale-free network. **(d)** A geometric random graph. **(e)** A hierarchical network.

**Small-world model** Watts and Strogatz [1998] introduced this model in order to generate networks with a connection topology ranging from regular (i.e. where each vertex is connected to its $k$ nearest neighbors) to random (Figure II.4b). In this model, edges of the regular graph are reconnected at random with probability $p$, where $p = 0$ corresponds to regularity and $p = 1$ corresponds to randomness. For intermediate values of $p$, the graph has a high clustering coefficient (see definition II.15) and small characteristic path length (see definition II.13), as would be expected of a random graph. Using this model, the authors showed that the neuronal network of *Caenorhabditis elegans* exhibits small-world characteristics, although this claim has been refuted (see geometric model below).

Studying the metabolic network of *Escherichia coli*, Wagner and Fell [2001] concluded that it exhibits characteristics of a small-world network. They further hypothesized that such an architecture would favor the minimization of response time in case of perturbations or of transition time between metabolic states. Arita [2004] employed a more realistic graph representation of the *E. coli* metabolic network, in which transfers of atoms between compounds were accounted for. Using this representation, it was shown that the characteristic path length (see definition II.13) was much longer than initially believed.

**Scale-free model** Barabási and Albert [1999] proposed the scale-free model to explain the topology of complex networks such as the world wide web or citation patterns in scientific literature (Figure II.4c). This model is characterized by the fact that its degree distribution $P(k)$ (i.e. the probability that a given node has degree $k$) decays as a power law, following $P(k) \sim k^{-\gamma}$, where $\gamma$ is between 2 and 4. In scale-free networks, a few highly connected nodes (vertices having a high degree) called *hubs* maintain the network's overall connectivity. This topology is particularly robust against random failure, as it was shown that failure of up to 45% of the nodes still allows for an essentially connected network, if the nodes are randomly chosen [Albert *et al.*, 2000]. Since this resilience property is ensured by hubs, it means that they are vulnerable to targeted attacks, however. Targeted attacks seek to eliminate nodes with high betweenness centrality (see definition II.16).

Barabási and Albert [1999] proposed that the power law of the degree distribution is explained by two factors. The first is network growth and refers to the fact that in this type of network new nodes are created continuously. The second factor is called *preferential attachment* and describes the fact that new nodes are not connected at random, being instead linked preferentially with already well connected nodes.

The scale-free model was found to be applicable to metabolic networks by Jeong

*et al.* [2000], who demonstrated that the metabolic networks of 43 organisms exhibited a scale-free topology, with $\gamma = 2.2$. Later, Barabási and Oltvai [2004] proposed that, apart from metabolic networks, most cellular networks can be described according to the scale-free model, including protein–protein interaction (PPI), signaling, and gene regulatory networks.

Several authors disagreed with the conclusion of Jeong *et al.* [2000] that metabolic networks are scale-free. For example, Tanaka [2005] argues that metabolic networks are scale-rich rather than scale-free, as the degree distribution of metabolites is observed on highly dissimilar scales between the full system level, where it indeed follows a power law, and the module level, where it is exponential.

In addition, Pereira-Leal *et al.* [2004] argued that the essential proteins in the baker's yeast PPI network form an exponential core. The authors suggest that the ancestral network possessed an exponential distribution and that relaxed constraints on preferential attachment enable the emergence of such exponential topologies. Pržulj *et al.* [2004] also disagreed with Barabási and Oltvai [2004] on PPI networks being scale-free (see geometric model below).

**Geometric model** In geometric graphs, vertices distributed in a two- or three-dimensional space are linked by edges if a certain distance criterion is met (Figure II.4d). Morita *et al.* [2001] modeled the neuronal network of *C. elegans* as a geometric graph. *C. elegans* is a nematode having a fixed number of cells (959 in the adult hermaphrodite and 1031 in the male). It is therefore an extremely interesting model organism in developmental biology, as well as in neurobiology since the adult hermaphrodite has precisely 302 neurons, of which 282 make up the somatic nervous system. The authors argued that the small-world model proposed by Watts and Strogatz [1998] did not account for the complete (i.e. fully connected) subgraphs observed on the complete neuronal "wiring diagram" of *C. elegans* which had been available since 1986 [White *et al.*, 1986].

However, geometric graphs are most commonly associated with PPI networks. Pržulj *et al.* [2004] were the first to model PPI networks using geometric graphs, showing that the interactomes of the baker's yeast and fruit fly showed a better fit against this model than against the scale-free model proposed by Barabási and Albert [1999]. They also hypothesized that only the noise in PPI networks is scale-free. An extension of this study further confirmed that geometric random graphs are better at modeling PPI networks for the 14 eukaryotic interactomes that were analyzed than random Erdős-Rényi or random scale-free graphs [Pržulj, 2007]. The same group proposed a development of the geometric model that integrates the concept of evolutionary dynamics [Pržulj *et al.*, 2010].

**Hierarchical model** Ravasz *et al.* [2002] refined the scale-free model proposed by Barabási and Albert [1999] into a hierarchical model simultaneously exhibiting scale-free topology and embedded modularity (Figure II.4e). This model is characterized by the fact that the clustering coefficient (see definition II.15) of a node of degree $k$ decays as $C(k) \sim k^{-1}$. The authors validated their model by measuring the clustering coefficient in the metabolic networks of 43 organisms. The study suggested that metabolic networks contain several large modules which, in turn, are made up of smaller but more integrated submodules.

### 3.1.3 Summary

Topological measures have the potential to yield a wealth of information on network structure. Moreover, metabolic networks are quite accurately described by certain network models. With respect to the aim of this thesis, however, motif extraction from heterogeneous biological networks cannot directly benefit from approaches focusing on network topology. Indirectly, topological measures may serve to refine motif extraction algorithms by adjusting the extraction strategy according to the overall network connectivity. Several possibilities are outlined in the Conclusions and perspectives chapter.

## 3.2 Network alignment

Network alignment consists in determining a mapping between the nodes of two (or more) input networks such that a given cost function is maximized. Since the underlying subgraph isomorphism problem is NP-complete, network alignment methods use heuristics to compare networks [Guzzi and Milenković, 2018].

*local network alignment*

*global network alignment*

Network alignment approaches can be either local or global. *Local network alignment* (LNA) identifies small network regions that likely represent highly conserved structures (Figure II.5a). In contrast, *global network alignment* (GNA) seeks mappings at the level of the whole input networks, which often results in suboptimally matched local structures (Figure II.5b). In general, existing LNA algorithms find topologically small but functionally conserved structures, whereas GNA algorithms find topologically large but poorly functionally conserved structures [Guzzi and Milenković, 2018].

For both local and global approaches, network alignment can be performed on two or more networks, corresponding to *pairwise alignment* (Figure II.6a) and *multiple alignment* (Figure II.6b), respectively.

Numerous local and global, pairwise and multiple network alignment algorithms have been proposed. The purpose of this section is not to enumerate or

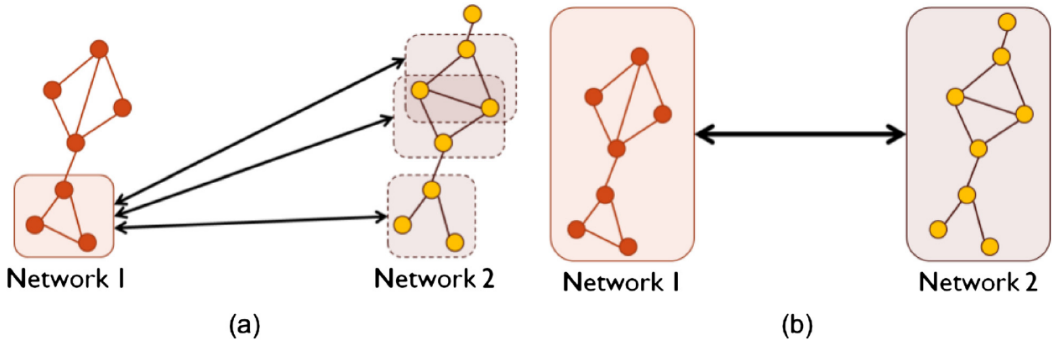



**Figure II.5 Local versus global network alignment.** **(a)** Local alignment. **(b)** Global alignment. Reproduced from Faisal *et al.* [2015] (licensed under CC BY 4.0).

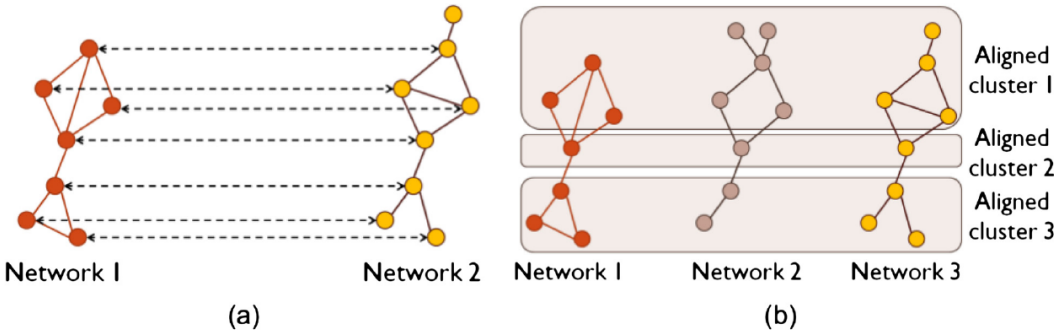



**Figure II.6 Pairwise versus multiple network alignment.** **(a)** Pairwise alignment. **(b)** Multiple alignment. Reproduced from Faisal *et al.* [2015] (licensed under CC BY 4.0).

compare such approaches, but to give the general idea of network alignment and its applications, and to present the currently open questions in the field. For more information, the interested reader may turn to Chen *et al.* [2009] for an integer programming formulation of pairwise alignment; to Clark and Kalita [2014] for a comparison of pairwise LNA algorithms; or to Mohammadi and Grama [2012]; Faisal *et al.* [2015]; Guzzi and Milenković [2018] for a general network alignment overview and a comparison of existing algorithms.

**Applications** Network alignment is commonly used as a complementary method to sequence alignment for transferring *functional annotation*. Recall from section I.3.2 that paralogs (homologous sequences separated by a duplication event) can be either in-paralogs (recent paralogs) or out-paralogs (ancient paralogs) with respect to a speciation event of reference. As paralogous sequences usually diverge in function after the duplication event, it is more likely for in-paralogs to be true func-

tional orthologs, since the duplication is more recent. When sequence similarity is not enough to identify true functional orthologs, other types of networks may be aligned to exclude out-paralogs [Mohammadi and Grama, 2012]. For example, an alignment of protein–protein interaction networks taking homology information into account may prove useful (Figure II.7).

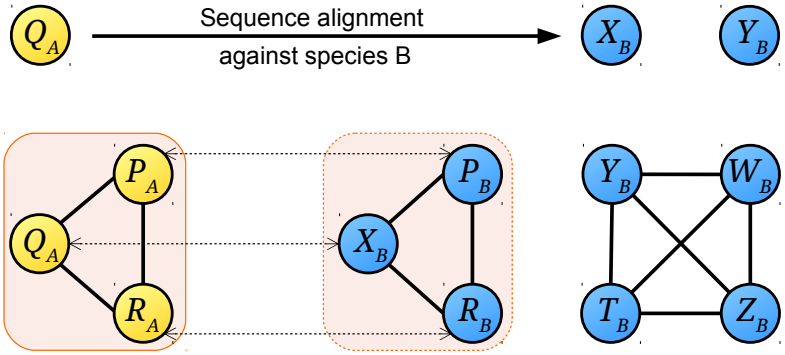



**Figure II.7 Network alignment may complement sequence alignment.** The query protein $Q_A$ in species $A$ has two homologous sequences $X_B$ and $Y_B$ in species $B$. If the protein–protein interaction networks of the two species are known, a network alignment that also takes sequence homology into account (here, $P_B$ and $R_B$ are homologues of $P_A$ and $R_A$, respectively) reveals that $X_B$ is a true functional ortholog for $Q_A$.

Additional applications of network alignment include the identification of conserved functional *modules* across different species, revealing *evolutionary relationships*, and *disease discovery* [Mohammadi and Grama, 2012; Faisal *et al.*, 2015].

**Open question** Here are summarized currently open questions in the field of network alignment (see Guzzi and Milenković [2018]).

Guzzi and Milenković [2018] discuss the possible reconciliation of local and global network alignment. Since there appears to be a conflict between functional (LNA) versus topological (GNA) fit, network aligners typically perform a single type of alignment. Although there exists one integrative algorithm to date that can be parametrized to perform either local or global alignment, it is not integrative in the sense that it does not improve either LNA or GNA compared to the other LNA- or GNA-specific aligners.

Another open question is related to the comparison of different alignments and to the evaluation of alignment quality. Although different metrics exist, focusing either on functional or topological quality, it would be beneficial if these (and possible new) measures could be integrated into a single unified framework.

In terms of applicability of alignment methods for biological networks to other

types of networks, it may be easier to adapt GNA algorithms because, unlike LNA approaches, they are directed at topological instead of functional network features. However, a limiting factor for their applicability to other domains is scalability. Currently existing network alignment methods would first need to be rendered more efficient.

**Summary** While network alignment comes closer to motif extraction than topological approaches, it is directly applicable to heterogeneous biological networks under very specific problem formulations (see section 4.2.1). The next section briefly surveys network mining approaches.

## 3.3 Network mining

Network (or graph) mining refers to the problem of searching for a particular pattern in a graph. Most graph mining approaches fall under two broad categories: frequent subgraph mining (FSM) and recurrent pattern mining. Parthasarathy *et al.* [2010]; Li *et al.* [2012] provide a general overview of graph mining in systems biology. For a general introduction to pattern mining, see Cheng *et al.* [2010].

**Frequent subgraph mining** A subgraph is *frequent* in a collection of graphs or in a single large network if it occurs with a frequency equal to or greater than a given threshold. However, exhaustive enumeration of all possible subgraphs in order to determine whether they are frequent is computationally intractable. Different approaches for frequent subgraph mining exist (see Jiang *et al.* [2013] for existing algorithms), the classical ones being Apriori-based and pattern growth (for more details, see Yan and Han [2006]):

- Apriori-based approaches start with small subgraphs that are extended at each iteration with an additional vertex or edge, using a breadth-first search[1] strategy. New subgraphs are created by joining existing smaller subgraphs.
- Pattern growth approaches use a depth-first search[2] strategy in which every newly discovered subgraph $g$ is extended recursively until every frequent subgraph that contains $g$ is discovered.

[1] Breadth-first search (BFS) is a graph traversal algorithm. Starting with a given vertex, BFS explores every direct neighbor of the starting vertex, then every directly neighboring vertex of a given neighbor of the starting node, and so on, until no more vertices can be explored.

[2] Depth-first search (DFS) is a graph traversal algorithm. Starting with a given vertex, DFS explores as far as possible in terms of depth (a neighbor of the starting node, then the first neighbor of this node, and so on) before backtracking.

**Recurrent pattern mining** This category includes graph mining algorithms for various patterns, such as coherent dense subgraphs [Hu *et al.*, 2005], frequent dense vertex sets [Li *et al.*, 2012], densest connected subgraphs [Wu *et al.*, 2016], etc.

**Examples** Below are a few examples of graph mining applications in systems biology.

Cakmak and Ozsoyoglu [2007] represented metabolic pathways as pathways of functionality templates, meaning graphs with Gene Ontology (GO) [GO Consortium, 2001] annotations instead of enzymes. They then mined for *frequent functionality patterns* (patterns made up of GO terms) in metabolic networks of different species, which allowed to infer previously unknown pathways.

Yan *et al.* [2007] proposed an algorithm for mining *frequent dense vertex sets* in coexpression graphs. The immediate applicability of this method is to detect potential transcriptional modules, given many microarray data sets.

Cheng and Yan [2017] modeled protein–RNA complexes as residue graphs, then mined for *common subgraphs* in protein–RNA interfaces in order to predict RNA binding sites. The study also pointed out residue patterns that might contribute to binding affinity.

Reinharz *et al.* [2018] developed a method for identifying conserved structural modules in three-dimensional RNA structures, based on interactions rather than sequence information. The methodology involved mining for *recurrent subgraphs* with a given topology.

**Summary** The graph mining methods described herein are implicitly applicable to a single network. The next section examines approaches specifically aimed at several networks.

## 4 Approaches for heterogeneous biological networks

*heterogeneous networks*

In systems biology, two networks are said to be *heterogeneous* if they contain different types of information describing distinct aspects of related processes for the same biological entity. For example, a set of heterogeneous networks would include at least two items such as the genomic context of an organism and any one of the following networks: its metabolic, coexpression, regulation, signaling, and protein–protein interaction networks.

Integrating heterogeneous biological data may help to elucidate particular aspects of an organism's lifestyle. For example, Tonon *et al.* [2011] proposed an integrative approach to the study of abiotic stress in brown algae of the genus *Ec-*

*tocarpus*. This approach, consisting in the integration of metabolomic, genomic, and transcriptomic data sets, allowed to uncover mechanisms of acclimation and adaptation to abiotic conditions. However, this section only discusses graph-based approaches for heterogeneous biological networks specifically aimed at pattern detection.

This section presents existing approaches for aligning or mining heterogeneous biological networks. Since both types of approaches result in identifying subgraphs across the input networks such that certain constraints are fulfilled, the approaches discussed herein are divided into pioneering works and general frameworks. The first category contains methods that have been proposed in order to solve a very specific problem, whereas methods in the second category are general-purpose and more easily adaptable to different types of biological data.

A brief discussion for each of the two categories summarizes the reasons for which a different strategy was adopted in this thesis. In particular, the output of each method is compared with the type of sought motif. Our aim is to detect metabolic and genomic patterns, defined as trails of reactions catalyzed by products of neighboring genes. Recall that, as opposed to paths, trails may contain repeated vertices, but not repeated arcs (see definition II.11). Hence, identifying trails instead of paths has the advantage of capturing metabolic routes that may contain cycles. On the one hand, a trail corresponds to a group of genes that are directly involved in a sequence of metabolic reactions. A subgraph, on the other hand, has the drawback of mixing together several metabolic routes.

## 4.1 Pioneering works

### 4.1.1 Correlated gene clusters

Ogata *et al.* [2000] proposed a heuristic graph comparison algorithm for extracting *functionally related enzyme clusters* (FRECs). A FREC is defined as a set of enzymes catalyzing successive reactions in a metabolic pathway such that the enzymes are encoded by genes in close locations on the chromosome.

*functionally related enzyme cluster*

In this comparison approach, both the order of genes on the chromosome and metabolic pathways are modeled as undirected graphs. The order of genes on the chromosome is represented as an undirected graph $G_1 = (V_1, E_1)$ with genes for vertices. $G_1$ takes into account the circularity or linearity of the chromosome while ignoring the direction of transcription. If the organism under study has several chromosomes, then $G_1$ has several connected components. A metabolic pathway is represented as an undirected graph $G_2 = (V_2, E_2)$ with enzymes for vertices. Two vertices are connected by an edge if the enzymes they represent are involved in

reactions sharing the same chemical compound as product and substrate, respectively. Since $G_2$ is undirected, all reactions are considered to be reversible.

The mapping between the two graphs $G_1$ and $G_2$ is given by a many-to-many correspondence function based on EC numbers between $V_1$ and $V_2$. The mapping is many-to-many because a given enzyme may catalyze several reactions and a given reaction may involve several enzymes (i.e. an enzyme complex that is the product of several genes).

Two gap parameters $\gamma_1$ and $\gamma_2$ are defined, representing the number of genes and enzymes that can be skipped in $G_1$ and $G_2$, respectively. Initially, every pair of corresponding vertices in $V_1$ and $V_2$ forms a cluster. Two clusters $C_i$ and $C_j$ are merged if there is a shortest path in both $G_1$ and $G_2$ between a vertex in $C_i$ and a vertex in $C_j$ such that the length of the path is at most $\gamma_1 + 1$ in $G_1$ and $\gamma_2 + 1$ in $G_2$, respectively. Clusters are merged according to this procedure until no more clusters can be merged. When this happens, the resulting clusters are FRECs.

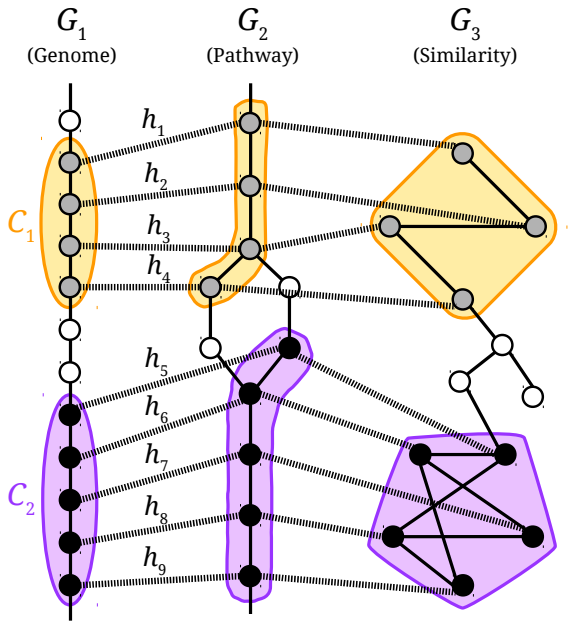



**Figure II.8 Correlated gene clusters.** $C_1$ (in yellow) and $C_2$ (in purple) are two correlated gene clusters, linked by hyperedges $h_1, \ldots, h_4$ and $h_5, \ldots, h_9$, respectively. Inspired by Nakaya *et al.* [2001].

Nakaya *et al.* [2001] extended this algorithm in order to handle multiple graphs with either genes or gene products for vertices. The mapping between vertices of two different graphs is established using hyperedges. This approach extends the notion of FRECs by defining and identifying *correlated gene clusters*. A correlated gene cluster is a set of corresponding vertices in the input graphs (i.e. vertices

*correlated gene cluster*

linked by hyperedges). In Figure II.8, $C_1$ and $C_2$ are two correlated gene clusters in the input graphs $G_1$, $G_2$, and $G_3$. This extension allows to determine simultaneous correlations between different data sets, such as genomic, metabolic, PPI, or coexpression data. In addition, if at least two among the input graphs represent gene order on the chromosome, the correspondence between their vertices is established on the basis of bidirectional best hits (see KEGG SSDB in section III.2.2).

### 4.1.2 Operon prediction

Observing that enzymes encoded by genes belonging to an operon tend to catalyze successive reactions, Zheng *et al.* [2002] developed a method for operon prediction using metabolic and genomic data.

Similarly to the previous method (see 4.1.1 above), metabolic pathways and gene order on the chromosome are both represented as undirected graphs. The correspondence between genes and enzymes is based on EC numbers.

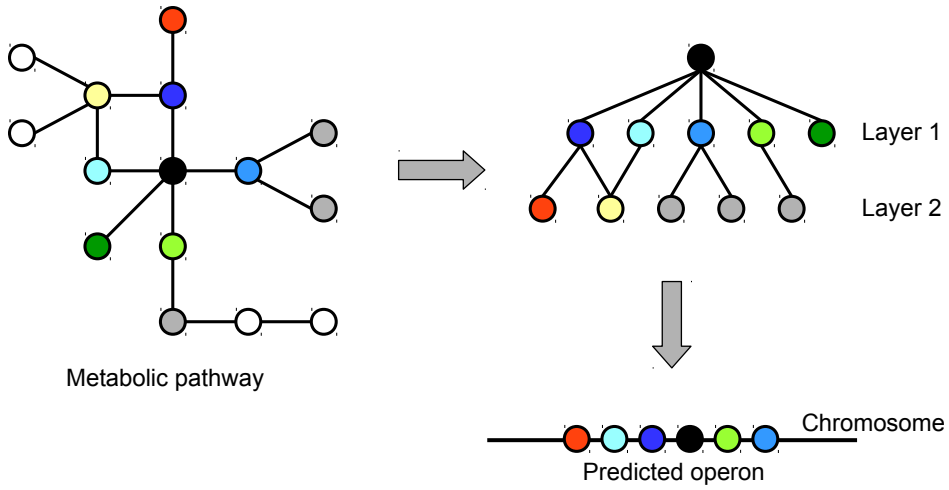



**Figure II.9 Graphical representation of breadth-first search (BFS) traversal.** Here, BFS starts with the vertex in black in the metabolic pathway. In this example, BFS is ran for a depth of 2. The first layer in the traversal "tree" on the right contains the direct neighbors of the black vertex. The second layer contains the direct neighbors of the vertices in the first layer. Inspired by Zheng *et al.* [2002].

The algorithm for operon prediction is a three-step process:

1. The **matching step** uses a modified version of breadth-first search (BFS) in which every vertex in the graph representing a metabolic pathway is, in turn, the starting vertex for traversal. For each starting vertex, a tree-like struc-

*traversal tree*

ture resulting from BFS traversal, called the *traversal tree*, is constructed[3]. It is checked whether nodes in the traversal tree are found within a same region of the chromosome. Since this method is aimed at predicting operons, not at identifying correlated gene clusters (see 4.1.1), BFS runs up to a predetermined (but configurable) depth of 3, meaning that only reactions up to three steps away from the root vertex are visited. For example, BFS is ran with a depth of 2 starting from the black vertex in Figure II.9. At the end of this step, putative operons are identified.

*BFS depth*

2. The **pruning step** is aimed at increasing the specificity of the algorithm and consists in eliminating genes at the extremities of putative operons identified during the matching step if they are separated from other genes in the group by at least two other genes.
3. The **merging step** (called *clustering step* by the authors) takes place at the very end, once the matching and pruning steps have been performed for every vertex in every metabolic pathway of the species under study. The merging step consists in merging overlapping clusters reported after the matching and pruning steps.

### 4.1.3 Evolutionary modules

Spirin *et al.* [2006] integrated metabolic networks and genomic associations in order to reveal evolutionary modules. *Evolutionary modules* are defined as regions of the metabolic network made up of highly connected reactions that are also highly associated from a comparative genomics standpoint. Two genes are said to be *associated* if, in different organisms, their neighborhoods are conserved, if they exhibit co-occurrence, and/or if they can be found fused together.

*evolutionary module*

*associated genes*

*integrated metabolic–genomic network*

The *integrated metabolic–genomic network* is an undirected graph with reactions for vertices, connected by two types of edges representing metabolic and genomic associations, respectively. A metabolic edge connects two reactions if they share a metabolite. Ubiquitous metabolites (such as ATP, phosphate, $H^+$, etc.) are excluded in order to avoid reaction over-connectivity. Two reactions are connected by a genomic edge if they are catalyzed by enzymes or enzyme subunits encoded by associated genes (see above).

Two algorithms that use the integrated metabolic–genomic network to search for clusters (evolutionary modules) are proposed. One is a Monte Carlo algorithm seeking to maximize the number of edges of both types (metabolic and genomic) for

[3]Stricly speaking, the traversal tree is not actually a tree, as a node can have multiple parents (see for example the yellow node in Figure II.9).

a set of reactions. The second algorithm identifies clusters where each pair of nodes is connected by two paths, made up of metabolic and genomic edges, respectively. This second algorithm relies on exact enumeration of connected components on metabolic edges. Successive refinements lead to the identification of clusters linked by both types of edges, as shown in Figure II.10.

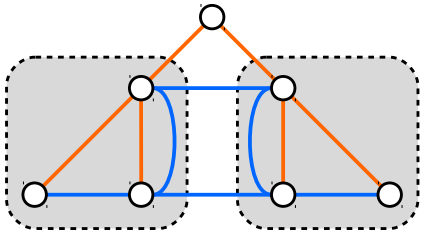


**Figure II.10 Illustration of cluster detection.** Orange and blue edges correspond to metabolic and genomic associations, respectively. In a first approximation, all seven vertices are in a connected component linked by orange edges. In a second approximation, the top vertex is excluded as it is not connected by blue edges to the remaining vertices. In a third approximation, the two clusters highlighted in gray are finally identified, as every pair of vertices in each cluster is connected by paths with orange, respectively blue, edges. Inspired by Spirin *et al.* [2006] (supplementary information).

### 4.1.4 Discussion

The approaches presented in sections 4.1.1 and 4.1.2 map genes to enzymes via EC numbers. (The mapping method is not mentioned in 4.1.3.) This is problematic, as not every reaction has an associated EC number, and some reactions only have partial EC numbers (see also section I.2.2). It is preferable to map genes to enzymes using already established associations. For KEGG, this information is available in KGML files (see section III.2.3).

It is important to note that reaction directionality is lost in all three approaches, because metabolic pathways are represented as undirected graphs.

The heuristic aspect of the algorithm proposed by Ogata *et al.* [2000] and Nakaya *et al.* [2001] (see 4.1.1) resides in the fact that only shortest paths between vertices of two clusters are computed. In the case of cycles as in Figure II.11, merging clusters is only possible if the gap parameters are adjusted accordingly. For the example in Figure II.11, $\gamma_1$ would need to be set at least to 1. This method is therefore not guaranteed to return (undirected) trails corresponding to cycles in the metabolic pathway.

Although designed for operon prediction, the method proposed by Zheng *et al.* [2002] (see 4.1.2) could be adapted to detect groups of neighboring genes such that

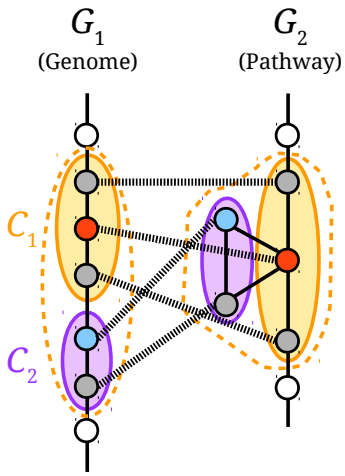



**Figure II.11 Correlated gene clusters are not maximal (see 4.1.1).** The correlated gene cluster $C_1$ (yellow solid contours) is not maximal, as the algorithm computes shortest paths and not cycles. The maximal correlated gene cluster is shown with a dashed yellow contour. Clusters $C_1$ and $C_2$ are merged if a shortest path exists in both graphs between the red and blue vertices. Since the shortest path in $G_1$ between the red and blue vertices has length 2, $\gamma_1$ needs to be at least 1.

their products are involved in neighboring reactions by relaxing the BFS depth parameter in the matching step and by removing the pruning step altogether. Nevertheless, reactions corresponding to a group of genes identified using this method do not necessarily form a metabolic route (a trail in the undirected case). For example, the gene products of the predicted operon in Figure II.9 are involved in a several metabolic routes forming a "branched" subgraph.

Searching for evolutionary modules using the method developed by Spirin *et al.* [2006] produces clusters of reactions linked by both metabolic and genomic associations (see 4.1.3). However, although the genomic edges among a cluster mean that genes involved in the reactions connected by such edges are neighbors on the chromosome, the metabolic edges do not necessarily correspond to a metabolic route. Using as example part of the metabolic pathway (with edge orientation) and the chromosome portion in Figure II.9, Figure II.12 shows a cluster in the integrated network where the metabolic edges do not correspond to a metabolic route.

## 4.2 General frameworks

### 4.2.1 Connectons

Boyer *et al.* [2005] designed a framework for extracting various motifs as common connected components from an undirected correspondence multigraph representing the input networks and the relations between them. Examples of motifs

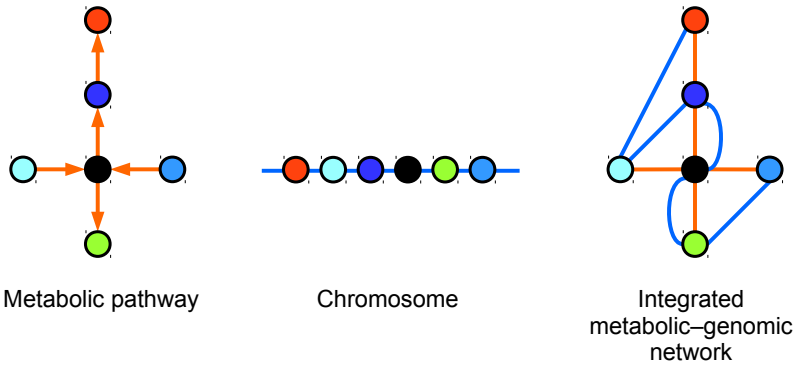



**Figure II.12 Example of an integrated metabolic–genomic network.** Metabolic edges in the integrated network are shown in orange, and genomic edges in blue. Since reaction directionality is lost, metabolic edges do not necessarily correspond to actual metabolic routes.

include syntons (neighboring genes for two or more species), metabolons (neighboring genes whose products are involved in connected metabolic reactions), and interactons (neighboring genes encoding physically interacting proteins).

*correspondence multigraph*

In the *correspondence multigraph*, vertices are connected by different types of edges. Edge type is defined according to a correspondence relation. For example, vertices might be reactions and two different types of edges between vertices may describe which reactions are connected in a metabolic pathway and which are catalyzed by products of neighboring genes. In this respect, the integrated metabolic–genomic network in the method proposed by Spirin *et al.* [2006] (see 4.1.3 above) is similar to the correspondence multigraph. By changing the correspondence relation, the multigraph can accommodate different types of data. For example, it can be used to represent interacting proteins in relation to gene order on the chromosome. As the name implies, *common connected components* are maximal subgraphs in the multigraph such that any two vertices are connected by paths consisting exclusively of edges of a given type, for all types of edges in the multigraph.

*common connected components*

While this method could in theory handle multiple input graphs, in practice the size of the correspondence multigraph is exponential in the number of networks when the correspondence between vertices is not one-to-one. The same group therefore proposed an improved framework that handles larger numbers of input networks by building an undirected *network alignment multigraph* on-the-fly [Deniélou *et al.*, 2009]. The concept of connecton was also introduced such that it generalizes syntons, metabolons, and interactons. A *connecton* is defined as a maximal subgraph in the multigraph such that, for each relation type, it is a connected component for that particular relation type.

*connecton*

A further development allowing the correspondence between aligned networks to be partial was employed for the detection of synteny blocks in bacteria [Deniélou *et al.*, 2011].

### 4.2.2 SIPPER

Meanwhile, Bordron *et al.* [2011] presented SIPPER, a method that was illustrated on the integrated genomic and metabolic network of *Escherichia coli*. SIPPER returns the $k$ shortest paths between two reactions.

*integrated network*

The *integrated network* is a directed weighted graph where each vertex is labeled with a gene–reaction pair. The mapping between reactions and genes is based on EC numbers. Arc weights in the integrated network represent the distance between genes within the genome. Arc weights are used to compute path length, which is defined as the ratio between the total weight of the path and the number of distinct reactions in the path. The shortest integrated path between two reactions is called a 1-SIP. SIPPER uses a heuristic algorithm to compute the $k$ shortest paths between a source and a destination reaction, thus yielding a subgraph of the integrated network called a $k$-SIP.

### 4.2.3 Longest path heuristic

Fertin *et al.* [2012] proposed a framework for the comparison of two heterogeneous biological networks, modeled by a directed graph $D$ and an undirected graph $G'$, respectively. For example, $D$ may represent a metabolic network and $G'$ may represent gene order on the chromosome, or a protein–protein interaction network.

The framework requires two simplifications, as it takes as input a directed acyclic graph (DAG) $D$ and an undirected graph $G$ on the same vertex set. In other words, instead of using a correspondence function between the vertex sets of two heterogeneous networks (see methods in 4.1.1, 4.1.2, 4.1.3, and 4.2.1) or applying joint double labels to vertices (see method in 4.2.2), this method requires both the correspondence function and the construction of an *additional undirected graph* $G$ on the same vertex set as the DAG $D$. An example of construction is given in Figure II.13.

*additional undirected graph*

The framework uses a heuristic algorithm for determining a longest path $P$ in $D$ such that $P$ induces a connected subgraph in $G$. Depending on the nature of the initial graph $G'$, the algorithm can be used to find paths of reactions catalyzed by products of neighboring genes, or by physically interacting proteins. Since the heuristic can only be applied on DAGs, a decomposition into DAGs [Blin *et al.*,

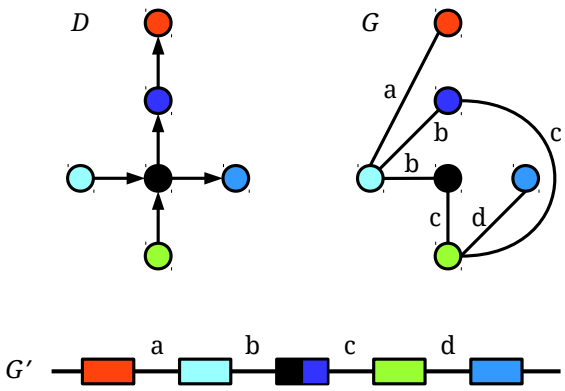



**Figure II.13 Construction of an additional undirected graph.** The algorithm proposed by Fertin *et al.* [2012] compares a directed acyclic graph $D$ and an undirected graph $G'$ by indirectly comparing $D$ and an additional undirected graph $G$, where both $D$ and $G$ have the same vertex set. A correspondence function associates a subset of vertices of $G'$ to every vertex in $D$. This correspondence function is used to construct $G$ by "translating" every edge in $G'$. Here, $D$ is a metabolic pathway (with reactions for vertices) and $G'$ is the order of genes on the chromosome (with genes for vertices). For the purpose of this example, edges in $G$ and $G'$ are labeled. The third gene (black and dark blue) is involved in both the black and dark blue reactions. When constructing $G$, edges $b$ and $c$ in $G'$ need to be "translated" accordingly. For instance, the edge $c$ in $G'$ links the black and dark blue genes with the green gene. Thus, this edge results in adding an edge between the black and green reactions in $G$, as well as an edge between the dark blue and green reactions (both are labeled $c$ in graph $G$).

2011] is necessary if the metabolic pathway contains cycles (which is almost always the case). Complexity results and proofs are provided in Fertin *et al.* [2015] for this problem that is NP-hard in the general case (i.e. when $D$ is not a DAG).

### 4.2.4 Discussion

The framework proposed by Boyer *et al.* [2005]; Deniélou *et al.* [2009, 2011] (see 4.2.1) extracts undirected subgraphs. In the context of metabolic pathways, this means that reaction directionality is lost.

The method proposed by Bordron *et al.* [2011] (see 4.2.2) extracts subgraphs consisting in the $k$ shortest paths between two reactions. This implies that reaction pairs need to be defined beforehand. Moreover, a post-processing step is also necessary because the shortest path between two reactions may include arcs with weights indicating that the genes involved in the reactions are too far apart to allow for any meaningful biological interpretation. For example, the 1-SIP in Figure 2a in Bordron *et al.* [2011] involves two genes separated by 43 other genes (the reactions are linked by an arc with weight 44).

The heuristic algorithm presented by Fertin *et al.* [2012] (see 4.2.3) is applicable to DAGs. This simplification was performed because the longest path problem is NP-hard in the general case [Fertin *et al.*, 2015]. However, metabolic pathways cannot be realistically modeled as DAGs because they typically contain cycles (a simple example being reversible reactions). Decomposing a directed graph into DAGs is not straightforward [Blin *et al.*, 2011] and may lead to loss of solutions. Avoiding solution loss would involve a post-processing merging step, where longest paths obtained in the decomposed graph would be concatenated with other partially overlapping paths (where applicable).

Interestingly, the algorithm proposed by Fertin *et al.* [2012] searches for a *longest* path, unlike other methods that focus on shortest paths. In the context of metabolic and genomic patterns, it is meaningful to search for maximal sequences of reactions catalyzed by products of neighboring genes. In Chapter IV, we therefore use the longest path problem formulation as a starting point for extracting "longest" trails.

## 5 Concluding remarks

After briefly introducing several notions of graph theory, this chapter gave an overview of graph-theoretical methods used in systems biology, namely topological, alignment, and mining approaches.

A particular emphasis was placed upon heterogeneous biological networks. We reviewed existing algorithms aimed at extracting patterns from such networks. It was found that none of these methods could be used nor adapted to extract patterns representing trails of reactions from a metabolic pathway such that the enzymes catalyzing the reactions are encoded by neighboring genes. These patterns would enable the exact identification of metabolic and genomic patterns.

For the purpose of extracting such patterns, an algorithm relying on trail finding is investigated in Chapter IV. Trail extraction conveys more biological meaning than subgraph extraction and richer information (in terms of reactions and cycles) than path extraction. Thus, in metabolic pathways modeled as directed graphs, trails have the ability to capture cycles, take reaction directionality into account, and guarantee that reactions in the trails correspond to actual metabolic routes.

Before proceeding to trail finding, however, the next chapter makes a necessary incursion through the KEGG knowledge base, as it is the primary source for the metabolic and genomic information used in the applications presented in this thesis.

# III

# The KEGG knowledge base: presentation and consistency issues

# 1 Introduction

Recall from Chapter I that we have chosen to use the KEGG (Kyoto Encyclopedia of Genes and Genomes)[1] knowledge base because it provides a global, top-down view of metabolism, as opposed to MetaCyc which goes into greater levels of detail on individual metabolic pathways.

Since its inception, the primary purpose of KEGG has been linking sequence data to biological function, at molecular as well as higher levels. Continually expanding since 1995, KEGG currently includes genomic, chemical, systems, and health information, making it a *de facto* reference for applications ranging from genome analysis to metabolic engineering.

Since the applications presented in this thesis rely on KEGG as the main source for metabolic and genomic information, the present chapter begins with an overview of the knowledge base. The overview gives a brief historical background on the beginnings of KEGG, then details the structure and role of the different components of the knowledge base. Finally, we discuss our contribution to detecting potential inconsistencies in KEGG.

# 2 Overview of the KEGG knowledge base

## 2.1 Historical context

The first organisms to have had their genomes sequenced were two viruses: the bacteriophage MS2, a single-stranded RNA virus sequenced in 1976, and the bacteriophage ΦX174, a single-stranded DNA virus sequenced the following year. In 1995, *Haemophilus influenza*, a pathogenic bacterium, was the first free-living organism to have had its genome completely sequenced.

As more efficient sequencing methods were being developed, Minoru Kanehisa, Professor at the Institute for Chemical Research, Kyoto University, anticipated the need to interpret and exploit genome sequence data. Having been part of the team that created GenBank in the 1980s, in 1995 Kanehisa began developing KEGG PATHWAY, a collection of manually drawn pathway maps. The first description of KEGG was published one year later [Kanehisa, 1996], when KEGG included information on pathways, genes, and compounds, interconnected via EC numbers. The paper stated that a major objective of KEGG was linking structural to functional data. As a visionary scientist, Kanehisa also predicted the emergence of metabolic engineering, which he referred to as *pathway engineering*, in the 21$^{st}$ century.

[1] KEGG website: https://www.kegg.jp

For the past two decades, KEGG has been constantly extended and enriched with new information, without deviating from its original purpose.

## 2.2 KEGG databases

As of the writing of this thesis, KEGG contains 18 databases, broadly categorized as systems information, genomic information, chemical information, and health information resources (Figure III.1). The databases are handled through an integrated distributed database retrieval system named DBGET/LinkDB [Fujibuchi *et al.*, 1998].

```
Systems information
    KEGG PATHWAY    Pathway maps, reference (total)                  525 (582,047)
    KEGG BRITE      Functional hierarchies, reference (total)        202 (208,660)
    KEGG MODULE     KEGG modules, reference (total)                  792 (474,998)

Genomic information
    KEGG ORTHOLOGY  KEGG Orthology (KO) groups                          22,126
    KEGG GENOME     KEGG organisms and selected viruses                  5,777
    KEGG GENES      Genes in KEGG organisms and other categories        26,476,450
    KEGG SSDB       Best hit relations within GENES                246,097,838,675
                    Bi-directional best hit relations within GENES  12,933,922,017

Chemical information (KEGG LIGAND)
    KEGG COMPOUND   Metabolites and other small molecules                18,335
    KEGG GLYCAN     Glycans                                             11,032
    KEGG REACTION   Biochemical reactions                               10,921
    KEGG RCLASS     Reaction class                                       3,108
    KEGG ENZYME     Enzyme nomenclature                                  7,214

Health information (KEGG MEDICUS)
    KEGG NETWORK    Disease-related network elements                       348
    KEGG VARIANT    Human gene variants                                    169
    KEGG DISEASE    Human diseases                                       2,094
    KEGG DRUG       Drugs                                               10,512
    KEGG DGROUP     Drug groups                                          2,062
    KEGG ENVIRON    Crude drugs and health-related substances              856
```

**Figure III.1 Overview of the KEGG knowledge base as of June 2018.** For each of the four categories are listed currently existing KEGG databases along with the number of entries in each database. The numbers in parentheses include computationally generated organism-specific entries. The values were retrieved from the statistics page (https://www.kegg.jp/kegg/docs/statistics.html) on June 14, 2018.

**Systems information** The systems information category of the KEGG knowledge base contains the following databases:

- KEGG PATHWAY (since 1995) is a collection of manually drawn pathway maps for primary and secondary metabolism (as well as global and overview maps), genetic information processing (such as transcription and translation), environmental information processing (such as signal transduction), cellular processes (such as the cell cycle), organismal systems (such as the immune system), human diseases, and drug development. As explained in section I.2.3, metabolic pathway maps in KEGG may group several metabolic pathways around a central metabolic process. Moreover, reference maps provide a global view on metabolism by cumulating every known metabolic variation for every sequenced organism. Metabolic pathway maps for a given species are thus subsets of the reference maps, in which only the reactions known to be performed by the given species are marked as present.

- KEGG BRITE (since 2005) is an ontology of functional hierarchies linking different biological entities, such as genes and proteins, compounds and reactions, or diseases and drugs [Kanehisa *et al.*, 2011].

- KEGG MODULE (since 2006) is a collection of functional units called *modules*, defined by boolean expression of orthology groups [Kanehisa *et al.*, 2013] (see KEGG ORTHOLOGY below). Functional units describe enzyme complexes and conserved subpathways in metabolic pathways, among others.

**Genomic information** The genomic information category of the KEGG knowledge base contains the following databases:

- KEGG ORTHOLOGY (since 2002) is a database of molecular function consisting of a collection of orthologs. KO (KEGG ORTHOLOGY) entries are defined as sequence similarity groups and assigned identifiers referred to as *K numbers*. Genome annotation in KEGG is done by assigning K numbers to individual genes in the KEGG GENES database (see below). The assignment of K numbers to genes involves both manual and automatic strategies [Kanehisa *et al.*, 2016b]. Thus, an advantage of genome annotation in KEGG over other sequence databases is that functional annotation performed using KO assignments is not associated to the sequence itself and does not entail its redefinition [Kanehisa, 2017]. See KEGG SSDB below for more details.

*K number*

- KEGG GENOME (since 2000) is a collection of organisms with complete genomes. Each species is designated by its three- or four-letter code [KEGG Organisms]. There are currently 5,777 species present in KEGG GENOME, of which ~8% eukaryotes, ~82% bacteria, ~5% archaea, and ~5% viruses.

- KEGG GENES (since 1995) contains the repertoire of genes (retrieved from RefSeq or GenBank) for all the species with complete genomes present in KEGG GENOME. The database currently contains over 26 million gene entries.

- KEGG SSDB (Sequence Similarity DataBase, since 2001) is a database resource on similarity of protein-coding genes and (bidirectional) best hits [Kanehisa *et al.*, 2002, 2013]. Amino acid sequence similarity is computed for all possible pairs of protein-coding genes for all complete genomes (currently more than 5,700) and stored in SSDB if a certain threshold is reached. In sequence analysis, a *bidirectional best hit* describes the relationship between a sequence $a$ in genome $A$ and another sequence $b$ in genome $B$, if $a$ is the best hit for the query $b$ against all sequences in genome $A$ and if $b$ is the best hit for the query $a$ against all sequences in $B$. Bidirectional best hits are widely employed as a strong indicator of orthology. Yet, Dalquen and Dessimoz [2013] have shown that this approach fails to detect orthologous sequences if gene duplication events took place after speciation. Although not explicitly stated, the (non-bidirectional) best hit information is probably used in SSDB along several other criteria such as presence of protein domains [Itoh *et al.*, 2002; Minowa *et al.*, 2003] to identify paralog sequences and to refine ortholog detection for the computational generation of paralog and ortholog clusters [Kanehisa *et al.*, 2004]. Internally, SSDB is used as a graph resource of genes connected by weighted arcs, where arc weight is a function of sequence similarity and arc orientation is given by best hit relations. Clique-like subgraphs in the SSDB graph are the basis for genome annotation and establishment of KO entries, followed by manual curation when discrepancies are detected [Kanehisa *et al.*, 2013] (see KEGG ORTHOLOGY above).

*bidirectional best hit*

**Chemical information** The chemical information category of the KEGG knowledge base contains the following databases, collectively referred to as KEGG LIGAND:

- KEGG COMPOUND (since 1995) contains compounds with biological roles (currently, over 18,000). To each compound is assigned a unique identifier starting with the letter C and followed by 5 digits, referred to as the *C number*. The database also offers the possibility to search for similar chemical structures [Hattori *et al.*, 2010]. Chemical similarity is evaluated by extracting maximal common subgraphs from graphs representing chemical structures [Hattori *et al.*, 2003].

*C number*

- KEGG GLYCAN (since 2003) contains experimentally determined glycan structures (currently, over 11,000) [Hashimoto *et al.*, 2006].

- KEGG REACTION (since 1998) is a collection of substrate–pair relations, representing mostly enzymatic reactions. To each reaction is assigned a unique identifier starting with the letter R and followed by 5 digits, referred to as the *R number*. Reactions are linked to enzyme K numbers (see KEGG ORTHOLOGY above). The database currently contains over 10,000 reaction entries.

*R number*

- KEGG RCLASS (since 2010) defines reaction classes that are subsequently used to classify R numbers from the KEGG REACTION database. A reaction class is a type of chemical transformation between pairs of substrates and products of a reaction. Reaction classes are described by the RDM patterns introduced by Hattori *et al.* [2003], where a RDM pattern represents changes at the reaction center (R), the difference region (D), and the matched region (M) of a substrate–product pair. RDM patterns express chemical transformations in terms of the 68 atom types present in KEGG[2], describing the atomic environment of carbon, nitrogen, oxygen, sulfur, phosphorus, and "other" atoms in chemical compounds. It is possible that a given R number is associated to several reaction classes. For example, a reaction $A + B \rightarrow C + D$ in which $C$ and $D$ are obtained from $A$ and $B$, respectively, would be associated to two reaction classes, describing the RDM patterns of the transformations $A \rightarrow C$ and $B \rightarrow D$, respectively.

- KEGG ENZYME (since 1995) is an implementation of the enzyme nomenclature (EC numbers) produced by IUBMB/IUPAC. Although the only official enzyme nomenclature, the EC number hierarchy presents some important limitations (see section I.2.2). EC numbers were the primary identifiers for the construction of pathways from complete genomes until 2002, when they were replaced with K numbers (see KEGG ORTHOLOGY above).

*EC number*

**Health information** The health information category of the KEGG knowledge base, collectively referred to as KEGG MEDICUS, contains information on drugs that are currently approved in Japan, Europe, and the United States (KEGG DRUG), on drug interactions (KEGG DGROUP[3]), on health-related substances (KEGG ENVIRON), on human diseases, viewed as perturbed molecular networks (KEGG

[2]KEGG atom types: https://www.kegg.jp/kegg/reaction/KCF.html

[3]KEGG DGROUP generalizes drug compounds in the same way that KEGG ORTHOLOGY generalizes pathway maps. It contains groups of drugs that are structurally and functionally related.

DISEASE), and on variations of perturbant agents in human diseases (KEGG NETWORK and KEGG VARIANT).

## 2.3 KGML format

In order to facilitate the exchange of pathway maps, KEGG uses KGML, its own XML-based markup language [KGML]. In a nutshell, the information contained within a KGML file describes how reactions and compounds are linked. Using the KGML terminology, reactions are linked through *relations*, whereas compounds are linked through *reactions*.

`relation`
`reaction`

Before going into more technical detail, a simple example will be examined. Consider the three reactions in Figure III.2, where the two reactions represented as green rectangles are performed and are connected through the compound `C04882`. The reaction represented as a white rectangle is performed by other species but is absent from the organism for which the pathway fragment is shown. The relevant part of the corresponding KGML file is presented in the listing in Figure III.3.

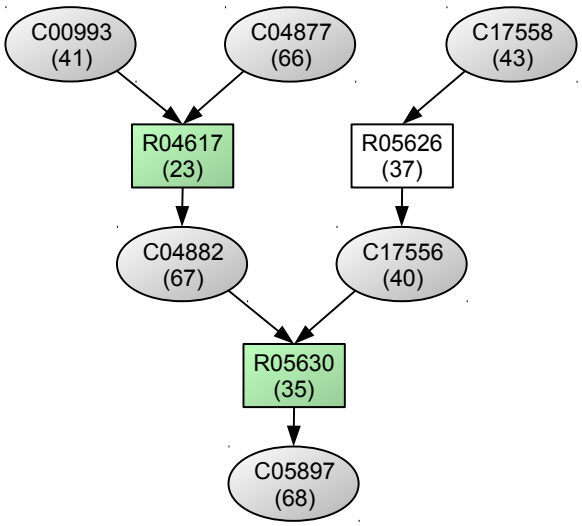



**Figure III.2 Portion of the peptidoglycan biosynthesis pathway in *Escherichia coli* (map `eco00550`).** Reactions (R numbers) are represented as rectangles, which are green if the reaction is performed or white if the reaction is absent. Compounds (C numbers) are represented as gray ovals. Numbers in parentheses represent unique internal KGML identifiers.

Entities such as compounds and reactions have a unique *numerical identifier* in KGML files (such as 23 for `R04617`). Although reactions and compounds have unique identifiers respectively called R and C numbers, an internal relabeling is preferable for multiplicity issues, such as the same compound appearing more than

`KGML id`

once in a given pathway map, or a reaction being the result of the chemical transformations described by more than a single R number.

In the KGML fragment in Figure III.3, the `relation` (text in green) between the entries with the identifiers 23 (`R04617`) and 35 (`R05630`) is established through the `compound` with the identifier 67 (`C04882`). The compounds are linked through `reaction` tags (text in red). For example, `R04617` is an `irreversible` reaction transforming the `substrate` compounds with the identifiers 66 (`C04877`) and 41 (`C00993`) into a `product` compound with the identifier 67 (`C04882`).

```
<!--  Creation date: Apr 10, 2017 09:30:15 +0900 (GMT+9)  -->
<pathway name="path:eco00550" org="eco" number="00550"
         title="Peptidoglycan biosynthesis"...>
  ...
  <entry id="41" name="cpd:C00993" type="compound"...>...</entry>
  <entry id="66" name="cpd:C04877" type="compound"...>...</entry>
  <entry id="43" name="cpd:C17558" type="compound"...>...</entry>
  <entry id="67" name="cpd:C04882" type="compound"...>...</entry>
  <entry id="40" name="cpd:C17556" type="compound"...>...</entry>
  <entry id="68" name="cpd:C05897" type="compound"...>...</entry>
  <entry id="23" name="eco:b0086" type="gene" reaction="rn:R04617"...>...</entry>
  <entry id="37" name="ko:K00887" type="ortholog" reaction="rn:R05626"...>...
  </entry>
  <entry id="35" name="eco:b0087" type="gene" reaction="rn:R05630"...>...</entry>
  ...
  <relation entry1="23" entry2="35" type="ECrel">
    <subtype name="compound" value="67"/>
  </relation>
  ...
  <reaction id="23" name="rn:R04617" type="irreversible">
    <substrate id="66" name="cpd:C04877"/>
    <substrate id="41" name="cpd:C00993"/>
    <product id="67" name="cpd:C04882"/>
  </reaction>
  <reaction id="35" name="rn:R05630" type="irreversible">
    <substrate id="40" name="cpd:C17556"/>
    <substrate id="67" name="cpd:C04882"/>
    <product id="68" name="cpd:C05897"/>
  </reaction>
  ...
</pathway>
```

**Figure III.3 Portion of the KGML file corresponding to the pathway fragment in Figure III.2.** Only the relevant parts of the KGML file corresponding to map `eco00550` have been extracted (omissions are indicated by ellipses).

In XML terminology, `<pathway>...</pathway>`, `<entry>...</entry>`, `<relation>...</relation>`, and `<reaction>...</reaction>` are called *XML elements*, with `<pathway>...</pathway>` being the *XML root*. An XML element can option-

ally have one or more *attributes*, representing name–value pairs. For the first `entry` in the listing in Figure III.3, `id`, `name`, and `type` are *attribute names*, their values being `"41"`, `"cpd:C00993"`, and `"compound"`, respectively.

The main elements in the KGML specification are `entry`, `relation`, and `reaction`, all three being direct child elements of the `pathway` root. As illustrated in the previous example, **`relation`** elements describe how reactions are connected, while **`reaction`** elements describe how compounds are linked through reactions.

*relation*

*reaction*

Biological entities such as reactions and compounds are specified in KGML using **`entry`** elements. There are several possible entry types, for example `ortholog` for KO groups, `enzyme` for enzymes, `reaction` for reactions, `gene` for gene products, or `compound` for chemical compounds including glycans. All the entry types among the ones listed, with the exception of `compound`, can have one or several *names*. This means, for example, that an entry of type `gene` can be a list of gene identifiers, which corresponds to the case where several gene products are involved in a reaction.

*entry*

Entries corresponding to reactions receive an additional attribute called **`reaction`**, not to be confounded with `reaction` elements which are shown in red in Figure III.3. For organism-specific pathway maps (such as `eco00550`), there are only two possible values for the `type` of an entry with a `reaction` attribute:

- **`gene`** if the reaction is present for the species in question, in which case the `name` attribute is a gene identifier, or a group of gene identifiers.
- **`ortholog`** if the species does not perform the reaction, in which case the `name` attribute is a KO group designated by a K number (see KEGG ORTHOLOGY in section 2.2), or a list of K numbers.

*gene*

*ortholog*

For example, consider the entries with the identifiers 23 and 37 in the listing in Figure III.3. The first of the two (identifier 23) describes a `reaction` of type `gene`, meaning that the reaction `R04617` (upper green rectangle in Figure III.2) is present in *E. coli* and is performed by the product of gene *b0086*. The other entry (identifier 37) describes a reaction of type `ortholog`, meaning that the reaction `R05626` (white rectangle in Figure III.2) belonging to the KO group `K00887` is absent from *E. coli*.

There are currently four types of KEGG pathway maps that can be retrieved in KGML format:

- Organism-specific pathway maps, linked to KEGG GENES entries. The pathway map prefix is the three- or four-letter organism code [KEGG Organisms], e.g. `eco00550` for the peptidoglycan biosynthesis pathway of *Escherichia coli* K-12 MG1655.

- Reference pathway maps linked to KEGG ORTHOLOGY entries (K numbers), with prefix `ko` (e.g. `ko00550`).
- Reference pathway maps linked to KEGG REACTION entries (R numbers), with prefix `rn` (e.g. `rn00550`).
- Reference pathway maps linked to KEGG ENZYME entries (EC numbers), with prefix `ec` (e.g. `ec00550`).

The four types of pathway maps can be retrieved directly through the KEGG web site, via the KEGG FTP, or using the KEGG REST API (see section 2.4 below).

## 2.4 REST API

The KEGG REST API offers the possibility to extract information from various KEGG databases and to download pathway maps in KGML format (see section 2.3 above) using HTTP requests. In case of a HTTP status code from the KEGG server indicating success, the response to the REST query is usually a text file (or an image). This renders KEGG particularly well-suited for programming purposes. The `CoMetGeNe` pipeline developed over the course of this thesis (see Chapter VI) makes extensive use of the KEGG REST API.

Below are a few basic usage examples of the KEGG REST API.

*find* **Example 1.** `http://rest.kegg.jp/find/genome/escherichia+coli`

This query lists all strains of *Escherichia coli* present in KEGG GENOME.

*list* **Example 2.** `http:/rest.kegg.jp/list/eco`

This query lists all genes of *Escherichia coli* K-12 MG1655 (`eco`).

**Example 3.** `http://rest.kegg.jp/get/eco:b0086+eco:b0087`

This query retrieves two entries in KEGG GENES for *E. coli* (`eco`), corresponding to the genes *b0086* and *b0087*. Gene name, definition, position on the chromosome,
*get* strand, and sequence data are available, among others. KEGG REST `get` queries usually accept up to 10 parameters that are concatenated with a plus sign, as in this example.

**Example 4.** `http:/rest.kegg.jp/list/pathway/eco`

This query lists all pathway maps of *E. coli* (`eco`).

**Example 5.** `http://rest.kegg.jp/get/eco00550/kgml`

This query retrieves the peptidoglycan biosynthesis pathway map of *E. coli* (`eco00550`) in KGML format.

**Example 6.** http://rest.kegg.jp/get/ko00550/kgml

This query retrieves the reference KEGG ORTHOLOGY peptidoglycan biosynthesis pathway map (ko00550) in KGML format.

**Example 7.** http://rest.kegg.jp/get/C04882/mol

This query retrieves the compound C04882 as an MDL Molfile (mol), a common format used in chemoinformatics.

**Example 8.** http://rest.kegg.jp/list/reaction

This query lists all entries in the KEGG REACTION database.

**Example 9.** http://rest.kegg.jp/get/R04617

This query retrieves the R04617 entry in the KEGG REACTION database.

The previous examples have shown how the KEGG REST API can be used to search (find and list) and retrieve entries (get) from a given KEGG database. A powerful feature of this API is the link command, allowing to cross-reference two databases. Some of the capabilities of the link command are demonstrated in the examples below.

*link*

**Example 10.** http://rest.kegg.jp/link/eco/eco00550

This query lists all *E. coli* (eco) genes whose products catalyze reactions in the peptidoglycan biosynthesis pathway (eco00550).

**Example 11.** http://rest.kegg.jp/link/pathway/eco:b0086

This query lists all *E. coli* (eco) pathways in which the product of gene *b0086* is involved.

**Example 12.** http://rest.kegg.jp/link/reaction/rn00550

This query lists all R numbers that are present in the reference peptidoglycan biosynthesis pathway map linked to KEGG REACTION (rn00550).

**Example 13.** http://rest.kegg.jp/link/reaction/enzyme

This query retrieves the associations between EC numbers and R numbers. Note that the correspondence is not one-to-one, as there exist R numbers with zero, one, or more associated EC numbers, as well as EC numbers with zero, one, or more associated R numbers.

**Example 14.** http://rest.kegg.jp/link/reaction/ko

This query retrieves the associations between K numbers and R numbers. Note that the correspondence is not one-to-one, as there exist R numbers with zero, one, or more associated K numbers, as well as K numbers with zero, one, or more associated R numbers.

## 3 Consistency issues in KEGG

This section describes actual, as well as potential inconsistencies in KEGG, as a result of several problems that I encountered while using the knowledge base from a programming perspective (see sections 2.3 and 2.4 above).

In late October 2017 I contacted KEGG through the feedback form on the website to report I had found (by chance) that three reactions for *Streptococcus pneumoniae* ST556 (`snd`), which were present in the pentose and glucuronate interconversions pathway (`snd00040`), were marked as absent in the ascorbate and aldarate metabolism pathway (`snd00053`). Almost three weeks later, they let me know the maps had been corrected. As it turned out, not only had they corrected the maps, but they had also suppressed three orthology (KO) groups in the process. This exchange prompted me to investigate orthology associations more carefully and to screen for different types of inconsistencies in KEGG in a systematic manner.

Previous works have already reported mostly annotation-related errors in public databases, including KEGG. Schnoes *et al.* [2009] reported varying degrees of functional misannotation in enzyme superfamilies and showed that the most frequent error was functional overprediction. Green and Karp [2005] examined the case of genes annotated with partial EC numbers (such as EC 4.2.1.-). At the time of the study, orthology (KO) groups had already been introduced in KEGG. Recall from section 2.2 that KO groups are sequence similarity groups to which individual genes are assigned. The genes in a given KO group would thus catalyze all the reactions associated to that group. The authors deduced that EC numbers also played a role in the establishment of KO groups, in the sense that all the genes in a given KO group $k$ were considered to be involved in all the reactions being assigned an EC number associated to $k$. While this reasoning is likely correct for complete EC numbers, in the case of partial EC numbers it leads to the incorrect functional characterization of the genes in such a KO group.

*KO group*

The issue reported by Green and Karp [2005] has since been addressed by KEGG in several ways. First, KO groups are continuously updated and refined [Kanehisa *et al.*, 2015]. Second, the assignment of reactions to KO groups takes place on a much finer scale than before[4]. Currently, 95% of all KO groups present in reference KEGG ORTHOLOGY maps have at most 5 associated reactions[5]. At the same time, reactions may be assigned to several KO groups, seemingly connected by logical *and* operators. A species with no associated gene to one of the KO groups assigned

[4] Personal observation

[5] A reaction is annotated with the K numbers of the genes that are involved in the reaction (see also section 3.2).

to a particular reaction is considered as not performing that reaction. Third, partial EC numbers are handled differently than complete EC numbers[6]. As more sequences are found to be similar with other genes assigned to a KO group, this group is divided into smaller and more specific similarity subgroups. Thus, even if a partial EC number is associated to a KO group, it does not define it entirely. Orthology groups with only one associated EC number being a partial EC number are generally associated to a very low number of reactions.

The types of consistency problems I draw attention to in this section, although related to the internal structure of the KEGG knowledge base, affect both its exploitation from a programming point of view, as well as its standard usage as a biological encyclopedia linking structure to function.

It is not always straightforward to decide whether apparent discrepancies observed between different KEGG databases are actual problems that should be signaled, or just particular instances of complex resource cross-linking. In the former case I report *actual* consistency issues (see section 3.1), whereas discrepancies in the latter case are reported as *potential* consistency issues (see section 3.2) warranting closer investigation.

Due to the intricate nature of KEGG and to an incomplete comprehension of the in-house procedures that are used during the maintenance and update of the knowledge base despite a thorough literature review (see section 2.2), the consistency issues signaled herein should only be considered preliminary results. Since this contribution is quite recent, the necessary steps for contacting the KEGG maintainers regarding the consistency issues reported herein will be taken in the near future.

## 3.1 Disconnected reactions in KEGG ORTHOLOGY maps

**Description** Certain reactions in KEGG pathway maps are disconnected from the rest of the pathway at the KGML level. This applies to both organism-specific and reference maps (`ko`, `rn`, and `ec`).

**Example** Figures III.4 and III.5 below illustrate the problem. Figure III.4 shows the same pathway as Figure III.2 in which an additional reaction, `R01150`, is present. Its product is the compound `C00993`, one of the two substrates of the reaction `R04617`. When accessing the pathway map online[7], no problem is apparent, as re-

[6] Personal observation

[7] The peptidoglycan biosynthesis pathway in *E. coli* is available at the following address: https://www.genome.jp/kegg-bin/show_pathway?eco00550. In June 2018, the latest version of this pathway map dates from April 10, 2017.

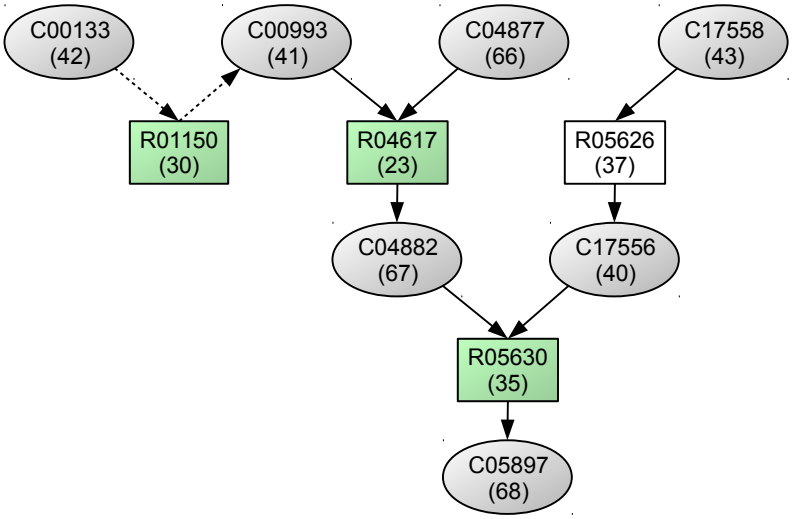



**Figure III.4 Portion of the peptidoglycan biosynthesis pathway in *Escherichia coli* (map `eco00550`).** Dashed arrows represent missing `reaction` KGML elements. See Figure III.2 for other explanations.

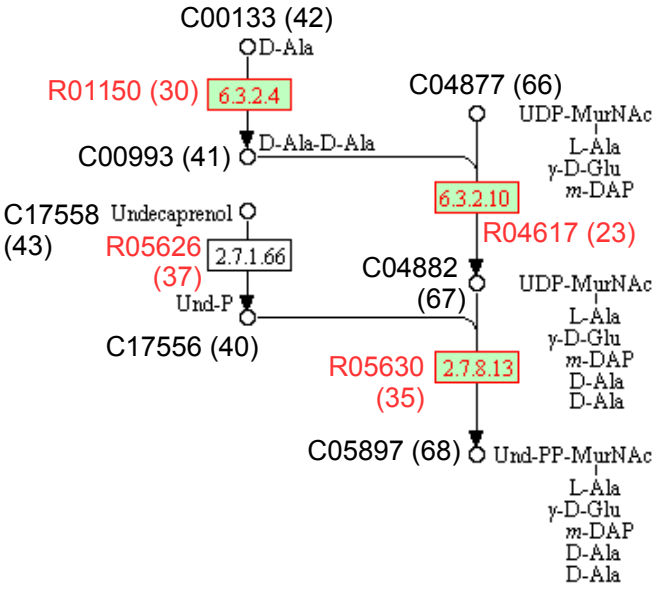



**Figure III.5 Portion of the peptidoglycan biosynthesis pathway in *E. coli*.** Adapted from KEGG PATHWAY, map `eco00550` (April 10, 2017 version). This is a fragment of Figure VII.13. Reactions are labeled in red with their corresponding R numbers. Compounds are labeled in black with their corresponding C numbers. Reactions and compounds are the same as those in Figure III.4. Numbers in parentheses represent KGML identifiers.

actions R01150 and R04617 are seemingly connected (Figure III.5). The problem, however, lies in the KGML file, where the compound C00993 does not perform the link between the two reactions (signaled by dashed arrows in Figure III.4).

In section 2.3 it was explained that KGML entries are linked by relation elements in the case of reactions (R numbers), and by reaction elements in the case of compounds (C numbers). The KGML file corresponding to the pathway map eco00550 used in this example has the correct relation linking reactions R01150 and R04617:

```
<relation entry1="30" entry2="23" type="ECrel">
  <subtype name="compound" value="41"/>
</relation>
```

However, the following reaction KGML element is missing:

```
<reaction id="30" name="rn:R01150" type="irreversible">
  <substrate id="42" name="cpd:C00133"/>
  <product id="41" name="cpd:C00993"/>
</reaction>
```

Interestingly, all peptidoglycan biosynthesis pathways that were examined manifested the problem of reaction R01150 being disconnected. The reason is the fact that all pathway maps in KEGG are drawn with KEGG ORTHOLOGY (KO) groups [Kanehisa, 2017]. It would appear then that this type of error initially took place at the level of reference KO maps and was then propagated to all species-specific maps (as well as rn and ec reference maps).

**Approach** The approach proposed in order to identify occurrences of disconnected reactions in KEGG pathway maps is to simply test for all KO maps whether entries with a reaction attribute have a corresponding reaction element in the same KGML file.

**Results** This approach allowed to determine all occurrences of reactions with missing links in KO maps. A total of 255 such instances were found (see Appendix A.1), of which 174 (68%) occur in metabolic pathways excluding global and overview maps (i.e., occur in maps with identifiers less than 01100).

**Discussion** The particular anomaly presented in Figures III.4 and III.5 is the reason for which the case study in section VII.5 uses data extracted from KEGG in September 2016 instead of June 2018. The previous version of the map eco00550

(May 28, 2015) had the reaction R01150 correctly linked to the rest of the pathway. (As will be shown in Chapter VII, this allowed to identify both trails in Figure VII.13, whereas with the current version only the trail highlighted in yellow is found.)

Although disconnected reactions do not affect KEGG users browsing through the website, they have a deep effect when handling the KGML files from a programming perspective. Methods relying on the information provided in KGML files lead to the construction of incomplete graphs in the case of reactions that are disconnected from the rest of the pathway. These graph-based models are typically used in bioinformatic studies, where accurate biological information and representation is critical for correct comprehension and interpretation.

### 3.2 Inconsistent reactions between pathway maps

**Description** Reactions may belong to more than a single pathway map. In some cases, they are marked as present in one pathway map, but absent from another, for the same species.

**Example** Figures III.6 and III.7 below show the same reaction, R02773 (in red), being present in the first pathway map (green rectangle) and absent in the second one (white rectangle). Both pathway maps belong to the same species, *Actinoplanes sp.* SE50/110 (ase). The difference is that in both cases the reaction R02773 is associated to different KO groups.

*KO group*

As explained in section 2.2, KO or orthology groups are similarity groups (in terms of amino acid sequence) to which genes are assigned when a new genome is annotated in KEGG. In the case of enzyme-coding genes, their products are involved in the catalysis of one or several reactions. The KO groups of enzyme-coding genes are therefore transferred to the catalyzed reactions, which explains why reactions are also associated to K numbers.

In the pathway in Figure III.6, the reaction R02773 is performed by the product of gene *ACPL_3667*. The K number associated to this gene is K20428 and corresponds to orthologs of gene *acbV* (see light blue circle in Figure III.8). In Figure III.7, however, the same reaction is associated to two other K numbers, K13308 and K21328, that have no associated genes in *Actinoplanes sp.*

Groups K20428 and K13308 are only associated to the reaction R02773 and both share the same definition, with K13308 corresponding to *desI* and *eryCIV* orthologs (see light and dark blue circles in Figure III.8).

The third KO group, K21328, contains orthologs of *calS13* and *atmS13* genes.

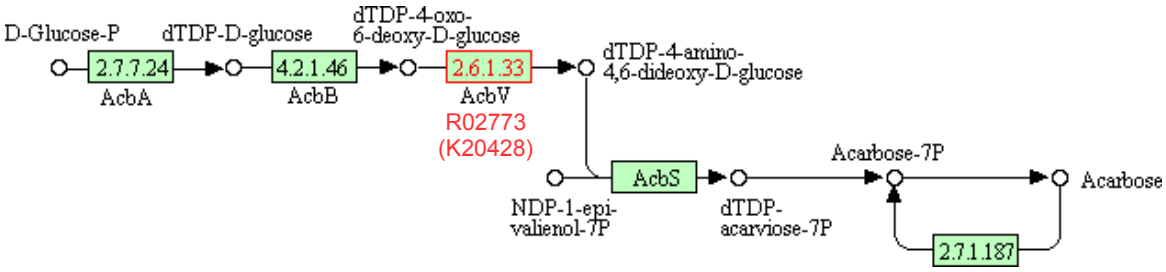



**Figure III.6 Portion of the acarbose and validamycin biosynthesis pathway in *Actinoplanes sp.* SE50/110.** Adapted from KEGG PATHWAY, map `ase00525` (January 24, 2017 version). The reaction `R02773` (in red) is present and is associated to the KO group `K20428`.

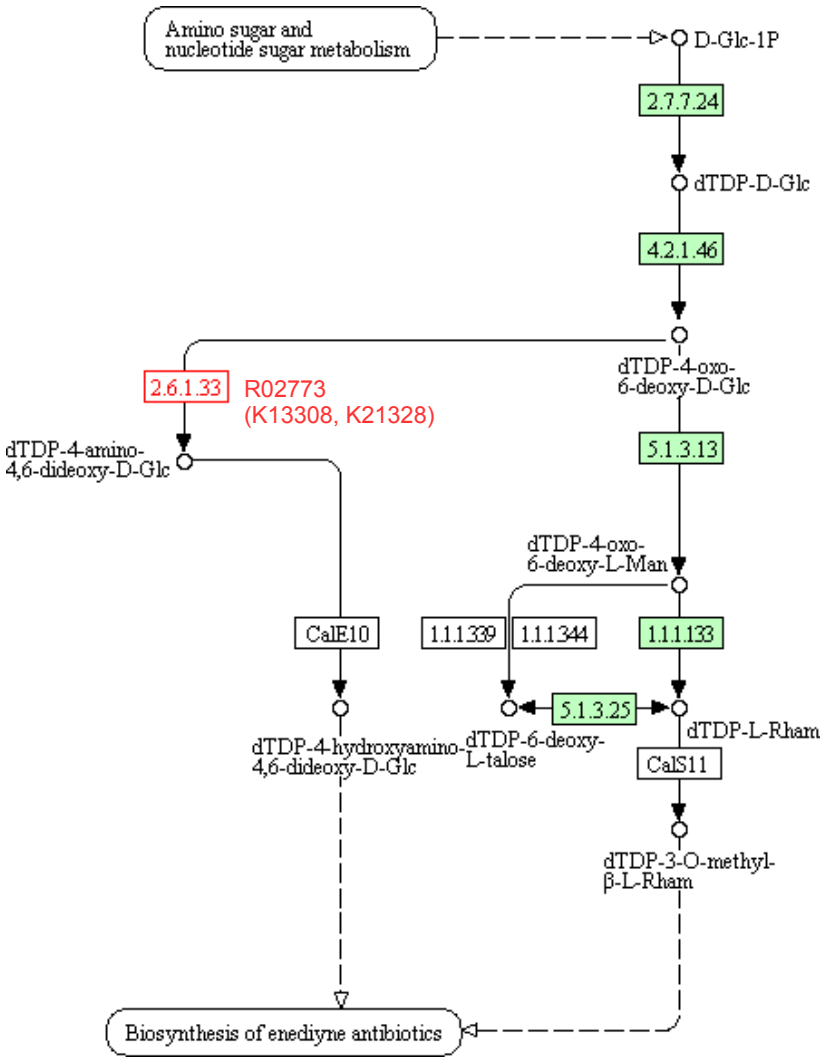



**Figure III.7 Portion of the polyketide sugar unit biosynthesis pathway in *Actinoplanes sp.*** Adapted from KEGG PATHWAY, map `ase00523` (March 3, 2017 version). The reaction `R02773` (in red) is absent and is associated to the KO groups `K13308` and `K21328`.

*calS13* has the same functional definition as *acbV* (for K20428), as well as *desI* and *eryCIV* (for K13308), whereas *atmS13* has a different functional definition (as shown by Singh *et al.* [2015] for another member of the Actinobacteria phylum). This third KO group (K21328) is associated to R02773 as well as another reaction, R11475 (see magenta circle in Figure III.8), which is absent from *Actinoplanes sp*[8].

Figure III.8 below summarizes the definition and composition of these three orthology groups.

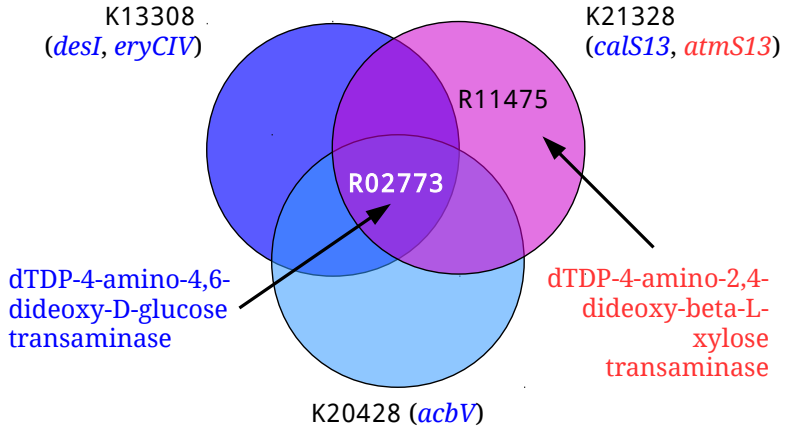



**Figure III.8 Definition of three orthology groups.** The KO groups K20428, K13308, and K21328 are represented by light blue, dark blue, and magenta circles, respectively. Each group contains orthologous genes of a given type, indicated in parentheses. The definition of each KO group corresponds to the ortholog color(s) in parentheses and is indicated by arrows. In the case of K21328, its bifunctional definition is indicated by a blue and red color code (hence magenta for the KO group as a whole).

Since the same reaction R02773 belongs to both pathway maps (Figures III.6 and III.7), marking it as present in one map but absent from the other appears inconsistent. In terms of orthology, only orthologous sequences of dTDP-4-amino-4,6-dideoxy-D-glucose transaminases seem to be involved in this reaction. Two alternative explanations can be proposed for the reaction R02773 being inconsistently marked as present or absent between the two pathway maps ase00525 (Figure III.6) and ase00523 (Figure III.7), respectively. The first possible explanation regards an overspecialization in the assignment of gene *ACPL_3667* to KO groups. In effect, this gene (an *acbV* ortholog) was uniquely assigned to group K20428. Perhaps that an overly strict assignment procedure overlooked its inclusion in one of the two other KO groups. The second possible explanation also regards an overspecialization, this time in terms of establishment of KO groups. When the reference map for

[8]Reaction R11475 is defined as being a dTDP-4-amino-2,4-dideoxy-beta-L-xylose:2-oxoglutarate aminotransferase.

the pathway in Figure III.7 was drawn, perhaps only orthologs in groups K13308 and K21328 were known as being able to catalyze R02773. If none of these explanations holds true, then it is the very concept of reaction (in terms of R numbers) that would have to be reexamined.

**Approach** In order to identify reactions being treated inconsistently between pathway maps in terms of presence and absence (such as reaction R02773 in the preceding example), it is necessary to define the properties of such reactions. The approach proposed here is a two-step process in which the first step identifies candidate reactions according to a very broad definition, whereas the second step allows to select reactions (among the proposed candidates) with respect to stricter criteria.

**Definition III.1.** Let $r$ be a reaction that may appear in $n$ pathway maps $P_r = \{p_1, \ldots, p_n\}$. Let $S$ be a species having a subset of pathway maps $P'_r \subseteq P_r$ such that $|P'_r| \geq 2$. If there exist two pathway maps $p_i$ and $p_j$ in $P'_r$ for species $S$ such that $r$ is present in $p_i$ but absent from $p_j$, the reaction $r$ is referred to as *potentially inconsistent* with respect to $S$.

*potentially inconsistent reaction*

The first step of the proposed approach consists in applying definition III.1 above for selecting candidate reactions that are present in some pathway maps but absent from others. Explanations may be found for such potentially inconsistent reactions when examining the associated EC numbers and KO groups. Several strategies may be used in the second step of the approach by formulating different definitions of (actually) inconsistent reactions, two of which are presented below.

Definition III.2 below was used to identify the reaction R02773 (see Figures III.6 and III.7) by selecting reactions (among potentially inconsistent candidates) with the same EC numbers but disjoint sets of associated KO groups.

**Definition III.2.** Let $r$ be a potentially inconsistent reaction for a given species $S$ and let $p_i$ and $p_j$ be two pathway maps of $S$ such that $r$ is present in $p_i$ but absent from $p_j$. Let $E_i$ and $E_j$ be the sets of EC numbers associated to $r$ in the pathway maps $p_i$ and $p_j$, respectively. Likewise, let $K_i$ and $K_j$ be the sets of K numbers (KO groups) associated to $r$ in the pathway maps $p_i$ and $p_j$, respectively. Then the reaction $r$ is referred to as *(actually) inconsistent* with respect to $S$ if $E_i = E_j$ and $K_i \cap K_j = \varnothing$.

*inconsistent reaction*

Definition III.3 below selects reactions (among potentially inconsistent candidates) with disjoint sets of associated EC numbers and KO groups.

**Definition III.3.** Let $r$ be a potentially inconsistent reaction for a given species $S$ and let $p_i$ and $p_j$ be two pathway maps of $S$ such that $r$ is present in $p_i$ but absent

from $p_j$. Let $E_i$ and $E_j$ be the sets of EC numbers associated to $r$ in the pathway maps $p_i$ and $p_j$, respectively. Likewise, let $K_i$ and $K_j$ be the sets of K numbers (KO groups) associated to $r$ in the pathway maps $p_i$ and $p_j$, respectively. Then the reaction $r$ is referred to as *(actually) inconsistent* with respect to $S$ if $E_i \cap E_j = \varnothing$ and $K_i \cap K_j = \varnothing$.

*inconsistent reaction*

**Results** The approach presented above was applied on all organism-specific pathway maps with the exception of global and overview maps (i.e., maps whose identifiers are greater than or equal to `01100`), for all species present in KEGG GENOME. Organism-specific maps were retrieved from the KEGG FTP in November 2017. Associations between K numbers, R numbers, and EC numbers were retrieved via the KEGG REST API using the `link` command (see section 2.4, examples 13 and 14). A total of 377,421 organism-specific pathway maps belonging to 5,084 species were analyzed.

Table III.1 below summarizes the findings using definition III.1 for potentially inconsistent reactions, and either definition III.2 or III.3 for inconsistent reactions. The table shows the number and percentage of species with at least one occurrence of (potentially) inconsistent reactions, as well as the total number of occurrences, the number of occurrences per species, and the number of unique (potentially) inconsistent reactions among all occurrences.

| | Definition III.1 | Definition III.2 | Definition III.3 |
|---|---|---|---|
| Nb. species affected | 4,910 | 1,515 | 4,762 |
| % species affected | 96.58% | 29.8% | 93.67% |
| Total nb. occurrences | 37,188 | 2,146 | 18,553 |
| Nb. occurrences/species | 7.31 | 0.42 | 3.9 |
| Unique reactions | 99 | 17* | 41† |

**Table III.1 Summary of (potentially) inconsistent reactions.** All organism-specific pathway maps present in KEGG in November 2017 were analyzed, with the exception of global and overview maps. The second column (Definition III.1) corresponds to potentially inconsistent reactions. The third (Definition III.2) and fourth (Definition III.3) columns correspond to inconsistent reactions, according to the respective definitions.

* The first occurrence (in any species) is listed in Appendix A.2.1.

† The first occurrence (in any species) is listed in Appendix A.2.2.

**Discussion** Reactions present in some organism-specific pathway maps but absent from others are disrupting for the biological comprehension of the metabolic pathways in which they are featured.

Identifying such reactions requires a specific definition taking into account the associations between reactions (R numbers), EC numbers, and KEGG orthology groups (K numbers). The definitions III.2 or III.3 presented in this section provide examples of the type of criteria that might be used. Other definitions of inconsistent reactions can be envisaged. For example, potentially inconsistent reactions can be screened in terms of intersections, such as selecting reactions with non disjoint sets of associated EC numbers and K numbers.

Once a working definition for inconsistently treated reactions (in terms of presence and absence from pathway maps of a given species) has been chosen, it will probably be necessary to examine orthology group definition and composition in detail (see Figures III.6, III.7, and III.8 for an example) and consult the existing literature in order to evaluate the correctness of qualifying a given reaction as inconsistent. This process would undoubtedly be made easier if more details concerning the procedure of assigning genes and reactions to KO groups were common knowledge.

## 4 Concluding remarks

This chapter gave an overview of KEGG (Kyoto Encyclopedia of Genes and Genomes), an important knowledge base whose main objective is linking sequence data to biological function. Over the years, KEGG has expanded to include various genomic, chemical, and health information, although the primary focus remains systems information, with a particular emphasis on pathway maps.

The overview detailed the role of the main KEGG databases and their interconnections. In addition, the KGML format, used for the exchange of KEGG pathway maps, was described. The KEGG REST API, a valuable resource for searching, retrieving, and cross-linking data from different KEGG databases, was also briefly commented.

Through extensive usage of the KEGG resource, certain anomalies related to the overall consistency of the knowledge base become apparent. This chapter presented two such consistency issues, the first one affecting the network information conveyed through KGML files, and the second one concerning the intricate relationship between reactions, orthology groups, and EC numbers.

The incursion through KEGG is significant in the context of this thesis, because the knowledge base served as the primary source for the biological data used in Chapter VII to illustrate the methods proposed in Chapters IV and V through the bioinformatics tool specifically developed for this purpose and introduced in Chapter VI.

# IV

# Trail finding

# 1 Introduction

*trail finding*

This chapter presents an exact method of graph mining in the context of heterogeneous biological networks. The method is termed *trail finding* and its purpose is to identify relevant patterns of biological interest. More specifically, it is used to detect metabolic and genomic patterns, defined as maximal trails of reactions catalyzed by products of neighboring genes. Recall from section II.4 that trails allow to capture cycles in metabolic pathways, while taking into account reaction directionality and guaranteeing that reactions in the trails correspond to actual metabolic routes.

We first explain the model used to represent biological networks. Next, we formally state the problem in graph theory terms. An overview of the trail finding method is given, followed by the detailed description of the algorithms that we propose. Finally, an improvement rendering the method more flexible is discussed.

In this thesis, trail finding focuses on metabolic pathways and genomic context. The method is however adaptable to other types of biological networks, requiring only minor adjustments to the model.

# 2 Model

A non-spontaneous metabolic reaction is catalyzed by one or several enzymes. A given enzyme can be encoded by one or several genes. Metabolic pathways and genomic context are regarded as networks of reactions and genes, respectively. The relation between metabolic pathways and their encoding genes is represented using a classical model involving two graphs and a correspondence function:

(i) Genomes (viewed as gene networks) are represented as undirected graphs with protein-coding genes for vertices (Figure IV.1a). Two protein-coding genes are connected by an edge if they are neighbors on the same strand of the same chromosome. For example, genes $Y$ and $Z$ are neighbors, therefore they are linked by the edge $(Y, Z)$.

(ii) Metabolic pathways are represented as directed graphs with reactions for vertices (Figure IV.1b). An arc leading from a reaction $r_i$ to another reaction $r_j$ signifies that $r_i$ produces a metabolite that is a substrate for $r_j$. For example, the arc $(r_4, r_9)$ translates the fact that the product of $r_4$ is a substrate for $r_9$. In order to avoid linking the same reaction $r$ to different parts of the pathway in case $r$ is present more than once, a relabeling of reactions with unique identifiers can be used. When using KEGG (Kyoto Encyclopedia of Genes and Genomes) [Kanehisa *et al.*, 2016a], the unique labels take the form of KGML

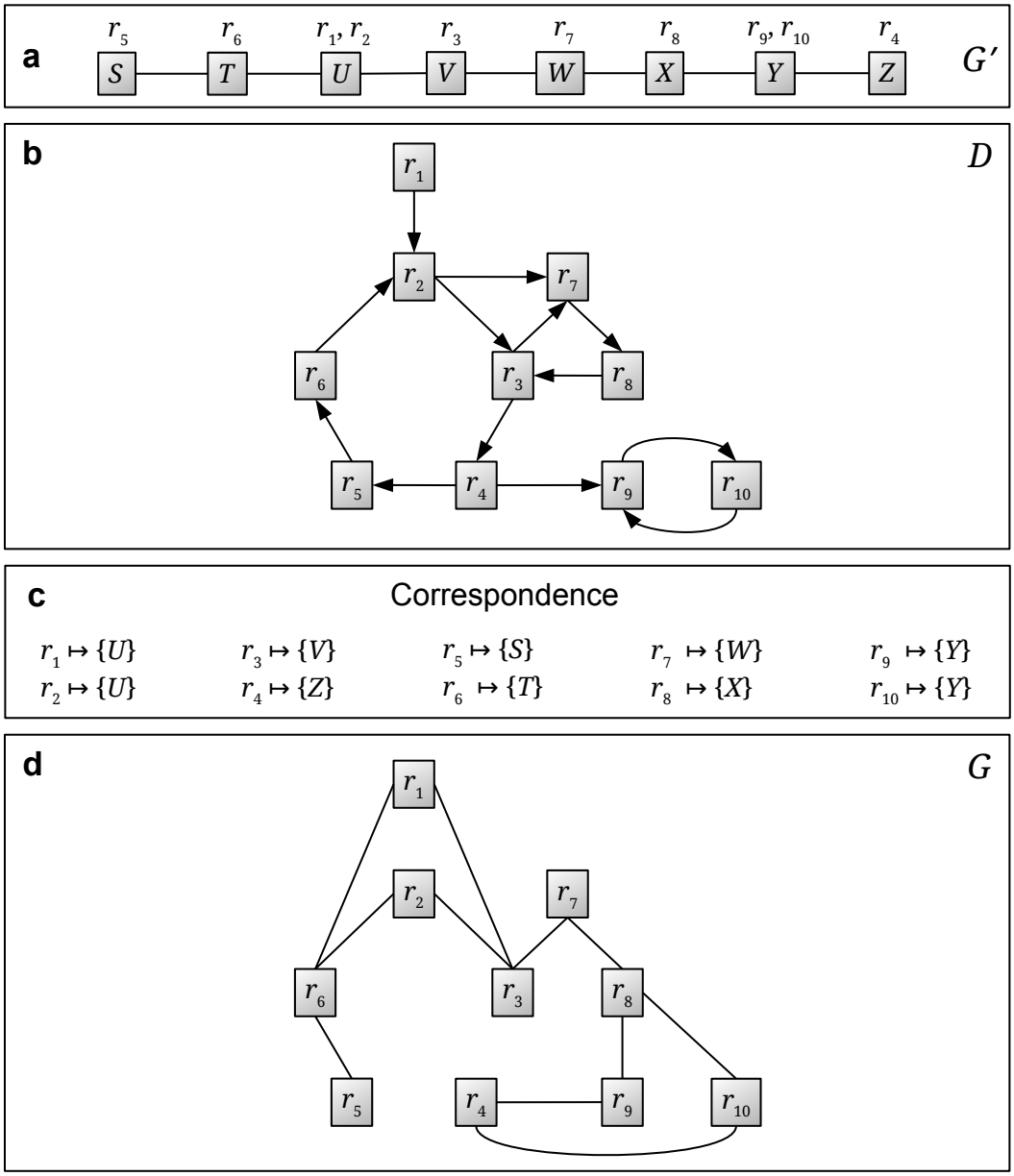



**Figure IV.1 Illustration of the model used to represent metabolic pathways and genomic context. (a)** The undirected graph $G'$ represents the gene order of a given species. The reactions that gene products catalyze are indicated above each gene. **(b)** The directed graph $D$ represents a metabolic pathway of the same species as in (a). **(c)** The correspondence between reactions in $D$ and genes in $G'$. **(d)** $G$ is an undirected graph with the same vertex set as $D$ built using the correspondence between reactions and genes. $G$ represents gene neighborhood with respect to the reactions that the gene products catalyze.

identifiers (see section III.2.3).

(iii) For a given species $S$, the relation between one of its metabolic pathways and its genome takes the form of a correspondence function associating genes to reactions: for any given reaction $r$, the correspondence function returns the set of genes of species $S$ that encode enzymes catalyzing reaction $r$ (e.g. in Figure IV.1c, $Z$ is the unique gene that encodes an enzyme catalyzing reaction $r_4$). This information can be found in knowledge bases such as KEGG which, for a given species, contains information on its metabolic pathways, the reactions that the species performs, and the genes associated to these reactions (see Chapter III).

The trail finding method requires two input graphs possessing the same vertex set. Thus, an additional undirected graph is constructed as described by Mohamed-Babou [2012] such that it reflects gene neighborhood with respect to the reactions that the gene products catalyze (Figure IV.1d). The additional graph links two reactions $r_i$ and $r_j$ with an edge if at least one of the genes encoding an enzyme involved in reaction $r_i$ is adjacent to a gene encoding an enzyme involved in reaction $r_j$. For example, genes $X$ and $Y$ are neighbors in $G'$ (Figure IV.1a). Gene $X$ codes for an enzyme involved in reaction $r_8$, while gene $Y$ codes for an enzyme involved in reactions $r_9$ and $r_{10}$. To reflect adjacency between genes $X$ and $Y$ in $G'$, reactions $r_8$ and $r_9$, respectively $r_8$ and $r_{10}$, are linked by an edge in $G$ (Figure IV.1d).

## 3 Problem formulation

*biological patterns*

Given a metabolic pathway and the genomic context for the same species, the patterns of biological interest captured by the trail finding method are maximal chains of reactions being catalyzed by products of neighboring genes.

The problem was initially formulated under the name of LONGEST SUPPORTED PATH (LSP) by Fertin *et al.* [2015], as follows:

> LONGEST SUPPORTED PATH (LSP)
> **Input:** A directed graph $D = (V, A)$, an undirected graph $G = (V, E)$.
> **Output:** A longest path $P$ in $D$ such that $G[V(P)]$ is connected.

The solution for LSP is thus a longest path in the directed graph $D$ inducing a connected subgraph in the undirected graph $G$. The vast majority of metabolic pathways, however, exhibit cycles (e.g. reversible reactions). Taking cycles into account requires that solutions be authorized to contain repeated vertices. Recall

from Chapter II that, contrary to paths, trails can contain repeated vertices, but not repeated arcs (see definitions II.9, II.10, and II.11).

We now define the concept of span and propose a new problem formulation that provides trails as solutions, instead of paths.

**Definition IV.1.** The *span* of a trail $T$ represents the number of distinct vertices in $T$. *span*

*Example.* If $T$ is the trail $(r_2, r_3, r_7, r_8, r_3, r_4)$ in Figure IV.1b, then the span of $T$ is 5, because vertex $r_3$ is repeated.

MAXIMUM SPAN SUPPORTED TRAIL (MaSST)
**Input:** A directed graph $D = (V, A)$, an undirected graph $G = (V, E)$, an arc $(u, v)$ in $D$.
**Output:** A trail of maximum span $T$ in $D$ passing through $(u, v)$ such that $G[V(T)]$ is connected.

Whereas LSP produces a path for every graph $D$, MaSST outputs trails of maximum span passing through arcs of $D$ if the vertex sets of these trails induce connected subgraphs in $G$. The choice of producing a trail for every arc in $D$ is deliberate in order to ensure that more than a single trail is retrieved per graph (see below).

For example, for graphs $D$ (Figure IV.1b) and $G$ (Figure IV.1d) and the arc $(r_1, r_2)$, MaSST outputs one of the two following trails of span 8: $t_1 = (r_1, r_2, r_3, r_7, r_8, r_3, r_4, r_9, r_{10})$ or $t_2 = (r_1, r_2, r_7, r_8, r_3, r_4, r_9, r_{10})$. Both $t_1$ and $t_2$ start with $r_1$ because this is the only way to include the arc $(r_1, r_2)$ in the trails. For any other arc in $D$, the output of MaSST is either of the two following trails of span 9: $t_3 = (r_5, r_6, r_2, r_3, r_7, r_8, r_3, r_4, r_9, r_{10})$ or $t_4 = (r_5, r_6, r_2, r_7, r_8, r_3, r_4, r_9, r_{10})$. (Alternatively, $t_3$ and $t_4$ may start with vertex $r_4$ followed by $r_5$, which does not change their span.) Since $t_3$ and $t_4$ must include arcs in $D$ other than $(r_1, r_2)$, maximizing their span implies passing through as many reactions as possible. For the graph $D$ (Figure IV.1b), the only way to accomplish this is if both trails start with vertex $r_5$ (or with vertex $r_4$ followed by $r_5$). Capturing trails of span 8 (either $t_1$ or $t_2$) as well as trails of span 9 (either $t_3$ or $t_4$) reveals that the genes involved in these partly overlapping trails are all neighbors on the chromosome. If only trails of span 9 were returned (either $t_3$ or $t_4$), the information that $r_1$ is catalyzed by the product of a gene in the same genomic context as the others would have been lost.

For practical purposes (see section 4.2 below), MaSST is solved by using the line graph of $D$ (see definition II.8).

**Definition IV.2.** Let $D$ be a directed graph and $L(D)$ be its line graph. Let $P = (a_1, a_2, \ldots, a_k)$ be a path in $L(D)$, where $a_i = (t_{i-1}, t_i), 1 \leq i \leq k$, are arcs in $D$. The *trail in D corresponding to P*, denoted $L^{-1}(P)$, is the trail $T = (t_0, t_1, t_2, \ldots, t_{k-1}, t_k)$. If $P$ is an empty path, then $L^{-1}(P)$ is an empty trail.

*corresponding trail*

*Example.* If $P$ is the path $((r_3, r_7), (r_7, r_8), (r_8, r_3))$ in the line graph $L(D)$ in Figure IV.2b, then $L^{-1}(P)$ is the trail $(r_3, r_7, r_8, r_3)$ in the directed graph $D$ in Figure IV.2a.

A problem formulation equivalent to MaSST, MAXIMUM SPAN SUPPORTED CORRESPONDING TRAIL (MaSSCoT), is further proposed:

> MAXIMUM SPAN SUPPORTED CORRESPONDING TRAIL (MaSSCoT)
> **Input:** A directed graph $D = (V, A)$, an undirected graph $G = (V, E)$, an arc $(u, v)$ in $D$.
> **Output:** A path $P$ in the line graph of $D$ such that $L^{-1}(P)$ has maximum span, passes through $(u, v)$, and $G[V(L^{-1}(P))]$ is connected.

LSP has been shown to be NP-hard in the general case [Fertin *et al.*, 2015]. The authors have shown that LSP remains NP-hard even if $D$ is acyclic and $G$ is a tree with diameter 4. We prove below that MaSST and MaSSCoT are also NP-hard in the general case. The proof makes use of MAXIMUM SPAN TRAIL (MaST), a problem formulation closely related to MaSST:

> MAXIMUM SPAN TRAIL (MaST)
> **Input:** A directed graph $D = (V, A)$, an undirected graph $G = (V, E)$.
> **Output:** A trail of maximum span $T$ in $D$ such that $G[V(T)]$ is connected.

**Proposition IV.1.** MaST is NP-hard.

*Proof.* LSP is NP-hard even if $D$ is acyclic and $G$ is a tree with diameter 4 [Fertin *et al.*, 2015]. Now, if $D$ is acyclic, then LSP and MaST have exactly the same solution. Thus MaST is NP-hard (even if $D$ is acyclic and $G$ is a tree with diameter 4). □

**Corollary IV.1 (of proposition IV.1).** MaSST is NP-hard.

*Proof.* Suppose that MaSST is polynomially tractable. Then, by applying it on all arcs of $D$ in turn, MaST can be solved in polynomial time as well. But MaST is NP-hard (proposition IV.1). □

**Lemma IV.1.** Let $D = (V, A)$ be a directed graph and $L(D)$ be its line graph. Let $P = (a_1, a_2, \ldots, a_k)$ be a path in $L(D)$, where $a_i = (t_{i-1}, t_i)$ for $i \in \{1, \ldots, k\}$ are edges in $D$. Then the unique vertex sequence $(t_0, t_1, t_2, \ldots, t_{k-1}, t_k)$ associated to $P$ is a trail in $D$.

*Proof.* By construction of $P$, the vertex sequence $T = (t_0, t_1, t_2, \ldots, t_{k-1}, t_k)$ is unique and is a walk in $D$. Since $P$ has no repeated vertices, $T$ contains no repeated arcs. $T$ is therefore a trail in $D$. □

**Corollary IV.2 (of proposition IV.1).** MaSSCoT is NP-hard.

*Proof.* A path in the line graph of a directed graph $D$ is a trail in $D$ (lemma IV.1). Given the MaSST problem formulation, let $T = L^{-1}(P)$ be the trail in $D$ corresponding to $P$. Then $T$ is the solution to the MaSSCoT problem formulation. MaSST and MaSSCoT are therefore equivalent. Since MaSST is NP-hard (corollary IV.1), it follows that MaSSCoT is also NP-hard. □

# 4 General approach

This section presents an overview of the trail finding method, before introducing the actual algorithm in section 5. The trail finding method solves MaSST with an exact approach that uses the MaSSCoT problem formulation internally. Trail finding starts off by reducing the input graphs $D$ and $G$ while ensuring no solution is lost (see 4.1). Next, trail finding in $D$ is replaced by path finding in the line graph of $D$ involving minimal path enumeration (see 4.2). Finally, partial paths enumerated in the line graph of $D$ are concatenated in order to produce a solution for the MaSST problem (see 4.3).

## 4.1 Graph reduction

Fertin *et al.* [2012] introduced the concept of a cover set of a path and proposed an algorithm to compute it. Briefly, given two graphs $D$ (directed) and $G$ (undirected) on the same vertex set $U$, as well as a path $P$ in $D$, the cover set of $P$ with respect to $D$ and $G$ is a maximal subset of $U$ containing only vertices that might extend $P$ into a path $P'$ such that $G[V(P')]$ and the undirected graph underlying $D[V(P')]$ stay connected. *cover set*

We have shown that, for a given arc $(u, v)$ in $D$, reducing the input graphs $D$ and $G$ to the cover set $S$ of $(u, v)$ and feeding these reduced graphs $D[S]$ and $G[S]$ as input to MaSST and MaSSCoT yields the same solution as providing $D$ and $G$ as

input. (The proof is provided in Appendix B.) In other words, graphs $D$ and $G$ can be reduced without loss of solutions.

## 4.2 Path finding in the line graph

The problem of trail enumeration in the directed graph $D$ modeling a metabolic pathway is naturally solved by performing path enumeration in the line graph $L(D)$. In other words, MaSST is solved using the MaSSCoT problem formulation. In effect, to a given path in $L(D)$ corresponds a unique trail in $D$, as shown in lemma IV.1. Figure IV.2b shows the line graph corresponding to the directed graph in Figure IV.2a.

Path enumeration in $L(D)$ is restricted to a minimum using the following three-step process:

1. The strongly connected components (SCCs, see definition II.7) of $L(D)$ and its condensation graph are computed, where a condensation graph results from replacing every SCC with a single vertex (Figure IV.2). Note that condensation graphs are acyclic by definition.

*entry point*
*exit point*

2. For every SCC of $L(D)$, vertices acting as entry points from predecessor SCCs, as well as vertices acting as exit points to successor SCCs are determined. For example, in Figure IV.2b, vertices $(r_2, r_3)$ and $(r_2, r_7)$ are entry points for the SCC $S_2$ when coming from the predecessor SCC $S_1$. Vertex $(r_3, r_4)$ in $S_2$ is an exit point when heading to the SCC $S_3$. In $S_3$, vertex $(r_4, r_9)$ is both an entry point when coming from predecessor $S_2$ and an exit point when heading to successor $S_4$. $S_1$ has no predecessor SCCs and $S_4$ has no successor SCCs.
3. For every SCC $X$ of $L(D)$, path enumeration is performed only between strictly necessary source and destination vertices, as follows: (i) if $X$ has at least one predecessor and one successor SCC, then paths are enumerated between all possible pairs of entry and exit points for these SCCs; (ii) if $X$ has no predecessor and at least one successor SCC, then paths are enumerated between every vertex of $X$ and exit points towards the successor SCC(s); (iii) if $X$ has at least one predecessor and no successor SCC, then paths are enumerated between entry points from the predecessor SCC(s) and every vertex of $X$; (iv) only if $X$ has no predecessor and no successor SCCs, paths are enumerated between every pair of vertices of $X$.

The paths obtained through step 3 above are evaluated in terms of span and length of their corresponding trails in $D$ and the best candidate paths among them

*best partial paths*

are retained. They are referred to as *best partial paths*. Among two partial paths $P$

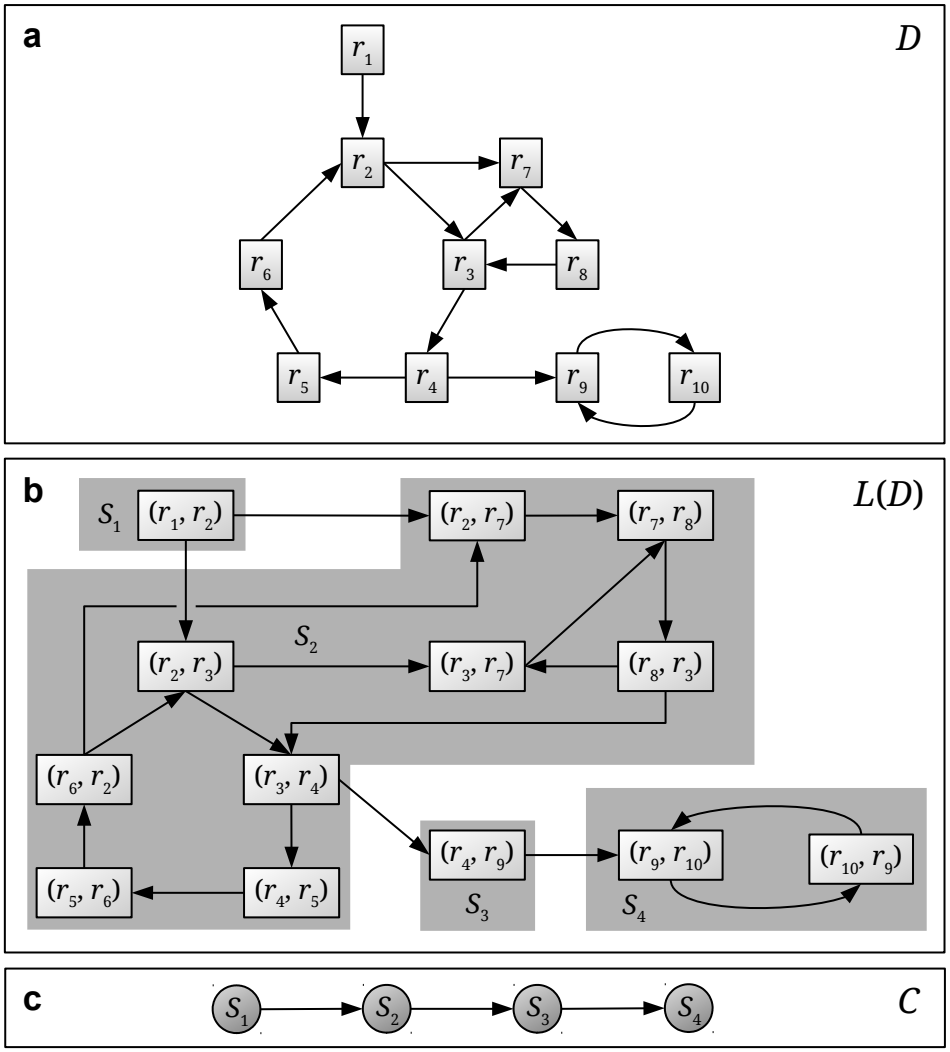



**Figure IV.2 Example of a directed graph, its line graph, and the condensation graph of the line graph.** **(a)** A directed graph $D$. **(b)** The directed graph $L(D)$ represents the line graph of the directed graph $D$ in (a). By definition of the line graph, vertices of $L(D)$ are arcs of $D$. Strongly connected components (SCCs) of $L(D)$ are shaded in gray and are assigned a label $S_i$. **(c)** The directed graph $C$ represents the condensation graph of the line graph $L(D)$ in (b), obtained by replacing every SCC of $L(D)$ with a single vertex.

and $P'$ in a SCC of the line graph, the best one is either the path with a corresponding trail in $D$ of maximum span or, in case both $L^{-1}(P)$ and $L^{-1}(P')$ have equal span, the path with a corresponding trail in $D$ of minimum length.

The interest of the maximum span and minimum length criteria is illustrated by the following example. Let $P = ((r_9, r_{10}))$ and $P' = ((r_9, r_{10}), (r_{10}, r_9))$ be two paths in the SCC $S_4$ of $L(D)$ in Figure IV.2b. Their corresponding trails in $D$ are $L^{-1}(P) = (r_9, r_{10})$ and $L^{-1}(P') = (r_9, r_{10}, r_9)$, respectively. While both $L^{-1}(P)$ and

$L^{-1}(P')$ have span 2, $L^{-1}(P)$ has length 1, whereas $L^{-1}(P')$ has length 2. Since $P'$ contributes no new information to the trail it yields, it is preferable to retain $P$ among the two candidate paths in the line graph.

### 4.3 Concatenation of partial paths

Every path in the condensation graph $C$ of $L(D)$ is "translated" into one or several paths in $L(D)$ by concatenating best partial paths in SCCs of $L(D)$. Let $q_i$ and $q_j$ be two consecutive vertices of a path $Q$ in $C$ of length at least 1. Let $S_i$ and $S_j$ be the SCCs in $L(D)$ corresponding to $q_i$ and $q_j$, respectively. Then $Q$ has more than one corresponding path in $L(D)$ if $S_i$ has at least two exit points when heading to the successor SCC $S_j$, or if $S_j$ has at least two entry points when coming from the predecessor SCC $S_i$.

For example, there are two paths in $L(D)$ (Figure IV.2b) corresponding to path $Q_1 = (S_1, S_2, S_3, S_4)$ in $C$ (Figure IV.2c): $P_1 = ((r_1, r_2), (r_2, r_7), (r_7, r_8), (r_8, r_3), (r_3, r_4), (r_4, r_9), (r_9, r_{10}))$ and $P_1' = ((r_1, r_2), (r_2, r_3), (r_3, r_7), (r_7, r_8), (r_8, r_3), (r_3, r_4), (r_4, r_9), (r_9, r_{10}))$. The corresponding trails in $D$ (Figure IV.2a) are $L^{-1}(P_1) = (r_1, r_2, r_7, r_8, r_3, r_4, r_9, r_{10})$ and $L^{-1}(P_1') = (r_1, r_2, r_3, r_7, r_8, r_3, r_4, r_9, r_{10})$, both with span 8. Note that if $P_1$ (respectively $P_1'$) passed through vertices $(r_4, r_5)$, $(r_5, r_6)$, and $(r_6, r_2)$ in $L(D)$ (Figure IV.2b), then $P_1$ (respectively $P_1'$) would be a trail in $L(D)$ instead of a path, which is not allowed. In effect, since path $Q_1$ in $C$ starts with $S_1$, the vertex $(r_1, r_2)$ in $L(D)$ needs to be the first in any path in $L(D)$ corresponding to $Q_1$ in $C$. Furthermore, paths in $S_2$ need to start with either vertex $(r_2, r_3)$ or $(r_2, r_7)$, as these are the only two vertices following $(r_1, r_2)$. Moreover, the path in $S_2$ would need to end with vertex $(r_3, r_4)$, as it is the only vertex leading to $S_3$, the third vertex of path $Q_1$ in $C$. It follows then that any walk in $L(D)$ starting with vertex $(r_1, r_2)$ and passing through vertices $(r_4, r_5)$, $(r_5, r_6)$, and $(r_6, r_2)$ would necessarily pass through vertex $(r_3, r_4)$ twice, which means it would be a trail instead of a path.

In order to determine the solution to the MaSST problem, all paths in the condensation graph of $L(D)$ are enumerated such that their corresponding paths in $L(D)$ contain the SCC possessing the input arc $(u, v)$ as vertex. If a path in $L(D)$ obtained by concatenating best partial paths contains vertex $(u, v)$, it is then evaluated in terms of its span by comparing it to the best current solution and by updating the current solution if necessary.

For example, let $(u, v) = (r_2, r_7)$ in the graph $D$ in Figure IV.2a. After translating path $Q_1 = (S_1, S_2, S_3, S_4)$ in $C$ to a path in $L(D)$, the best current solution $P_1$ has span 8 as shown above. Note that $P_1$ is a solution since it includes the input arc $(r_2, r_7)$ and $G[V(L^{-1}(P_1))]$ is connected (see Figure IV.1d). Now, suppose the path $Q_2 =$

($S_2$, $S_3$, $S_4$) in $C$ (Figure IV.2c) is enumerated. There is one corresponding path in $L(D)$ (Figure IV.2b) passing through $(r_2, r_7)$, obtained by concatenation of best partial paths in $S_2$, $S_3$, and $S_4$. The best partial path in $S_2$ ends in vertex $(r_3, r_4)$ (which is an exit point when heading toward $S_3$) and may start with any vertex in $S_2$, provided the corresponding trail in $D$ has maximum span. The path in $L(D)$ corresponding to $Q_2$ is therefore $P_2 = ((r_5, r_6), (r_6, r_2), (r_2, r_7), (r_7, r_8), (r_8, r_3), (r_3, r_4), (r_4, r_9), (r_9, r_{10}))$, for which $L^{-1}(P_2)$ has span 9. When $P_1$ and $P_2$ are compared, the best current solution now becomes $P_2$ because $P_2$ has maximum span and because $G[V(L^{-1}(P_2))]$ is connected (see Figure IV.1d).

## 5 Algorithm HNET

The trail finding method is embodied by HNET (***H**eterogeneous **Net**work mining*), an algorithm that solves the MaSST problem using the MaSSCoT formulation internally (Algorithm 1).

**Algorithm 1** HNET($D$, $G$, $(u, v)$)

**Input:** A directed graph $D = (V, A)$, an undirected graph $G = (V, E)$, an arc $(u, v)$ in $D$.

**Output:** A trail $T$ of maximum span in $D$ that includes $(u, v)$ such that $G[V(T)]$ is connected, or $\varnothing$ if no such trail exists.

```
D, G ← GraphReduction(D, G, (u, v))
L(D) ← LineGraph(D)
C ← CondensationGraph(L(D))
𝒜 ← AccessPoints(L(D))
ℬ ← PartialPaths(L(D), 𝒜)
Let a ∈ V(C) such that the SCC of L(D) corresponding to a contains (u, v)
P ← ∅
for all s ∈ V(C) do
    for all t ∈ V(C) do
        for all Q in EnumeratePaths(C, s, t) do
            if a ∈ V(Q) then
                for all P′ in FindPaths(L(D), Q, ℬ) do
                    if (u, v) ∈ V(P′) and G[V(L^{-1}(P′))] is connected then
                        P ← BestPath(P, P′)
return L^{-1}(P)
```

Unlike the heuristic solution introduced in Fertin *et al.* [2012] to the LSP problem, HNET is an exact algorithm as it is guaranteed to return a trail of maximum span in $D$ passing through the input arc $(u, v)$ such that $G[V(L^{-1}(P))]$ is connected, if such a trail exists. However, HNET is not exhaustive with respect to the more

general problem of determining *all* trails passing through $(u, v)$ such that these trails induce connected subgraphs in $G$. Since HNET solves the MaSST problem, it means that if several trails of maximum span pass through a given arc $(u, v)$ in $D$, then only one such trail is reported as solution.

The bottleneck in HNET is path enumeration at line 5, which can be exponential with respect to the size of the graph (recall that MaSST and MaSSCoT are NP-hard). The worst-case scenario occurs when all possible paths are enumerated between all pairs of vertices in a SCC. This scenario occurs in two distinct cases which nonetheless rarely arise in practice. The first case is that of SCCs of $D$ that are completely disconnected from the rest of the graph. Sequences of reactions in metabolic pathways that are completely disconnected from the rest of the pathway are typically very short (on average, shorter than 2 reactions for the 50 species in Table VII.1) and therefore not limiting for exhaustive path enumeration. The second case is when $D$ is strongly connected, corresponding to the infrequent situation in which a chain of reactions leads from any reaction $r_i$ to any other reaction $r_j$ of a given metabolic pathway, and vice versa.

An overview of algorithm HNET is given in subsection 5.1, followed by detailed descriptions of the sub-algorithms used by HNET (subsections 5.2, 5.3, and 5.4).

## 5.1 Overview

In the following, assume: $D = (V, A)$ is a directed graph; $(u, v)$, an arc in $D$; $G = (V, E)$, an undirected graph; $L(D)$, the line graph of $D$; and $C$, the condensation graph of $L(D)$.

Algorithm GRAPHREDUCTION (line 1) returns the reduced graphs $D$ and $G$ (see section 4.1 above). For graphs $D$ and $G$ in Figure IV.1 (panels b and d), the reduced and unreduced graphs are the same. LINEGRAPH (line 2) returns the line graph $L(D)$ of the reduced input graph (for example, $L(D)$ in Figure IV.2b is the line graph of graph $D$ in Figure IV.2a). CONDENSATIONGRAPH (line 3) returns the condensation graph of $L(D)$, i.e. the directed acyclic graph obtained by replacing every SCC of $L(D)$ by a single vertex (for example, graph $C$ in Figure IV.2c is the condensation graph of graph $L(D)$ in Figure IV.2b).

Algorithm ACCESSPOINTS (see section 5.2 below) determines entry and exit points for every SCC $X$ of $L(D)$, from SCCs that are predecessors of $X$ and toward SCCs that are successors of $X$ (see section 4.2 above, step 2). This information is stored in a data structure $\mathcal{A}$ that the algorithm returns at line 4. Algorithm PARTIALPATHS (see section 5.3 below) then uses $\mathcal{A}$ to compute best paths in every SCC $X$ of $L(D)$ (in terms of span of their corresponding trails in $D$) between all possible

pairs of source and destination vertices. Source vertices are entry points from predecessor SCCs if $X$ has predecessors, and vertices of $X$ otherwise. Conversely, destination vertices are exit points to successor SCCs if $X$ has successors, and vertices of $X$ otherwise. These paths, called *best partial paths*, are stored in a data structure $\mathcal{B}$ that the algorithm returns at line 5 (see section 4.2 above, step 3).

*best partial paths*

At line 6, HNET determines $a$, the vertex of $C$ whose corresponding SCC in $L(D)$ contains the input arc $(u, v)$ as a vertex. Next, all possible paths in $C$ are enumerated (lines 8–14) and, if they contain vertex $a$, the corresponding paths in $L(D)$ are obtained by concatenation of best partial paths stored in $\mathcal{B}$. The best current solution is updated accordingly.

A path $P$ in $L(D)$ qualifies as a best current solution if the trail in $D$ corresponding to $P$, $L^{-1}(P)$, fulfills the following conditions:

*best current solution*

(i) It contains the input arc $(u, v)$;
(ii) Its vertex set induces a connected subgraph in $G$;
(iii) It has maximum span so far.

Algorithm ENUMERATEPATHS at line 10 returns all paths starting with vertex $s$ and ending in vertex $t$ in the condensation graph $C$. If $s$ and $t$ are the same vertex, the algorithm returns either one. Algorithm FINDPATHS (see section 5.4 below) at line 12 returns all paths in $L(D)$ corresponding to path $Q$ in the condensation graph $C$, obtained by concatenation of best partial paths stored in $\mathcal{B}$. Given two paths in $L(D)$, algorithm BESTPATH at line 14 returns the best current path, i.e. the path whose corresponding trail in $D$ has greater span than the other (see section 4.3 above).

Finally, HNET returns the trail in $D$ corresponding to a best solution (line 15), effectively solving the MaSST problem. An additional consistency check is performed as detailed in Mohamed-Babou [2012] to ensure that the trail $L^{-1}(P)$ also "makes sense" when passing from $G$ to the initial graph $G'$ (see section 2 and Figure IV.1 above). We check whether vertices in $G'$ corresponding to the vertex set of the trail are connected. Note that Mohamed-Babou [2012] describes the check that needs to be performed to ensure consistency between a solution returned by the heuristic implementation of LSP (see section 3) and an additional graph $G$, constructed as detailed in section 2.

## 5.2 Algorithm ACCESSPOINTS

For every strongly connected component (SCC, see definition II.7) $X$ of $L(D)$, all its entry and exit points are determined with respect to possible predecessor and

successor SCCs of $X$ in $L(D)$ using algorithm 2 (AccessPoints) presented below. First, predecessor and successor SCCs, as well as entry and exit points, are formally defined below.

*predecessor SCC*

**Definition IV.3.** Let $X$ be a SCC in a directed graph $D$. A SCC $W$ in $D$ is a *predecessor* of $X$ if there exists an arc $(w, x)$ from a vertex $w$ in $W$ to a vertex $x$ in $X$. In this case, $x$ is an *entry point in* $X$ when coming from the predecessor SCC $W$.

*entry point*

*Example.* The SCC $S_1$ has no predecessor SCC in the line graph $L(D)$ in Figure IV.2b. Vertices $(r_2, r_3)$ and $(r_2, r_7)$ are entry points for the SCC $S_2$ when coming from the predecessor SCC $S_1$.

*successor SCC*

**Definition IV.4.** Let $X$ be a SCC in a directed graph $D$. A SCC $Y$ in $D$ is a *successor* of $X$ if there exists an arc $(x, y)$ from a vertex $x$ in $X$ to a vertex $y$ in $Y$. In this case, $x$ is an *exit point in* $X$ when heading toward the successor SCC $Y$.

*exit point*

*Example.* The SCC $S_4$ has no successor SCC in the line graph $L(D)$ in Figure IV.2b. Vertex $(r_3, r_4)$ is an exit point for the SCC $S_2$ when heading toward the successor SCC $S_3$. In $S_3$, vertex $(r_4, r_9)$ is both an entry point when coming from the predecessor SCC $S_2$ and an exit point when heading toward the successor SCC $S_4$.

*access points*

*Remark.* Entry and exit points for a given SCC are collectively referred to as *access points*.

Algorithm 2 below returns entry and exit point information for every SCC in $D$ (defined hereafter).

For every SCC $X$ of the input graph $D$, algorithm 2 (AccessPoints) determines entry point information (lines 3–8, see definition IV.5) and exit point information (lines 9–14, see definition IV.6). Access point information for $X$ is stored in the data structure $\mathcal{A}$ at line 15. If $X$ has no predecessor SCC, then all vertices of $X$ are implicitly considered to be entry points for $X$ (line 8), with the predecessor of $X$ being undefined ($\perp$). Similarly, if $X$ has no successor SCC, then all vertices of $X$ are implicitly considered to be exit points for $X$ (line 14), with the successor of $X$ being undefined ($\perp$).

**Definition IV.5.** Let $X$ be a SCC in a directed graph $D$. The set of all tuples $(W, \{x_1, \ldots, x_k\})$ where $x_i$ is an entry point of $X$ when coming from a predecessor SCC $W$ represents *entry point information* for $X$ and is denoted $I_X$.

**Definition IV.6.** Let $X$ be a SCC in a directed graph $D$. The set of all tuples $(Y, \{x'_1, \ldots, x'_{k'}\})$ where $x'_i$ is an exit point of $X$ when heading toward a successor SCC $Y$ represents *exit point information* for $X$ and is denoted $O_X$.

**Algorithm 2** ACCESSPOINTS($D$)

**Input:** A directed graph $D = (V, A)$.
**Output:** A data structure $\mathcal{A}$ storing entry and exit point information for every SCC in $D$.

```
1:  A ← ∅
2:  for all X in StronglyConnectedComponents(D) do
3:      if there exists at least one predecessor of X in D then
4:          I_X ← ∅
5:          for all W predecessor of X do
6:              I_X ← I_X ∪ (W, {x ∈ X | (w, x) ∈ A, w ∈ W})
7:      else
8:          I_X ← (⊥, {x ∈ X})
9:      if there exists at least one successor of X in D then
10:         O_X ← ∅
11:         for all Y successor of X do
12:             O_X ← O_X ∪ (Y, {x ∈ X | (x, y) ∈ A, y ∈ Y})
13:     else
14:         O_X ← (⊥, {x ∈ X})
15:     A[X] ← (I_X, O_X)
16: return A
```

Suppose algorithm 2 takes as input the graph $L(D)$ in Figure IV.3. Then:

- For the SCC $S_1$, entry point information is $I_{S_1} = (\bot, \{(1,2),(2,1)\})$, since $S_1$ has no predecessor SCC. Exit point information for $S_1$ is $O_{S_1} = (S_2, \{(2,1)\}) \cup (S_3, \{(1,2)\})$, meaning that $(2,1)$ is an exit point for $S_1$ when heading to the successor SCC $S_2$ and that $(1,2)$ is an exit point for $S_1$ when heading to $S_3$.
- For the SCC $S_2$, entry point information is $I_{S_2} = (S_1, \{(1,5)\})$ and exit point information is $O_{S_2} = (S_4, \{(1,5)\})$, meaning that vertex $(1,5)$ is both an entry point for $S_2$ when coming from the predecessor SCC $S_1$, and an exit point when heading to the successor SCC $S_4$.
- Similarly, for the SCC $S_3$, entry point information is $I_{S_3} = (S_1, \{(2,3)\})$ and exit point information is $O_{S_3} = (S_4, \{(2,3)\})$.
- For the SCC $S_4$, entry point information is $I_{S_4} = (S_3, \{(3,4)\}) \cup (S_2, \{(5,3)\})$ and exit point information is $O_{S_4} = (\bot, \{(3,4),(4,5),(5,3)\})$, since $S_4$ has no successor SCC.

## 5.3 Algorithm PARTIALPATHS

Algorithm 3 (PARTIALPATHS) below determines best partial paths for every SCC of the line graph $L(D)$ between all possible pairs of access points in $X$. Partial paths and best partial paths are formally defined below.

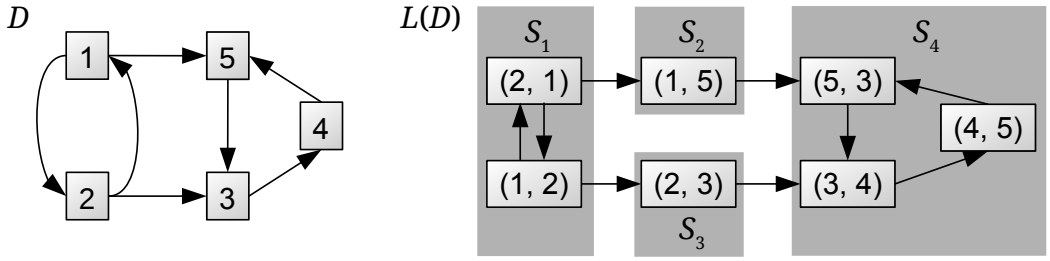



**Figure IV.3 Example of a directed graph and its line graph.** $L(D)$ is the line graph of $D$. SCCs of $L(D)$ are shaded in gray and are assigned a label $S_i$.

*partial path*

**Definition IV.7.** A *partial path* in a SCC $X$ of a line graph $L(D)$ is a path in $X$ between an entry point $s$ when coming from a predecessor SCC $W$ and an exit point $t$ when heading toward a successor SCC $Y$.

*Remark.* If $X$ has no predecessor SCCs, any vertex in $X$ can play the role of an entry point for a partial path in $X$. Similarly, if $X$ has no successor SCCs, any vertex in $X$ can play the role of an exit point for a partial path in $X$.

*best partial path*

**Definition IV.8.** Given a set $\mathcal{P}$ of partial paths in a SCC $X$ of a line graph $L(D)$ between an entry point $s$ when coming from a predecessor SCC $W$ and an exit point $t$ when heading toward a successor SCC $Y$, a *best partial path* is a path $P$ in $\mathcal{P}$ such that its corresponding trail in $D$ has maximum span, or maximum span and minimum length, in case several paths in $\mathcal{P}$ have corresponding trails in $D$ of maximum span.

**Algorithm 3** PARTIALPATHS($L(D)$, $\mathcal{A}$)

**Input:** A line graph $L(D)$ and a data structure $\mathcal{A}$ storing entry and exit point information for every SCC in $L(D)$ as specified in algorithm ACCESSPOINTS.
**Output:** A data structure $\mathcal{B}$ storing a best partial path for every quintuplet $(X, W, Y, s, t)$ (see text).

```
1: B ← ∅
2: for all X in StronglyConnectedComponents(L(D)) do
3:     (I_X, O_X) ← A[X]
4:     for all (W, s) ∈ I_X do
5:         for all (Y, t) ∈ O_X do
6:             for all P in EnumeratePaths(X, s, t) do
7:                 EvaluatePath(P, X, W, Y, s, t, B)
8: return B
```

Algorithm 3 (PARTIALPATHS) retrieves access point information for every SCC $X$ of $L(D)$ at line 3. All paths in every SCC of $L(D)$ between every entry point

$s$ when coming from a predecessor SCC $W$ and every exit point $t$ when heading toward a successor SCC $Y$ are enumerated at lines 4–6, and a best partial path among them is retained at line 7. (Recall that algorithm ENUMERATEPATHS at line 6 takes as input a graph $G$ and two vertices $s$ and $t$ in $V(G)$, and returns all paths in $G$ between $s$ and $t$. If $s$ and $t$ are the same vertex, then algorithm ENUMERATEPATHS returns either one.)

Algorithm 3 (PARTIALPATHS) returns a data structure $\mathcal{B}$ (line 8) storing best partial paths for every SCC of $L(D)$. $\mathcal{B}$ is an array indexed by quintuplets of the form $(X, W, Y, s, t)$. For each such quintuplet, $\mathcal{B}$ stores a best partial path $P$ in the SCC $X$ of $L(D)$ between vertices $s$ and $t$, when coming from a predecessor SCC $W$ and when heading toward a successor SCC $Y$. Initially empty, $\mathcal{B}$ is updated using algorithm EVALUATEPATH (see below) such that, when execution of algorithm PARTIALPATHS is finished, $\mathcal{B}$ contains only best partial paths for every quintuplet $(X, W, Y, s, t)$. If the predecessor $W$ is undefined ($\perp$), then $s$ is also undefined and $\mathcal{B}$ stores a best partial path in $X$ from any vertex in $X$ to $t$. Similarly, if $Y$ is undefined, then $t$ is also undefined and $\mathcal{B}$ stores a best partial path in $X$ from $s$ to any vertex of $X$. If both $W$ and $Y$ are undefined, then $\mathcal{B}$ stores a best partial path in $X$ from any vertex to any other vertex in $X$.

Partial paths in SCCs of $L(D)$ are evaluated in terms of span, length, and type.

### 5.3.1 Path evaluation in terms of span and length

Every partial path in a SCC of a line graph $L(D)$ between possible pairs of entry and exit points is evaluated in terms of the span (see definition IV.1) and length of its corresponding trail in the directed graph $D$. As explained in subsection 4.2, paths with corresponding trails of maximum span *and* minimum length are to be preferred. Algorithm 4 (BESTPATH) below shows how the selection is made.

**Algorithm 4** BESTPATH($P$, $P'$)

**Input:** Two paths $P$ and $P'$ in a line graph $L(D)$.
**Output:** The path among $P$ and $P'$ whose corresponding trail in $D$ has maximum span or, in case both trails in $D$ corresponding to $P$ and $P'$ have equal span, the one whose corresponding trail in $D$ has minimum length.

1: **if** SPAN($L^{-1}(P)$) > SPAN($L^{-1}(P')$) **then**
2: **return** $P$
3: **if** SPAN($L^{-1}(P)$) = SPAN($L^{-1}(P')$) **and** $|L^{-1}(P)| < |L^{-1}(P')|$ **then**
4: **return** $P$
5: **return** $P'$

### 5.3.2 Path evaluation in terms of path type

The type of a partial path $P$ in a SCC $X$ of $L(D)$ between vertices $s$ and $t$ in $X$ reflects the role that $s$ and $t$ play in relation to the access points of $X$. More specifically, if the trail in $D$ corresponding to $P$ has maximum span, $P$ can be a path in $X$:

(a) Between entry point $s$ when coming from a SCC $W$ and an arbitrary vertex $t$;
(b) Between an arbitrary vertex $s$ and exit point $t$ when heading toward a SCC $Y$;
(c) Between entry point $s$ when coming from a SCC $W$ and exit point $t$ when heading to a SCC $Y$.

**Algorithm 5** EVALUATEPATH($P$, $X$, $W$, $Y$, $s$, $t$, $\mathcal{B}$)

**Input:** A path $P$ in the SCC $X$ of a line graph between vertices $s$ and $t$ in $X$, when coming from SCC $W$ and heading toward SCC $Y$, and a data structure $\mathcal{B}$ storing the best partial paths so far.

**Output:** $\mathcal{B}$ is updated with $P$ for the quintuplet $\mathcal{Q} = (X, W, Y, s, t)$ if $P$ is a better partial path than the one currently stored in $\mathcal{B}$ for $\mathcal{Q}$. $\mathcal{B}$ is updated to retain a path in $X$ whose corresponding trail has maximum span so far: (a) in $X$ between entry point $s$ and any vertex of $X$; (b) in $X$ between any vertex of $X$ and exit point $t$; (c) in $X$ between entry point $s$ and exit point $t$.

```
1: if W ≠ ⊥ then
2:     EVALUATEPATHAUX(P, X, W, ⊥, s, ⊥, B) /* case (a) */
3: if Y ≠ ⊥ then
4:     EVALUATEPATHAUX(P, X, ⊥, Y, ⊥, t, B) /* case (b) */
5: if W ≠ ⊥ and Y ≠ ⊥ then
6:     EVALUATEPATHAUX(P, X, W, Y, s, t, B) /* case (c) */
```

Algorithm 5 (EVALUATEPATH) distinguishes partial paths in $L(D)$ according to their type, as explained above. Internally, it uses a helper procedure named EVALUATEPATHAUX (algorithm 6) in order to determine whether the partial path stored in $\mathcal{B}$ for quintuplet $\mathcal{Q} = (X, W, Y, s, t)$ should be updated.

## 5.4 Algorithm FINDPATHS

Algorithm 7 (FINDPATHS) below starts out by initializing $\mathscr{P}$, a list that will store paths in $L(D)$ corresponding to a path $Q$ in the condensation graph (line 1). At line 2, $\mathscr{P}$ is given as an input/output parameter to the recursive algorithm CONCATENATEPARTIALPATHS (algorithm 8, see below). As CONCATENATEPARTIALPATHS recurses, best partial paths stored in $\mathcal{B}$ are concatenated and the resulting paths in $L(D)$ corresponding to a path $Q$ in the condensation graph of $L(D)$ are stored in $\mathscr{P}$. When recursion finishes, FINDPATHS returns the list $\mathscr{P}$ (line 3).

**Algorithm 6** EVALUATEPATHAUX($P$, $X$, $W$, $Y$, $s$, $t$, $\mathcal{B}$)

**Input:** A path $P$ in the SCC $X$ of a line graph between vertices $s$ and $t$ in $X$, when coming from SCC $W$ and heading toward SCC $Y$, and a data structure $\mathcal{B}$ storing the best partial paths so far.

**Output:** $\mathcal{B}$ is updated with $P$ for the quintuplet $\mathcal{Q} = (X, W, Y, s, t)$ if there is no partial path stored in $\mathcal{B}$ for $\mathcal{Q}$ or if $P$ is a better partial path than the one currently stored in $\mathcal{B}$ for $\mathcal{Q}$.

1: **if** $\mathcal{B}[(X, W, Y, s, t)] = \varnothing$ **then**
2: $\quad\mathcal{B}[(X, W, Y, s, t)] \leftarrow P$
3: **else if** $P =$ BESTPATH($P$, $\mathcal{B}[(X, W, Y, s, t)]$) **then**
4: $\quad\mathcal{B}[(X, W, Y, s, t)] \leftarrow P$

Algorithm 8 (CONCATENATEPARTIALPATHS) recursively extends a partial solution $P$ with a best partial path $P'$ stored in $\mathcal{B}$, the data structure returned by algorithm 3 (PARTIALPATHS).

Recursion proceeds for every index of a path $Q$ in the condensation graph of $L(D)$ and stops when the index exceeds the length of the path. Whenever this happens, it means that a path in $L(D)$ has been retrieved by concatenation of best partial paths in $\mathcal{B}$ and can be appended to the list $\mathscr{P}$ of paths in the line graph corresponding to the path $Q$ in the condensation graph (lines 1–2). The list of paths $\mathscr{P}$ is initialized to the empty set in algorithm 7 (FINDPATHS), before algorithm 8 (CONCATENATEPARTIALPATHS) is invoked.

If recursion does not stop for a given index $i$, CONCATENATEPARTIALPATHS proceeds to determine the SCC $X$ corresponding to the vertex at position $i$ in $Q$ ($Q_i$) at line 4. Next, the SCC $W$ acting as predecessor of $X$ is determined at lines 6–11, along with all vertices in $X$ acting as sources in relation to $W$ ($P_W$). Similarly, the SCC $Y$ acting as successor of $X$ is determined at lines 12–17, along with all vertices in $X$ acting as sinks in relation to $Y$ ($S_Y$).

Recursion actually takes place at lines 19–22. For every pair of vertices $(s, t)$ representing an entry point for the SCC $X$ when coming from $W$ and an exit point for $X$ when heading toward $Y$, respectively, the best partial path $P'$ stored in $\mathcal{B}$ for the quintuplet $(X, W, Y, s, t)$ is retrieved at line 21. CONCATENATEPARTIALPATHS is then called for the next value of the index $i$ and the path resulting from the concatenation of $P$ and $P'$, denoted by $P \sqcup P'$, at line 22.

When recursion finishes due to the index $i$ being greater than the length of path $Q$ in the condensation graph, the current path $P$ in $L(D)$ is added to the list of paths $\mathscr{P}$ at lines 1–2. The current call to CONCATENATEPARTIALPATHS is popped off the execution stack and the algorithm resumes to using $P$, the path in $L(D)$ it started with before concatenating $P'$. This way, a new pair $(s, t)$ of vertices in $X$

**Algorithm 7** FINDPATHS($L(D)$, $Q$, $\mathcal{B}$)

**Input:** A line graph $L(D)$, a path $Q$ in the condensation graph of $L(D)$, and a data structure $\mathcal{B}$ storing best partial paths for all SCCs of $L(D)$ as specified in algorithm PARTIALPATHS.

**Output:** All paths in $L(D)$ corresponding to path $Q$ in the condensation graph of $L(D)$.

1: $\mathscr{P} \leftarrow \varnothing$
2: CONCATENATEPARTIALPATHS($L(D)$, $Q$, 1, $\varnothing$, $\mathscr{P}$, $\mathcal{B}$)
3: **return** $\mathscr{P}$

**Algorithm 8** CONCATENATEPARTIALPATHS($L(D)$, $Q$, $i$, $P$, $\mathscr{P}$, $\mathcal{B}$)

**Input:** A line graph $L(D)$, a path $Q$ in the condensation graph of $L(D)$, an index $i$ between 1 and $|V(Q)| + 1$, a path $P$ in $L(D)$ obtained by concatenation of partial paths for the first $i - 1$ vertices of $Q$, an input/output list $\mathscr{P}$ storing paths in $L(D)$ corresponding to $Q$ obtained by concatenation of best partial paths in $\mathcal{B}$, and a data structure $\mathcal{B}$ storing best partial paths for all SCCs of $L(D)$ as specified by algorithm PARTIALPATHS.

**Output:** $\mathscr{P}$ contains paths in $L(D)$ corresponding to path $Q$ in the condensation graph of $L(D)$. The paths in $\mathscr{P}$ are obtained by concatenation of best partial paths stored in $\mathcal{B}$.

1: **if** $i = |V(Q)| + 1$ **then**
2: $\quad \mathscr{P} \leftarrow \mathscr{P} \cup P$
3: **else**
4: $\quad$ Let $X$ be the SCC in $L(D)$ corresponding to vertex $Q_i$ in $C$
5:
6: $\quad$ **if** $i = 1$ **then**
7: $\quad\quad W \leftarrow \bot$
8: $\quad\quad P_W \leftarrow \{\bot\}$
9: $\quad$ **else**
10: $\quad\quad$ Let $W$ be the SCC in $L(D)$ corresponding to vertex $Q_{i-1}$ in $C$
11: $\quad\quad P_W \leftarrow \{x \in X \mid (w, x)$ an arc in $L(D)$ with $w \in W\}$
12: $\quad$ **if** $i = |V(Q)|$ **then**
13: $\quad\quad Y \leftarrow \bot$
14: $\quad\quad S_Y \leftarrow \{\bot\}$
15: $\quad$ **else**
16: $\quad\quad$ Let $Y$ be the SCC in $L(D)$ corresponding to vertex $Q_{i+1}$ in $C$
17: $\quad\quad S_Y \leftarrow \{x \in X \mid (x, y)$ an arc in $L(D)$ with $y \in Y\}$
18:
19: $\quad$ **for all** $s \in P_W$ **do**
20: $\quad\quad$ **for all** $t \in S_Y$ **do**
21: $\quad\quad\quad P' \leftarrow \mathcal{B}[(X, W, Y, s, t)]$
22: $\quad\quad\quad$ CONCATENATEPARTIALPATHS($L(D)$, $Q$, $i + 1$, $P \sqcup P'$, $\mathscr{P}$, $\mathcal{B}$)

(respectively acting as sources when coming from $W$, and as sinks when heading toward $Y$) can be examined.

## 6 Allowing for skipped vertices

The MaSST and MaSSCoT formulations imply that solutions consist of strictly neighboring reactions catalyzed by products of strictly neighboring genes. As in a previous graph-based approach for the integration of heterogeneous biological data in another context [Boyer *et al.*, 2005], a preprocessing step was added to algorithm HNET (algorithm 1) in order to allow for non contiguous reactions and/or genes. The preprocessing step consists in modifying the input graphs by adding arcs (respectively edges) between vertices separated by at most $\delta_D$ other reactions (respectively $\delta_G$ other genes). $\delta_D$ and $\delta_G$ are referred to as the *gap parameters*. Their value should be set quite low (e.g. at most 3) for ensuring that the trails produced by HNET are relevant from a biological point of view.

*gap parameters*

For example, black solid edges corresponding to $\delta_G = 0$ link genes $A$ through $E$ in the undirected graph $G$ in Figure IV.4. One gene can be skipped if $\delta_G$ is set to 1, in which case the edge set of $G$ includes the dashed black edges. Finally, if two genes can be skipped ($\delta_G = 2$), the edge set of $G$ also includes the dotted blue edges.

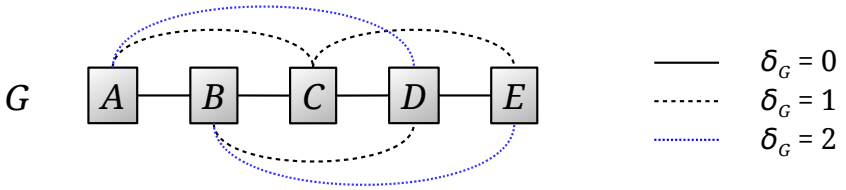



**Figure IV.4 Illustration of the gap parameter $\delta_G$.** If $\delta_G$ is positive, supplementary edges need to be added to $G$.

In a similar fashion to $\delta_G$, the gap parameter $\delta_D$ allows to skip reactions through the introduction of supplementary arcs in $D$.

## 7 Concluding remarks

This chapter presented a method for identifying trails of reactions in a metabolic pathway catalyzed by products of neighboring genes, easily adaptable to other types of biological networks with only minor modifications to the underlying model. We have therefore presented a generic method, applicable to different

kinds of biological data. The problem was formulated in graph theory terms and the exact algorithm HNET was proposed for trail finding. Although the problem is polynomially intractable in the general case, in practice HNET performs quite well as it reduces the computationally expensive operation of path enumeration to the strictly necessary minimum. The trail finding method proposed herein is hence very promising when applied to heterogeneous biological networks such as metabolic pathways and genomic context. The next chapter lays out the theoretical framework for exploiting HNET trails.

# V

# Trail grouping

# 1 Introduction

The previous chapter presented trail finding, a method that identifies trails of reactions being catalyzed by products of neighboring genes for a given species. The present chapter shows how these species-specific patterns of metabolic and genomic organization can be exploited in order to detect the conservation of such patterns across a vast array of different species.

For simplicity, let trails of reactions catalyzed by products of neighboring genes for a given species, as identified using the trail finding method presented in Chapter IV, be called *metabolic and genomic patterns* for a given species.

*metabolic and genomic pattern*

We first explain the comparative approach for metabolic and genomic patterns, delineating three possible solutions and justifying the choice for trail grouping. Next, we introduce the underlying concept in trail grouping, namely reaction sets. Subsequently, two methods of trail grouping, focusing on the conservation of metabolic and genomic context, respectively, are presented and discussed. Two complex trail grouping situations that arise in practice are also illustrated. Finally, this chapter concludes with a general discussion about trail grouping.

*conservation*

# 2 Comparative approach

The objective of the comparative approach is to exploit metabolic and genomic patterns obtained for several species in order to analyze their degree of conservation, both in terms of metabolic, as well as genomic, context. This section discusses three possible solutions for comparing metabolic and genomic patterns. Each solution is evaluated in terms of detection of metabolic and genomic context, as well as ability to capture the conservation of such patterns across multiple species.

The three solutions discussed here are illustrated on the example in Figure V.1. Assume the following:

- The trail $T = (r_1, r_2, r_3)$ was identified for species $S_1$ and $S_2$;
- The trail $T' = (r_1, r_2', r_3)$ was identified for species $S_3$;
- Species $S_1$ and $S_2$ do not perform the reaction $r_2'$;
- Species $S_3$ does not perform the reaction $r_2$.

## 2.1 Trail pooling

Trails identified by the trail finding method can be pooled together in order to determine which trails are common to several species. Trail pooling specifically

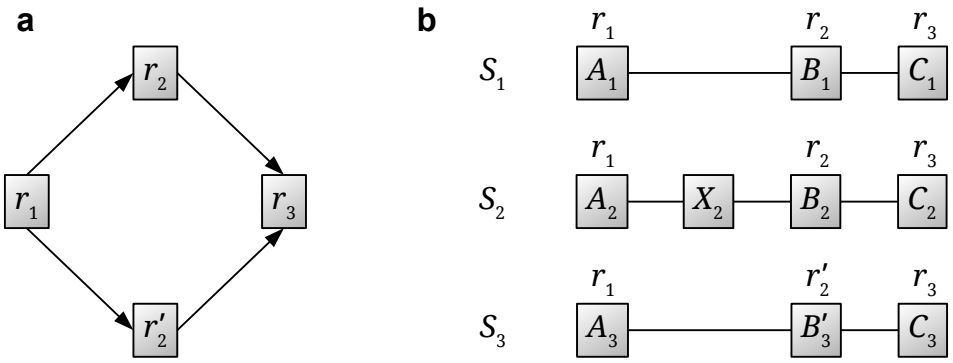



**Figure V.1 A metabolic pathway and the genomic context for three species.** **(a)** A metabolic pathway with two possible metabolic routes: $T = (r_1, r_2, r_3)$ and $T' = (r_1, r'_2, r_3)$. **(b)** Genomic context for three species $S_1$, $S_2$, and $S_3$. Genes belonging to the same chromosomal strand are shown as rectangles. Neighboring genes are linked by solid black edges. Reactions in which gene products are involved are specified above each gene, with the exception of gene $X_2$ of species $S_2$ which does not encode an enzyme. Reactions belong to the pathway in (a).

consists in pooling trails in order to obtain the set of all trails identified for every analyzed species, regardless of the values of the gap parameters (see section IV.6). These values would however become available when investigating a particular trail in the pool, for every species possessing the trail.

**Detection of metabolic and genomic patterns** Using the example in Figure V.1, trail pooling would detect the trail $T$ as being shared by species $S_1$ and $S_2$. $S_1$ performs the reactions in $T$ using products of strictly neighboring genes, whereas for species $S_2$ one gene was omitted. It would also be reported that species $S_3$ possesses the trail $T'$.

**Conservation of metabolic and genomic patterns** This approach would however fail to show that $T'$ is alternative metabolic route to $T$. Although the trails $T$ and $T'$ share the reactions $r_1$ and $r_3$ and although the genes involved in $r_1$ and $r_3$ are neighbors for all three species (if one or two genes are skipped), this metabolic and genomic pattern would not be apparent using a trail pooling approach.

## 2.2 Trail clustering

Trails obtained via trail finding can be clustered by using a particular clustering method and a particular (dis)similarity measure for trails. Depending on the selected method and measure, different results may be obtained.

For the purpose of this example, assume hierarchical clustering is chosen and that the Jaccard distance is used for cluster establishment. The Jaccard distance uses the ratio between the number of reactions present in two given trails and the total number of reactions involved in the two trails. This distance measure leads to the establishment of clusters reflecting trail similarity in terms of shared reactions. While on the surface it might seem promising, trail clustering exhibits two main disadvantages, discussed below.

**Detection of metabolic and genomic patterns** Using the example in Figure V.1, the Jaccard distance between trails $T$ and $T'$ is calculated as follows:

$$D_J = 1 - \frac{|T \cap T'|}{|T \cup T'|}$$

where $|T \cap T'|$ denotes the number of reactions shared between the trails $T$ and $T'$ (here, 2) and $|T \cup T'|$ denotes the total number of reactions involved in the two trails (here, 4). Hence, for this example, the Jaccard distance between the trails $T$ and $T'$ is 0.5. If the cutoff for hierarchical clustering is at least 0.5, the two trails are clustered together. Since hierarchical clustering is an exploratory approach, the cutoff value is context-dependent, meaning its value is chosen in accordance with the majority of the data to be clustered. For instance, if the trails to be clustered are highly similar (i.e. their respective Jaccard distances are closer to 0 rather than 1), the cutoff value is likely to be smaller than 0.5, which in turn means that trails $T$ and $T'$ will belong to different clusters.

**Conservation of metabolic and genomic patterns** Trail clustering does not allow a direct view on the whole array of species under study. It other words, while trail clustering manages to capture metabolic and genomic patterns if the cutoff value is chosen accordingly, it does not reflect their inter-specific degree of conservation. The only way to obtain the species distribution for trails from a given cluster is to investigate each of its trails in turn. For the example in Figure V.1, the trails $T$ and $T'$ belong to the same cluster assuming the cutoff value is at least 0.5. Although being in the same cluster means these trails are similar with respect to reaction composition, it is not known in advance that they occur in species $S_1$, $S_2$, and $S_3$.

## 2.3 Trail grouping

This approach examines trails of a given reference species in terms of their metabolic and genomic conservation across the remaining species under study. Instead

of directly comparing trails of the reference species to trails of the other species, trail grouping determines:

(a) whether reactions involved in trails of the reference species are catalyzed by products of neighboring genes in other species, and
(b) whether genes of the reference species involved in reactions in a given trail have neighboring functionally similar genes in other species. (*Functionally similar genes* encode enzymes that catalyze the same reaction.)

*functionally similar genes*

For the example in Figure V.1, the objectives (a) and (b) above can be accomplished by examining the metabolic context for the trails $T$ and $T'$, as well as the genomic context for species $S_1$, $S_2$, and $S_3$ for genes involved in these two trails (Table V.1). As the exact details take up the rest of this chapter, conjecture for now that trail grouping attains the previously stated objectives.

| Reaction | $S_1$ | $S_2$ | $S_3$ |
|---|---|---|---|
| $r_1$ | $A_1$ | $A_2$ | $A_3$ |
| $r_2$ | $B_1$ | $B_2$ | — |
| $r'_2$ | — | — | $B'_2$ |
| $r_3$ | $C_1$ | $C_2$ | $C_3$ |

**Table V.1 Metabolic and genomic context with respect to Figure V.1.** For every reaction involved in the two trails $T$ and $T'$ in Figure V.1a, it is shown which genes in species $S_1$, $S_2$, and $S_3$ in Figure V.1b encode the required enzymes.

**Detection of metabolic and genomic patterns** Although trails of the reference species obtained via trail finding are not directly compared to trails of the remaining species under study, by choosing an appropriate manner of summarizing metabolic and genomic information (similar to Table V.1), it is possible to determine whether a target species shares a common trail with the reference species, partially or entirely. For example, the trail $T$ occurs in species $S_1$ and $S_2$ entirely, whereas only the reactions $r_1$ and $r_3$ from the trail $T'$ occur in these species (Figure V.1 and Table V.1).

**Conservation of metabolic and genomic patterns** As explained in the previous paragraph, metabolic and genomic patterns are detected whether the matches between the various species are partial or complete. This allows to effectively study inter-specific variations at both the metabolic and genomic levels or, in other words, the conservation of metabolic and genomic patterns.

### 2.4 Summary

The three possible solutions for a comparative approach capable of exploiting trail finding results were evaluated according to their ability to detect metabolic and genomic patterns, as well as conserved such patterns. The conclusions are summarized in Table V.2.

By definition, all three approaches are able to detect metabolic and genomic patterns, although trail pooling does not contribute any new knowledge from this point of view. Trail clustering promisingly brings together similar trails but fails at detecting conservation. Finally, trail grouping fulfills both criteria. The rest of this chapter describes the theoretical framework for trail grouping.

| Approach | Detection | Conservation |
|---|---|---|
| Trail pooling | (yes) | no |
| Trail clustering | yes | no |
| Trail grouping | yes | yes |

**Table V.2 Comparison of trail pooling, trail clustering, and trail grouping.** Summary of the three comparative approaches with respect to the detection and conservation of metabolic and genomic patterns.

## 3 Reaction sets

For reasons detailed in this section, trail grouping treats trails as *reaction sets*, meaning that the order of reactions is not taken into account and that repeated reactions are ignored. In Figure V.2, trails $T_1 = (r_2, r_7, r_8, r_3, r_4)$ and $T_2 = (r_2, r_3, r_7, r_8, r_3, r_4)$ both have the same corresponding reaction set $\{r_2, r_3, r_4, r_7, r_8\}$. Henceforth, reaction sets corresponding to trails produced by the HNET algorithm (see section IV.5) will be called HNET *reaction sets*.

HNET *reaction set*

The definition of conserved metabolic and genomic patterns (in terms of metabolic and gene neighborhoods) needs to be able to accommodate slight variations between species.

One such variation is encountering a different reaction order between trails. For example, if trails $(r_9, r_{10})$ and $(r_{10}, r_9)$ are identified for two different species for the pathway in Figure V.2, these trails naturally constitute a conserved pattern for the two species.

Another variation that needs to be taken into account is best illustrated with the example of trails $T_1$ and $T_2$ above. If these trails are obtained for different species, the common denominator is that both species perform the same five reactions us-

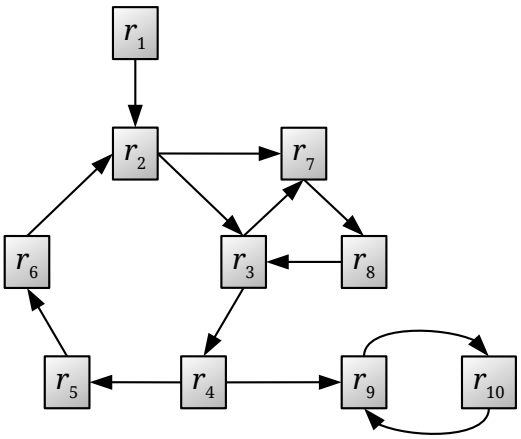



**Figure V.2 Example of a metabolic pathway.** Vertices represent reactions. This figure is identical to Figure IV.1b.

ing products of neighboring genes, irrespective of reaction order and of whether reaction $r_3$ is repeated.

Another example of variation that should not prevent the identification of conserved patterns is related to reactions (or genes) that are present in trails of some, but not all, of the species. For example, suppose the trails $(r_2, r_3, r_7)$ and $(r_3, r_7, r_8)$ are identified for two different species for the pathway in Figure V.2. The fact that reactions $r_3$ and $r_7$ are common to both trails and are catalyzed by products of neighboring genes for both species should be identified as a conserved pattern.

The necessity of accommodating these types of trail variations explains the choice for processing HNET trails as HNET reaction sets during the present trail grouping step.

## 4 Theoretical framework for trail grouping

Let $\mathcal{P}$ be the panel of species under study. Trail grouping requires the designation of a *reference species* $S$ among the species in $\mathcal{P}$. Trails of the reference species obtained via trail finding are processed as HNET reaction sets in order to detect conservation of their metabolic and genomic patterns across the remaining species in $\mathcal{P}$.

*reference species*

Let $R_S$ be the set of all HNET reaction sets of $S$. Note that reaction sets in $R_S$ are not disjoint. From a biological standpoint, $R_S$ represents the pool of trails of the reference species obtained through trail finding, viewed in terms of HNET reaction sets.

In order for trail grouping to accommodate genomic variations between species,

*neighboring genes*

it is considered that two genes of a given species are *neighbors* if they are separated by at most three other genes on the same strand of the same chromosome.

The remainder of this section presents two methods for trail grouping:

*grouping by reactions*

- *Trail grouping by reactions* consists in grouping reactions of the reference species according to the HNET reaction sets they belong to. This method focuses more on conserved metabolic, rather than genomic, patterns.

*grouping by genes*

- *Trail grouping by genes* consists in grouping HNET reaction sets of the reference species according to its gene order. This method focuses more on conserved genomic, rather than metabolic, patterns.

Assume the trail $t = (r_6, r_2, r_3, r_7, r_8)$ has been identified by the trail finding method in the pathway in Figure V.2 for a reference species $S$. The genomic contexts of species $S$ and another species $S_1$ are shown in Figure V.3. Both methods will be illustrated on this example.

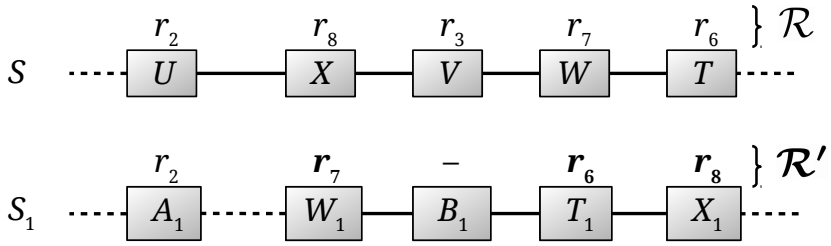



**Figure V.3 Gene neighborhood for species $S$ and $S_1$.** Genes belonging to the same chromosomal strand are shown as rectangles. Neighboring and non neighboring genes are linked with continuous and dotted edges, respectively. Reactions in which gene products are involved are specified above each gene, with the exception of gene $B_1$ of species $S_1$ which does not code for an enzyme. Reactions belong to the pathway in Figure V.2. $\mathcal{R}$ represents a HNET reaction set of $S$. $\mathcal{R}' = \{r_6, r_7, r_8\}$ designates a maximal subset of $\mathcal{R}$ such that genes of $S_1$ involved in reactions in $\mathcal{R}'$ (in bold) are neighbors.

## 4.1 Grouping by reactions

Grouping trails by reactions for the reference species $S$ consists in constructing a table $T_S^r$ where rows represent reactions in every HNET reaction set of $S$ and columns represent the remaining species in $\mathcal{P}$. Table $T_S^r$ reflects conserved metabolic patterns between the reference species and the rest of the panel through the three possible symbols that can be assigned to each cell. These symbols allow to easily distinguish which reactions of the reference species are not present

in the other species (blanks), and which are catalyzed by products of neighboring (crosses) and non neighboring (dots) genes of the other species.

For example, for the trail $t = (r_6, r_2, r_3, r_7, r_8)$ in Figure V.2 and the gene neighborhood in Figure V.3 for the reference species $S$ and another species $S_1$, $T_S^r$ is represented by the first ($\mathcal{R}$) and fourth ($S_1$) columns in Table V.3. Reaction $r_3$ is not performed by species $S_1$. Reactions $r_6$, $r_7$, and $r_8$ are performed by neighboring genes of $S_1$ ($T_1$, $W_1$, and $X_1$, respectively), whereas reaction $r_2$ involves the product of a distant gene.

| $\mathcal{R}$ | $S$ genes | $S_1$ genes | $S_1$ |
|---|---|---|---|
| $\boldsymbol{r_6}$ | $\boldsymbol{T}$ | $\boldsymbol{T_1}$ | $\times$ |
| $r_2$ | $U$ | $A_1$ | . |
| $r_3$ | $V$ | $-$ | |
| $\boldsymbol{r_7}$ | $\boldsymbol{W}$ | $\boldsymbol{W_1}$ | $\times$ |
| $\boldsymbol{r_8}$ | $\boldsymbol{X}$ | $\boldsymbol{X_1}$ | $\times$ |
| $\boldsymbol{\mathcal{R}'}$ | **Neighboring** | **Neighboring** | |

**Table V.3 Trail grouping by reactions for the reference species $S$ against another species $S_1$ (column $S_1$).** This is an extended version of the trail grouping by reactions table $T_S^r$, where columns "$S$ genes" and "$S_1$ genes" have been added for convenience. Entries in bold in columns $\mathcal{R}$, "$S$ genes", and "$S_1$ genes" respectively designate $\mathcal{R}'$ and neighboring genes in $S$ and $S_1$ (see table footer). $\mathcal{R}$ represents a HNET reaction set of $S$. Symbols in column $S_1$ represent conserved metabolic patterns between species $S$ and $S_1$ for reactions in $\mathcal{R}$. Roughly speaking, $\mathcal{R}'$ designates a maximal subset of $\mathcal{R}$ such that genes of $S_1$ involved in reactions in $\mathcal{R}'$ are neighbors (see text for formal definitions).

Rows in table $T_S^r$ represent reactions in $R_S$ and are ordered by HNET reaction sets of $S$. Note that a given reaction performed by species $S$ appears several times in $T_S^r$ if it belongs to several HNET reaction sets. Columns represent the remaining species in $\mathcal{P}$ and are ordered according to evolutionary distance to $S$, such that species phylogenetically closer to $S$ have lower column indexes than species phylogenetically distant from $S$.

Let $T_S^r[i, j]$ denote the cell in $T_S^r$ on row $i$ and column $j$. Let $r_i$ denote the reaction of species $S$ corresponding to row $i$ in $T_S^r$. Let $S_1$ denote the species corresponding to column $j$ in $T_S^r$. Let $\mathcal{R} \subseteq R_S$ denote the HNET reaction set of species $S$ to which reaction $r_i$ belongs. For the example presented above, the HNET reaction set of species $S$ that is investigated is $\mathcal{R} = \{r_2, r_3, r_6, r_7, r_8\}$ (see the first column ($\mathcal{R}$) in Table V.3).

Let $\mathcal{R}'$ denote a maximal subset of $\mathcal{R}$ such that the genes of $S_1$ involved in $\mathcal{R}'$ are neighbors. For the above example, the subset $\mathcal{R}'$ is $\{r_6, r_7, r_8\}$ (see $\mathcal{R}'$, i.e.

entries in bold in the first column ($\mathcal{R}$) in Table V.3) because reactions in $\mathcal{R}'$ involve the neighboring genes $T_1$, $W_1$, and $X_1$, respectively, in species $S_1$ (even though gene $B_1$ is skipped).

One of the following three symbols is assigned to each cell $T^r_S[i, j]$:

- a cross ($\times$) if $r_i \in \mathcal{R}'$.
- a dot (.) if $r_i \in \mathcal{R} - \mathcal{R}'$ and $r_i$ is performed by species $S_1$.
- a blank if $r_i \in \mathcal{R} - \mathcal{R}'$ and $r_i$ is not performed by species $S_1$.

For the above example (see the fourth column ($S_1$) in Table V.3), the cells corresponding to reactions in $\mathcal{R}'$ receive a cross symbol ($\times$). Since reaction $r_2$ in $\mathcal{R}$ is performed in $S_1$ by gene $A_1$ and does not belong to $\mathcal{R}'$, the corresponding cell on column $S_1$ in $T^r_S$ receives a dot symbol (.). Finally, reaction $r_3$ is absent from $S_1$, therefore the corresponding cell receives a blank. The interpretation is that reactions $r_6$, $r_7$, and $r_8$ are performed in species $S_1$ by products of neighboring genes. Reaction $r_3$ is absent from $S_1$, whereas the gene involved in $r_2$ is not a neighbor of genes involved in reactions $r_6$, $r_7$, and $r_8$.

## 4.2 Grouping by genes

*functionally similar genes*

Two genes encoding enzymes involved in the same metabolic reaction are referred to as *functionally similar genes*. Functionally similar genes in two species can be either analogues (products of convergent evolution) or homologues (products of divergent evolution).

Grouping trails by genes consists in constructing a table $T^g_S$ where rows represent genes of the reference species $S$ involved in HNET reaction sets shared by $S$ and at least one other species in $\mathcal{P}$, and columns represent the remaining species in $\mathcal{P}$. Table $T^g_S$ reflects conserved genomic patterns between the reference species and the rest of the panel through the two possible symbols that can be assigned to each cell. These symbols allow to easily distinguish genes of $S$ with neighboring (crosses) and non neighboring (dots) functionally similar genes in other species.

For example, for the trail $t = (r_6, r_2, r_3, r_7, r_8)$ in Figure V.2 and the gene neighborhood in Figure V.3 for the reference species $S$ and another species $S_1$, $T^g_S$ is represented by the second ($\mathcal{G}$) and fourth ($S_1$) columns in Table V.4. Genes $X_1$, $W_1$, and $T_1$ of $S_1$ respectively have the neighboring functionally similar genes $X$, $W$, and $T$ in the reference species $S$ (hence the cross symbols).

Let $R_{S_1}$ be the set of all HNET reaction sets for species $S_1 \in \mathcal{P} - \{S\}$. Let $R$ be the set of HNET reaction sets defined by:

| $\mathcal{R}$ | $\mathcal{G}$ | $\mathcal{H}$ | $S_1$ |
|---|---|---|---|
| $r_2$ | $U$ | $A_1$ | . |
| $\boldsymbol{r_8}$ | $\boldsymbol{X}$ | $\boldsymbol{X_1}$ | $\times$ |
| $r_3$ | $V$ | $-$ | . |
| $\boldsymbol{r_7}$ | $\boldsymbol{W}$ | $\boldsymbol{W_1}$ | $\times$ |
| $\boldsymbol{r_6}$ | $\boldsymbol{T}$ | $\boldsymbol{T_1}$ | $\times$ |
| | $\boldsymbol{\mathcal{G}'}$ | $\boldsymbol{\mathcal{H}'}$ | |

**Table V.4 Trail grouping by genes for the reference species $S$ against another species $S_1$ (column $S_1$).** This is an extended version of the trail grouping by genes table $T_S^g$, where columns $\mathcal{R}$ and $\mathcal{H}$ have been added for convenience. Entries in bold in columns $\mathcal{G}$ and $\mathcal{H}$ respectively designate $\mathcal{G}'$ and $\mathcal{H}'$ (see table footer). $\mathcal{R}$ represents a HNET reaction set of $S$. $\mathcal{G}$ represents a group of neighboring genes of $S$ whose products catalyze the respective reactions in $\mathcal{R}$. Symbols in column $S_1$ in $T_S^g$ represent conserved genomic patterns between species $S$ and $S_1$ for genes in $\mathcal{G}$. Roughly speaking, $\mathcal{H}$ designates genes in $S_1$ involved in reactions in $\mathcal{R}$; $\mathcal{H}'$ designates neighboring genes in $\mathcal{H}$ involved in reactions in $\mathcal{R}$. $\mathcal{H}'$ maximizes the number of genes in $\mathcal{G}'$, where genes in $\mathcal{H}'$ and $\mathcal{G}' \subseteq \mathcal{G}$ are involved in the same reactions in $\mathcal{R}$ (see text for formal definitions).

$$R = R_S \cap \Big( \bigcup_{S_1 \in \mathcal{P} - \{S\}} R_{S_1} \Big)$$

Hence, $R$ represents the set of HNET reaction sets common to $S$ and at least one other species in $\mathcal{P}$. Let $G_S$ be the set of genes of the reference species $S$ that are involved in reactions belonging to HNET reaction sets of $R$. From a biological standpoint, $G_S$ represents the pool of genes of the reference species encoding enzymes involved in HNET reaction sets common to $S$ and at least one other species in $\mathcal{P}$.

Rows in table $T_S^g$ represent genes from $G_S$ and are ordered by chromosome and strand, according to the position of genes on the strand. Columns represent the remaining species in $\mathcal{P}$ and are ordered according to evolutionary distance to $S$, such that species phylogenetically closer to $S$ have lower column indexes than species phylogenetically distant from $S$.

Let $S_1$ denote the species corresponding to column $j$ in $T_S^g$. Let $\mathcal{G}$ be a subset of $G_S$ such that genes in $\mathcal{G}$ are neighbors on the same strand and chromosome of $S$. For the example presented above, the gene group of species $S$ that is investigated is $\mathcal{G} = \{U, X, V, W, T\}$ (see the second column ($\mathcal{G}$) in Table V.4).

Let $\mathcal{R}$ be the set of reactions in all HNET reaction sets in which the genes in $\mathcal{G}$ are involved. Formally, $\mathcal{R}$ is the set of all reactions $r$ such that:

(a) there exists a reaction set $h$ of species $S$ such that $r \in h$, and
(b) there exists a gene $g \in \mathcal{G}$ such that $g$ is involved in $r$.

In other words, given a group $\mathcal{G}$ of neighboring genes of $S$, $\mathcal{R}$ is the set of reactions in trails common to $S$ and at least one other species in $\mathcal{P}$ such that reactions in $\mathcal{R}$ are catalyzed by products of genes in $\mathcal{G}$. For the above example, $\mathcal{R}$ is $\{r_2, r_3, r_6, r_7, r_8\}$ (see the first column ($\mathcal{R}$) in Table V.4).

Let $\mathcal{H}$ be the set of genes of $S_1$ involved in reactions in $\mathcal{R}$. That is, given $\mathcal{R}$, the genome for species $S_1$, and the correspondence between reactions in $\mathcal{R}$ and genes of $S_1$, $\mathcal{H}$ is the set of genes in $S_1$ (along with their position on the chromosome) such that every gene in $\mathcal{H}$ is involved in at least one reaction in $\mathcal{R}$. For the above example, $\mathcal{H} = \{A_1, X_1, W_1, T_1\}$ (see the third column ($\mathcal{H}$) in Table V.4).

Let $\mathcal{H}' \subseteq \mathcal{H}$ be neighboring genes in $\mathcal{H}$, and let $\mathcal{G}' \subseteq \mathcal{G}$ such that genes in $\mathcal{H}'$ and $\mathcal{G}'$ are involved in the same reactions in $\mathcal{R}$. $\mathcal{H}'$ is chosen such as to maximize $|\mathcal{G}'|$, i.e. the number of genes in $\mathcal{G}$ involved in the same reactions as neighboring genes in $\mathcal{H}$.

For the above example, gene $A_1$ is not a neighbor of gene $W_1$, therefore $\mathcal{H}'$ must be a strict subset of $\mathcal{H}$. There are several possible strict non-empty subsets of $\mathcal{H}$ of neighboring genes, other than singletons: $\{W_1, T_1\}$, $\{W_1, X_1\}$, $\{T_1, X_1\}$, and $\{W_1, T_1, X_1\}$. The subset of $\mathcal{H}$ that is of interest is $\mathcal{H}' = \{W_1, T_1, X_1\}$, as it maximizes the number of genes in $\mathcal{G}$ involved in reactions in $\mathcal{R}$; $\mathcal{G}'$ is thus $\{X, W, T\}$ (see $\mathcal{H}'$ and $\mathcal{G}'$, i.e. entries in bold in the third ($\mathcal{H}$) and second ($\mathcal{G}$) columns, respectively, in Table V.4). The genes in $\mathcal{H}'$ can be considered neighboring because only gene $B_1$ needs to be skipped as it does not encode an enzyme. Thus the subset of reactions of $\mathcal{R}$ catalyzed by genes in $\mathcal{H}'$ is $\{r_6, r_7, r_8\}$.

Let $T_S^g[i, j]$ denote the cell in $T_S^g$ on row $i$ and column $j$, where $i$ is the index in $G_S$ of a gene $g_i$ in $\mathcal{G}$. One of the following two symbols is assigned to each cell $T_S^g[i, j]$:

- a cross ($\times$) if $g_i \in \mathcal{G}'$.
- a dot ($.$) if $g_i \in \mathcal{G} - \mathcal{G}'$.

For the above example, cells for genes $U$ and $V$ receive a dot symbol ($.$), whereas cells for genes $X$, $W$, and $T$ receive a cross symbol ($\times$) (see the second ($\mathcal{G}$) and fourth ($S_1$) columns in Table V.4). The interpretation is that genes $X$, $W$, and $T$ of the reference species are involved in reactions catalyzed by neighboring genes in species $S_1$.

# 5 Special situations

This section presents two case studies of complex situations that arise in practice when dealing with biological data. They are illustrated using the method of trail grouping by genes.

## 5.1 The number of genes in the reference species is maximized

Here is addressed the aspect of the formal definition of $T_S^g$ requiring that $\mathcal{H}'$ maximizes $|\mathcal{G}'|$. In other words, given a group $\mathcal{G}$ of genes of the reference species $S$ involved in reactions of a HNET reaction set of $S$, $\mathcal{G}'$ is the maximum subset of genes in $\mathcal{G}$ having neighboring functionally similar genes in the target species.

As before, consider the trail $t = (r_6, r_2, r_3, r_7, r_8)$ was obtained for a reference species $S$ for the pathway in Figure V.2. The genomic context of species $S$ and another species $S_2$ is shown in Figure V.4. Trail grouping by genes is represented by the second ($\mathcal{G}$) and fourth ($S_2$) columns in Table V.5.

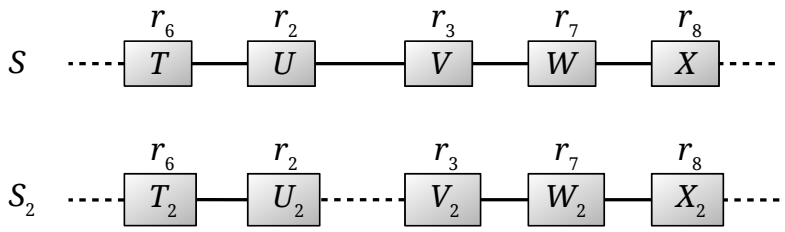



**Figure V.4 Gene neighborhood for species $S$ and $S_2$.** Genes belonging to the same chromosomal strand are shown as rectangles. Neighboring and non neighboring genes are linked with continuous and dotted edges, respectively. Reactions in which gene products are involved are specified above each gene. Reactions belong to the pathway in Figure V.2.

| $\mathcal{R}$ | $\mathcal{G}$ | $\mathcal{H}$ | $S_2$ |
|---|---|---|---|
| $r_6$ | $T$ | $T_2$ | . |
| $r_2$ | $U$ | $U_2$ | . |
| $\boldsymbol{r_3}$ | $\boldsymbol{V}$ | $\boldsymbol{V_2}$ | $\times$ |
| $\boldsymbol{r_7}$ | $\boldsymbol{W}$ | $\boldsymbol{W_2}$ | $\times$ |
| $\boldsymbol{r_8}$ | $\boldsymbol{X}$ | $\boldsymbol{X_2}$ | $\times$ |
| | $\boldsymbol{\mathcal{G}'}$ | $\boldsymbol{\mathcal{H}'}$ | |

**Table V.5 Trail grouping by genes for the reference species $S$ against another species $S_2$ (column $S_2$).** The HNET trail under study is $t = (r_6, r_2, r_3, r_7, r_8)$, obtained for the pathway in Figure V.2. For more details see Table V.4.

The gene group of species $S$ that is investigated here is $\mathcal{G} = \{T, U, V, W, X\}$ (see the second column ($\mathcal{G}$) in Table V.5). The set of reactions from all HNET reaction sets in which genes in $\mathcal{G}$ are involved is $\mathcal{R} = \{r_2, r_3, r_6, r_7, r_8\}$ (see the first column ($\mathcal{R}$) in Table V.5).

The set of genes of species $S_2$ involved in reactions in $\mathcal{R}$ is $\mathcal{H} = \{T_2, U_2, V_2, W_2, X_2\}$ (see the third column ($\mathcal{H}$) in Table V.5). All genes in $\mathcal{G}$ are neighbors in the reference species $S$. However, for species $S_2$, genes in $\mathcal{H}$ are separated into two groups of neighboring genes: $\{T_2, U_2\}$ and $\{V_2, W_2, X_2\}$, respectively (Figure V.4).

The genes in subset $\mathcal{H}' \subseteq \mathcal{H}$ must be neighbors for species $S_2$, therefore $\mathcal{H}'$ is either $\{T_2, U_2\}$ or $\{V_2, W_2, X_2\}$. If $\mathcal{H}' = \{T_2, U_2\}$, then the reactions catalyzed in $\mathcal{R}$ by genes in $\mathcal{H}'$ are $\{r_6, r_2\}$, and the genes of the reference species involved in these reactions are $\mathcal{G}' = \{T, U\}$. If $\mathcal{H}' = \{V_2, W_2, X_2\}$, then the reactions in $\mathcal{R}$ catalyzed by genes in $\mathcal{H}'$ are $\{r_3, r_7, r_8\}$, and the genes of the reference species involved in these reactions are $\mathcal{G}' = \{V, W, X\}$. The correct choice for $\mathcal{H}'$ is therefore $\mathcal{H}' = \{V_2, W_2, X_2\}$, as it corresponds to $|\mathcal{G}'| = 3$ instead of $|\mathcal{G}'| = 2$ (see $\mathcal{H}'$, i.e. entries in bold in the third column ($\mathcal{H}$) in Table V.5).

The subset $\mathcal{G}'$ indicates how cells in $T_S^g$ on the column corresponding to species $S_2$ are filled; as can be seen in Table V.5, cells for genes $T$ and $U$ receive a dot symbol ($.$), whereas cells for genes $V$, $W$, and $X$ receive cross symbols ($\times$). The interpretation is that genes $V$, $W$, and $X$ of the reference species are involved in reactions catalyzed by neighboring functionally similar genes in species $S_2$. The same is true of the other two genes $T$ and $U$, however. The reason $T_S^g$ shows these two genes as not having neighboring functionally similar genes in species $S_2$ is twofold. On the one hand, $\{T, U\}$ is not the maximum subset of $\mathcal{G}$ having neighboring functionally similar genes in species $S_2$ (as shown, the maximum subset is $\mathcal{G}' = \{V, W, X\}$). On the other hand, even though all genes in $\mathcal{G}$ have neighboring functionally similar genes in species $S_2$, the sets of genes $\{T_2, U_2\}$ and $\{V_2, W_2, X_2\}$ in $S_2$ are not neighbors on the chromosome. It would therefore be misleading to indicate that genes of the reference species $V$, $W$, and $X$, as well as genes $T$ and $U$, have neighboring functionally similar genes in species $S_2$.

Maximizing the number of neighboring genes of the reference species in the context of trail grouping by genes represents a greedy strategy. It increases the probability of detecting large similar conserved patterns at the genomic level across the remaining species under study.

## 5.2 The enzyme–reaction association is not one-to-one

This example elaborates on the previous one (see section 5.1 above). As the preceding example, it shows that, for trail grouping by genes, the number of neighboring genes of the reference species is maximized. Unlike the previous example, here are illustrated the cases where the product of one gene is involved in several reactions, and where one reaction involves the products of several genes.

Consider the trail $t = (r_6, r_2, r_3, r_7)$ was obtained for a reference species $S$ for the pathway in Figure V.2. The genomic context of species $S$ and another species $S_3$ is shown in Figure V.5. Trail grouping by genes is represented by the second ($\mathcal{G}$) and fourth ($S_3$) columns in Table V.6.

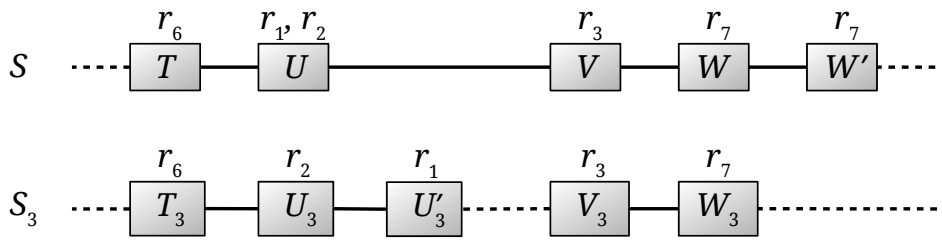



**Figure V.5 Gene neighborhood for species $S$ and $S_3$.** Genes belonging to the same chromosomal strand are shown as rectangles. Neighboring and non neighboring genes are linked with continuous and dotted edges, respectively. Reactions in which gene products are involved are specified above each gene. Reactions belong to the pathway in Figure V.2.

| $\mathcal{R}$ | $\mathcal{G}$ | $\mathcal{H}$ | $S_3$ |
|---|---|---|---|
| $r_6$ | $T$ | $T_3$ | . |
| $r_1$<br>$r_2$ | $U$ | $U'_3$<br>$U_3$ | . |
| $\boldsymbol{r_3}$ | $\boldsymbol{V}$ | $\boldsymbol{V_3}$ | $\times$ |
| $\boldsymbol{r_7}$ | $\boldsymbol{W}$<br>$\boldsymbol{W'}$ | $\boldsymbol{W_3}$ | $\times$<br>$\times$ |
| | $\boldsymbol{\mathcal{G}'}$ | $\boldsymbol{\mathcal{H}'}$ | |

**Table V.6 Trail grouping by genes for the reference species $S$ against another species $S_3$ (column $S_3$).** The HNET trail under study is $t = (r_6, r_2, r_3, r_7)$, obtained for the pathway in Figure V.2. For more details see Table V.4.

Figure V.5 shows that the reactions $r_1$ and $r_2$ are performed by the product of a unique gene $U$ in the reference species $S$, whereas in species $S_3$ the two reactions involve the two separate genes $U'_3$ and $U_3$. Conversely, the reaction $r_7$ is catalyzed

by two enzymes encoded by the genes $W$ and $W'$ in the reference species $S$, whereas species $S_3$ performs this reaction using solely the product of a gene $W_3$.

The gene group of species $S$ that is investigated here is $\mathcal{G} = \{T, U, V, W, W'\}$ (see the second column ($\mathcal{G}$) in Table V.6). Recall that in trail grouping by genes are considered reactions from *all* HNET reaction sets in which genes in $\mathcal{G}$ are involved. This includes the reaction $r_1$. $\mathcal{R}$ is therefore the set $\{r_1, r_2, r_3, r_6, r_7\}$ (see the first column ($\mathcal{R}$) in Table V.6).

The set of genes of species $S_3$ involved in reactions in $\mathcal{R}$ is $\mathcal{H} = \{T_3, U_3, U'_3, V_3, W_3\}$ (see the third column ($\mathcal{H}$) in Table V.6). All genes in $\mathcal{G}$ are neighbors in the reference species $S$. However, for species $S_3$, genes in $\mathcal{H}$ are separated into two groups of neighboring genes: $\{T_3, U_3, U'_3\}$ and $\{V_3, W_3\}$, respectively (Figure V.5).

The genes in subset $\mathcal{H}' \subseteq \mathcal{H}$ must be neighbors for species $S_3$, therefore $\mathcal{H}'$ is either $\{T_3, U_3, U'_3\}$ or $\{V_3, W_3\}$. If $\mathcal{H}' = \{T_3, U_3, U'_3\}$, then the reactions in $\mathcal{R}$ catalyzed by genes in $\mathcal{H}'$ are $\{r_6, r_1, r_2\}$, and the genes of the reference species involved in these reactions are $\mathcal{G}' = \{T, U\}$. If $\mathcal{H}' = \{V_3, W_3\}$, then the reactions in $\mathcal{R}$ catalyzed by genes in $\mathcal{H}'$ are $\{r_3, r_7\}$, and the genes of the reference species involved in these reactions are $\mathcal{G}' = \{V, W, W'\}$. The correct choice for $\mathcal{H}'$ is therefore $\mathcal{H}' = \{V_3, W_3\}$, as it corresponds to $|\mathcal{G}'| = 3$ instead of $|\mathcal{G}'| = 2$ (see $\mathcal{H}'$, i.e. entries in bold in the third column ($\mathcal{H}$) in Table V.6).

The subset $\mathcal{G}'$ indicates how cells in $T_S^g$ on the column corresponding to species $S_3$ are filled; as can be seen in Table V.6, cells for genes $T$ and $U$ receive a dot symbol ($.$), whereas cells for genes $V$, $W$, and $W'$ receive cross symbols ($\times$). The interpretation is that genes $V$, $W$, and $W'$ of the reference species are involved in reactions catalyzed by neighboring genes in species $S_3$, and are the maximum subset in $\mathcal{G}$ having neighboring functionally similar genes in species $S_3$. Although the genes $T$ and $U$ are equally neighbors in $S$ and have neighboring functionally similar genes in $S_3$, the functionally similar genes to $T$ and $U$ are separated on the chromosome from the functionally similar genes to $V$, $W$, and $W'$.

This example shows that trail grouping is a robust method, capable of handling the complex biological associations between metabolism and genomic context.

## 6 Discussion

Following trail finding, trail grouping is a second step leading from metabolic and genomic patterns for a single species (HNET trails) to the identification of potentially interesting conserved metabolic and genomic patterns in interspecies comparisons. In order to capture the most relevant conserved patterns across multiple species, it is fundamentally important to go beyond strictly matching patterns by

accommodating possible trail variations, such as trail directionality, reaction order, repetition of reactions, as well as different but overlapping sets of reactions and/or neighboring genes. The necessity of incorporating these variations for establishing conserved interspecies patterns requires processing trails as HNET reaction sets during the trail grouping step.

Once trail grouping has identified potentially interesting conserved patterns, the metabolic and genomic patterns conserved across multiple species can be analyzed on a case-by-case basis. During this third analysis step, HNET reaction sets should be considered in their metabolic context and hence treated yet again as trails. In their metabolic context, trails contain information on cycles and reaction directionality, and correspond to actual metabolic routes.

To provide a powerful and flexible way to analyze trails obtained through trail finding as explained in the previous chapter, two trail grouping methods are proposed, respectively termed trail grouping by genes and by reactions.

On the one hand, trail grouping by genes is restricted to genes of the reference species that are involved in HNET reaction sets common to at least one other species. This approach has the distinct advantage of keeping together neighboring genes that potentially make up for more than a single trail for the reference species (an example is given in section VII.5).

On the other hand, trail grouping by reactions identifies all HNET reaction sets for the reference species, which makes it possible to retrieve valuable information in the form of alternative reactions that might have been filtered out when grouping trails by genes. Suppose the reference species is the only species in the selected panel to perform a given metabolic route $M$, while also sharing some reactions with other species in the panel. If the shared reactions as well as those specific to the metabolic route $M$ involve neighboring genes in the reference species, then the specific route $M$, while not visible when grouping trails by genes, will be present in trail grouping by reactions. Consider the case of species $S_3$ in Figure V.1, chosen as reference species. It is the only species among $S_1$, $S_2$, and $S_3$ for which the HNET trail $T' = (r_1, r'_2, r_3)$ was identified. This trail is present (as a HNET reaction set) when performing grouping by reactions for the reference species $S_3$.

Notice that from trail grouping by genes alone it is not possible to decide whether the reactions catalyzed by genes that receive a dot symbol (.) in the column corresponding to a species $S_1$ other than the reference species are absent from $S_1$ or performed by products of non neighboring genes. Trail grouping by reactions however distinguishes the two cases by assigning a dot symbol to reactions that are not catalyzed by products of neighboring genes in $S_1$, or a blank if the reaction in question is absent from $S_1$.

## 7 Concluding remarks

This chapter introduced trail grouping, a theoretical framework for the identification of conserved metabolic and genomic patterns across multiple species. With respect to a given reference species, the two proposed methods, trail grouping by reactions and by genes, identify conserved metabolic and genomic patterns, respectively. Jointly, the two methods allow to flexibly exploit trails detected using the trail finding method (Chapter IV). After a brief presentation of `CoMetGeNe` (Chapter VI), a pipeline designed to perform trail finding and trail grouping, concrete examples of (conserved) metabolic and genomic patterns will be presented and discussed in Chapter VII.

# VI

# The CoMetGeNe pipeline

# 1 Introduction

The previous chapters introduced trail finding (Chapter IV) and trail grouping (Chapter V), two methods that were designed during this thesis to identify conserved metabolic and genomic patterns across multiple species. This chapter describes how CoMetGeNe, especially created for this purpose, achieves trail finding and trail grouping in practice.

CoMetGeNe

CoMetGeNe, short for ***Co****nserved* ***Met****abolic and* ***Ge****nomic* ***Ne****ighborhoods*, is a Python pipeline implementing trail finding and trail grouping. Given one or several query species, CoMetGeNe detects sequences of reactions in metabolic pathways of the query species such that the reactions are catalyzed by products of neighboring genes. Trail grouping allows to group CoMetGeNe trails obtained for a given reference species by either reactions or genes.

# 2 Trail finding

Trail finding can be performed for one or several species using the convenient command-line interface proposed by the script CoMetGeNe.py (and the accompanying script CoMetGeNe_launcher.py for parallel execution). The only required information is the species to be analyzed (designated by its three- or four-letter KEGG identifier [KEGG Organisms]) and the directory where metabolic pathways of the species in question will be stored. The listing in Figure VI.1 details the command-line interface for CoMetGeNe.py and offers a usage example.

By default, the gap parameters $\delta_D$ and $\delta_G$ are set to 0, meaning that no reactions or genes, respectively, are skipped (see section IV.6). Optionally, other values can be assigned to these parameters using the options -dD and -dG in the listing in Figure VI.1. The trails produced by CoMetGeNe.py can be saved in an optional output file using the option -o in the listing in Figure VI.1.

## 2.1 Automatic data retrieval

CoMetGeNe.py automatically extracts the necessary metabolic and genomic information from KEGG using the KEGG REST API. Metabolic pathways are stored in KGML format (see section III.2.3 and [KGML]) in a user-specified directory (see DIR in the listing in Figure VI.1). Only metabolic pathway maps, excluding global and overview maps, are extracted (i.e., maps whose KEGG identifiers are greater than or equal to 01100 are excluded). Genomic information is stored in binary format. In addition, information linking EC numbers to R numbers is equally re-

```
usage: CoMetGeNe.py [-h] [--delta_G NUMBER] [--delta_D NUMBER]
                    [--timeout SECONDS] [--output OUTPUT] [--skip-import]
                    ORG DIR

Determines maximum trails of reactions for the specified organisms such that the
genes encoding the enzymes involved in the trails are neighbors.

A trail of reactions is a sequence of reactions that can repeat reactions
(vertices), but not arcs between reactions.

Metabolic pathways and genomic information are automatically retrieved from the
KEGG knowledge base.


Required arguments:
  ORG                   query organism (three- or four-letter KEGG code, e.g.
                        'eco' for Escherichia coli K-12 MG1655). See full list
                        of KEGG organism codes at
                        http://rest.kegg.jp/list/genome
  DIR                   directory storing metabolic pathways for the query
                        organism ORG or where metabolic pathways for ORG will
                        be downloaded

Optional arguments:
  -h, --help            show this help message and exit
  --delta_G NUMBER, -dG NUMBER
                        the NUMBER of genes that can be skipped (default: 0)
  --delta_D NUMBER, -dD NUMBER
                        the NUMBER of reactions that can be skipped (default: 0)
  --timeout SECONDS, -t SECONDS
                        timeout in SECONDS (default: 300)
  --output OUTPUT, -o OUTPUT
                        output file
  --skip-import, -s     skips importing metabolic pathways from KEGG,
                        attempting to use locally stored KGML files if they
                        are present under the specified directory (DIR)

Example: running

    python2 CoMetGeNe.py eco data/ -dG 2 -o eco.out

downloads metabolic pathways for species 'eco' to directory 'data/'. Trail
finding is performed, allowing two genes to be skipped at most (-dG 2).
Reactions cannot be skipped (-dD is 0 by default). Maximum trails of reactions
such that the reactions are catalyzed by products of neighboring genes are saved
in the output file 'eco.out'.
```

**Figure VI.1** Command-line options for `CoMetGeNe.py`

trieved and stored in binary format for subsequent runs; it is used exclusively for trail output.

Storing metabolic pathways and genomic information for a given species allows to perform trail finding without re-downloading the same data for subsequent executions, e.g. when `CoMetGeNe.py` is ran for the same species but with different gap parameters.

### 2.2 Blacklisted pathways

Since the underlying problem formulation for trail finding is NP-hard (see section IV.3), `CoMetGeNe.py` uses a configurable timeout (defaulting to 5 minutes) for analyzing a given metabolic pathway (see option `-t` in the listing in Figure VI.1). If this timeout is reached without producing any results, then the pathway in question is "blacklisted", i.e. it is added to a list of exclusions for the species and combination of gap parameters for which the analysis could not be finished. This prevents `CoMetGeNe` from further attempting to analyze the given pathway for subsequent executions if the gap parameters increase. For example, a pathway that is blacklisted for ($\delta_D = 2$, $\delta_G = 2$) will not be further analyzed for $(\delta_D, \delta_G) \in \{(2,2),(2,3),(3,2),(3,3)\}$. The blacklist is stored locally as a text file.

### 2.3 Parallel execution

An important speedup is attained if `CoMetGeNe.py` is ran in parallel using the accompanying script `CoMetGeNe_launcher.py`. Restrictions inherent to KEGG limit pathway and genomic information retrieval to 3 and 2 threads, respectively. Trail finding in `CoMetGeNe` can, however, take full advantage of the maximum number of physical threads.

Although `CoMetGeNe_launcher.py` does not provide a command-line interface, it can be easily configured to perform multithreaded trail finding. Thus, the desired list of species, the values of the gap parameters $\delta_D$ and $\delta_G$, as well as the directories storing metabolic pathways and trail finding results can be specified by modifying one or several variables.

## 3 Trail grouping

Once `CoMetGeNe` results are available for several species, trail grouping can be performed in order to identify conserved interspecies metabolic and genomic patterns, as described in sections V.4.1 and V.4.2. The script `grouping.py` provides this functionality and offers the possibility to save the tables $T'_S$ (trail grouping by

reactions) and $T_S^g$ (trail grouping by genes) for a given reference species in CSV format.

Three binary files are created when grouping trails by either reactions or genes. They contain pathway data, genomic information, and parsed `CoMetGeNe` results that can be reused when choosing another species as reference.

The listing in Figure VI.2 details the command-line interface for `grouping.py` and offers a usage example.

```
usage: grouping.py [-h] [--output OUTPUT] {genes,reactions} RESULTS KGML ORG

Groups CoMetGeNe trails by either genes or reactions, optionally producing
a CSV file.

Required arguments:
  {genes,reactions}     type of trail grouping to perform (possible values:
                        'genes' or 'reactions')
  RESULTS               directory storing CoMetGeNe results
  KGML                  directory containing input KGML files
  ORG                   reference species (KEGG organism code)

Optional arguments:
  -h, --help            show this help message and exit
  --output OUTPUT, -o OUTPUT
                        output file (CSV)

KGML needs to contain a subdirectory for every species for which a result file
is present in RESULTS. The subdirectory names need to be the three- or four-
letter KEGG codes for the species in question (e.g. 'bsu', 'eco', 'pae',
etc.). Each species subdirectory is expected to contain metabolic pathways in
KGML format.

Example: running

    python2 grouping.py genes results/ data/ eco -o grouping_gene_eco.csv

will perform trail grouping by genes for the reference species 'eco'. The
CoMetGeNe results are stored in 'results/', and the KGML files are available in
'data/'. A CSV file is produced ('grouping_gene_eco.csv').
```

**Figure VI.2** Command-line options for `grouping.py`

Note that phylogenetic relationships are not established automatically. Trail grouping as implemented by `grouping.py` displays species in Table VII.1 in phylogenetic order for any given reference species among the ones in the table. If other species are present in the data set, however, they are ordered lexicographically and a warning invites the user to manually define a phylogeny for the new species under study.

## 4 Requirements and availability

CoMetGeNe is a cross-platform pipeline written in Python. It requires Python 2.7 and the Python libraries lxml[1] and NetworkX[2].

In order to automatically extract metabolic pathway maps and genomic information from KEGG, CoMetGeNe needs an active internet connection. A multi-core CPU is recommended for faster (multithreaded) trail finding.

The CoMetGeNe pipeline is freely available under a MIT license and can be obtained at https://cometgene.lri.fr.

## 5 Concluding remarks

This chapter presented CoMetGeNe, a robust implementation of the trail finding and trail grouping methods, described in the previous two chapters.

The next chapter discusses several findings detected using CoMetGeNe, advancing it as an exploratory tool that allows biologists and bioinformaticians to easily identify conserved metabolic and genomic patterns between species they choose to study.

[1] lxml is available at the following address: http://lxml.de
[2] NetworkX is available at the following address: https://networkx.github.io

# VII

# Identification of metabolic and genomic patterns

# 1 Introduction

The previous chapter introduced `CoMetGeNe`, a pipeline implementing both trail finding (Chapter IV) and trail grouping (Chapter V). The present chapter illustrates several metabolic and genomic patterns identified using `CoMetGeNe` on a selected bacterial data set. Investigating the degree of conservation of these metabolic and genomic patterns reveals insights as well as surprising findings regarding links between genomic organization and metabolic architecture. Unexpectedly, careful analysis of `CoMetGeNe` results also calls attention to existing annotation problems in public knowledge bases.

This introduction presents the data set on which `CoMetGeNe` was executed (see 1.1), as well as an overview of `CoMetGeNe` results (see 1.2). Section 1.3 explains important aspects concerning the figures in this chapter.

The rest of the chapter discusses several examples of conserved metabolic and genomic patterns detected for *Bacillus subtilis* and *Escherichia coli*, in increasing order of relevance.

## 1.1 Bacterial data set

We have chosen to focus on prokaryotes because of their propensity for organization of genes into operons [Moreno-Hagelsieb, 2015]. Although eukaryotes exhibit gene clustering to a certain extent [Hurst *et al.*, 2004], such an organization is quite infrequent.

While the organization of prokaryotic genes into operons has long been known and studied, `CoMetGeNe` does not focus specifically on operons. It uncovers them if the resulting proteins are involved in consecutive steps in a metabolic pathway, but it also uncovers genes that are adjacent to operons if the proteins they encode belong to the same trail of reactions. For example, `CoMetGeNe` identifies a trail of six reactions for *E. coli* in the valine, leucine, and isoleucine biosynthesis pathway (`eco00290`) representing the conversion of threonine into leucine (data not shown). This trail involves five genes of *E. coli*, four of which constitute the *ilvMEDA* region of the *ilvLGMEDA* operon. The fifth gene, *ilvC*, is not part of this operon as its transcription is regulated by expression of *ilvY* [Wek and Hatfield, 1988].

For this study, a data set of 50 bacterial species spanning major phyla of the bacterial tree of life was chosen (Table VII.1). The data set is therefore representative of the whole bacterial domain.

Recall from Chapter V that when trail grouping is performed for the reference species, the remaining species in the data set are ordered by increasing evolutionary

| Species | Strain | Class | KEGG code |
|---|---|---|---|
| *Escherichia coli* | K-12 MG1655 | $\gamma$-proteobacteria | eco |
| *Yersinia pestis* | CO92 (biovar Orientalis) | $\gamma$-proteobacteria | ype |
| *Vibrio cholerae* | O395 | $\gamma$-proteobacteria | vco |
| *Shewanella putrefaciens* | CN-32 | $\gamma$-proteobacteria | spc |
| *Pseudomonas aeruginosa* | PAO1 | $\gamma$-proteobacteria | pae |
| *Xylella fastidiosa* | 9a5c | $\gamma$-proteobacteria | xfa |
| *Ralstonia solanacearum* | GMI1000 | $\beta$-proteobacteria | rso |
| *Neisseria meningitidis* | MC58 (serogroup B) | $\beta$-proteobacteria | nme |
| *Acidithiobacillus ferrivorans* | — | Acidithiobacillia | afi |
| *Agrobacterium radiobacter* | — | $\alpha$-proteobacteria | ara |
| *Rickettsia rickettsii* | Iowa | $\alpha$-proteobacteria | rrj |
| *Geobacter sulfurreducens* | PCA | $\delta$-proteobacteria | gsu |
| *Nitrospira defluvii* | — | Nitrospira | nde |
| *Acidobacterium capsulatum* | — | Acidobacteriales | aca |
| *Desulfurispirillum indicum* | — | Chrysiogenetes | din |
| *Fusobacterium nucleatum* | subsp. *nucleatum* ATCC 25586 | Fusobacteriia | fnu |
| *Denitrovibrio acetiphilus* | — | Deferribacteres | dap |
| *Thermodesulfatator indicus* | — | Thermodesulfobacteria | tid |
| *Aquifex aeolicus* | — | Aquificae | aae |
| *Bacillus subtilis* | subsp. *subtilis* 168 | Bacilli | bsu |
| *Listeria monocytogenes* | EGD-e | Bacilli | lmo |
| *Staphylococcus aureus* | subsp. *aureus* N315 (MRSA/VSSA) | Bacilli | sau |
| *Lactobacillus acidophilus* | NCFM | Bacilli | lac |
| *Streptococcus pneumoniae* | ST556 | Bacilli | snd |
| *Clostridium perfringens* | 13 | Clostridia | cpe |
| *Mycoplasma pneumoniae* | M129 | Mollicutes | mpn |
| *Synechocystis sp.* | PCC 6803 | Cyanobacteria (phylum) | syn |
| *Prochlorococcus marinus* | subsp. *marinus* CCMP1375 | Cyanobacteria (phylum) | pma |
| *Chloroflexus aurantiacus* | — | Chloroflexia | cau |
| *Bifidobacterium breve* | ACS-071-V-Sch8b | Actinobacteria | bbv |
| *Corynebacterium glutamicum* | ATCC 13032 (Kyowa Hakko) | Actinobacteria | cgl |
| *Mycobacterium tuberculosis* | H37Rv | Actinobacteria | mtv |
| *Streptomyces coelicolor* | — | Actinobacteria | sco |
| *Deinococcus radiodurans* | — | Deinococci | dra |
| *Thermus thermophilus* | HB27 | Thermi | tth |
| *Fimbriimonas ginsengisoli* | — | Fimbriimonadia | fgi |
| *Acetomicrobium mobile* | — | Synergistia | amo |
| *Thermotoga maritima* | MSB8 | Thermotogae | tmm |
| *Caldisericum exile* | — | Caldisericia | cex |
| *Dictyoglomus thermophilum* | — | Dictyoglomia | dth |
| *Fibrobacter succinogenes* | — | Fibrobacteria | fsu |
| *Gemmatimonas aurantiaca* | — | Gemmatimonadetes | gau |
| *Chlorobium phaeobacteroides* | DSM 266 | Chlorobia | cph |
| *Bacteroides fragilis* | YCH46 | Bacteroidia | bfr |
| *Rhodopirellula baltica* | — | Planctomycetia | rba |
| *Chlamydia pneumoniae* | CWL029 | Chlamydiia | cpn |
| *Opitutus terrae* | — | Opitutae | ote |
| *Borrelia burgdorferi* | N40 | Spirochaetia | bbn |
| *Elusimicrobium minutum* | — | Elusimicrobia | emi |
| *Helicobacter pylori* | 26695 | $\varepsilon$-proteobacteria | heo |

**Table VII.1** The data set of 50 bacterial species chosen for this study

distance to the reference species. The ordering of species in the data set with respect to the reference species uses the phylogeny in Figure 2 by Rinke *et al.* [2013].

Note that phylogeny, especially bacterial phylogeny, is an ever-moving field. Yarza *et al.* [2014] have pointed out that, in the case of proteobacteria, only Alphaproteobacteria, Betaproteobacteria, and Gammaproteobacteria form a monophyletic group. Deltaproteobacteria and Epsilonproteobacteria show an important divergence from this monophyletic group in terms of 16S ribosomal RNA sequence, which led the authors to propose that the Proteobacteria phylum rank be reconsidered. More recently, Parks *et al.* [2017] have shown that the Deltaproteobacteria class is polyphyletic. The phylogenetic relationships used in trail grouping between the species in Table VII.1 do not incorporate these recent findings, as the work presented herein has been started before the latest relevant study.

## 1.2 Overview of `CoMetGeNe` results

Trail finding and trail grouping were performed on the bacterial data set in Table VII.1, with gap parameters $\delta_D$ and $\delta_G$ ranging from 0 to 3 (see section IV.6). Genome size varies between 1062 and 8300 genes, with an average of approximately 3270 genes. In total, 3709 pathways were extracted (74 pathways per species, on average). Metabolic and genomic data used in the examples presented in sections 2, 3, and 4 were extracted from KEGG in June 2018. The data for the case study in section 5 was extracted in September 2016, for reasons explained in section III.3.1.

Using the metabolic and genomic information extracted from KEGG in June 2018, a total of 4179 `CoMetGeNe` trails were identified. Of these, 2620 (62.7%) occur solely in a single species. The number of trails per species varies between 19 and 501, with an average of 201 trails. Table VII.2 shows trail span distribution (recall that the span of a `CoMetGeNe` trail represents the number of distinct reactions in the trail). The majority of trails are short, consisting of up to three distinct reactions. Other trails, however, have as many as 35 unique reactions, e.g. for the fatty acid biosynthesis pathway in *Bifidobacterium breve* (`bbv00061`) and *Streptococcus pneumoniae* (`snd00061`). A total of 121 out of 3709 pathways were blacklisted, amounting to 3.3% of the data set (see section VI.2.2).

| Trail span | Percentage of trails |
|---|---|
| 1−3 | 56.4% |
| 4−10 | 38.7% |
| 11−35 | 4.9% |

**Table VII.2** Distribution of trail span

The trail finding run time for CoMetGeNe for the whole data set of 50 bacterial species (Table VII.1) was under 4 hours and 30 minutes when using 8 threads.[1] The trail finding run time does not take into account the time required to automatically retrieve data from KEGG, as this is dependent upon the Internet connexion speed and upon the number and size of the selected genomes. In the experimental setup used in this thesis, metabolic pathways and genomic information were retrieved in 12 and 76 minutes, respectively. When each of the species in the data set is taken in turn as reference species, trail grouping by reactions and by genes takes approximately one hour in total. Thus, data retrieval from KEGG for the data set in Table VII.1, followed by trail finding and trail grouping, amounted to approximately 7 hours.

Considering the quantity of metabolic and genomic data to be retrieved and analyzed, as well as the exponential nature of the HNET algorithm (see section IV.5) due to MaSST and MaSSCoT being NP-hard (see section IV.3), the total trail finding run time (including data retrieval) for the selected data set was quite satisfactory, amounting to less than 6 hours. Moreover, CoMetGeNe execution time is linear with respect to the number of species to analyze.

## 1.3 Figure information

For the figures in this chapter illustrating trail grouping, the colors used in the table headers represent the bacterial superphylum. Alphaproteobacteria, Betaproteobacteria, Gammaproteobacteria, and Deltaproteobacteria are highlighted in pink; Terrabacteria, in brown; Sphingobacteria (FCB bacteria), in yellow; and Planctobacteria (PVC bacteria), in light green.

The naming scheme for *E. coli* genes uses the Blattner identifiers or *b* numbers. Consecutive *b* numbers usually reflect neighboring genes (e.g. the genes *b0086* and *b0087* are consecutive).

The naming scheme for *B. subtilis* genes has the form *BSUXXXX0*, where *X* is a digit. Increments of 10 in identifiers of *B. subtilis* genes usually reflect neighboring genes (e.g. the genes *BSU28300* and *BSU28310* are consecutive).

[1] The test machine was a quad-core 2.6 GHz Intel Xeon E5-2623 v4 (Broadwell) with 10 MB L3 cache and 64 GB of RAM, running under Ubuntu GNU/Linux 16.04.3 LTS. Although the test machine has 64 GB of main memory, running CoMetGeNe on a single thread only requires approximately 100 MB of RAM.

## 2 Branching in metabolic pathways

Figure VII.1 shows a `CoMetGeNe` trail for *Bacillus subtilis* in the valine, leucine, and isoleucine biosynthesis pathway, representing the conversion of pyruvate into a precursor of leucine. `CoMetGeNe` produced this trail by skipping the reaction `R04441` (EC 4.2.1.9), with gap parameter $\delta_D$ set to 1.

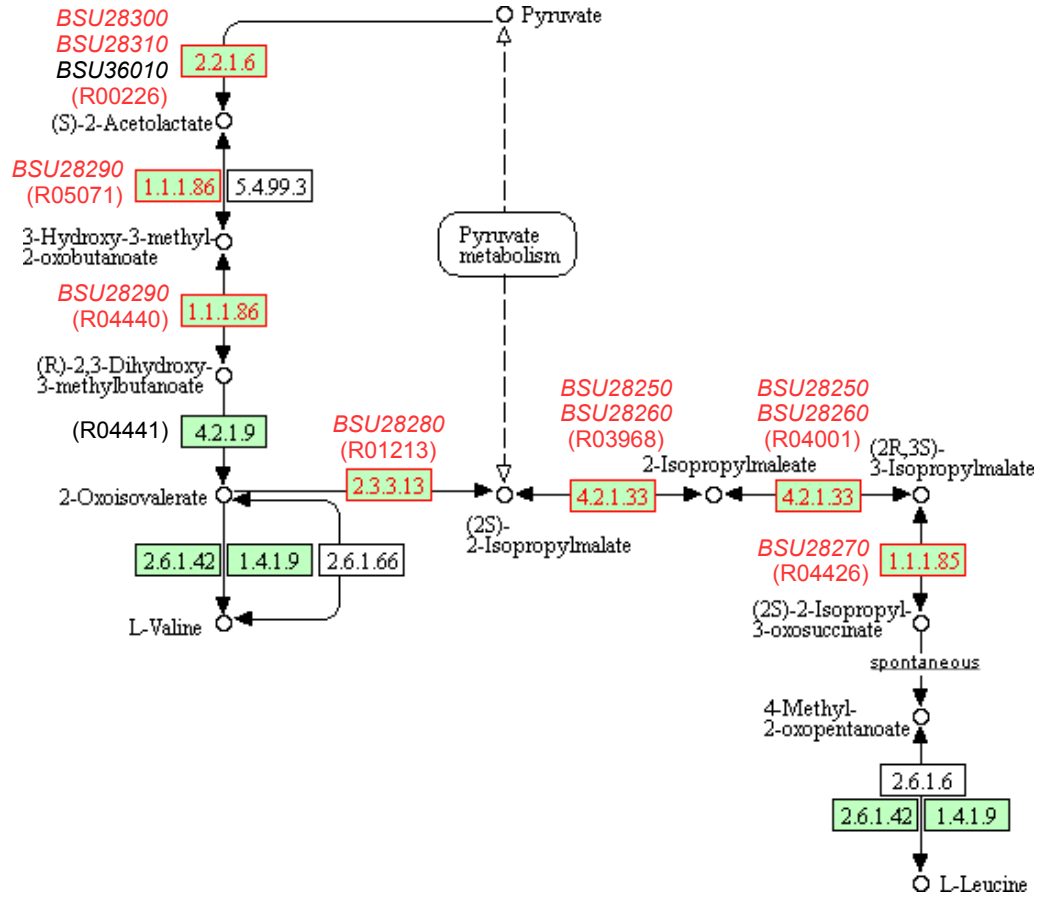



**Figure VII.1 Partial view of the valine, leucine, and isoleucine biosynthesis pathway in *Bacillus subtilis*.** Adapted from KEGG PATHWAY, map `bsu00290` (March 7, 2017 version). Shown here is a `CoMetGeNe` trail consisting in the reactions with red contours. Reactions in the trail are labeled with the corresponding KEGG reaction identifiers (R numbers) and with the gene identifiers of the genes involved in the reactions. The reaction `R04441` performing the enzymatic activity 4.2.1.19 was skipped ($\delta_D = 1$). Genes with black identifiers do not belong to the gene group in Figure VII.2. Genes with red identifiers are neighbors on the negative strand of the *B. subtilis* chromosome.

Figures VII.2 and VII.3 respectively show the corresponding grouping by genes and by reactions for *B. subtilis* as reference species and 30 other bacteria from the

data set (members of the Terrabacteria superphylum, Gammaproteobacteria, and 8 other species). Trail grouping by genes and by reactions for the full data set is presented in Figures C.1 and C.2, respectively.

| *B. subtilis* gene | lmo | sau | lac | snd | cpe | mpn | syn | pma | cau | bbv | cgl | mtv | sco | dra | tth | fgi | cex | gau | cpn | bbn | emi | heo | fnu | eco | ype | vco | spc | pae | xfa | rrj |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *BSU28250* | x | x | . | . | . | . | . | x | x | . | . | x | . | . | . | . | . | . | . | . | x | . | . | x | x | x | x | . | . | . |
| *BSU28260* | x | x | . | . | . | . | . | x | x | . | . | x | . | . | . | . | . | . | . | . | x | . | . | x | x | x | x | . | . | . |
| *BSU28270* | x | x | . | . | . | . | . | . | . | . | . | x | . | . | . | . | . | . | . | . | . | . | . | x | x | x | x | . | . | . |
| *BSU28280* | x | x | . | . | . | . | . | . | x | . | . | . | . | . | x | . | . | . | . | . | . | . | . | x | x | x | x | . | x | . |
| *BSU28290* | x | x | . | x | . | . | . | . | x | . | x | x | x | x | x | x | . | . | . | . | . | . | . | . | . | . | . | x | x | . |
| *BSU28300* | x | x | . | x | . | . | . | . | x | x | x | x | x | x | x | x | . | . | . | . | . | . | . | . | . | . | . | x | x | . |
| *BSU28310* | x | x | . | x | . | . | . | . | x | x | x | x | x | x | x | x | . | . | . | . | . | . | . | . | . | . | . | x | x | . |

**Figure VII.2 Group of homologous genes involved in the trail in Figure VII.1.** Columns in gray correspond to species without neighboring functionally similar genes to the genes in *B. subtilis* involved in this trail. Cells in light yellow represent species that have neighboring functionally similar genes to at least two *B. subtilis* genes involved in the trail, but not for the gene *BSU28280* involved in the reaction `R01213` (EC 2.3.3.13). Colors in the table header designate the bacterial superphylum (see section 1.3 for details). See Figure C.1 for the grouping by genes corresponding to this trail for all the species in the data set.

| reaction | *B. subtilis* gene | lmo | sau | lac | snd | cpe | mpn | syn | pma | cau | bbv | cgl | mtv | sco | dra | tth | fgi | cex | gau | cpn | bbn | emi | heo | fnu | eco | ype | vco | spc | pae | xfa | rrj |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R03968 | *BSU28250 BSU28260* | x | x | | x | | | . | x | x | . | . | x | . | . | . | . | | | | | x | | | x | x | x | x | x | . | |
| R04001 | *BSU28250 BSU28260* | x | x | | x | | | . | x | x | . | . | x | . | . | . | . | | | | | x | | | x | x | x | x | x | . | |
| R04426 | *BSU28270* | x | x | | x | | | . | . | . | . | . | x | . | . | . | . | | | | | | | | x | x | x | x | x | . | |
| R01213 | *BSU28280* | x | x | | x | | | . | . | x | . | . | . | . | . | x | . | | | | | | | | x | x | x | x | . | x | |
| R05071 | *BSU28290* | x | x | | . | | | . | . | x | x | x | x | x | x | x | x | | | | | | . | | . | . | . | . | . | x | |
| R04440 | *BSU28290* | x | x | | . | | | . | . | x | x | x | x | x | x | x | x | | | | | | . | | . | . | . | . | . | x | |
| R00226 | *BSU28300 BSU28310 BSU36010* | x | x | | . | | | . | . | x | . | x | x | x | x | x | x | | | | | | | | . | . | . | . | . | x | |

**Figure VII.3 Group of reactions defining the trail in Figure VII.1.** The cells in gray correspond to species lacking all or a vast majority of reactions from this trail. Cells in light yellow represent species that have neighboring functionally similar genes to *B. subtilis* genes involved in at least two reactions in the trail, but not in reaction `R01213` (EC 2.3.3.13). Cells in blue and orange correspond to species having neighboring functionally similar genes to *B. subtilis* genes involved in the last and first three reactions in the trail, respectively. Colors in the table header designate the bacterial superphylum (see section 1.3 for details). See Figure C.2 for the grouping by reactions corresponding to this trail for all the species in the data set.

A total of six species perform all the reactions in the trail in Figure VII.1 using products of neighboring functionally similar genes to *B. subtilis* genes involved in this trail (species highlighted in purple in Figure C.1). In contrast, 12 species in the data set (almost 25%) do not have neighboring functionally similar genes to genes in *B. subtilis* involved in this trail. They are highlighted in gray in Figures VII.2 and

C.1. Figure VII.3 shows that, among these species:

- *Synechocystis sp.* PCC 6803 (`syn`) performs every reaction in this `CoMetGeNe` trail using products of distant genes.
- *Elusimicrobium minutum* (`emi`) and *Helicobacter pylori* (`heo`) only perform two of the reactions in the trail. In effect, both species have been shown to require certain amino acids for growth, including valine, leucine and isoleucine [Herlemann *et al.*, 2009; Reynolds and Penn, 1994].
- The nine remaining species lack every reaction in the trail.

From the trail grouping by reactions in Figures VII.3 and C.2 it is apparent that the first three (`R00226`, `R05071`, `R04440`) as well as the last three reactions (`R03968`, `R04001`, `R04426`) in the trail are often performed by products of neighboring functionally similar genes to genes in *B. subtilis* (species highlighted in orange and blue, respectively, in the two figures). The reaction in between, `R01213` (EC 2.3.3.13), involves the product of gene *BSU28280* in *B. subtilis*. Among the species in the data set having at least two neighboring functionally similar genes to *B. subtilis* genes involved in the trail in Figure VII.1:

- Slightly more than half of the species (19 out of 36 species) perform `R01213` using the product of a gene neighboring other genes involved in the trail (Figure C.2).
- Slightly less than half of the species (17 out of 36 species) perform `R01213` using the product of distant genes from other neighboring genes involved in the trail (see the species highlighted in light yellow in Figures VII.3 and C.2).

These observations naturally lead to inquire into likely reasons for which the genes whose products are involved in either the first three of the last three reactions in the `CoMetGeNe` trail in Figure VII.1 seem to be constrained to be adjacent, unlike the gene whose product catalyzes the reaction `R01213`.

A possible explanation is the fact that reaction `R01213` is the branching point between the valine and the leucine biosynthesis pathways (Figure VII.1). The metabolite 2-oxoisovalerate can serve as substrate for either valine or 2-isopropylmalate. It would therefore make sense that certain species optimize valine biosynthesis by keeping the required genes in physical proximity on the chromosome, while others focus on leucine production once the branching point between the two pathways is reached.

This example suggests that branching points in metabolic pathways sometimes imply certain gene arrangements that favor a particular branch.

# 3 Conserved metabolic and genomic sub-patterns

Figure VII.4 shows a `CoMetGeNe` trail for *Bacillus subtilis* in the purine metabolism pathway, representing the conversion of glutamine and phosphoribosyl pyrophosphate (PRPP) into inosine monophosphate (IMP), an important intermediate in purine metabolism.

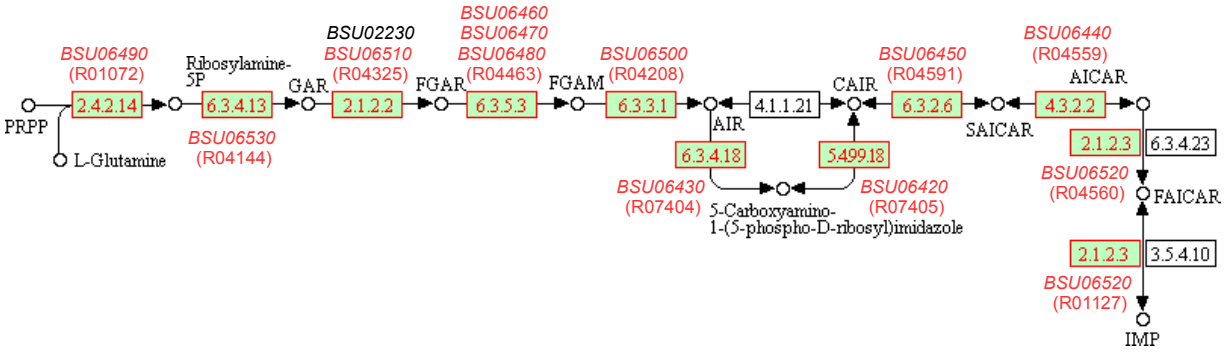



**Figure VII.4 Partial view of the purine metabolism pathway in *Bacillus subtilis*.** Adapted from KEGG PATHWAY, map `bsu00230` (April 11, 2018 version). Shown here is a `CoMetGeNe` trail consisting in the reactions with red contours. Reactions in the trail are labeled with the corresponding KEGG reaction identifiers (R numbers) and with the gene identifiers of the genes involved in the reactions. Genes with black identifiers do not belong to the gene group in Figure VII.5. Genes with red identifiers are neighbors on the positive strand of the *B. subtilis* chromosome.

Figures VII.5 and VII.6 respectively show the corresponding trail grouping by genes and by reactions for *B. subtilis* as reference species and 27 other bacteria from the data set. Trail grouping by genes and by reactions for the full data set is presented in Figures C.3 and C.4, respectively. Figure C.3 shows that approximately one fifth of the species in the data set present neighboring functionally similar genes encoding most of the enzymes for the different steps in the trail in Figure VII.4. These species are present in Figure VII.5 and are generally close to *B. subtilis* from a phylogenetic point of view, as indicated by the cross symbols ($\times$) concentrated mainly at the left of figure.

Several species receive only dot symbols (`.`) in the grouping by genes in Figure VII.5, meaning they either do not have neighboring functionally similar genes to genes of *B. subtilis* involved in the trail, or that they do not perform the reactions in the trail. As it turns out from the trail grouping by reactions presented in Figure VII.6, for the six species highlighted in gray:

- The trail is entirely absent for *Mycoplasma pneumoniae* (`mpn`), *Caldisericum exile* (`cex`), *Chlamydia pneumoniae* (`cpn`), and *Borrelia burgdorferi* (`bbn`).

| *B. subtilis* gene | lmo | sau | lac | snd | cpe | mpn | amo | tmm | cex | dth | fsu | cpn | ote | bbn | emi | heo | din | fnu | dap | eco | ype | vco | spc | pae | xfa | rrj | gsu |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *BSU06420* | x | x | x | x | x | . | x | . | . | . | x | . | . | . | x | . | . | x | . | . | . | . | . | . | . | . | . |
| *BSU06430* | x | x | x | x | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . |
| *BSU06440* | x | . | . | x | . | . | . | . | . | x | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | x |
| *BSU06450* | x | x | x | x | x | . | x | x | . | . | . | . | . | . | x | . | x | x | . | . | . | . | . | . | . | . | . |
| *BSU06460* | x | x | x | x | x | . | x | x | . | x | . | . | . | . | x | . | x | x | x | . | . | . | . | . | . | . | x |
| *BSU06470* | x | x | x | x | x | . | x | x | . | x | . | . | . | . | x | . | x | x | x | . | . | . | . | . | . | . | x |
| *BSU06480* | x | x | x | x | x | . | x | x | . | x | . | . | . | . | x | . | x | x | x | . | . | . | . | . | . | . | x |
| *BSU06490* | x | x | x | x | x | . | x | x | . | x | . | . | x | . | x | . | . | x | x | . | . | . | . | . | . | . | x |
| *BSU06500* | x | x | x | x | x | . | x | x | . | x | . | . | x | . | x | . | . | x | . | x | x | x | x | x | x | . | . |
| *BSU06510* | x | x | x | x | x | . | x | x | . | x | . | . | . | . | x | . | . | x | . | x | x | x | x | x | x | . | . |
| *BSU06520* | x | x | x | x | x | . | x | x | . | . | . | . | . | . | x | . | . | x | . | . | . | . | . | . | . | . | . |
| *BSU06530* | x | x | x | x | x | . | x | x | . | x | x | . | . | . | x | . | . | x | . | . | . | . | . | . | . | . | . |

**Figure VII.5 Group of homologous genes involved in the trail in Figure VII.4.** Colors in the table header designate the bacterial superphylum (see section 1.3 for details). See Figure C.3 for the grouping by genes corresponding to this trail for all the species in the data set.

| reaction | *B. subtilis* gene | lmo | sau | lac | snd | cpe | mpn | amo | tmm | cex | dth | fsu | cpn | ote | bbn | emi | heo | din | fnu | dap | eco | ype | vco | spc | pae | xfa | rrj | gsu |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R07405 | *BSU06420* | x | x | x | x | x | | x | . | | . | x | | . | | x | | . | x | x | . | . | . | . | . | . | | x |
| R07404 | *BSU06430* | x | x | x | x | | | | . | | | | | | | | | | | | . | . | . | . | . | . | | |
| R04559 | *BSU06440* | x | . | . | x | . | | . | . | | x | . | | . | | . | . | . | . | . | . | . | . | . | . | . | | . |
| R04591 | *BSU06450* | x | x | x | x | x | | x | x | | . | . | | . | | x | | . | x | . | . | . | . | . | . | . | . | . |
| R04463 | *BSU06460 BSU06470 BSU06480* | x | x | x | x | x | | x | x | | x | . | | . | | x | | . | x | . | . | . | . | . | . | . | | . |
| R01072 | *BSU06490* | x | x | x | x | x | | x | x | | x | . | | x | | x | | . | x | . | . | . | . | . | . | . | | . |
| R04208 | *BSU06500* | x | x | x | x | x | | x | x | | x | . | | x | | x | | x | x | . | . | . | . | . | . | . | | . |
| R04325 | *BSU02230 BSU06510* | x | x | x | x | x | | x | x | | x | . | | . | | x | | x | x | . | . | . | . | . | . | . | | . |
| R04560 | *BSU06520* | x | x | x | x | x | | x | x | | . | . | | . | | x | | . | x | x | x | x | x | x | x | x | | x |
| R01127 | *BSU06520* | x | x | x | x | x | | x | x | | . | . | | . | | x | | . | x | x | x | x | x | x | x | x | | x |
| R04144 | *BSU06530* | x | x | x | x | x | | x | x | | x | x | | . | | x | . | x | x | x | x | x | x | x | x | x | | x |

**Figure VII.6 Group of reactions defining the trail in Figure VII.4.** Cells in gray correspond to species lacking all or a vast majority of reactions from this trail. Cells in light yellow correspond to species that do not perform the reaction `R07404`, if these species possess neighboring functionally similar genes for at least two reactions in the trail. Cells in blue correspond to the maximum set of reactions among the reactions in the trail that are common to different species and performed by neighboring functionally similar genes in these species. Cells in orange correspond to reactions performed by products of neighboring genes in Gammaproteobacteria. Colors in the table header designate the bacterial superphylum (see section 1.3 for details). See Figure C.4 for the grouping by reactions corresponding to this trail for all the species in the data set.

- The trail is partially absent for *Helicobacter pylori* (heo) and *Rickettsia rickettsii* (rrj), with only two and one reaction present, respectively.

The reaction R07404 (EC 6.3.4.18) stands out among the reactions in Figure VII.6 because it is absent in several of the species exhibiting neighboring genes involved in at least two reactions in the trail. The ten species in question are highlighted in light yellow in Figure VII.6 and perform the reaction R07405 (EC 5.4.99.18), immediately following R07404. The two reactions R07404 (EC 6.3.4.18) and R07405 (EC 5.4.99.18) provide an alternative route leading from aminoimidazole ribotide (AIR) to 5'-phosphoribosyl-4-carboxy-5-aminoimidazole (CAIR) instead of the direct route represented by the reaction R04209 (EC 4.1.1.21 in Figure VII.4). It has been shown that EC 4.1.1.21 is the alternative present in vertebrates to convert AIR to CAIR, whereas bacteria prefer the other alternative involving EC 6.3.4.18 and EC 5.4.99.18 [Firestine *et al.*, 1994]. The enzymatic activity EC 6.3.4.18 is also reported absent in the ten species for the superpathway of purine nucleotides *de novo* biosynthesis II (DENOVOPURINE2-PWY) in MetaCyc. As the information on EC 6.3.4.18 is coherent between KEGG and MetaCyc, it would appear that, in the case of the ten species highlighted in light yellow in Figure VII.6, no suitable candidate gene has yet been determined as encoding the enzyme performing this step.

Cells highlighted in blue in Figure VII.6 correspond to the maximum set of reactions in the trail that are catalyzed by products of functionally similar genes in several species:

- *Listeria monocytogenes* (lmo), *Staphylococcus aureus* (sau), *Lactobacillus acidophilus* (lac), *Streptococcus pneumoniae* (snd), and *Clostridium perfringens* (cpe), all members of the Firmicutes phylum;
- *Acetomicrobium mobile* (amo), a member of the Synergistetes phylum;
- *Thermotoga maritima* (tmm), a member of the Thermotogae phylum;
- *Elusimicrobium minutum* (emi), a member of the Elusimicrobia phylum;
- *Fusobacterium nucleatum* (fnu), a member of the Fusobacteria phylum.

As shown in Figure VII.6, the eight reactions highlighted in blue always involve neighboring genes, whereas the remaining reactions might involve neighboring genes for the species in question. While gene order conservation is to be expected to some extent for closely related species (*B. subtilis* and the five other Firmicutes), it is not clear how or why the same pattern occurs in the four other species listed above. Since not much is known about *A. mobile* (amo) and *E. minutum* (emi), additional information on these species' environment and lifestyle might contribute to explain the conserved metabolic and genomic pattern detected here.

Additionally, a more intriguing conserved metabolic and genomic pattern among closely related species exists. Upon initial consideration, it would appear that only the three reactions highlighted in orange in Figure VII.6 are catalyzed by products of neighboring functionally similar genes to *BSU06520* and *BSU06530* in the six Gammaproteobacteria species in the data set. From the trail grouping by genes in Figure VII.5, however, it can be seen that these six species also have neighboring functionally similar genes to *BSU06500* and *BSU06510*. Closer inspection reveals an interesting *sub-pattern of conserved metabolic and genomic organization* for the six Gammaproteobacteria, as illustrated in Figures VII.7 and VII.8.

*metabolic and genomic sub-pattern*

| reaction | *B. subtilis* gene | eco | ype | vco | spc | pae | xfa |
|---|---|---|---|---|---|---|---|
| R07405 | *BSU06420* | *b0523* | *YPO3076* | *VC0395_A2468* | *Sputcn32_1041* | *PA5425* | *XF_2672* |
| R07404 | *BSU06430* | *b0522* | *YPO3077* | *VC0395_A2467* | *Sputcn32_1040* | *PA5426* | *XF_2671* |
| R04559 | *BSU06440* | *b1131* | *YPO1636* | *VC0395_A0644* | *Sputcn32_2235* | *PA2629 PA3516 PA3517* | *XF_1553* |
| R04591 | *BSU06450* | *b2476* | *YPO3059* | *VC0395_A0811* | *Sputcn32_0608* | *PA1013* | *XF_0205* |
| R04463 | *BSU06460 BSU06470 BSU06480* | *b2557* | *YPO2921* | *VC0395_A0395* | *Sputcn32_2642* | *PA3763* | *XF_1423* |
| R01072 | *BSU06490* | *b2312* | *YPO2772* | *VC0395_A0525* | *Sputcn32_2437* | *PA3108* | *XF_1949* |
| R04208 | *BSU06500* | *b2499* | *YPO2828* | *VC0395_A1819* | *Sputcn32_1596* | *PA0945* | *XF_0587* |
| R04325 | *BSU02230* ***BSU06510*** | *b1849* ***b2500*** | *YPO1775* ***YPO2829*** | *VC0395_A0850* ***VC0395_A1820*** | *Sputcn32_1001* ***Sputcn32_1595*** | ***PA0944*** *PA3451* | *XF_0585* |
| R04560 | *BSU06520* | *b4006* | *YPO3728* | *VC0395_A2653* | *Sputcn32_3401* | *PA4854* | *XF_1975* |
| R01127 | *BSU06520* | *b4006* | *YPO3728* | *VC0395_A2653* | *Sputcn32_3401* | *PA4854* | *XF_1975* |
| R04144 | *BSU06530* | *b4005* | *YPO3729* | *VC0395_A2652* | *Sputcn32_3402* | *PA4855* | *XF_1976* |

**Figure VII.7 The group of reactions shown in Figure VII.6 for Gammaproteobacteria.** For each species, identifiers of genes encoding the enzymes involved each reaction are shown in the corresponding cells. Gene identifiers in bold for `R04325` designate the genes neighboring those involved in the reaction `R04208`. Rows with the same background color correspond to reactions from the trail in Figure VII.4 catalyzed by products of neighboring genes. Color-coded reactions are shown in their metabolic context in Figure VII.8.

Thus, for each of the six species of $\gamma$-proteobacteria in Figure VII.7:

- The three reactions highlighted in orange are catalyzed by two neighboring genes (one gene for `R04144` and a second gene for both `R04560` and `R01127`).
- The two reactions highlighted in light orange are catalyzed by products of neighboring genes. For *Escherichia coli* (`eco`), *Yersinia pestis* (`ype`), *Shewanella putrefaciens* (`spc`), and *Pseudomonas aeruginosa* (`pae`), a distant second gene is equally involved in reaction `R04325`. For *Xylella fastidiosa* (`xfa`), only one gene

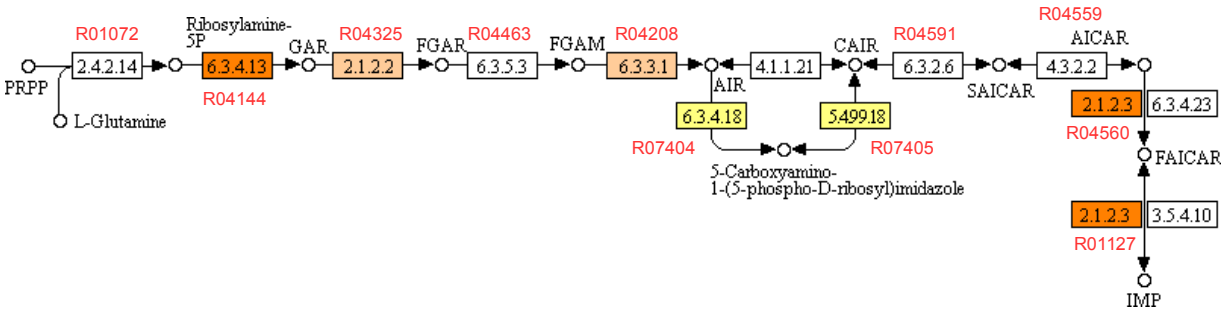



**Figure VII.8 Partial view of the general purine metabolism pathway.** Adapted from KEGG PATHWAY, `map00230` (April 11, 2018 version). The reactions involved in the trail in Figure VII.4 are labeled with their respective R numbers. Reactions with the same color are catalyzed by products of neighboring genes in the six Gammaproteobacteria in Figure VII.7. Color-coded reactions are the same as in Figure VII.7.

(*XF_0585*) is involved in this reaction. The genes *XF_0585* and *XF_0587* in *X. fastidiosa* are not strict neighbors, being separated by the gene *XF_0586* which encodes a hypothetical protein.

- The two reactions highlighted in yellow are catalyzed by products of neighboring genes.

Although the genes involved in the 7 out of the 11 reactions present in Figure VII.8 highlighted with the same color code as in Figure VII.7 are not neighbors between themselves for the six species, they represent pairs of neighboring genes. Moreover, since the six species in question are closely related in terms of phylogeny, it seems highly probable that the six extant (current-day) species of Gammaproteobacteria preserved this particular genomic organization, having inherited it from a common ancestor.

This example identifies two different conserved metabolic and genomic patterns among closely related species. The first pattern involves strictly neighboring functionally similar genes (cells highlighted in blue in Figure VII.6), whereas the second one is actually a *sub-pattern* involving pairs of neighboring functionally similar genes for groups of two or three reactions (Figures VII.7 and VII.8).

*metabolic and genomic sub-pattern*

## 4 Discovery of unexpected gene ordering patterns

Figure VII.9 shows a `CoMetGeNe` trail for *Escherichia coli* in the glycine, serine, and threonine metabolism pathway (`eco00260`), representing the conversion of aspartate into threonine. `CoMetGeNe` produced this trail by skipping the reaction `R02291` (EC 1.2.1.11), with gap parameter $\delta_D$ set to 1.

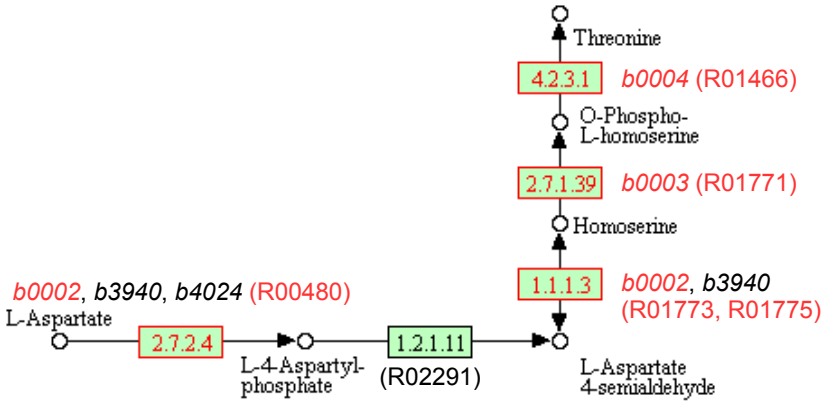



**Figure VII.9 Partial view of the glycine, serine, and threonine metabolism pathway in *Escherichia coli*.** Adapted from KEGG PATHWAY, map `eco00260` (October 26, 2017 version). Shown here is a `CoMetGeNe` trail consisting in the reactions with red contours. Reactions in the trail are labeled with the corresponding KEGG reaction identifiers (R numbers) and with the gene identifiers of the genes involved in the reactions. The reaction `R02291` performing the enzymatic activity 1.2.1.11 was skipped ($\delta_D = 1$). Genes with black identifiers do not belong to the gene group in Figure VII.10. Genes with red identifiers are neighbors on the positive strand of the *E. coli* chromosome.

Figures VII.10 and VII.11 respectively show the corresponding grouping by genes and by reactions for *E. coli* as reference species and 30 other bacteria from the data set. Trail grouping by genes and by reactions for the full data set is presented in Figures C.5 and C.6, respectively. In the case of the 11 species highlighted in light yellow in Figure VII.10, functionally similar genes to *b0003* are not neighbors of functionally similar genes to *b0002* and *b0004*. The relevant genomic context for these species and two additional ones, *Denitrovibrio acetiphilus* (`dap`) and *Rhodopirellula baltica* (`rba`), is shown in Figure VII.12.

Figure VII.11 shows that, of the species highlighted in light yellow in Figure VII.10, *Caldisericum exile* (`cex`), *Gemmatimonas aurantiaca* (`gau`), and *Bacteroides fragilis* (`bfr`) do not perform reaction `R01771` (EC 2.7.1.39), in which the product of gene *b0003* is involved (species highlighted in gray). Only *Lactobacillus acidophilus* (`lac`) conserved the functionally similar gene *LBA1211* as a neighbor of the gene performing the reaction `{R01773, R01775}` (see also Figure VII.12). The functionally similar genes to *b0003* for the other species highlighted in light yellow in Figure VII.10 exist, but they are located farther on the bacterial chromosome.

Figure VII.12 shows that strictly neighboring functionally similar genes involved in reactions `{R01773, R01775}` (EC 1.1.1.3, in green) and `R01466` (EC 4.2.3.1, in blue) are conserved for *Pseudomonas aeruginosa* (`pae`), *Ralstonia solanacearum* (`rso`), *Acidithiobacillus ferrivorans* (`afi`), *Nitrospira defluvii* (`nde`), and *Desulfurispirillum in-*

| *E. coli* gene | ype | pae | xfa | rso | nme | afi | gsu | nde | din | dap | aae | bsu | sau | lac | mpn | syn | pma | tth | fgi | cex | fsu | gau | cph | bfr | rba | cpn | ote | bbn | emi | heo |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *b0002* | x | x | x | x | . | x | . | x | x | x | . | x | x | x | . | . | . | x | . | x | . | x | . | x | . | . | x | . | x | . |
| *b0003* | x | . | x | . | . | . | . | . | . | . | . | x | x | . | . | . | . | . | x | . | . | . | x | . | . | . | . | . | x | . |
| *b0004* | x | x | x | x | . | x | . | x | x | . | . | x | x | x | . | . | . | x | x | x | . | x | x | x | . | . | x | . | x | . |

**Figure VII.10 Group of homologous genes involved in the trail in Figure VII.9.** Eleven of the species in the data set (highlighted in light yellow) either do not have functionally similar genes to *b0003*, or are not contiguous with genes functionally similar to *b0002* and *b0004*. Colors in the table header designate the bacterial superphylum (see section 1.3 for details). See Figure C.5 for the grouping by genes corresponding to this trail for all the species in the data set.

| reaction | *E. coli* gene | ype | pae | xfa | rso | nme | afi | gsu | nde | din | dap | aae | bsu | sau | lac | mpn | syn | pma | tth | fgi | cex | fsu | gau | cph | bfr | rba | cpn | ote | bbn | emi | heo |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R00480 | *b0002 b3940 b4024* | x | . | x | . | . | . | . | x | x | x | . | . | . | . | | . | . | . | . | x | . | x | . | x | . | . | x | | . | . |
| {R01773, R01775} | *b0002 b3940* | x | x | x | x | . | x | . | x | x | x | . | x | x | x | | . | . | x | . | x | . | x | . | x | . | | . | | x | . |
| R01771 | *b0003* | x | . | x | . | . | . | | . | . | | . | x | x | x | | . | . | . | x | | . | | x | | | | . | | x | . |
| R01466 | *b0004* | x | x | x | x | . | x | . | x | x | . | . | x | x | . | | . | . | x | x | x | . | x | x | x | . | | x | | x | . |

**Figure VII.11 Group of reactions defining the trail in Figure VII.9.** Cells highlighted in gray correspond to the three species among the ones highlighted in light yellow in Figure VII.10 that do not perform reaction `R01771` (catalyzed by the product of gene *b0003* in *E. coli*). Colors in the table header designate the bacterial superphylum (see section 1.3 for details). See Figure C.6 for the grouping by reactions corresponding to this trail for all the species in the data set.

*dicum* (`din`). Interestingly, bi-functional enzymes catalyzing both reactions `R00480` (EC 2.7.2.4, in yellow) and `{R01773, R01775}` (EC 1.1.1.3, in green) are present for *E. coli* (`eco`), *C. exile* (`cex`), *G. aurantiaca* (`gau`), and *B. fragilis* (`bfr`).

Intriguingly, in species *N. defluvii* (`nde`), *D. indicum* (`din`), and *B. fragilis* (`bfr`), the genes involved in reactions `R00480` (EC 2.7.2.4, in yellow) and `R01466` (EC 4.2.3.1, in blue) are separated by a gene whose product is involved in the reaction `R01518` (EC 5.4.2.12, in red).

The bacterial data set was examined in order to determine whether other species exhibit a similar gene ordering pattern. Only *D. acetiphilus* (`dap`) and *R. baltica* (`rba`) have neighboring genes involved in `R01518` and other reactions from the trail in Figure VII.9. The common denominator for all five species seems to be that the genes whose products catalyze reactions `R01518` (EC 5.4.2.12, in red) and `R00480` (EC 2.7.2.4, in yellow) are strict neighbors (Figure VII.12). Reaction `R01518` makes use of a phosphomutase activity for transferring a phosphate group within the same molecule (phosphoglycerate), whereas `R00480` employs a phosphotransferase activity for adding a phosphate group to aspartate using ATP.

Although there is no obvious link between the two reactions aside from the transfer of a phosphate group, it could be an instance of genomic hitchhiking [Ro-

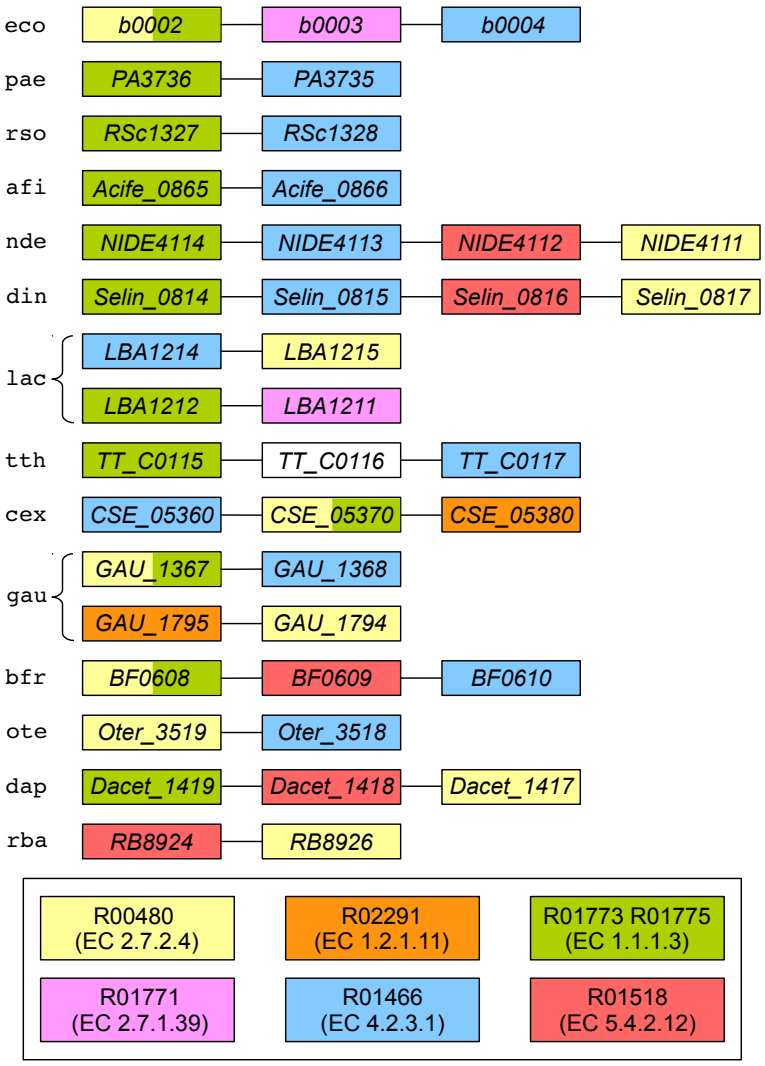



**Figure VII.12 Genomic context for genes involved in the trail in Figure VII.9.** Two additional reactions are shown: `R02291` (EC 1.2.1.11) linking reactions `R00480` (EC 2.7.2.4) and {`R01773`, `R01775`} (EC 1.1.1.3), and `R01518` (EC 5.4.2.12) representing a phosphoglycerate mutase activity farther along the glycine, serine, and threonine metabolism pathway. Neighboring genes are linked by an edge. Genes are color-coded according to the reactions in which the enzymes they encode take part. Two pairs of neighboring genes on different strands of the bacterial chromosome are shown for *L. acidophilus* (`lac`) and *G. aurantiaca* (`gau`). The gene in white in *Thermus thermophilus* (`tth`) codes for a hypothetical protein. *D. acetiphilus* (`dap`) and *R. baltica* (`rba`) exhibit a similar gene ordering pattern to *N. defluvii* (`nde`), *D. indicum* (`din`), and *B. fragilis* (`bfr`) (see text).

gozin *et al.*, 2002]. This means that operons sometimes contain functionally unrelated genes that nonetheless share similar expression requirements with the rest of the operon. It is possible that gene *apgM* (encoding the enzyme involved in reac-

tion R01518, in red) benefits from the expression levels of the genes involved in the trail in Figure VII.9. At any rate, a physiological and/or biochemical reason for the coexpression of *apgM* and the gene involved in R00480 (in yellow) seems to exist, since the two genes are neighbors across the bacterial domain, as reported in the STRING database [Szklarczyk *et al.*, 2014] (see Figure C.7).

In light of these observations, two hypotheses can be formulated:

(a) This particular genomic arrangement pattern has occurred independently several times during the evolution of extant bacterial species, or
(b) The ancestor of the bacterial domain exhibited this exact genomic arrangement, which was subsequently lost.

Hypothesis (b) cannot be excluded as not enough evidence is available to do so, but it can be considered less likely [Panchen, 1982] than hypothesis (a), which is the most parsimonious.

This example is an interesting instance of trail grouping by genes featuring an intriguing motif of absence of neighboring functionally similar genes in an important number of species. Upon closer investigation, an unexpected gene ordering pattern is uncovered for five of the species in the data set.

## 5 Case study: Exploring steps of peptidoglycan biosynthesis

Figure VII.13 illustrates trail finding by CoMetGeNe on the well-studied biological process of peptidoglycan biosynthesis [Barreteau *et al.*, 2008]. Peptidoglycan is the main constituent of the bacterial cell wall, providing its structural strength and determining cell shape. Manifesting an important diversity at both the chemical and architectural levels [Vollmer *et al.*, 2008; Turner *et al.*, 2014], peptidoglycan is present in the vast majority of bacteria.

The yellow and purple trails in Figure VII.13, recovered in the peptidoglycan biosynthesis pathway of *Escherichia coli* (eco00550), represent the conversion of UDP-N-acetylmuramate (UDP-MurNAc) into a precursor of DAP-type peptidoglycan and into a precursor of lysine-type peptidoglycan, respectively. Figure VII.14 shows the genes encoding the enzymes involved in these trails: *murE* (*b0085*), *murF* (*b0086*), *mraY* (*b0087*), *murD* (*b0088*), *murG* (*b0090*), *murC* (*b0091*), and *ddlB* (*b0092*). Note that both trails produced by CoMetGeNe were obtained by skipping gene *ftsW* (*b0089*), with the gap parameter $\delta_G$ set to 1.

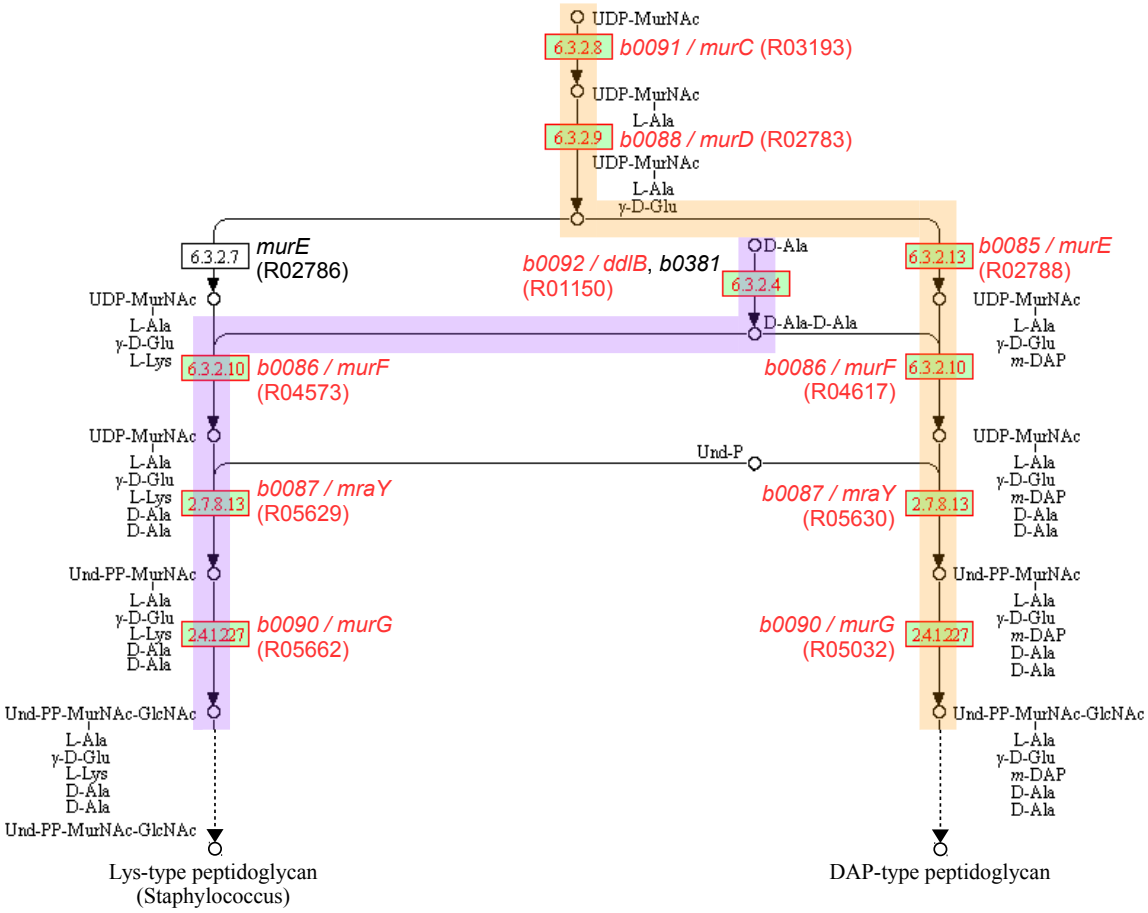



**Figure VII.13 Partial view of the peptidoglycan biosynthesis pathway in *Escherichia coli*.** Adapted from KEGG PATHWAY, map `eco00550` (May 28, 2015 version). Shown here are two `CoMetGeNe` trails, highlighted in yellow and purple. The gap parameter $\delta_G$ was set to one (thus allowing to skip one gene). Reactions in the trails are labeled with the corresponding KEGG reaction identifiers (R numbers) and with the Blattner identifiers and gene names of the genes involved in the reactions. Genes with black identifiers do not belong to the gene group in Figure VII.15. Genes with red identifiers are neighbors on the positive strand of the *E. coli* chromosome (see Figure VII.14). Dashed arrows from a metabolite $m$ to another metabolite $m'$ signify that a chain of reactions, omitted in this figure for clarity, leads from $m$ to $m'$.


**Figure VII.14 Genomic context for *E. coli* genes involved in the trails in Figure VII.13.** The gene in gray is not involved in the trails. The genes are located on the positive strand of the bacterial chromosome.

| *E. coli* gene | gsu | aae | sau | mpn | syn | pma | fgi | rba | ote |
|---|---|---|---|---|---|---|---|---|---|
| *b0085* | x | . | . | . | . | . | x | . | x |
| *b0086* | . | . | . | . | . | . | x | . | x |
| *b0087* | x | . | x | . | . | . | x | . | x |
| *b0088* | x | . | x | . | . | . | x | . | x |
| *b0090* | x | . | . | . | . | . | x | . | x |
| *b0091* | x | . | . | . | . | . | . | . | . |
| *b0092* | x | . | x | . | . | . | x | . | x |

**Figure VII.15 Group of homologous genes involved in the trails in Figure VII.13.** See Figure C.8 for the grouping by genes corresponding to these trails for all the species in the data set.

| reaction | *E. coli* gene | gsu | aae | sau | mpn | syn | pma | fgi | rba | ote |
|---|---|---|---|---|---|---|---|---|---|---|
| R02788 | *b0085* | x | . | | | . | . | x | . | x |
| R04617 | *b0086* | | . | . | | . | . | x | | x |
| R05630 | *b0087* | x | . | x | | . | . | x | | x |
| R02783 | *b0088* | x | . | x | | . | . | x | | x |
| R05032 | *b0090* | x | . | . | | . | . | x | | x |
| R03193 | *b0091* | x | . | . | | . | . | | | |

**Figure VII.16 Group of reactions defining the trail in yellow in Figure VII.13.** Cells in gray designate missing reactions. See Figure C.9 for the grouping by reactions corresponding to this trail for all the species in the data set.

| reaction | *E. coli* gene | gsu | aae | sau | mpn | syn | pma | fgi | rba | ote |
|---|---|---|---|---|---|---|---|---|---|---|
| R04573 | *b0086* | | . | x | | . | . | x | | x |
| R05629 | *b0087* | x | . | . | | . | . | x | | x |
| R05662 | *b0090* | x | . | . | | . | . | x | | x |
| R01150 | *b0092* *b0381* | x | . | x | | . | . | x | . | x |

**Figure VII.17 Group of reactions defining the trail in purple in Figure VII.13.** Cells in gray designate missing reactions. See Figure C.10 for the grouping by reactions corresponding to this trail for all the species in the data set.

The skipped gene encodes the FtsW protein, which plays an essential role in cell division [Boyle *et al.*, 1997]. Moreover, it has been shown that FtsW is also a transporter of peptidoglycan precursors across the inner membrane [Mohammadi *et al.*, 2011]. It is therefore interesting that the gene encoding this transporter, although not included in the trail, is found in the same neighborhood as other peptidoglycan biosynthesis genes. This underlines the capacity of the trail finding method to identify trails of reactions that are compatible with their genomic context.

Trail grouping was performed for *E. coli* (`eco`) as reference species. Figure VII.15 illustrates the portion of table $T^g_{\texttt{eco}}$ (trail grouping by genes) corresponding to the trails in Figure VII.13, for *E. coli* and 9 other bacterial species presenting interesting features. Likewise, Figures VII.16 and VII.17 illustrate the portions of table $T^r_{\texttt{eco}}$ (trail grouping by reactions) corresponding to the trails highlighted in yellow and in purple in Figure VII.13, respectively. Trail grouping for the full data set is presented in Figure C.8 (trail grouping by genes), Figure C.9 (grouping by reactions for the yellow trail), and Figure C.10 (grouping by reactions for the purple trail).

Trail grouping by genes identifies genes of the reference species with neighboring functionally similar genes in other species. The degree of conservation of gene neighborhood for the genes involved in a given trail is proportional to the number of cross symbols ($\times$) in $T^g_S$ for the reference species $S$. The number of crosses in $T^g_{\texttt{eco}}$ (Figure C.8) confirms that the trails in Figure VII.13 are frequently found for the species in the data set, albeit with varying degrees of conservation of gene neighborhood. This finding represents a positive control, being consistent with the fact that most bacteria possess peptidoglycan cell walls. Cells with dot symbols (`.`) in $T^g_{\texttt{eco}}$ (Figures VII.15 and C.8) do not allow to distinguish between non neighboring and missing genes. However, Figures VII.16 and VII.17 identify species with missing reactions (in gray in the figures) with respect to *E. coli*: *Geobacter sulfurreducens* (`gsu`), *Staphylococcus aureus* (`sau`), *Mycoplasma pneumoniae* (`mpn`), *Fimbriimonas ginsengisoli* (`fgi`), *Rhodopirellula baltica* (`rba`), and *Opitutus terrae* (`ote`). The remaining species perform all the reactions but do not necessarily have contiguous genes coding for the required enzymes. Among the six species with missing reactions with respect to *E. coli*, *M. pneumoniae* (`mpn`) is a negative control, as it is well-known that it is devoid of a cell wall [Waites and Talkington, 2004]; the five other species are discussed below.

## 5.1 Incomplete annotations

*G. sulfurreducens* (`gsu`), a Deltaproteobacterium [Caccavo *et al.*, 1994] with a peptidoglycan dry weight fraction of 4% [Mahadevan *et al.*, 2006], is reportedly missing

reactions R04617 (Figure VII.16) and R04573 (Figure VII.17), which should be catalyzed by MurF (Figure VII.13).

The KEGG GENES entry *GSU3073* is annotated as *murF*[2]. Regardless, this gene is not associated to either of the reactions R04617 or R04573 in the pathway map gsu00550. As of the writing of this thesis (June 2018), the latest version of the pathway map gsu00550[3] dates from April 10, 2017. *GSU3073* is located in the same gene neighborhood as the other genes encoding the enzymes for the reactions in Figures VII.16 and VII.17. Moreover, as revealed by CoMetGeNe, every other reaction in the two trails in Figure VII.13 is performed by enzymes encoded by neighboring genes.

The functional annotation *murF* for the gene *GSU3073* is confirmed by performing a protein BLAST [Altschul *et al.*, 1997] for the *E. coli* MurF query sequence against *G. sulfurreducens* (NCBI taxon 35554). The matching protein WP_010943698 (40% identity, 98% query cover, E-value 1e−76) corresponds to the gene *GSU3073* via the identical protein YP_006589581.

If two reactions can in theory be catalyzed by a unique enzyme, both reactions do not necessarily occur in a given species that produces the enzyme in question. For *G. sulfurreducens*, it is expected that it synthesizes peptidoglycan [Mahadevan *et al.*, 2006] using the metabolic route leading to DAP-type peptidoglycan (instead of staphylococcal lysine-type peptidoglycan). This metabolic route passes through the reaction R04617 in Figure VII.13.

The missing reaction R04617 for *G. sulfurreducens* (gsu) is hence an instance of incomplete annotation in the KEGG knowledge base in the sense that the gene *GSU3073* has not yet been associated to the reaction R04617.

## 5.2 Alternative metabolic routes

*S. aureus* (sau) is a Gram-positive bacterium [Willey *et al.*, 2008], well-known to produce lysine-type peptidoglycan (dashed arrow in Figure VII.13) instead of DAP-type peptidoglycan. This is accomplished using the alternative route passing through reactions R02783 (EC 6.3.2.9) and R02786 (EC 6.3.2.7). The metabolic route leading to lysine-type peptidoglycan in *Staphylococcus* shares the two reactions catalyzed by MurC (R03193) and MurD (R02783) with the route leading to DAP-type peptidoglycan. Equivalents of the other four reactions in the trail highlighted in yellow exist in lysine-type peptidoglycan biosynthesis and are performed by the same enzymes (MurE, MurF, MraY, and MurG) on UDP-MurNAc substrates having lysine (instead of DAP) residues (Figure VII.13).

[2]KEGG GENES entry for *GSU3073*: http://www.genome.jp/dbget-bin/www_bget?gsu:GSU3073

[3]The peptidoglycan biosynthesis pathway in *G. sulfurreducens* is available at the following address: https://www.genome.jp/kegg-bin/show_pathway?gsu00550.

As illustrated in Figure VII.16, two genes among those involved in peptidoglycan biosynthesis in *S. aureus* are neighbors (*mraY* and *murD*). From Figure VII.17, it can be seen that the genes corresponding to *murF* and *ddlB* in *E. coli* (which is *ddlA* in *S. aureus*) are also neighbors. Recall that trail grouping by reactions (see section V.4.1) determines *a* maximal subset of reactions $\mathcal{R}'$ from a given trail (assessed as a HNET reaction set $\mathcal{R}$) of the reference species such that the reactions in $\mathcal{R}'$ are catalyzed by products of neighboring genes in the target species. For the reaction set $\mathcal{R}$ in Figure VII.17, there are two such maximal subsets, `{R04573, R01150}` and `{R05629, R05662}`. (See section 3 for another example.)

This example shows that missing reactions for a given organism with respect to the reference species may indicate the existence of an alternative metabolic route for the organism in question with respect to the reference species.

### 5.3 A possibly erroneous ORF prediction

*F. ginsengisoli* (`fgi`), a member of the recent Armatimonadetes phylum, is reportedly missing the reaction `R03193` (EC 6.3.2.8 in Figure VII.13) which should be catalyzed by MurC (Figure VII.16). This species has nevertheless been described as synthesizing DAP-type peptidoglycan [Im *et al.*, 2012]. Moreover, *F. ginsengisoli* performs every other reaction in the trail highlighted in yellow in Figure VII.13 using products of neighboring genes (Figure VII.18). We have therefore proceeded to a protein BLAST [Altschul *et al.*, 1997] search against *F. ginsengisoli* (NCBI taxon 1005039) with the MurC sequence of *Chthonomonas calidirosea*, another member of the Armatimonadetes phylum, as query.


**Figure VII.18 Genomic context for *F. ginsengisoli* genes involved in the trails in Figure VII.13.** The genes in gray are not involved in the trails. The genes are located on the negative strand of the bacterial chromosome. *4783*, *4784*, and *4785* stand for *OP10G_4783*, *OP10G_4784*, and *OP10G_4785*, respectively. The gene in green (labeled *4784*) is annotated as *ddl*.

The search was inconclusive, as the best match (WP_025227986 with 39% identity, 71% query cover, E-value 9e−67) corresponds to the gene *OP10G_4783* which encodes a hypothetical protein roughly half the size of MurC and with no known domains (see *4783* in Figure VII.18). The second best match (AIE88152 with 47% identity, 34% query cover, E-value 8e−39) corresponds to the gene *OP10G_4784* which is a D-alanine–D-alanine ligase (*ddl*), being involved in the reaction `R01150` in the peptidoglycan biosynthesis pathway (Figure VII.13 and *4784* in Figure VII.18).

The functional assignment *ddl* is not a genomic (RefSeq or GenBank) annotation, but a K number assignment. Recall that the KEGG ORTHOLOGY (KO) database assigns a K number (or KO identifier) to an individual gene if this gene is determined to be an ortholog of sequences from the KO group designated by the K number (see section III.2.2).

Intriguingly, the gene *OP10G_4784* has been annotated in GenBank as a UDP-N-acetylmuramate–L-alanine ligase, which describes the role of MurC. Furthermore, in addition to the expected *ddl*-specific domains due to the KO assignment in KEGG, *OP10G_4784* also exhibits a *Mur_ligase_C* domain annotation, corresponding to the C-terminal Mur ligase domain. MurC proteins should however possess additional middle and/or catalytic domains. These findings led to investigate the possibility of *OP10G_4784* being a fusion between *murC* and *ddl*. Although the STRING database [Szklarczyk *et al.*, 2014] reports that fusions of *murC* and *ddl* occur frequently in the Chlamydiae phylum, it does not appear to be the case for *OP10G_4784* due to missing Mur ligase domains and different sequence size with respect to *murC–ddl* fusions in Chlamydiae.

Interestingly, a Mur ligase catalytic domain is reported for the short neighboring gene *OP10G_4785* (RefSeq: WP_084179698), labeled as *4785* in Figure VII.18. Furthermore, *OP10G_4785* has been annotated as a UDP-N-acetylmuramate–L-alanine ligase (MurC) in GenBank. Its surrounding genes are *ddl* (*OP10G_4784*) and *murG* (*OP10G_4786*), which is the established genomic context for *murC* in bacteria that maintain the genes involved in peptidoglycan biosynthesis organized into operons. A KEGG ortholog search for *OP10G_4785* reveals longer *murC* ortholog sequences in other species. Two hypotheses are therefore possible:

(a) The activity EC 6.3.2.8 is performed jointly by products of genes *OP10G_4784* and *OP10G_4785* in *F. ginsengisoli* (`fgi`), or
(b) The open reading frame (ORF) for *OP10G_4784* was incorrectly predicted, the *ddl* coding sequence erroneously including a *Mur_ligase_C* domain that may in fact belong to *OP10G_4785*.

Hypothesis (a) does not seem to be likely because bacteria typically have only one gene encoding the MurC enzyme. Hypothesis (b) on the other hand describes a situation that can arise in practice due to the automatic processes involved in genome annotation. If hypothesis (b) above is verified, the redefined coding sequence neighboring *ddl* is likely *murC*. The dashed red line in Figure VII.19 between the C-terminal region of the Mur ligase (*Mur_ligase_C*) and the N-terminal region of the D-alanine–D-alanine ligase (*Dala_Dala_lig_N*) domains signifies that another stop codon for the ORF of *OP10G_4784* might be found in this inter-domain region of

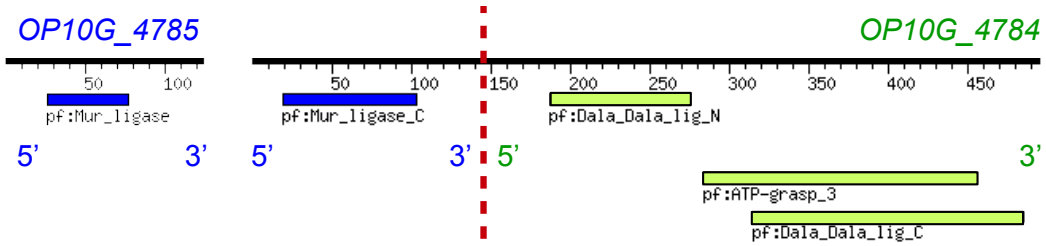



**Figure VII.19 Possible incorrect ORF prediction in *OP10G_4784*.** Adapted from KEGG SSDB motif search. Genes *OP10G_4784* and *OP10G_4785* are shown as thick horizontal lines (in black). Underneath each gene are shown the associated Pfam domains with an E-value less than 1e−07. The dashed line in red represents the incorrect ORF prediction hypothesis.

approximately 80 amino acid residues (corresponding to approximately 240 nucleotides).

Note that the gene *OP10G_4784* is shown at the left of *OP10G_4785* in Figure VII.18, and at the right in Figure VII.19. This is normal, as Figure VII.18 shows how genes are organized on the negative chromosomal strand (3′ → 5′, with 3′ at the left by convention), whereas Figure VII.19 shows genes in the 5′ → 3′ direction. (Refer to Figure I.9 and section I.3.1 for more information.)

This analysis shows that missing reactions with respect to the reference species may indicate the existence of incorrect genomic annotations.

## 5.4 Outdated annotations

*R. baltica* (`rba`), as other Planctomycetes, has been thought to be lacking peptidoglycan [Fuerst and Sagulenko, 2011]. Consistent with annotations in KEGG reflecting the existing genome annotations, `CoMetGeNe` only identifies one reaction among the six in the trail highlighted in yellow in Figure VII.13 as being present in *R. baltica*. In addition, no peptidoglycan biosynthesis genes are currently listed in the STRING database [Szklarczyk *et al.*, 2014] for other Planctomycetes beside members of the *Planctomyces* genus. However, Jeske *et al.* [2015] have biochemically demonstrated that sugar and peptide components of peptidoglycan are present in Planctomycetes. The study also uses an *in silico* approach to identify candidate peptidoglycan biosynthesis genes in *R. baltica* and other Planctomycetes.

The fact that the findings of this study are yet to be reflected in existing annotations indicates the difficulty of validating proposed gene function. Consequently, `CoMetGeNe` correctly identifies the only reaction in the trail highlighted in yellow in Figure VII.13 that is associated to an annotated gene in *R. baltica* (`rba`).

## 5.5 Missing annotations

*O. terrae* (`ote`), a member of the subdivision 4 of the Verrucomicrobia phylum, had been thought to be one of the very few exceptions of free-living bacteria without peptidoglycan [Yoon, 2011]. Using `CoMetGeNe`, it was however determined that all reactions in the trail highlighted in yellow in Figure VII.13 are present in *O. terrae* (Figure VII.16), with the exception of reaction `R03193` which should be catalyzed by MurC. Furthermore, the five present reactions are catalyzed by products of neighboring genes.

These `CoMetGeNe` results are in agreement with the data obtained by Rast *et al.* [2017], who have recently challenged the concept of free-living bacteria lacking peptidoglycan. They proved that members of the Opitutaceae family do possess peptidoglycan cell walls. We propose the candidate *murC* gene in *O. terrae* to be *Oter_2637*, following a protein BLAST [Altschul *et al.*, 1997] for the *E. coli* MurC query sequence (WP_012375453 with 29% identity, 94% query cover, E-value 5e−41).

This is an instance of missing annotation from public knowledge bases.

## 5.6 Summary

This case study illustrated two `CoMetGeNe` trails detected in the peptidoglycan biosynthesis pathway of *E. coli*, identified by skipping one gene. Both trails correspond to the same group of genes, retrieved for the reference species *E. coli* when grouping its `CoMetGeNe` trails by genes. The analysis of this case study was conducted by focusing on missing reactions with respect to the reference species.

Perhaps counter-intuitively, missing reactions do not always translate to species that lack a particular metabolic route, as is the case for *M. pneumoniae* (`mpn` in Figures VII.16 and VII.17). When a target species performs some, but not all, of the reactions in a trail of the reference species, the missing reaction(s) may indicate that an alternative metabolic route exists in the target species with respect to the reference species (see section 5.2). It was shown here that missing reactions with respect to the reference species may also signal incomplete annotations, such as the gene *GSU3073* in *G. sulfurreducens* (see section 5.1), or even outdated (see section 5.4) or missing annotations, as is the case for MurC in *O. terrae* (see section 5.5). Finally, in some rare cases, missing reactions may also point out possible annotation errors at the genomic level, as is the case of the gene *OP10G_4784* in *F. ginsengisoli* (see section 5.3).

## 6 Concluding remarks

This chapter demonstrated how trail finding (Chapter IV) and trail grouping (Chapter V) are performed using `CoMetGeNe` (Chapter VI) on the metabolic pathways and genomic contexts of a selection of representative bacterial species.

Several instances of conserved metabolic and genomic patterns were discussed, revealing the existence of strong relationships between metabolic architecture and genome structure. In some situations, links between metabolic and genomic context are detected as conserved metabolic and genomic patterns, although the biochemical rationale for these associations is not readily apparent. A case was made for the attentive investigation of missing reactions with respect to the reference species. It was shown that divergent conserved metabolic and genomic patterns may indicate that certain species possess alternative metabolic routes with respect to the reference species. In other cases, however, missing reactions indicate potential annotation problems in public knowledge bases. Several concrete reannotations were suggested in the case study.

The trail finding and trail grouping methodologies (as well as their implementation represented by `CoMetGeNe`) are thus exploratory tools that may help provide insights into metabolic evolution and the links between metabolic and genomic contexts. The findings presented in this chapter emphasize the discovery aspect of trail finding and trail grouping as performed by `CoMetGeNe`, leading to the formulation of several biological hypotheses.

The next chapter proposes an alternative definition of conserved metabolic and genomic patterns by modulating the definition of metabolic patterns in terms of similarity of chemical reactions.

# VIII

## Toward the integration of reaction signatures

# 1 Introduction

The previous chapters (Chapter IV through Chapter VII) were aimed at detecting *metabolic and genomic patterns*, defined as trails (see definition II.11) of reactions being catalyzed by products of neighboring genes. The trail grouping methodology (Chapter V) helps to uncover similar metabolic and genomic patterns across multiple species. In such cases, metabolic and genomic patterns are said to be *conserved*. The definition of conserved patterns allows for flexibility in the sense that strict matching is not enforced. Thus, reaction and/or gene order may differ. Moreover, all reactions and/or functionally similar genes are not required to be present.

Therefore, while trail grouping allows to group together several similar `CoMetGeNe` trails, their similarity is based on the simple presence or absence of reactions. A less naïve measure for trail similarity would be based on the nature of the chemical transformations performed by reactions in the trails. The ability to qualify trails as being chemically similar and to quantify this similarity would enable the identification of "extended" metabolic and genomic patterns in which reactions may be different as long as they perform the same chemical transformations.

*chemical, metabolic, and genomic pattern*

We introduce the term ***chemical**, metabolic, and genomic pattern* to describe a group of `CoMetGeNe` trails in which reactions are chemically similar. In other words, two or more trails of reactions catalyzed by products of neighboring genes form a chemical, metabolic, and genomic pattern if the chemical transformations performed by the reactions in the trails are similar.

Numerous measures exist for evaluating chemical similarity [Bender and Glen, 2004]. The work presented in this chapter relies on a descriptor of atom neighborhood (section 2) that is subsequently used to compute reaction signatures (section 3). Two methods based on reaction signatures are then proposed in order to determine the chemical similarity of `CoMetGeNe` trails. The first one is a qualitative approach (section 4), while the second one is quantitative (section 5).

# 2 Signature molecular descriptor

Introduced by Faulon *et al.* [2003], the *signature molecular descriptor* is a description scheme for chemical compounds, in which a molecule is characterized by the neighborhood of each of its atoms up to a given distance. The underlying representation of a compound is the *molecular graph*, which is an undirected graph where vertices represent atoms and edges represent bonds between atoms.

*molecular graph*

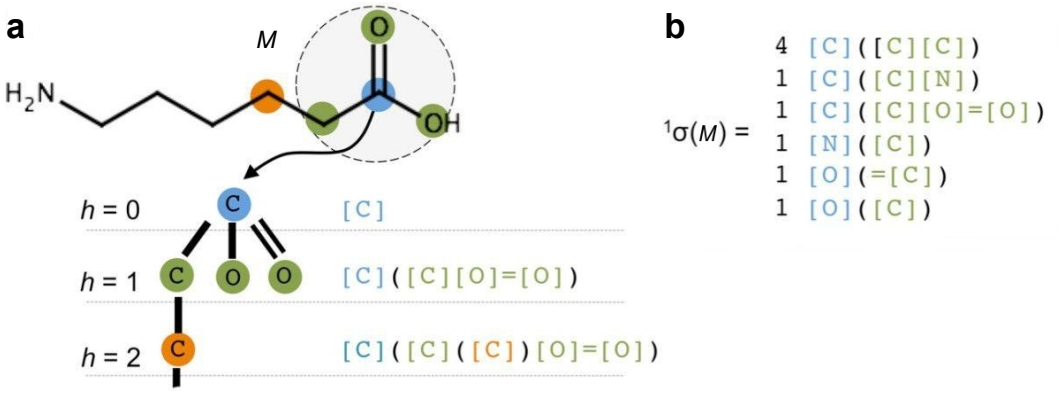



**Figure VIII.1 Computation of the molecular signature of a compound *M*.** Adapted from Carbonell *et al.* [2011b] (licensed under CC BY 2.0). To simplify, hydrogen atoms are not shown in the signatures. **(a)** Atomic signatures are determined for every atom in the molecule. The signature of the carbon atom in blue is shown for heights $h = 0$ (the carbon atom itself), $h = 1$ (the carbon atom in blue surrounded by the three carbon atoms in green), and $h = 2$ (the carbon atom in blue surrounded by the three carbon atoms in green at a distance of 1, and by the carbon atom in orange at a distance of 2). **(b)** The molecular signature of the compound $M$ (in a) of height 1, ${}^1\sigma(M)$, is shown. The molecular signature of $M$ contains lexicographically sorted atomic signatures for every atom in $M$, accompanied by their counts.

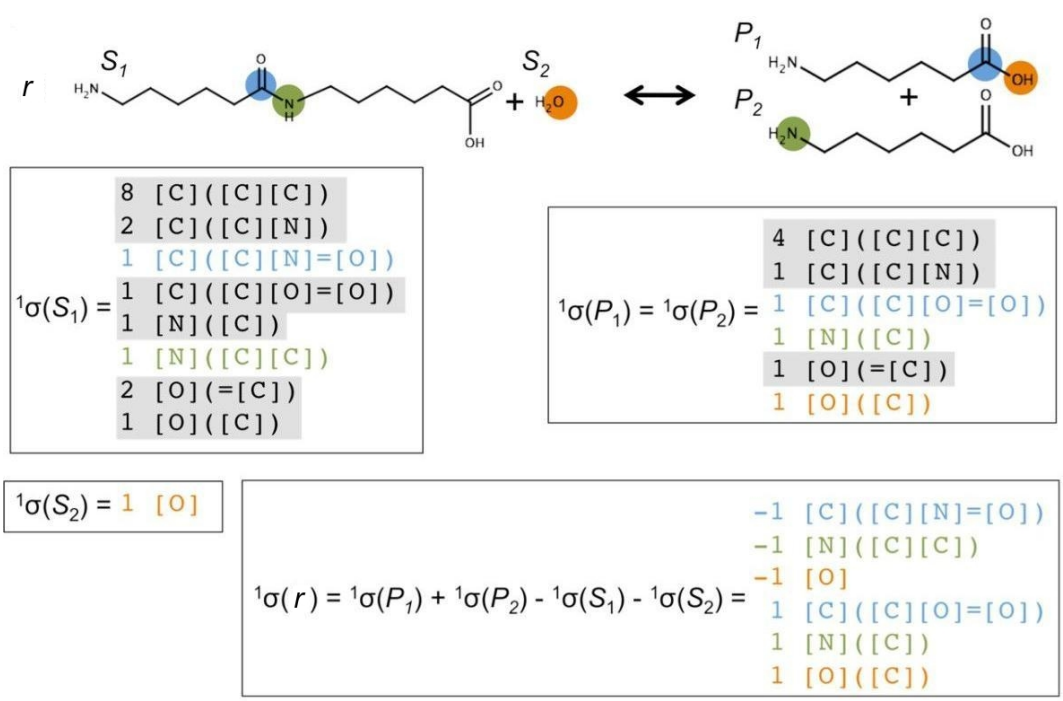



**Figure VIII.2 Computation of the reaction signature of a reaction *r*.** Adapted from Carbonell *et al.* [2011b] (licensed under CC BY 2.0). To simplify, hydrogen atoms are not shown in the signatures. The reaction signature of height 1 for the reaction $r : S_1 + S_2 \rightarrow P_1 + P_2$, ${}^1\sigma(r)$, is given by the subtraction (net difference) of substrates from products in terms of descriptors of height 1.

*atomic signature*

**Atomic signature** The *atomic signature* of height $h$ of an atom $a$ is a description of the neighborhood of atom $a$ up to a maximal distance $h$. For example, the atomic signature of the carbon atom in blue in Figure VIII.1a of height 1 is `[C]([C][O]=[O])`, meaning that the neighbors of the carbon atom in blue at a distance of 1 are another carbon atom (in green in the figure) and two oxygen atoms, one of them having a double bond with the carbon atom in blue.

Note that it is also possible to determine the signature of a bond of a given height. Similar to the signature of an atom, the signature of a bond $b$ describes its neighborhood up to a given distance in terms of surrounding bonds. For simplicity, this section only refers to atomic neighborhood.

*molecular signature*

**Molecular signature** The *molecular signature* of height $h$ of a compound $M$, denoted ${}^h\sigma(M)$, is the lexicographically sorted set of every atomic signature of height $h$ in $M$, preceded by its count (i.e. number of occurrences in $M$). Figure VIII.1b shows the molecular signature of height 1 for the compound in Figure VIII.1a. For example, the atomic signatures of the oxygen atoms in the molecule occur only once: `[O](=[C])` for the oxygen atom linked by a double bond to the atom carbon in blue in Figure VIII.1a, and `[O]([C])` for the oxygen atom linked by a simple bond. The atomic signature `[C]([C][C])` appears four times in the molecular signature ${}^1\sigma(M)$, because $M$ contains four carbon atoms surrounded by two other carbon atoms (the ones in green and orange, and the two other carbon atoms immediately at their left).

The molecular signature can be computed using the MolSig software [Carbonell *et al.*, 2013], which returns compound signatures in a SMILES-like format [Weininger, 1988].

*reaction signature*

**Reaction signature** The *reaction signature* of height $h$ of a reaction $r$ is obtained by subtracting the molecular signatures of height $h$ of substrates of $r$ from the molecular signatures of height $h$ of products of $r$. Formally, reactions have the general equation $r : s_1S_1 + \ldots + s_nS_n \rightarrow p_1P_1 + \ldots + p_mP_m$, where $s_i$ and $p_j$ are the stoichiometric coefficients of substrates $S_i$ and products $P_j$, respectively. Then, the reaction signature of height $h$ of a reaction $r$ is defined as the following vector [Carbonell *et al.*, 2011b]:

$$ {}^h\sigma(r) = \left( \sum_{P_j \in r} p_j \, {}^h\sigma(P_j) - \sum_{S_i \in r} s_i \, {}^h\sigma(S_i) \right) $$

For example, Figure VIII.2 shows how the reaction signature of height 1 is computed for a reaction featuring the compound in Figure VIII.1 as product.

**Quantification of reaction similarity** In general, it can be assumed that two reactions sharing the same reaction signature of a given height perform the same type of chemical transformation[1]. Under this assumption, reaction signatures only offer a qualitative measure of chemical similarity: two reactions either have the same signature, or they do not. In order to quantify the similarity between two reactions, Carbonell *et al.* [2011b] adapted the Tanimoto similarity coefficient to reaction signatures as follows:

$$ {}^{h}T_c(r_i, r_j) = \frac{|{}^{h}\sigma(r_i) \cdot {}^{h}\sigma(r_j)|}{|{}^{h}\sigma(r_i)|^2 + |{}^{h}\sigma(r_j)|^2 - |{}^{h}\sigma(r_i) \cdot {}^{h}\sigma(r_j)|} $$

${}^{h}T_c(r_i, r_j)$ is a real number between 0 and 1, with 0 for complete dissimilarity and 1 when the two reactions share the same signature.

## 3 Computation of reaction signatures

In order to compute reaction signatures, chemical compounds were first retrieved from KEGG in MDL Molfile format (see the KEGG REST query number 7 in section III.2.4). As described in Sorokina *et al.* [2015], compounds were first standardized by applying protonation and aromatization[2] as needed using the Molconvert utility[3]. Next, signature molecular descriptors [Carbonell *et al.*, 2013] for the compounds were computed using the MolSig software[4] for diameters ranging between 0 and 9 (see below).

*signature diameter*

The *diameter* of a signature is a concept used in the MolSig software to abstract the type of signature for a given height. Thus, even diameters refer to atomic neighborhood while odd diameters refer to bond neighborhood. An even diameter $d$ means that the molecular signature of a compound is computed by describing the neighborhood of every atom in the compound up to a maximum height $d/2$. Similarly, an odd diameter $d$ means that the neighborhood of every bond in the compound is described up to a maximum height $(d-1)/2$.

In other words, molecular signature descriptors were computed for compounds in KEGG for heights ranging from 0 (diameters 0 and 1 for atom and bond neighborhood, respectively) to 4 (diameters 8 and 9).

[1] This is often not the case for low height signatures, as shall be seen in sections 4 and 5.

[2] Protonation and aromatization refer to the addition of hydrogen atoms and the formation of aromatic systems, respectively.

[3] Molconvert can be obtained from the ChemAxon website (`https://chemaxon.com`) as part of the Marvin suite.

[4] MolSig can be obtained from `http://molsig.sourceforge.net`.

In addition, stoichiometric coefficients were normalized for the reactions having equations that contain literals. For example, reaction R03042 has the following definition:

```
Polyphosphate(n) + H2O <=> Polyphosphate(n-1) + Orthophosphate
```

The associated equation for reaction R03042 is:

```
C00404(n) + C00001 <=> C00404(n-1) + C00009
```

For the above example, the normalization consists in replacing $n$ with 1.

As explained in section VI.2.1, CoMetGeNe retrieves only metabolic pathway maps from KEGG. In total, 2,438 reactions are present in the KGML files (see section III.2.3) that were analyzed. Since it is only possible to compute reaction signatures if the structures of all participating compounds are available, reaction signatures were computed for 2,251 of the reactions present in the analyzed KGML files as described in section 2 above. Figure VIII.3 shows the breakdown of reactions present in KEGG.

*empty signature*

Among reaction signatures, some are *empty* because the difference of signature molecular descriptors between products and substrates is zero. Figure VIII.4 shows the example of an isomerisation reaction that falls within this category.

Among the 2,438 reactions present in KGML files, 1,468 (60%) belong to CoMetGeNe trails. Table VIII.1 shows the number of reaction signatures for reactions in CoMetGeNe trails with computable and non-empty signatures, as well as the average number of reactions associated to a given reaction signature.

# 4 Sets of reaction signatures

## 4.1 Approach

This approach consists in associating sets of reaction signatures to CoMetGeNe trails (see Figure VIII.5 for an overview).

Recall from section V.3 that CoMetGeNe trails are transformed into reaction sets in order to allow for variations in terms of reaction and/or gene order, as well as composition. Once reaction sets are determined, the next step is to "translate" them into *sets of reaction signatures*. A formal definition follows.

*set of reaction signatures*

**Definition VIII.1.** Let $\mathcal{R} = \{r_1, r_2, \ldots, r_n\}$ be a reaction set associated to a CoMetGeNe trail. The *set of reaction signatures* associated to $\mathcal{R}$ at diameter $d$ is the set $\bigcup_{i \in \{1,\ldots,n\}} {}^{d}\sigma(r_i)$, where ${}^{d}\sigma(r)$ is the reaction signature of a reaction $r$ at diameter $d$.

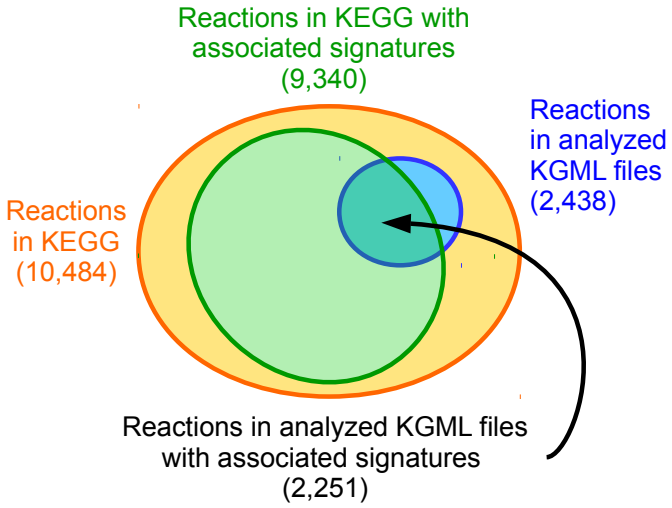



**Figure VIII.3 Breakdown of reactions in KEGG with respect to reaction signatures.** Among the 10,484 reactions present in KEGG (March 2017), 9,340 (89%) have an associated signature and 2,438 (23%) are present in the KGML files analyzed by `CoMetGeNe`. Among the 2,438 reactions present in KGML files, 2,251 (representing 92% of the 2,438 reactions in KGML files) have an associated signature.

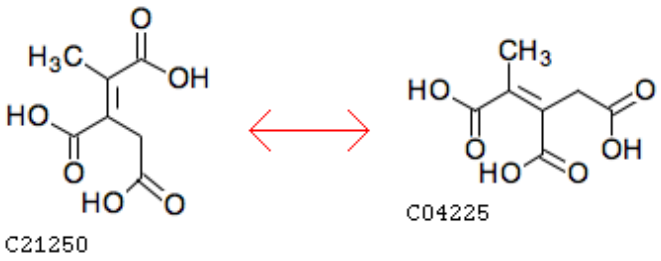



**Figure VIII.4 Reaction `R11264` (isomerisation of 2-methylaconitate).** 2-methyl-trans-aconitate (`C21250`) is transformed into cis-2-methylaconitate (`C04225`). The associated reaction signature is empty. Reproduced from KEGG REACTION.

| Signature diameter | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| #signatures | 72 | 266 | 531 | 774 | 931 | 1018 | 1081 | 1122 | 1150 | 1173 |
| #reactions | 373 | 991 | 1267 | 1298 | 1348 | 1351 | 1354 | 1354 | 1354 | 1354 |
| #react./sign. | 5.18 | 3.73 | 2.39 | 1.68 | 1.45 | 1.33 | 1.25 | 1.21 | 1.18 | 1.15 |

**Table VIII.1 Reaction signature statistics for reactions in `CoMetGeNe` trails.** For every signature diameter between 0 and 9, are shown the number of non-empty reaction signatures (#signatures), the number of reactions in `CoMetGeNe` trails with computable and non-empty signatures (#reactions), and the average number of reactions per signature (#react./sign.).

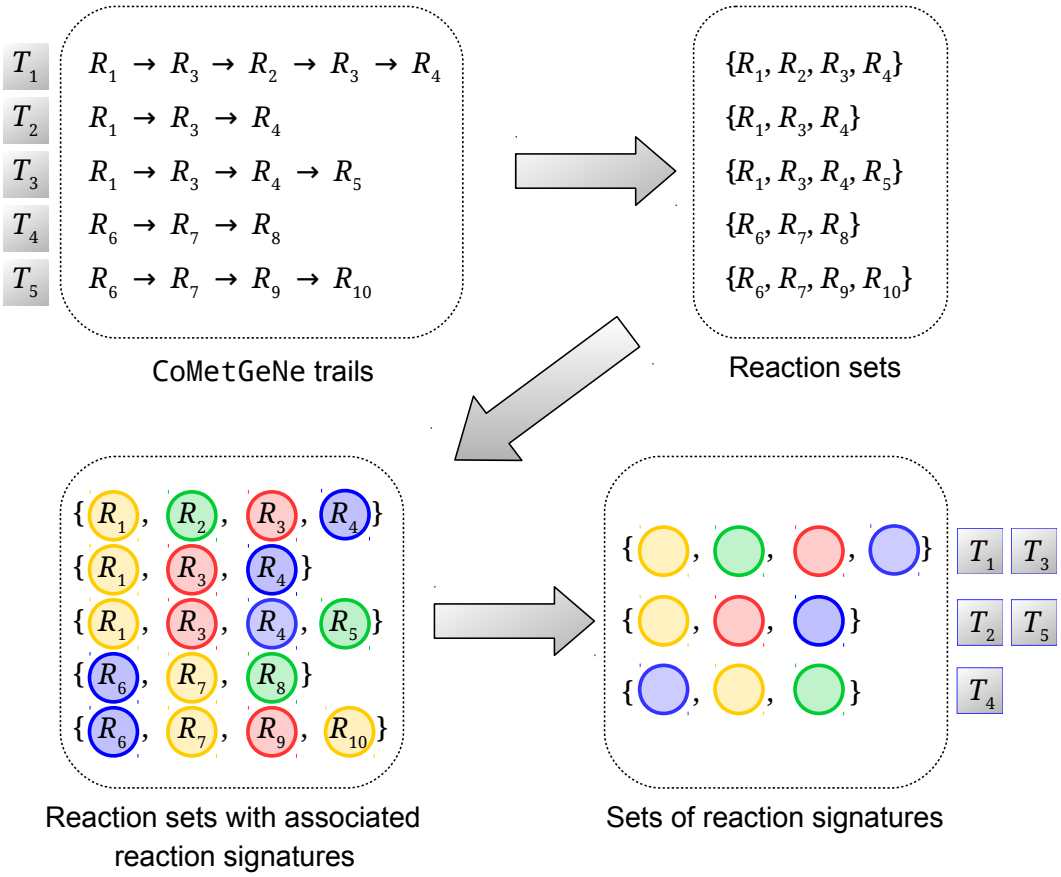



**Figure VIII.5 Overview of the approach consisting in transforming reaction sets into sets of reaction signatures.** Trail composition and reaction directionality are abstracted by transforming `CoMetGeNe` trails ($T_1$ to $T_5$) into reaction sets. In this approach, reaction signatures are integrated by determining the set of reaction signatures corresponding to a given `CoMetGeNe` reaction set. In this example, the reaction sets associated to trails $T_1$ and $T_3$ have the same corresponding set of reaction signatures. It is also the case for the reaction sets associated to trails $T_2$ and $T_5$.

A reaction set has a unique corresponding set of reaction signatures. However, a given set of reaction signatures may correspond to several reaction sets. For example, the reaction sets associated to trails $T_2$ and $T_5$ in Figure VIII.5 have the same corresponding set of reaction signatures (consisting in the yellow, red, and blue signatures). This property enables a qualitative evaluation of reaction set similarity, and hence of trail similarity.

*complete set of reaction signatures*

Two measures need to be taken in order to avoid to incorrectly qualify two reaction sets as similar. The first measure consists in only considering *complete* sets of reaction signatures. In a complete set of reaction signatures, every reaction in the associated reaction set(s) has a computable signature. The second measure consists

in ignoring sets of reaction signatures that contain empty signatures (see section 3). A set of reaction signatures without empty signatures is referred to as *valid*.

*valid set of reaction signatures*

## 4.2 Results

To the 4,179 trails produced by `CoMetGeNe` (see section VII.1.2) correspond 3,712 reaction sets (for an average of 1.13 `CoMetGeNe` trails per reaction set). Sets of reaction signatures corresponding to `CoMetGeNe` reaction sets were determined according to definition VIII.1 above, for signature diameters between 0 and 9. Additionally, sets of reaction signatures that were incomplete or contained empty signatures were ignored (see above).

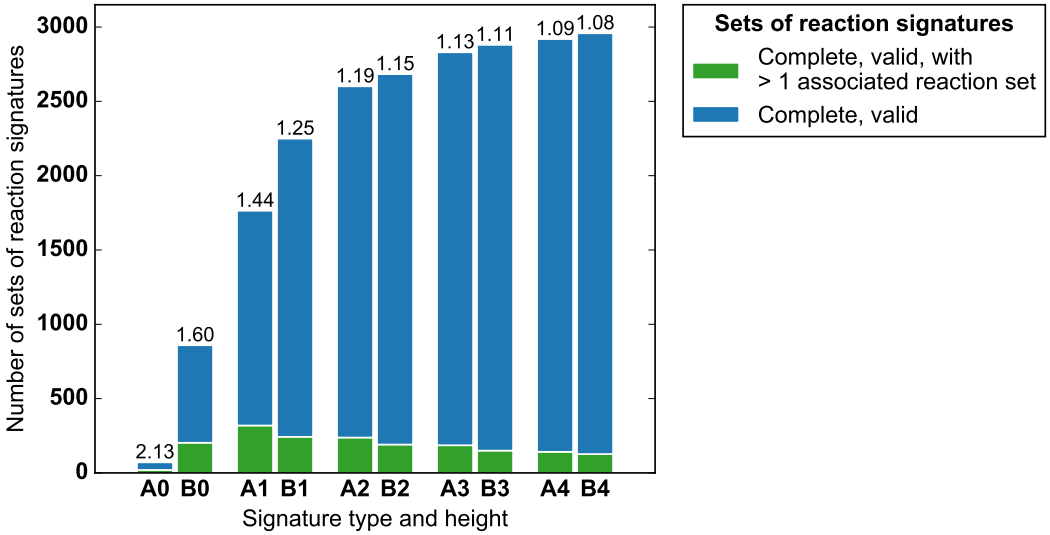



**Figure VIII.6 Sets of reaction signatures derived from reaction sets.** A set of reaction signatures is associated to every `CoMetGeNe` reaction set. Sets of reaction signatures in which a reaction signature exists (can be computed) for every reaction in the `CoMetGeNe` reaction set are referred to as *complete* sets of reaction signatures. Here are shown only complete sets of reaction signatures in which every signature is non-empty (i.e. *valid* sets of reaction signatures). Bar labels designate the signature height for atom (A) and bond (B) neighborhood. Values above each bar represent the mean number of reaction sets (with computable and non-empty signatures for every reaction) corresponding to one complete set of reaction signatures without any empty signatures.

Figure VIII.6 shows the number of complete and valid sets of reaction signatures for different values of the signature diameter. Among the total number of sets of reaction signatures, the complete and valid sets represent between 31.56% (atom type signature of height 0) and 88.01% (bond type signature of height 4), with an average of 75.3%.

The average number of reactions sets with computable and non-empty signatures per complete and valid set of reaction signatures is indicated above each bar in Figure VIII.6.

The sets of reaction signatures of interest are sets corresponding to at least two reaction sets (in green in Figure VIII.6). These sets of reaction signatures amount to between 4.29% and 4.86% (for bond and atom type signatures, respectively, of height 4), and 23.63% and 28.17% (for bond and atom type signatures, respectively, of height 0) of the total number of complete and valid sets of reaction signatures at each diameter.

## 4.3 Examples

### 4.3.1 Partially overlapping trails in different species

Figure VIII.7 shows a portion of the propanoate metabolism pathway (`map00640`) representing the conversion of propanoate into succinate. Two reaction sets corresponding to trails obtained for *Escherichia coli* and *Vibrio cholerae* were found to have the same associated set of reaction signatures in which reactions `R04424` (in blue) and `R11263` (in orange) share the same reaction signature. The set of reaction signatures was obtained for both atom and bond neighborhood signature types, for heights from 1 to 4.

This example highlights the interest of using sets of reaction signatures. In addition to the metabolic and genomic patterns meaning that the two mostly overlapping trails are catalyzed by products of neighboring genes for both species, the corresponding set of reaction signatures shows that the non-overlapping reactions (`R04424` for *E. coli* and `R11263` for *V. cholerae*) perform the same type of chemical transformation.

Trail grouping (described in Chapter V) is not able to capture this pattern. Indeed, grouping by reactions results in a reaction set for *E. coli* that includes the reaction `R04424`, and in a reaction set for *V. cholerae* that includes the reaction `R11263`. However, the two tables $T^r_{\texttt{eco}}$ and $T^r_{\texttt{vco}}$ need to be manually compared in order to determine trail overlap. Moreover, trail grouping does not evaluate chemical similarity between trails.

Nevertheless, there is an advantage to the extraction of simpler metabolic and genomic patterns (without the chemical aspect). The reaction `R11264` (in orange) is performed by *V. cholerae* using the product of a gene that is found in the same genomic context as the genes involved in the other reactions in the trail. This reaction, however, is an isomerisation, meaning that its substrate and product have the same chemical formula but with a different arrangement of atoms. For this reason,

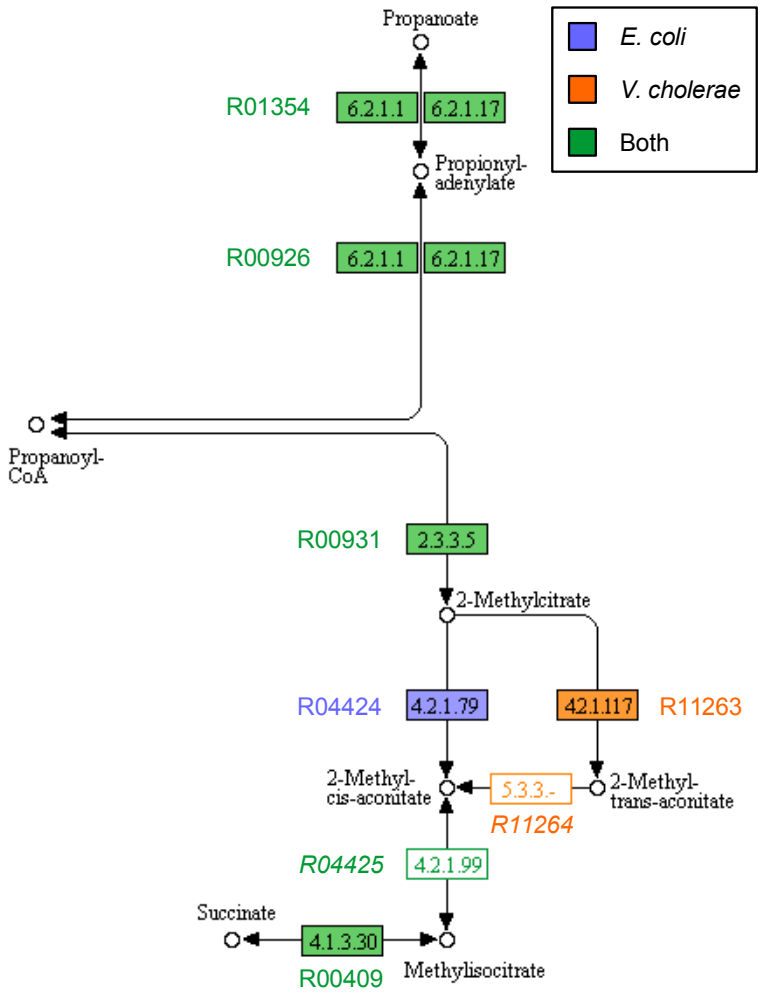



**Figure VIII.7 Partial view of the propanoate metabolism pathway.** Adapted from KEGG PATHWAY `map00640` (November 1, 2017 version). Shown here is a set of reaction signatures with two associated `CoMetGeNe` reaction sets. The reaction set in green and blue corresponds to a `CoMetGeNe` trail in *Escherichia coli*, and the one in green and orange to *Vibrio cholerae*. Reactions are labeled with the corresponding KEGG reaction identifiers (R numbers). Reactions with empty rectangles (labels in italics) are not part of the set of reaction signatures. Reaction `R04425` (in green) is performed by both species using the products of distant genes. Reaction `R11264` (in orange), having an empty signature for diameters from 0 to 9, is performed by *V. cholerae* using the product of a gene that shares the same neighborhood as the other genes involved in the green and orange trail.

the associated reaction signature is empty (see Figure VIII.4). Whereas a chemical, metabolic, and genomic pattern would not be able to capture this transformation, a simpler metabolic and genomic pattern may include reactions with empty signatures.

### 4.3.2 Non-overlapping trails in the same species

Figure VIII.8 shows two `CoMetGeNe` trails obtained for *Bacteroides fragilis* in the tricarboxylic acid cycle (panel a) and in the valine, leucine, and isoleucine biosynthesis pathway (panel b), respectively.

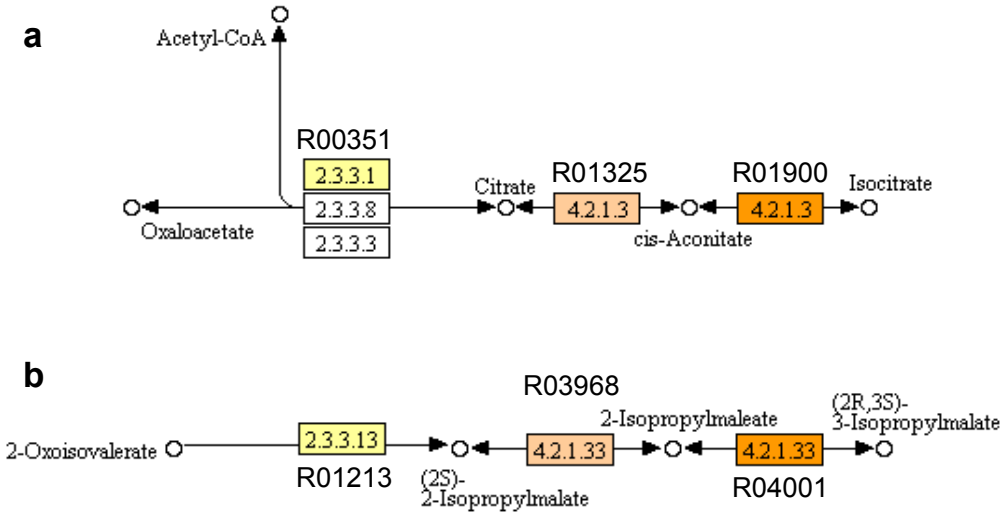



**Figure VIII.8 A chemical, metabolic, and genomic pattern for *Bacteroides fragilis*.** Shown here is a set of reaction signatures with two associated reaction sets (one in each pathway). Reactions are labeled with the corresponding KEGG reaction identifiers (R numbers). Reactions with the same color have the same signature of height 1 for atom neighborhood. **(a)** Partial view of the tricarboxylic acid cycle. Adapted from KEGG PATHWAY, map `bfr00020` (June 7, 2018 version). **(b)** Partial view of the valine, leucine, and isoleucine biosynthesis pathway. Adapted from KEGG PATHWAY, map `bfr00290` (March 7, 2017 version).

The two trails, and consequently their corresponding reaction sets, are disjoint. However, both reaction sets are associated to the same set of reaction signatures of height 1 for atom neighborhood. Reactions having the same signature are displayed with the same color in Figure VIII.8.

Interestingly, while the two pairs of reactions in yellow and light orange are found to be similar using EC-BLAST [Rahman *et al.*, 2014], this is not the case for the reactions in dark orange (`R01900` and `R04001`). EC-BLAST is a fingerprint-based chemical similarity search tool for reactions. Similarity is expressed as a score between 0 (no similarity) and 1 (maximum similarity). One of the methods proposed by EC-BLAST is bond similarity, in which fingerprints of bond change patterns are compared. This method reports that the pairs of reactions in yellow and light orange have bond similarity scores of 0.91 and 1.00, respectively. Another search method, based on comparing reaction center information, is the only one to report

that reactions R01900 and R04001 are similar, with a relatively low score of 0.61.

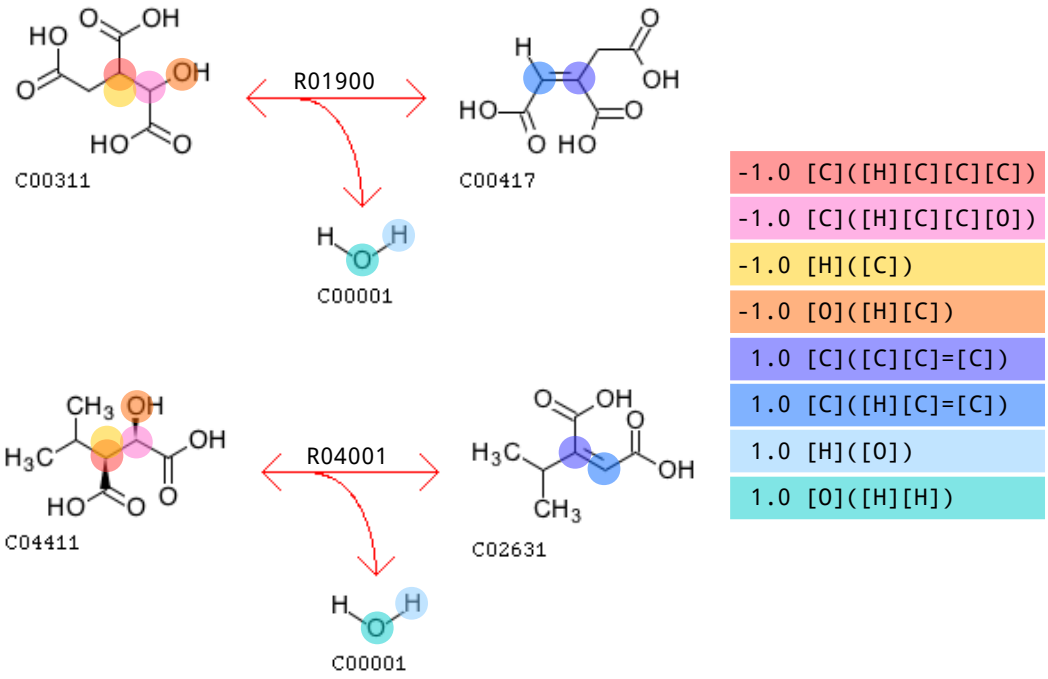



**Figure VIII.9 Reactions R01900 and R04001 have the same signature of height 1 for atom neighborhood.** Each descriptor in the reaction signature (right) is color-coded, with the corresponding atoms highlighted in the same color for the two reactions. Adapted from KEGG REACTION.

As shown in Figure VIII.9, although the substructures of the chemical compounds involved in these reactions differ, their overall structure is similar. Both substrates and products consist of three-carbon backbones with carboxyl groups at one end. Additionally, the product backbones exhibit a double carbon–carbon bond. Since the reaction signature for R01900 and R04001 is essentially based on backbone atoms and their direct neighbors, it follows that substructures do not play an important role in the specificity of these two chemical transformations. It is therefore relevant to describe the two reactions as similar.

This example illustrates how sets of reaction signatures may help uncover subtler metabolic and genomic organization patterns. Two distinct CoMetGeNe trails obtained for the same species in different pathways may in fact perform the same types of chemical transformations. If this is the case, it can be hypothesized that the genes involved in reactions having the same signature originate from a gene duplication event. Indeed, KEGG SSDB (see section III.2.2) reports that the genes in *B. fragilis* involved in the reactions in light orange in Figure VIII.8 are paralogs[5].

[5] https://www.kegg.jp/ssdb-bin/ssdb_paralog?org_gene=bfr:BF3755

# 5 Sets of reaction signature clusters

## 5.1 Approach

We propose associating sets of reaction signature clusters to `CoMetGeNe` trails (see Figure VIII.10 for an overview of the approach).

The similarity between reaction signatures is quantified using the Tanimoto coefficient (see section 2). Since this coefficient is a real number between 0 and 1, the distance between two reaction signatures ${}^{d}\sigma(r_i)$ and ${}^{d}\sigma(r_j)$ at diameter $d$ is expressed as $1 - {}^{d}T_c(r_i, r_j)$, where ${}^{d}T_c(r_i, r_j)$ is the Tanimoto coefficient applied to the two reaction signatures. Average linkage hierarchical clustering is performed in order to group together similar reaction signatures. Conservative cutoff thresholds ranging between 0.01 and 0.10 are used in order to avoid "over-clustering". Since reaction signatures represent the first level of reaction similarity, clustering them in an overly relaxed manner may lead to clusters that describe quite different chemical transformations. The idea is to group together comparable chemical transformations that do not have the same reaction signature. Overdoing this grouping process risks to be devoid of biochemical meaning.

Similarly to the first approach (see section 4.1 above), `CoMetGeNe` trails are transformed into their corresponding reaction sets. Reaction sets are then "translated" into *sets of reaction signature clusters*. A formal definition follows.

*set of reaction signature clusters*

**Definition VIII.2.** Let $\mathcal{R} = \{r_1, r_2, \ldots, r_n\}$ be a reaction set associated to a `CoMetGeNe` trail. The *set of reaction signature clusters* associated to $\mathcal{R}$ for a given cutoff threshold $t$ is the set

$$\bigcup_{i \in \{1,\ldots,n\}} \mathcal{C}_t({}^{d}\sigma(r_i))$$

where $\mathcal{C}_t({}^{d}\sigma(r_i))$ is the cluster of reaction signatures obtained for threshold $t$ that contains the signature ${}^{d}\sigma(r_i)$ of reaction $r_i$ at diameter $d$.

A reaction set has a unique corresponding set of reaction signature clusters. However, a given set of reaction signature clusters may correspond to several reaction sets. For example, the reaction sets associated to trail $T_2$ and $T_5$ in Figure VIII.10 have the same corresponding set of reaction signature clusters. It consists in the yellow, red, and blue clusters, each of them containing several similar reaction signatures (in the figure, the similarity between reaction signatures is represented by signatures of the same color). This property enables a quantitative evaluation of reaction set similarity, and hence of trail similarity, in which the quantitative aspect is given by the Tanimoto coefficient between two reaction signatures.

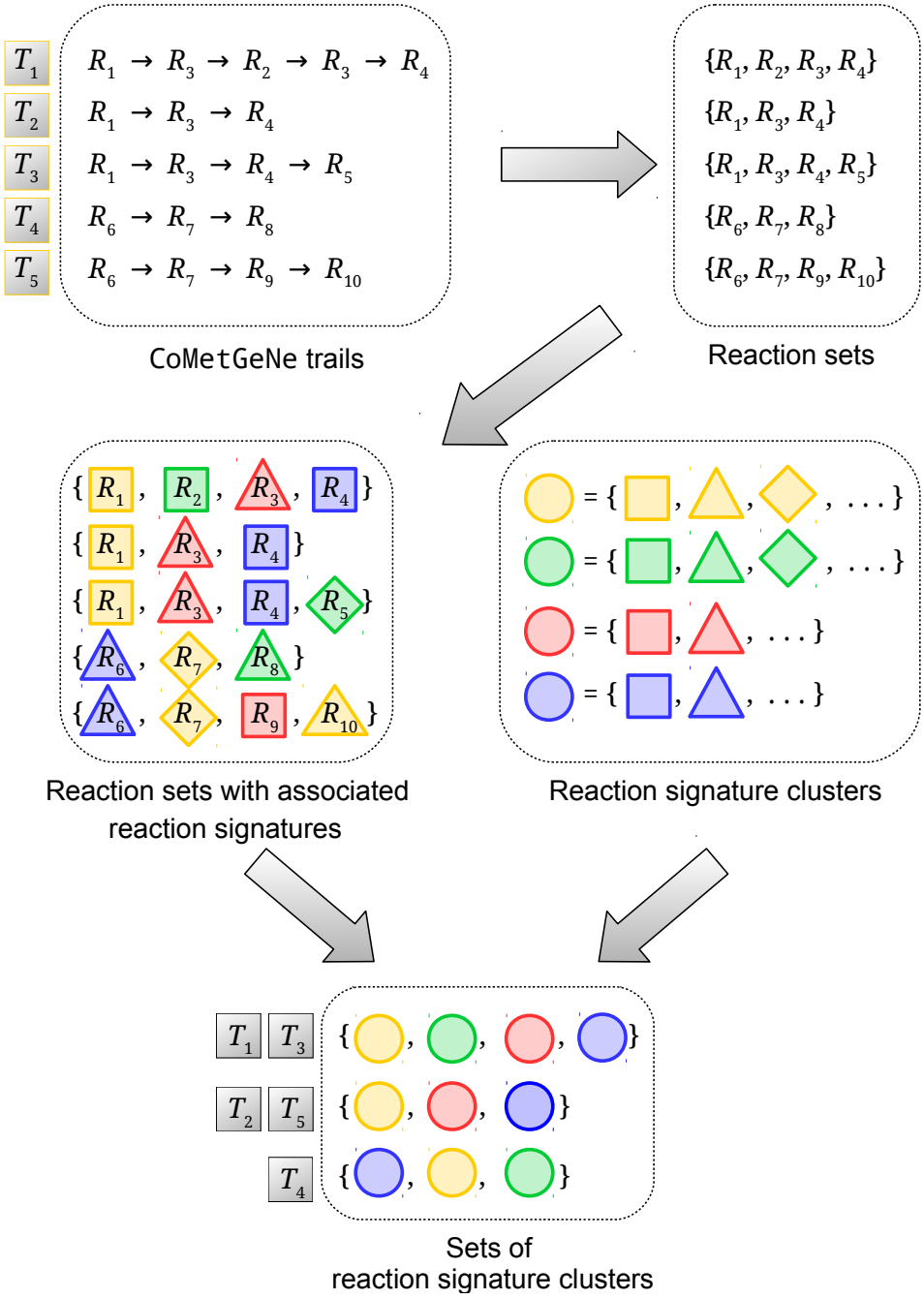



**Figure VIII.10 Overview of the approach consisting in transforming reaction sets into sets of reaction signature clusters.** `CoMetGeNe` trails ($T_1$ to $T_5$) are transformed into reaction sets. Clusters of reaction signatures are established on the basis of reaction signature similarity. In this approach, reaction signatures are integrated by determining the set of reaction signature clusters corresponding to a given `CoMetGeNe` reaction set. In this example, the reaction sets associated to trails $T_1$ and $T_3$ have the same corresponding set of reaction signature clusters. It is also the case for the reaction sets associated to trails $T_2$ and $T_5$.

As in the case of the previous approach, measures need to be taken to prevent dissimilar reaction sets from being qualified as similar. The first measure is to only consider sets of reaction signature clusters associated to reaction sets in which every reaction has a computable signature. Such a set of reaction signature clusters is referred to as *complete*. The second measure consists in ignoring sets of reaction signature clusters containing clusters that feature empty signatures. A set of reaction signature clusters without empty signatures is referred to as *valid*.

*complete set of reaction signature clusters*

*valid set of reaction signature clusters*

## 5.2 Results

Tanimoto coefficients were computed for every possible pair of reactions present in `CoMetGeNe` trails, for signature diameters ranging from 0 through 9. Bottom-up hierarchical clustering was then performed on reaction signatures, using average linkage and a cutoff threshold ranging between 0.01 and 0.10 by increments of 0.01. As explained in the previous section (see 5.1), these conservative values have been chosen in order to avoid the over-clustering of reaction signatures and, consequently, the loss of biological meaning.

Table VIII.2 shows averages over the ten cutoff thresholds for the number of reaction signature clusters, the percentage of clusters among them that contain a single reaction signature (i.e., singleton clusters), and the number of reaction signatures per cluster. If the values obtained for diameters 0 and 1 are ignored, as signatures of height 0 are very general, this table confirms that the clustering is indeed minimal. Thus, on average for the ten cutoff thresholds, at least 96% of all clusters are singletons, and the average number of reaction signatures per cluster is between 1.01 and 1.06. Higher values of the cutoff threshold would result in less singleton clusters with the effect that clusters would contain more reaction signatures, on average.

Figure VIII.11 shows the number of complete and valid sets of reaction signature clusters for different values of the signature diameter, averaged for clustering cutoff thresholds between 0.01 and 0.10. Among the total number of sets of reaction signature clusters, the complete and valid sets represent between 45.53% (atom type signature of height 0) and 91.63% (bond type signature of height 4), with an average of 82.36%.

The average number of reaction sets with computable and non-empty signatures per complete and valid set of reaction signature clusters is indicated above each bar in Figure VIII.11. These values are averaged across the ten cutoff thresholds. They are very close to the average number of reaction sets per set of reaction signatures (see Figure VIII.6).

| Diameter | <#clusters> ± SD | <%singleton> ± SD | <#signatures/cluster> |
|---|---|---|---|
| 0 | 58.70 ± 4.78 | 90.65% ± 2.11% | 1.23 |
| 1 | 247.20 ± 13.09 | 94.12% ± 4.05% | 1.08 |
| 2 | 520.00 ± 11.00 | 98.15% ± 1.73% | 1.02 |
| 3 | 729.50 ± 34.86 | 96.23% ± 2.82% | 1.06 |
| 4 | 902.50 ± 30.18 | 97.42% ± 2.39% | 1.03 |
| 5 | 981.10 ± 29.80 | 97.01% ± 2.52% | 1.04 |
| 6 | 1041.70 ± 34.52 | 96.98% ± 2.50% | 1.04 |
| 7 | 1090.30 ± 21.01 | 97.36% ± 1.66% | 1.03 |
| 8 | 1113.60 ± 28.07 | 97.30% ± 2.09% | 1.03 |
| 9 | 1158.30 ± 12.76 | 98.88% ± 0.96% | 1.01 |

**Table VIII.2 Statistics for clusters of reaction signatures corresponding to reactions in `CoMetGeNe` trails.** For every signature diameter between 0 and 9 are shown the average number of reaction signature clusters ± standard deviation (<#clusters> ± SD), the percentage of singleton clusters ± standard deviation (<%singleton> ± SD), and the average number of reaction signatures per cluster (<#signatures/cluster>). Values are averaged for clustering cutoff thresholds between 0.01 and 0.10. Singleton clusters are clusters of reaction signatures with a single associated reaction signature.

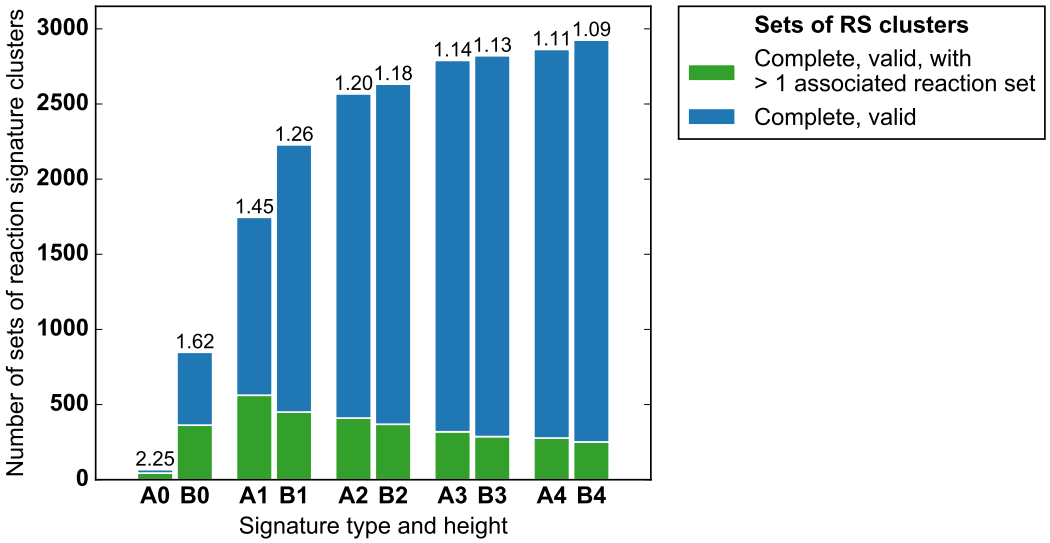



**Figure VIII.11 Sets of reaction signature clusters derived from reaction sets.** Reaction signatures are clustered on the basis of a distance criterion that reflects the similarity between signatures. Every `CoMetGeNe` reaction set in which all reactions have computable reaction signatures is then assigned to a set of reaction signature clusters. Such a set is said to be a *complete* set of reaction signature clusters. Here are shown, for every signature diameter, the mean numbers of complete sets of reaction signature clusters in which every reaction signature is non-empty (i.e. *valid* sets of reaction signature clusters), averaged for clustering cutoff thresholds between 0.01 and 0.10. Bar labels designate the signature height for atom (A) and bond (B) neighborhood. Values above each bar represent the mean number of reaction sets (with computable and non-empty signatures for every reaction) corresponding to one complete set of reaction signature clusters without any empty signatures.

The sets of reaction signature clusters of interest are sets corresponding to at least two reaction sets (in green in Figure VIII.11). On average, these sets of reaction signature clusters amount to between 8.59% and 9.67% (for bond and atom type signatures, respectively, of height 4), and 42.77% and 64.96% (for bond and atom type signatures, respectively, of height 0) of the total number of complete and valid sets of reaction signature clusters at each diameter. These percentages are roughly two times greater than those corresponding to Figure VIII.6 and representing the fraction of sets of reaction signatures of interest. Thus, although the clustering of reaction signatures is minimal as illustrated in Table VIII.2, it is also effective, since the ratio of interesting sets is significantly higher than in the previous approach.

## 5.3 Metabolic building blocks

Figure VIII.12 shows two `CoMetGeNe` trails obtained for *Acetomicrobium mobile* in the pathways for arginine (top) and lysine (bottom) biosynthesis. Each color represents a pair of similar reactions. For signature diameters between 2 and 4, every pair of reactions shares the same signature (the Tanimoto coefficient is 1). The similarity coefficients in the figure refer to a signature diameter of 5. If clustering of similar reaction signatures had not been performed, these trails would not have been reported as similar at diameter 5. For diameters 6 through 9, the trails are no longer identified as similar.

Apart from representing metabolic and genomic patterns, these trails are also chemically similar. Although this information is available using lower diameters (2, 3, or 4), the example illustrates how higher diameters can still yield meaningful biological information when only minimal clustering is performed.

Moreover, this chemical, metabolic, and genomic pattern is an example of metabolic building block, or module. In the literature, metabolic *modules* are seen as successive enzymatic steps performing similar chemical transformations [Muto *et al.*, 2013; Sorokina *et al.*, 2015].

*A. mobile* is the only species among those in Table VII.1 to possess these two `CoMetGeNe` trails. The genes involved in these two trails are the same, meaning that *A. mobile* uses products of the same genes to synthesize arginine and lysine. Interestingly, this is not the case for the other species in the data set. Among them, for example, all Terrabacteria use a parallel metabolic route to obtain arginine, and a different route altogether for lysine biosynthesis.

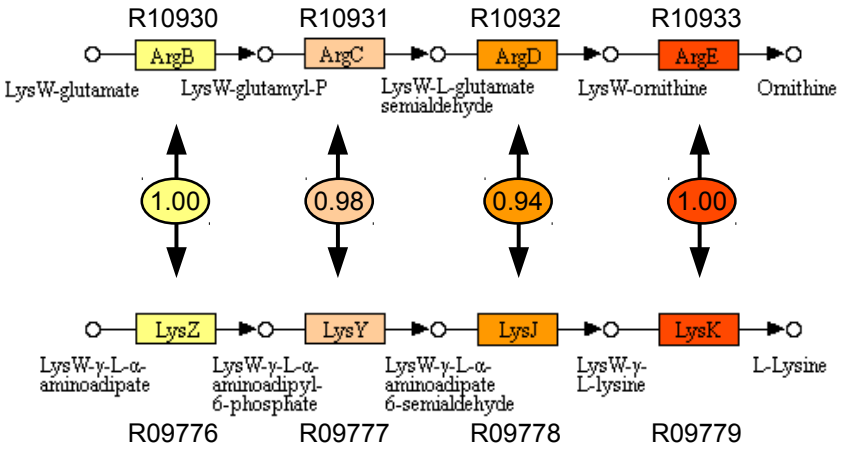



**Figure VIII.12 A chemical, metabolic, and genomic pattern for *Acetomicrobium mobile*.** Top: Partial view of the arginine biosynthesis pathway in *A. mobile*. Adapted from KEGG PATHWAY, map `amo00220` (July 20, 2017 version). Bottom: Partial view of the lysine biosynthesis pathway in *A. mobile*. Adapted from KEGG PATHWAY, map `amo00300` (June 23, 2017 version). Shown here is a set of reaction signature clusters with two associated `CoMetGeNe` reaction sets (one in each pathway). Reactions are labeled with the corresponding KEGG reaction identifiers (R numbers). Reactions with the same color have similar or identical signatures at diameter 5, as indicated by the Tanimoto coefficients between each pair of similar reactions.

# 6 Discussion

**Metabolic modularity** Perhaps the most natural application for chemical similarity is to detect enzymatic "building blocks" revealing metabolic modularity.

The two approaches presented in this chapter uncover chemical, metabolic, and genomic patterns that often translate to metabolic modules. Approaches that perform the exhaustive enumeration of possible sequences of reaction steps define modules as successive reactions performing the same type of transformations, as has been done by Muto *et al.* [2013] (using fingerprint-based signatures) and Sorokina *et al.* [2015] (using reaction signatures). In this thesis, a *metabolic module* is a chemical, metabolic, and genomic pattern. In other words, most definitions in the literature require modules to fulfill two criteria (chemical similarity and metabolic context), whereas in this thesis they fulfill three criteria: chemical similarity, metabolic context, and genomic context.

*metabolic module*

**Sets of reaction signatures** The approach consisting in associating sets of reaction signatures to `CoMetGeNe` trails (see section 4) is a *qualitative* approach. It allows to establish chemical similarity between reaction sets corresponding to `CoMetGeNe`

trails that may be either partly overlapping (see 4.3.1) or disjoint (see 4.3.2).

Sets of reaction signatures of interest are those corresponding to at least two reaction sets. As can be seen from Figure VIII.6, most sets of reaction signatures of interest are found for heights 1 and 2, especially for atom type signatures.

Of the ten signature diameters that were tested, those corresponding to height 0 are very generic, as they only provide the remaining counts of atoms or bonds after subtracting molecular signatures of the substrates from those of the reaction products.

Signatures of heights 3 and 4 are very specific, describing atom and bond neighborhood at up to 3 and 4 surrounding atoms and bonds, respectively. Although very few sets of reaction signatures at these heights have more than one corresponding reaction set, they capture almost identical reactions, for example reactions in which coenzymes are either $NAD^+$/NADH or $NADP^+$/NADPH[6], or reactions hydrolyzing purine mononucleotides[7]. Consequently, sets of reaction signatures at theses heights containing reactions in disjoint reaction sets often describe parallel metabolic blocks involving the same types of chemical transformations.

Sets of reaction signatures complement trail grouping in `CoMetGeNe` by identifying chemical, metabolic, and genomic patterns, that is, metabolic and genomic patterns in which reactions perform similar chemical transformations. If they involve disjoint `CoMetGeNe` trails and occur in the same species (see Figure VIII.8), chemical, metabolic, and genomic patterns highlight metabolic modularity. If occurring in different species and involving partially overlapping (see Figure VIII.7) or disjoint `CoMetGeNe` trails, such chemical, metabolic, and genomic patterns may reveal subtler conservation aspects of metabolic and genomic organization, while also offering insights into metabolic evolution.

When two reactions have the same signature at lower heights (e.g. 1 and 2), but not at greater heights (e.g. 3 and 4), it means that the reactions are similar, but not "that similar". The very difficulty in explaining the difference between such reactions points out the fact that the limitation of using sets of reaction signatures is their inability to quantify reaction similarity. This limitation is addressed by a second approach that makes use of sets of reaction signature clusters (see below).

**Sets of reaction signature clusters** The approach consisting in associating sets of reaction signature clusters to `CoMetGeNe` trails (see section 5) is a *quantitative*

[6]For instance, reactions `R10221` and `R01528` have the same signature of height 4: https://www.genome.jp/dbget-bin/www_bget?R10221+R01528

[7]For instance, reactions `R00183` and `R01227` have the same signature of height 4: https://www.genome.jp/dbget-bin/www_bget?R00183+R01227

approach. It allows to establish chemical similarity between reaction sets corresponding to CoMetGeNe trails in which reactions may not share the same reaction signatures.

Similarly to the previous approach, most sets of reaction signatures of interest are found for lower heights (1 and 2). However, as shown in the example (see 5.3), higher diameters may contain interesting trails due to clustering of similar reaction signatures.

By quantifying reaction similarity, this method provides a solution to the problematic situation in which reaction signatures are distinct although chemically close. Trails that would otherwise be considered different may be found similar when passing from sets of reaction signatures to sets of reaction signature clusters.

Since the quantification of reaction similarity is performed through clustering, care needs to be taken in the choice of the cutoff threshold. Small values were used in this application (ranging from 0.01 to 0.10) in order to avoid over-clustering. However, manual analysis of several groups of trails hinted to the fact that, in some situations, thresholds up to 0.20 may be pertinent.

Many chemical, metabolic, and genomic patterns identified using sets of reaction signature clusters fall within the category of "metabolic building blocks", representing successive reactions that generally occur in distinct pathways and lead to the production of similar compounds through similar chemical transformations. In this thesis, modules are further restricted to metabolic building blocks that involve products of neighboring genes.

## 7 Concluding remarks

This chapter proposed a refinement of the concept of metabolic and genomic patterns. When taking into account reaction similarity, certain CoMetGeNe trails reveal chemical, metabolic, and genomic patterns. These patterns still represent reactions that are catalyzed by products of neighboring genes, with the distinction that the transformations they perform are chemically similar.

Chemical similarity is evaluated using two approaches, the first one qualitative (sets of reaction signatures) and the second one quantitative (sets of reaction signature clusters). Both approaches reuse the concept of reaction sets introduced for trail grouping.

The qualitative approach consists in associating reaction sets to sets of reaction signatures. Thus, several reaction sets associated to a single set of reaction signatures indicate that the reaction sets in question are chemically similar.

The quantitative approach consists in clustering similar reaction signatures and

in associating reaction sets to sets of reaction signature clusters. In this approach, reaction sets are treated as similar if the reactions they contain perform somewhat different, but chemically close transformations.

Intuitively, chemical, metabolic, and genomic patterns allow for an extension of trail grouping in which reaction sets are replaced by sets of either reaction signatures or reaction signature clusters. In practice, the advantage of such a trail grouping method is that it complements "classical" trail grouping by revealing metabolic modules. Although several definitions of metabolic modules exist, in this approach they are seen as elementary building blocks of metabolism linking genomic organization to metabolic function. More specifically, conserved chemical, metabolic, and genomic patterns reflect the fact that the organization of genomic context is conserved for several species in order to perform a given *type* of metabolic function.

# Conclusions and perspectives

This thesis fits within the field of systems biology and addresses a problem related to heterogeneous biological networks. It focuses on the relationship between metabolism and genomic context through a graph mining approach.

It is well-known that succeeding enzymatic steps involving products of genes in close proximity on the chromosome translate an evolutionary advantage in maintaining this neighborhood relationship at both the metabolic and genomic levels. We therefore chose to focus on the detection of neighboring reactions being catalyzed by products of neighboring genes, where the notion of neighborhood may be modulated by allowing the omission of several reactions and/or genes. More specifically, the sought motifs are trails of reactions (that is, reaction sequences in which reactions may be repeated) being catalyzed by products of neighboring genes. For simplicity, these motifs are called *metabolic and genomic patterns*.

The particular choice of extracting trails is motivated by three aspects. The first one is the fact that cycles are ubiquitous in metabolism and the only way to capture them is to repeat the reactions that serve as entry and exit points to and from cycles, respectively. The second and third aspects are related to the biological significance of the extracted motifs. First, by representing motifs in directed graphs, trails incorporate reaction directionality. Various approaches extract subgraphs in the undirected case, which results in ignoring reaction directionality. This means that there may be no metabolic routes corresponding to the extracted motifs. Then, trails of reactions translate metabolic routes. Thus, by the very problem definition, neighboring genes involved in a given trail are guaranteed to be involved in the corresponding metabolic route. If subgraphs were extracted, the neighboring genes involved in the reactions defining the subgraphs would not necessarily be

neighbors for the different corresponding metabolic routes.

In addition to the identification of metabolic and genomic patterns, we also investigate the degree of *conservation* of such patterns among multiple species. Similarly to the notion of metabolic and genomic neighborhood, a flexible definition of pattern conservation is adopted. Thus, when evaluating conservation, the order of reactions in trails and the order of functionally similar genes on the chromosome may differ between species. Moreover, the conservation may be partial, meaning that the composition of trails and genomic contexts may vary, with some species having only conserved part of a metabolic and genomic pattern detected in other organisms.

The exploration of the relationship between metabolism and genomic context is therefore captured by the two main objectives of this thesis: the detection of metabolic and genomic patterns for a single species on the one hand, and the study of conserved metabolic and genomic patterns among multiple species on the other hand.

## Contributions

**Trail finding** In order to detect metabolic and genomic patterns for a given species, we propose a heterogeneous graph mining methodology called trail finding (see Chapter IV). The underlying graph model may be easily modified in order to accommodate different types of biological data, such as metabolic pathways and protein–protein interaction networks. We present the exact algorithm HNET which performs trail enumeration in a metabolic pathway. Trail enumeration in a directed graph is naturally solved through path enumeration in its line graph. The scope of this computationally expensive operation is decreased by applying a reduction to the input graphs, and is further restricted to only enumerate paths between vertices that are susceptible to be part of the sought solution.

**Trail grouping** In order to detect conserved metabolic and genomic patterns between several species, we propose a methodology called trail grouping (see Chapter V). In order to account for variations between similar trails in terms of reaction and/or gene order, as well as their respective presence or absence, trail grouping transforms trails into reaction sets. Two approaches are proposed for evaluating the conservation of trails belonging to a designated reference species: trail grouping by reactions, focusing on the conservation of metabolic patterns, and trail grouping by genes, focusing on the conservation of genomic patterns. Both approaches

construct tables akin to phylogenetic profiles for reactions sets or groups of neighboring genes involved in trails of the reference species. Jointly, these profiles allow to compare the degree of trail conservation among the species under study.

**`CoMetGeNe`** The trail finding and trail grouping methodologies are implemented in an easy-to-use open-source pipeline called `CoMetGeNe`, short for ***Co**nserved **Met**abolic and **Ge**nomic **Ne**ighborhoods* (see Chapter VI). `CoMetGeNe`, available at the address `https://cometgene.lri.fr`, is used to analyze a set of 50 bacterial species spanning major phyla of the bacterial tree of life (see Chapter VII), showing that the trail finding and trail grouping methodologies serve as exploratory tools for investigating the links between metabolic and genomic contexts. We highlight the discovery aspect of our approach, showing that the identified metabolic and genomic patterns may lead to biological insights, to the formulation of biological hypotheses, as well as to the detection of annotation problems in public knowledge bases as a side effect.

A paper summarizing these contributions has been submitted to *BMC Bioinformatics* on June 25, 2018 [Zaharia *et al.*, 2018]. It describes the trail finding and trail grouping methodologies, presents the `CoMetGeNe` pipeline, and outlines examples of biological applications (sections VII.4 and VII.5).

**Extension to chemical similarity** The notion of metabolic and genomic patterns can be extended to account for the chemical similarity between several trails (see Chapter VIII). These extended patterns are called *chemical, metabolic, and genomic patterns* and reflect the fact that the nature of the chemical transformations is another factor in the relationship between metabolism and the genome. One of the possible definitions of chemical similarity is used to compute reaction signatures. We then propose two approaches that extend the grouping of `CoMetGeNe` trails. The first one is qualitative and consists in deciding whether two reactions are similar, whereas the second one allows to quantify reaction similarity. The two approaches reuse the concept of reaction sets, by associating them to either sets of reaction signatures, or to sets of reaction signature clusters. Existing studies on metabolic modularity usually define metabolic modules as sequences of chemically similar enzymatic transformations. We show that chemical, metabolic, and genomic patterns correspond to a particular type of metabolic modules in which the genes encoding the enzymes are neighbors.

**Detection of consistency issues in KEGG** Through the extensive use of the KEGG knowledge base (see Chapter III) during the course of this thesis, several consistency issues became apparent. Two cases are illustrated, the first concerning disconnected reactions in pathway maps, and the second concerning reactions being inconsistently marked as present and absent between pathway maps of the same species. In both cases, a general approach allowing to systematically identify such occurrences is outlined. Whereas the first issue is immediately noticeable, the second one may be subject to interpretation depending on the definition of inconsistent treatment of reactions. Nevertheless, we report these issues since such discrepancies between various KEGG databases may have an important impact on the bioinformatic community relying on KEGG as a reference resource for linking genome to function.

## Perspectives

**Trail finding** It would be interesting to tailor existing approaches (see section II.4) that extract undirected subgraphs to perform a post-processing step checking whether these motifs correspond to actual metabolic routes. If this is the case, an additional filtering step would only retain the routes involving neighboring genes. These modified methods could then be benchmarked against `CoMetGeNe` in terms of pattern detection and execution time.

In addition, network topology (see section II.3.1) may be used to adjust the trail finding strategy. A first possibility is to directly perform path finding in metabolic pathways or their subgraphs instead of passing through the line graph (see section IV.4.2) if no cycles exist. Deciding whether cycles are present can be performed using a depth-first search in which back edges indicate cycles. A second possibility referring to connectivity aspects would be to only focus on trails passing through reactions playing important roles in a given metabolic pathway. Two examples are hub reactions, meaning reactions with high degree (see definition II.14), and critical reactions for the overall network connectivity, meaning reactions with high betweenness centrality (see definition II.16). Such an approach might prove useful if the whole metabolic network would be considered for trail finding instead of isolated pathways.

**Trail grouping and reaction similarity** As explained in Chapter VIII, the definition of chemical, metabolic, and genomic patterns may be used for an extended version of trail grouping in which reaction sets would be replaced by either sets of reaction signatures or sets of reaction signature clusters. As explained previously,

this approach has the benefit of detecting metabolic modules in which the genes that encode enzymes performing similar transformations are neighbors. Although the general approach is outlined in Chapter VIII, the theoretical framework for trail grouping still needs to be modified accordingly. The preliminary results that were presented have been obtained using a proof-of-concept piece of software code that requires extensive improvements prior to its integration into `CoMetGeNe`.

Reaction similarity is established through the use of the signature molecular descriptor. To simplify, the signature of a reaction is given by the atoms and bonds that are not common to its substrates and products. However, this approach does not guarantee that the remaining atoms and bonds are actually the ones that were modified during the reaction. For this purpose, atom-to-atom mapping approaches should be considered.

As shown in Chapter VIII, reactions with empty signatures are problematic. As a precautionary measure, reaction sets having reactions with empty signatures have been excluded from the analysis. This constraint could however be removed, which would result in additional chemical, metabolic, and genomic patterns that might prove meaningful. Assuming that reaction signatures as well as similarity coefficients are precomputed for several diameters, another possible solution for handling reactions with empty signatures is to consider that two such reactions share the same signature if they both have a maximum similarity coefficient for higher values of the signature diameter. This would in turn impact the quantitative approach (see section VIII.5), for which the clustering method should be refined.

**Visualization** Although `CoMetGeNe` is an easy-to-use pipeline, it does not offer any visualization options. From a user's perspective, it would be practical to have an integrated viewer that highlights the obtained trails (for trail finding) or reactions in slightly different trails in different species (for trail grouping). Another point of interest (currently lacking from KEGG) is the ability to link in a one-step process the definition of a given reaction to the gene(s) involved in this reaction. During my teaching activities, I conceived two projects[1] addressing these aspects and presented them to two groups of first year Master's students in bioinformatics. Their implementation (in Java) indicates that elegant visualization solutions can be realistically envisaged. Such solutions would simplify the biological interpretation of metabolic and genomic patterns detected using `CoMetGeNe` by automatically highlighting them in an integrated viewer.

[1] Visualization and superposition of metabolic pathways (2016/2017): https://www.lri.fr/~zaharia/EdC2016/Projet_EdC_2016_2017.pdf. KEGG browser (2017/2018): https://www.lri.fr/~zaharia/EdC2017/Projet_EdC_2017_2018.pdf.

# Bibliography

Albert, R., Jeong, H., and Barabási, A.-L. Error and attack tolerance of complex networks. *Nature*, 406(6794):378, 2000. [Cited on page 38.]

Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K., and Walter, P. *Molecular biology of the cell*. Garland Science, fifth edition, 2008. [Cited on pages 18 and 19.]

Algfoor, Z. A., Sunar, M. S., and Kolivand, H. A comprehensive study on pathfinding techniques for robotics and video games. *International Journal of Computer Games Technology*, 2015:7, 2015. [Cited on page 30.]

Altenhoff, A. M. and Dessimoz, C. Inferring orthology and paralogy. In *Evolutionary genomics: statistical and computational methods*, volume 1, pages 259–279. Springer, 2012. [Cited on pages 24 and 25.]

Altman, T., Travers, M., Kothari, A., Caspi, R., and Karp, P. D. A systematic comparison of the MetaCyc and KEGG pathway databases. *BMC Bioinformatics*, 14 (1):112, 2013. [Cited on page 10.]

Altschul, S. F., Madden, T. L., Schäffer, A. A., Zhang, J., Zhang, Z., Miller, W., and Lipman, D. J. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*, 25(17):3389–3402, 1997. [Cited on pages 25, 143, 144, and 147.]

Alves, R., Chaleil, R. A., and Sternberg, M. J. Evolution of enzymes in metabolism: A network perspective. *Journal of Molecular Biology*, 320(4):751–770, 2002. [Cited on pages 1 and 14.]

Arita, M. The metabolic world of *Escherichia coli* is not small. *Proceedings of the National Academy of Sciences*, 101(6):1543–1547, 2004. [Cited on pages 11 and 38.]

Asensio, N. C., Giner, E. M., De Groot, N. S., and Burgas, M. T. Centrality in the host–pathogen interactome is associated with pathogen fitness during infection. *Nature Communications*, 8:14092, 2017. [Cited on pages 35 and 36.]

Azé, J., Gentils, L., Toffano-Nioche, C., Loux, V., Gibrat, J.-F., Bessières, P., Rouveirol, C., Poupon, A., and Froidevaux, C. Towards a semi-automatic functional annotation tool based on decision-tree techniques. In *BMC Proceedings*, volume 2, page S3. BioMed Central, 2008. [Cited on page 27.]

Balakrishnan, R. and Ranganathan, K. *A textbook of graph theory*. Springer Science & Business Media, second edition, 2012. [Cited on pages 30 and 32.]

Bang-Jensen, J. and Gutin, G. Z. *Digraphs: theory, algorithms and applications*. Springer Science & Business Media, second edition, 2008. [Cited on page 30.]

Barabási, A.-L. and Albert, R. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999. [Cited on pages 38, 39, and 40.]

Barabási, A.-L. and Oltvai, Z. N. Network biology: understanding the cell's functional organization. *Nature Reviews Genetics*, 5(2):101, 2004. [Cited on pages 37 and 39.]

Barreteau, H., Kovač, A., Boniface, A., Sova, M., Gobec, S., and Blanot, D. Cytoplasmic steps of peptidoglycan biosynthesis. *FEMS Microbiology Reviews*, 32(2): 168–207, 2008. [Cited on page 139.]

Bastard, K., Smith, A. A. T., Vergne-Vaxelaire, C., Perret, A., Zaparucha, A., De Melo-Minardi, R., Mariage, A., Boutard, M., Debard, A., Lechaplais, C., *et al.* Revealing the hidden functional diversity of an enzyme family. *Nature Chemical Biology*, 10(1):42, 2014. [Cited on page 10.]

Bender, A. and Glen, R. C. Molecular similarity: a key technique in molecular informatics. *Organic & Biomolecular Chemistry*, 2(22):3204–3218, 2004. [Cited on page 150.]

Blin, G., Fertin, G., Mohamed-Babou, H., Rusu, I., Sikora, F., and Vialette, S. Algorithmic aspects of heterogeneous biological networks comparison. In *International Conference on Combinatorial Optimization and Applications*, pages 272–286. Springer, 2011. [Cited on pages 52 and 54.]

Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., and Hwang, D.-U. Complex networks: Structure and dynamics. *Physics Reports*, 424(4-5):175–308, 2006. [Cited on page 34.]

Bordron, P., Eveillard, D., and Rusu, I. Integrated analysis of the gene neighbouring impact on bacterial metabolic networks. *IET systems biology*, 5(4):261–268, 2011. [Cited on pages 52 and 53.]

Boyer, F., Morgat, A., Labarre, L., Pothier, J., and Viari, A. Syntons, metabolons and interactons: an exact graph-theoretical approach for exploring neighbourhood between genomic and functional data. *Bioinformatics*, 21(23):4209–4215, 2005. [Cited on pages 26, 50, 53, and 97.]

Boyle, D. S., Khattar, M. M., Addinall, S. G., Lutkenhaus, J., and Donachie, W. D. *ftsW* is an essential cell-division gene in *Escherichia coli*. *Molecular Microbiology*, 24(6):1263–1273, 1997. [Cited on page 142.]

Braakman, R. and Smith, E. The compositional and evolutionary logic of metabolism. *Physical Biology*, 10(1):011001, 2012. [Cited on page 17.]

Caccavo, F., Lonergan, D. J., Lovley, D. R., Davis, M., Stolz, J. F., and McInerney, M. J. *Geobacter sulfurreducens* sp. nov., a hydrogen- and acetate-oxidizing dissimilatory metal-reducing microorganism. *Applied and Environmental Microbiology*, 60 (10):3752–3759, 1994. [Cited on page 142.]

Cakmak, A. and Ozsoyoglu, G. Mining biological networks for unknown pathways. *Bioinformatics*, 23(20):2775–2783, 2007. [Cited on page 44.]

Carbonell, P. and Faulon, J.-L. Molecular signatures-based prediction of enzyme promiscuity. *Bioinformatics*, 26(16):2012–2019, 2010. [Cited on page 17.]

Carbonell, P., Lecointre, G., and Faulon, J.-L. Origins of specificity and promiscuity in metabolic networks. *Journal of Biological Chemistry*, 286(51):43994–44004, 2011a. [Cited on page 17.]

Carbonell, P., Planson, A.-G., Fichera, D., and Faulon, J.-L. A retrosynthetic biology approach to metabolic pathway design for therapeutic production. *BMC Systems Biology*, 5(1):122, 2011b. [Cited on pages 151, 152, and 153.]

Carbonell, P., Carlsson, L., and Faulon, J.-L. Stereo signature molecular descriptor. *Journal of Chemical Information and Modeling*, 53(4):887–897, 2013. [Cited on pages 152 and 153.]

Cech, T. R. The ribosome is a ribozyme. *Science*, 289(5481):878–879, 2000. [Cited on page 8.]

Cech, T. R. The RNA worlds in context. *Cold Spring Harbor Perspectives in Biology*, 4 (7):a006742, 2012. [Cited on page 18.]

Chen, L. and Vitkup, D. Predicting genes for orphan metabolic activities using phylogenetic profiles. *Genome Biology*, 7(2):R17, 2006. [Cited on page 26.]

Chen, L., Wang, R.-S., and Zhang, X.-S. Alignment of biomolecular networks. In *Biomolecular Networks: Methods and Applications in Systems Biology*, chapter 7, pages 205–229. Wiley Online Library, 2009. [Cited on page 41.]

Cheng, H., Yan, X., and Han, J. Mining graph patterns. In Aggarwal, C. C. and Wang, H., editors, *Managing and Mining Graph Data*, chapter 12, pages 365–392. Springer, 2010. [Cited on page 43.]

Cheng, W. and Yan, C. A graph approach to mining biological patterns in the binding interfaces. *Journal of Computational Biology*, 24(1):31–39, 2017. [Cited on page 44.]

Chesler, E. J. and Langston, M. A. Combinatorial genetic regulatory network analysis tools for high throughput transcriptomic data. In *Systems Biology and Regulatory Genomics*, pages 150–165. Springer, 2007. [Cited on page 33.]

Clark, C. and Kalita, J. A comparison of algorithms for the pairwise alignment of biological networks. *Bioinformatics*, 30(16):2351–2359, 2014. [Cited on page 41.]

Crick, F. Central dogma of molecular biology. *Nature*, 227(5258):561, 1970. [Cited on page 18.]

Dalquen, D. A. and Dessimoz, C. Bidirectional best hits miss many orthologs in duplication-rich clades such as plants and animals. *Genome Biology and Evolution*, 5(10):1800–1806, 2013. [Cited on page 59.]

Dasgupta, S., Papadimitriou, C. H., and Vazirani, U. V. *Algorithms*. McGraw-Hill Higher Education, 2006. [Cited on page 31.]

Deniélou, Y.-P., Boyer, F., Viari, A., and Sagot, M.-F. Multiple alignment of biological networks: A flexible approach. In *Annual Symposium on Combinatorial Pattern Matching*, pages 263–273. Springer, 2009. [Cited on pages 26, 51, and 53.]

Deniélou, Y.-P., Sagot, M.-F., Boyer, F., and Viari, A. Bacterial syntenies: an exact approach with gene quorum. *BMC Bioinformatics*, 12(1):193, 2011. [Cited on pages 26, 52, and 53.]

Díaz-Mejía, J. J., Pérez-Rueda, E., and Segovia, L. A network perspective on the evolution of metabolism by gene duplication. *Genome Biology*, 8(2):R26, 2007. [Cited on pages 14 and 16.]

Drillon, G., Carbone, A., and Fischer, G. SynChro: a fast and easy tool to reconstruct and visualize synteny blocks along eukaryotic chromosomes. *PLoS One*, 9(3): e92621, 2014. [Cited on page 26.]

Durek, P. and Walther, D. The integrated analysis of metabolic and protein interaction networks reveals novel molecular organizing principles. *BMC Systems Biology*, 2(1):100, 2008. [Cited on page 35.]

Enright, A. J. and Ouzounis, C. A. Functional associations of proteins in entire genomes by means of exhaustive detection of gene fusions. *Genome biology*, 2(9): research0034–1, 2001. [Cited on page 26.]

Erdős, P. and Rényi, A. On random graphs. *Publicationes Mathematicae*, 6:290–297, 1959. [Cited on page 36.]

Faisal, F. E., Meng, L., Crawford, J., and Milenković, T. The post-genomic era of biological network alignment. *EURASIP Journal on Bioinformatics and Systems Biology*, 2015(1):3, 2015. [Cited on pages 41 and 42.]

Fani, R. and Fondi, M. Origin and evolution of metabolic pathways. *Physics of Life Reviews*, 6(1):23–52, 2009. [Cited on pages 13 and 15.]

Faulon, J.-L., Visco, D. P., and Pophale, R. S. The signature molecular descriptor. 1. Using extended valence sequences in QSAR and QSPR studies. *Journal of Chemical Information and Computer Sciences*, 43(3):707–720, 2003. [Cited on pages 17 and 150.]

Faust, K., Croes, D., and van Helden, J. Metabolic pathfinding using RPAIR annotation. *Journal of Molecular Biology*, 388(2):390–414, 2009. [Cited on page 13.]

Faust, K., Croes, D., and van Helden, J. Prediction of metabolic pathways from genome-scale metabolic networks. *Biosystems*, 105(2):109–121, 2011. [Cited on page 11.]

Fertin, G., Mohamed-Babou, H., and Rusu, I. Algorithms for subnetwork mining in heterogeneous networks. In *International Symposium on Experimental Algorithms*, pages 184–194. Springer, 2012. [Cited on pages 52, 53, 54, 83, 87, 213, and 214.]

Fertin, G., Komusiewicz, C., Mohamed-Babou, H., and Rusu, I. Finding supported paths in heterogeneous networks. *Algorithms*, 8(4):810–831, 2015. [Cited on pages 53, 54, 80, and 82.]

Finn, R. D., Coggill, P., Eberhardt, R. Y., Eddy, S. R., Mistry, J., Mitchell, A. L., Potter, S. C., Punta, M., Qureshi, M., Sangrador-Vegas, A., *et al.* The Pfam protein families database: towards a more sustainable future. *Nucleic Acids Research*, 44 (D1):D279–D285, 2015. [Cited on page 27.]

Firestine, S. M., Poon, S.-W., Mueller, E. J., Stubbe, J., and Davisson, V. J. Reactions catalyzed by 5-aminoimidazole ribonucleotide carboxylases from *Escherichia coli* and *Gallus gallus*: a case for divergent catalytic mechanisms? *Biochemistry*, 33(39): 11927–11934, 1994. [Cited on page 133.]

Forst, C. V. and Schulten, K. Evolution of metabolisms: a new method for the comparison of metabolic pathways using genomics information. *Journal of Computational Biology*, 6(3-4):343–360, 1999. [Cited on page 11.]

Fuerst, J. A. and Sagulenko, E. Beyond the bacterium: planctomycetes challenge our concepts of microbial structure and function. *Nature Reviews Microbiology*, 9 (6):403, 2011. [Cited on page 146.]

Fujibuchi, W., Goto, S., Migimatsu, H., Uchiyama, I., Ogiwara, A., Akiyama, Y., and Kanehisa, M. DBGET/LinkDB: an integrated database retrieval system. In *Pacific Symposium on Biocomputing '98*, volume 98, pages 683–694, 1998. [Cited on page 57.]

Gabaldón, T. and Koonin, E. V. Functional and evolutionary implications of gene orthology. *Nature Reviews Genetics*, 14(5):360, 2013. [Cited on page 25.]

Galperin, M. Y. and Koonin, E. V. From complete genome sequence to "complete" understanding? *Trends in Biotechnology*, 28(8):398–406, 2010. [Cited on page 25.]

Gehrmann, T. and Reinders, M. J. Proteny: discovering and visualizing statistically significant syntenic clusters at the proteome level. *Bioinformatics*, 31(21):3437–3444, 2015. [Cited on page 26.]

Gerlee, P., Lundh, T., Zhang, B., and Anderson, A. Gene divergence and pathway duplication in the metabolic network of yeast and digital organisms. *Journal of The Royal Society Interface*, 6(41):1233–1245, 2009. [Cited on page 16.]

GO Consortium. Creating the Gene Ontology resource: design and implementation. *Genome Research*, 11(8):1425–1433, 2001. http://www.geneontology.org. [Cited on pages 27 and 44.]

Green, M. and Karp, P. Genome annotation errors in pathway databases due to semantic ambiguity in partial EC numbers. *Nucleic Acids Research*, 33(13):4035–4039, 2005. [Cited on page 66.]

Guzzi, P. H. and Milenković, T. Survey of local and global biological network alignment: the need to reconcile the two sides of the same coin. *Briefings in Bioinformatics*, 19(3):472–481, 2018. [Cited on pages 40, 41, and 42.]

Hashimoto, K., Goto, S., Kawano, S., Aoki-Kinoshita, K. F., Ueda, N., Hamajima, M., Kawasaki, T., and Kanehisa, M. KEGG as a glycome informatics resource. *Glycobiology*, 16(5):63R–70R, 2006. [Cited on page 60.]

Hattori, M., Okuno, Y., Goto, S., and Kanehisa, M. Development of a chemical structure comparison method for integrated analysis of chemical and genomic information in the metabolic pathways. *Journal of the American Chemical Society*, 125(39):11853–11865, 2003. [Cited on pages 59 and 60.]

Hattori, M., Tanaka, N., Kanehisa, M., and Goto, S. SIMCOMP/SUBCOMP: chemical structure search servers for network analyses. *Nucleic Acids Research*, 38 (suppl_2):W652–W656, 2010. [Cited on page 59.]

He, X. and Zhang, J. Why do hubs tend to be essential in protein networks? *PLoS Genetics*, 2(6):e88, 2006. [Cited on page 36.]

Hegyi, H. and Gerstein, M. The relationship between protein structure and function: a comprehensive survey with application to the yeast genome. *Journal of Molecular Biology*, 288(1):147–164, 1999. [Cited on page 27.]

Herlemann, D., Geissinger, O., Ikeda-Ohtsubo, W., Kunin, V., Sun, H., Lapidus, A., Hugenholtz, P., and Brune, A. Genomic analysis of *Elusimicrobium minutum*, the first cultivated representative of the phylum *Elusimicrobia* (formerly termite group 1). *Applied and Environmental Microbiology*, 75(9):2841–2849, 2009. [Cited on page 130.]

Horowitz, N. H. On the evolution of biochemical syntheses. *Proceedings of the National Academy of Sciences*, 31(6):153–157, 1945. [Cited on page 14.]

Hu, H., Yan, X., Huang, Y., Han, J., and Zhou, X. J. Mining coherent dense subgraphs across massive biological networks for functional discovery. *Bioinformatics*, 21(suppl_1):i213–i221, 2005. [Cited on page 44.]

Hurst, L. D., Pál, C., and Lercher, M. J. The evolutionary dynamics of eukaryotic gene order. *Nature Reviews Genetics*, 5(4):299, 2004. [Cited on page 124.]

Im, W.-T., Hu, Z.-Y., Kim, K.-H., Rhee, S.-K., Meng, H., Lee, S.-T., and Quan, Z.-X. Description of *Fimbriimonas ginsengisoli* gen. nov., sp. nov. within the *Fimbriimonadia* class nov., of the phylum *Armatimonadetes*. *Antonie Van Leeuwenhoek*, 102(2): 307–317, 2012. [Cited on page 144.]

Itoh, M., Nakaya, A., and Kanehisa, M. Identification of ortholog groups in KEGG/SSDB by considering domain structures. *Genome Informatics*, 13:342–343, 2002. [Cited on page 59.]

Iwasaki, W. and Takagi, T. Rapid pathway evolution facilitated by horizontal gene transfers across prokaryotic lineages. *PLoS Genetics*, 5(3):e1000402, 2009. [Cited on page 17.]

Jensen, R. A. Enzyme recruitment in evolution of new function. *Annual Reviews in Microbiology*, 30(1):409–425, 1976. [Cited on page 15.]

Jeong, H., Tombor, B., Albert, R., Oltvai, Z. N., and Barabási, A.-L. The large-scale organization of metabolic networks. *Nature*, 407(6804):651, 2000. [Cited on pages 38 and 39.]

Jeske, O., Schüler, M., Schumann, P., Schneider, A., Boedeker, C., Jogler, M., Bollschweiler, D., Rohde, M., Mayer, C., Engelhardt, H., Spring, S., and Jogler, C. Planctomycetes do possess a peptidoglycan cell wall. *Nature Communications*, 6:7116, 2015. [Cited on page 146.]

Jiang, C., Coenen, F., and Zito, M. A survey of frequent subgraph mining algorithms. *The Knowledge Engineering Review*, 28(1):75–105, 2013. [Cited on page 43.]

Jiang, Y., Oron, T. R., Clark, W. T., Bankapur, A. R., D'Andrea, D., Lepore, R., Funk, C. S., Kahanda, I., Verspoor, K. M., Ben-Hur, A., *et al.* An expanded evaluation of protein function prediction methods shows an improvement in accuracy. *Genome Biology*, 17(1):184, 2016. [Cited on page 25.]

Jothi, R., Przytycka, T. M., and Aravind, L. Discovering functional linkages and uncharacterized cellular pathways using phylogenetic profile comparisons: a comprehensive assessment. *BMC Bioinformatics*, 8(1):173, 2007. [Cited on page 26.]

Kanehisa, M. Toward pathway engineering: a new database of genetic and molecular pathways. *Science & Technology Japan*, 59:34–38, 1996. [Cited on page 56.]

Kanehisa, M. Chemical and genomic evolution of enzyme-catalyzed reaction networks. *FEBS Letters*, 587(17):2731–2737, 2013. [Cited on pages 1 and 10.]

Kanehisa, M. Enzyme annotation and metabolic reconstruction using KEGG. *Protein Function Prediction: Methods and Protocols*, pages 135–145, 2017. [Cited on pages 58 and 69.]

Kanehisa, M., Goto, S., Kawashima, S., and Nakaya, A. The KEGG databases at GenomeNet. *Nucleic Acids Research*, 30(1):42–46, 2002. [Cited on page 59.]

Kanehisa, M., Goto, S., Kawashima, S., Okuno, Y., and Hattori, M. The KEGG resource for deciphering the genome. *Nucleic Acids Research*, 32(suppl_1):D277–D280, 2004. [Cited on page 59.]

Kanehisa, M., Goto, S., Sato, Y., Furumichi, M., and Tanabe, M. KEGG for integration and interpretation of large-scale molecular data sets. *Nucleic Acids Research*, 40(D1):D109–D114, 2011. [Cited on page 58.]

Kanehisa, M., Goto, S., Sato, Y., Kawashima, M., Furumichi, M., and Tanabe, M. Data, information, knowledge and principle: back to metabolism in KEGG. *Nucleic Acids Research*, 42(D1):D199–D205, 2013. [Cited on pages 58 and 59.]

Kanehisa, M., Sato, Y., Kawashima, M., Furumichi, M., and Tanabe, M. KEGG as a reference resource for gene and protein annotation. *Nucleic Acids Research*, 44 (D1):D457–D462, 2015. [Cited on page 66.]

Kanehisa, M., Furumichi, M., Tanabe, M., Sato, Y., and Morishima, K. KEGG: new perspectives on genomes, pathways, diseases and drugs. *Nucleic Acids Research*, 45(D1):D353–D361, 2016a. [Cited on page 78.]

Kanehisa, M., Sato, Y., and Morishima, K. BlastKOALA and GhostKOALA: KEGG tools for functional characterization of genome and metagenome sequences. *Journal of Molecular Biology*, 428(4):726–731, 2016b. [Cited on page 58.]

Karp, P. D. Call for an enzyme genomics initiative. *Genome Biology*, 5(8):401, 2004. [Cited on page 10.]

KEGG FTP. An academic or a commercial license is required in order to access the FTP. ftp://ftp.bioinformatics.jp. Accessed 15 June 2018. [Cited on pages 64, 74, and 201.]

KEGG Organisms. List of KEGG organisms with complete genomes. http://www.kegg.jp/kegg/catalog/org_list.html. Accessed 12 June 2018. [Cited on pages 58, 63, and 118.]

KEGG REST API. Specification. https://www.kegg.jp/kegg/rest/keggapi.html. Accessed 12 June 2018. [Cited on pages 64, 74, 118, and 208.]

Keller, M. A., Piedrafita, G., and Ralser, M. The widespread role of non-enzymatic reactions in cellular metabolism. *Current Opinion in Biotechnology*, 34:153–161, 2015. [Cited on pages 9 and 11.]

Kern, D. and Zuiderweg, E. R. The role of dynamics in allosteric regulation. *Current Opinion in Structural Biology*, 13(6):748–757, 2003. [Cited on page 8.]

KGML. KEGG Markup Language specification. https://www.kegg.jp/kegg/xml/docs. Accessed 14 June 2018. [Cited on pages 61 and 118.]

Khersonsky, O. and Tawfik, D. S. Enzyme promiscuity: a mechanistic and evolutionary perspective. *Annual Reviews of Biochemistry*, 79:471–505, 2010. [Cited on page 17.]

Kliebenstein, D. J. A role for gene duplication and natural variation of gene expression in the evolution of metabolism. *PLoS One*, 3(3):e1838, 2008. [Cited on page 16.]

Koonin, E. V. Orthologs, paralogs, and evolutionary genomics. *Annual Review of Genetics*, 39:309–338, 2005. [Cited on page 23.]

Koonin, E. V. and Starokadomskyy, P. Are viruses alive? The replicator paradigm sheds decisive light on an old but misguided question. *Studies in history and philosophy of science part C: Studies in history and philosophy of biological and biomedical sciences*, 59:125–134, 2016. [Cited on page 19.]

Koschützki, D. Network centralities. In Junker, B. H. and Schreiber, F., editors, *Analysis of biological networks*, chapter 4, pages 65–84. Wiley Online Library, 2008. [Cited on page 34.]

Kotera, M., Hattori, M., Oh, M.-A., Yamamoto, R., Komeno, T., Yabuzaki, J., Tonomura, K., Goto, S., and Kanehisa, M. RPAIR: a reactant-pair database representing chemical changes in enzymatic reactions. *Genome Informatics*, 15:P062, 2004. [Cited on page 13.]

Kristensen, D. M., Wolf, Y. I., Mushegian, A. R., and Koonin, E. V. Computational methods for gene orthology inference. *Briefings in Bioinformatics*, 12(5):379–391, 2011. [Cited on page 25.]

Kuepfer, L., Sauer, U., and Blank, L. M. Metabolic functions of duplicate genes in *Saccharomyces cerevisiae*. *Genome Research*, 15(10):1421–1430, 2005. [Cited on page 16.]

Lacroix, V. *Identification de motifs dans les réseaux métaboliques*. PhD thesis, Université Claude Bernard-Lyon I, 2007. [Cited on page 33.]

Lacroix, V., Cottret, L., Thébault, P., and Sagot, M.-F. An introduction to metabolic networks and their structural analysis. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 5(4):594–617, 2008. [Cited on pages 6, 7, 11, and 12.]

Lazcano, A. and Miller, S. L. On the origin of metabolic pathways. *Journal of Molecular Evolution*, 49(4):424–431, 1999. [Cited on page 16.]

Lee, D., Redfern, O., and Orengo, C. Predicting protein function from sequence and structure. *Nature Reviews Molecular Cell Biology*, 8(12):995, 2007. [Cited on page 27.]

Lemoine, F., Labedan, B., and Lespinet, O. SynteBase/SynteView: a tool to visualize gene order conservation in prokaryotic genomes. *BMC Bioinformatics*, 9(1): 536, 2008. [Cited on page 26.]

Lespinet, O. and Labedan, B. Orphan enzymes? *Science*, 307(5706):42–42, 2005. [Cited on page 10.]

Li, W., Hu, H., Huang, Y., Li, H., Mehan, M. R., Nunez-Iglesias, J., Xu, M., Yan, X., and Zhou, X. J. Pattern mining across many massive biological networks. In Koyutürk, M., Subramaniam, S., and Grama, A., editors, *Functional Coherence of Molecular Networks in Bioinformatics*, chapter 6, pages 137–170. Springer, 2012. [Cited on pages 43 and 44.]

Lilley, D. M. The origins of RNA catalysis in ribozymes. *Trends in Biochemical Sciences*, 28(9):495–501, 2003. [Cited on pages 8 and 18.]

Lobb, B. and Doxey, A. C. Novel function discovery through sequence and structural data mining. *Current Opinion in Structural Biology*, 38:53–61, 2016. [Cited on pages 24 and 27.]

Mahadevan, R., Bond, D. R., Butler, J. E., Esteve-Núñez, A., Coppi, M. V., Palsson, B. O., Schilling, C. H., and Lovley, D. Characterization of metabolism in the Fe(III)-reducing organism *Geobacter sulfurreducens* by constraint-based modeling. *Applied and Environmental Microbiology*, 72(2):1558–1568, 2006. [Cited on pages 142 and 143.]

Marri, P. R., Bannantine, J. P., and Golding, G. B. Comparative genomics of metabolic pathways in *Mycobacterium* species: gene duplication, gene decay and lateral gene transfer. *FEMS Microbiology Reviews*, 30(6):906–925, 2006. [Cited on page 16.]

Mazière, P., Granier, C., and Molina, F. A description scheme of biological processes based on elementary bricks of action. *Journal of Molecular Biology*, 339(1):77–88, 2004. [Cited on page 27.]

Mazière, P., Parisey, N., Beurton-Aimar, M., and Molina, F. Formal TCA cycle description based on elementary actions. *Journal of Biosciences*, 32(1):145–155, 2007. [Cited on page 28.]

McDonald, A. G. and Tipton, K. F. Fifty-five years of enzyme classification: advances and difficulties. *The FEBS Journal*, 281(2):583–592, 2014. [Cited on page 10.]

Meléndez-Hevia, E., Montero-Gómez, N., and Montero, F. From prebiotic chemistry to cellular metabolism—The chemical evolution of metabolism before Darwinian natural selection. *Journal of Theoretical Biology*, 252(3):505–519, 2008. [Cited on page 18.]

Mercier, J., Josso, A., Médigue, C., and Vallenet, D. GROOLS: reactive graph reasoning for genome annotation through biological processes. *BMC Bioinformatics*, 19(1):132, 2018. [Cited on page 28.]

Milenković, T. and Pržulj, N. Topological characteristics of molecular networks. In Koyutürk, M., Subramaniam, S., and Grama, A., editors, *Functional Coherence of Molecular Networks in Bioinformatics*, chapter 2, pages 15–48. Springer, 2012. [Cited on pages 36 and 37.]

Mills, C. L., Beuning, P. J., and Ondrechen, M. J. Biochemical functional predictions for protein structures of unknown or uncertain function. *Computational and Structural Biotechnology Journal*, 13:182–191, 2015. [Cited on page 27.]

Minowa, Y., Katayama, T., Nakaya, A., Goto, S., and Kanehisa, M. Classification of protein sequences into paralog and ortholog clusters using sequence similarity profiles of KEGG/SSDB. *Genome Informatics*, 14:528–530, 2003. [Cited on page 59.]

Mitchell, A., Chang, H.-Y., Daugherty, L., Fraser, M., Hunter, S., Lopez, R., McAnulla, C., McMenamin, C., Nuka, G., Pesseat, S., *et al.* The InterPro protein families database: the classification resource after 15 years. *Nucleic Acids Research*, 43(D1): D213–D221, 2014. [Cited on page 27.]

Mohamed-Babou, H. *Comparaison de réseaux biologiques*. PhD thesis, Université de Nantes, 2012. [Cited on pages 33, 80, 89, and 214.]

Mohammadi, S. and Grama, A. Biological network alignment. In Koyutürk, M., Subramaniam, S., and Grama, A., editors, *Functional Coherence of Molecular Networks in Bioinformatics*, chapter 5, pages 15–48. Springer, 2012. [Cited on pages 41 and 42.]

Mohammadi, T., Van Dam, V., Sijbrandi, R., Vernet, T., Zapun, A., Bouhss, A., Diepeveen-de Bruin, M., Nguyen-Distèche, M., De Kruijff, B., and Breukink, E. Identification of FtsW as a transporter of lipid-linked cell wall precursors across the membrane. *The EMBO Journal*, 30(8):1425–1432, 2011. [Cited on page 142.]

Moreira, D. and López-García, P. Ten reasons to exclude viruses from the tree of life. *Nature Reviews Microbiology*, 7(4):306, 2009. [Cited on page 19.]

Moreno-Hagelsieb, G. The power of operon rearrangements for predicting functional associations. *Computational and Structural Biotechnology Journal*, 13:402–406, 2015. [Cited on pages 26 and 124.]

Moreno-Hagelsieb, G. and Hudy-Yuffa, B. Estimating overannotation across prokaryotic genomes using BLAST+, UBLAST, LAST and BLAT. *BMC Research Notes*, 7(1):651, 2014. [Cited on page 25.]

Moreno-Hagelsieb, G. and Santoyo, G. Predicting functional interactions among genes in prokaryotes by genomic context. In *Prokaryotic Systems Biology*, pages 97–106. Springer, 2015. [Cited on page 25.]

Morita, S., Oshio, K.-i., Osana, Y., Funabashi, Y., Oka, K., and Kawamura, K. Geometrical structure of the neuronal network of *Caenorhabditis elegans*. *Physica*

*A: Statistical Mechanics and its Applications*, 298(3-4):553–561, 2001. [Cited on page 39.]

Muto, A., Kotera, M., Tokimatsu, T., Nakagawa, Z., Goto, S., and Kanehisa, M. Modular architecture of metabolic pathways revealed by conserved sequences of reactions. *Journal of Chemical Information and Modeling*, 53(3):613–622, 2013. [Cited on pages 1, 13, 166, and 167.]

Nakaya, A., Goto, S., and Kanehisa, M. Extraction of correlated gene clusters by multiple graph comparison. *Genome Informatics*, 12:44–53, 2001. [Cited on pages 46 and 49.]

Nelson, D. L. and Cox, M. M. *Lehninger principles of biochemistry*. W. H. Freeman, fourth edition, 2005. [Cited on page 11.]

Newman, M. E., Strogatz, S. H., and Watts, D. J. Random graphs with arbitrary degree distributions and their applications. *Physical Review E*, 64(2):026118, 2001. [Cited on page 36.]

Nobeli, I., Favia, A. D., and Thornton, J. M. Protein promiscuity and its implications for biotechnology. *Nature Biotechnology*, 27(2):157, 2009. [Cited on pages 8 and 17.]

Ogata, H., Fujibuchi, W., Goto, S., and Kanehisa, M. A heuristic graph comparison algorithm and its application to detect functionally related enzyme clusters. *Nucleic Acids Research*, 28(20):4021–4028, 2000. [Cited on pages 45 and 49.]

Omelchenko, M. V., Galperin, M. Y., Wolf, Y. I., and Koonin, E. V. Non-homologous isofunctional enzymes: a systematic analysis of alternative solutions in enzyme evolution. *Biology Direct*, 5(1):31, 2010. [Cited on page 23.]

Overbeek, R., Fonstein, M., D'Souza, M., Pusch, G. D., and Maltsev, N. Use of contiguity on the chromosome to predict functional coupling. *In Silico Biology*, 1 (2):93–108, 1999. [Cited on page 26.]

Pál, C., Papp, B., and Lercher, M. J. Adaptive evolution of bacterial metabolic networks by horizontal gene transfer. *Nature Genetics*, 37(12):1372, 2005. [Cited on page 17.]

Panchen, A. The use of parsimony in testing phylogenetic hypotheses. *Zoological Journal of the Linnean Society*, 74(3):305–328, 1982. [Cited on page 139.]

Parks, D. H., Rinke, C., Chuvochina, M., Chaumeil, P.-A., Woodcroft, B. J., Evans, P. N., Hugenholtz, P., and Tyson, G. W. Recovery of nearly 8,000 metagenome-assembled genomes substantially expands the tree of life. *Nature Microbiology*, 2 (11):1533, 2017. [Cited on page 126.]

Parthasarathy, S., Tatikonda, S., and Ucar, D. A survey of graph mining techniques for biological datasets. In Aggarwal, C. C. and Wang, H., editors, *Managing and Mining Graph Data*, chapter 18, pages 547–580. Springer, 2010. [Cited on page 43.]

Pavlopoulos, G. A., Secrier, M., Moschopoulos, C. N., Soldatos, T. G., Kossida, S., Aerts, J., Schneider, R., and Bagos, P. G. Using graph theory to analyze biological networks. *BioData Mining*, 4(1):10, 2011. [Cited on page 34.]

Pearson, W. R. and Lipman, D. J. Improved tools for biological sequence comparison. *Proceedings of the National Academy of Sciences*, 85(8):2444–2448, 1988. [Cited on page 25.]

Pellegrini, M., Marcotte, E. M., Thompson, M. J., Eisenberg, D., and Yeates, T. O. Assigning protein functions by comparative genome analysis: protein phylogenetic profiles. *Proceedings of the National Academy of Sciences*, 96(8):4285–4288, 1999. [Cited on page 26.]

Pereira-Leal, J. B., Audit, B., Peregrin-Alvarez, J. M., and Ouzounis, C. A. An exponential core in the heart of the yeast protein interaction network. *Molecular Biology and Evolution*, 22(3):421–425, 2004. [Cited on page 39.]

Pržulj, N. Biological network comparison using graphlet degree distribution. *Bioinformatics*, 23(2):e177–e183, 2007. [Cited on page 39.]

Pržulj, N., Corneil, D. G., and Jurisica, I. Modeling interactome: scale-free or geometric? *Bioinformatics*, 20(18):3508–3515, 2004. [Cited on page 39.]

Pržulj, N., Kuchaiev, O., Stevanović, A., and Hayes, W. Geometric evolutionary dynamics of protein interaction networks. In *Biocomputing 2010*, pages 178–189. World Scientific, 2010. [Cited on page 39.]

Radivojac, P., Clark, W. T., Oron, T. R., Schnoes, A. M., Wittkop, T., Sokolov, A., Graim, K., Funk, C., Verspoor, K., Ben-Hur, A., *et al.* A large-scale evaluation of computational protein function prediction. *Nature Methods*, 10(3):221, 2013. [Cited on page 25.]

Rahman, S. A., Cuesta, S. M., Furnham, N., Holliday, G. L., and Thornton, J. M. EC-BLAST: a tool to automatically search and compare enzyme reactions. *Nature Methods*, 11(2):171, 2014. [Cited on page 160.]

Rance, B., Gibrat, J.-F., and Froidevaux, C. An adaptive combination of matchers: application to the mapping of biological ontologies for genome annotation. In *International Workshop on Data Integration in the Life Sciences*, pages 113–126. Springer, 2009. [Cited on page 27.]

Rast, P., Glöckner, I., Boedeker, C., Jeske, O., Wiegand, S., Reinhardt, R., Schumann, P., Rohde, M., Spring, S., Glöckner, F. O., *et al.* Three novel species with peptidoglycan cell walls form the new genus *Lacunisphaera* gen. nov. in the family Opitutaceae of the verrucomicrobial subdivision 4. *Frontiers in Microbiology*, 8: 202, 2017. [Cited on page 147.]

Ravasz, E., Somera, A. L., Mongru, D. A., Oltvai, Z. N., and Barabási, A.-L. Hierarchical organization of modularity in metabolic networks. *Science*, 297(5586): 1551–1555, 2002. [Cited on page 40.]

Ream, D. C., Bankapur, A. R., and Friedberg, I. An event-driven approach for studying gene block evolution in bacteria. *Bioinformatics*, 31(13):2075–2083, 2015. [Cited on page 26.]

Reinharz, V., Soulé, A., Westhof, E., Waldispühl, J., and Denise, A. Mining for recurrent long-range interactions in RNA structures reveals embedded hierarchies in network families. *Nucleic Acids Research*, 46(8):3841–3851, 2018. [Cited on page 44.]

Reynolds, D. J. and Penn, C. W. Characteristics of *Helicobacter pylori* growth in a defined medium and determination of its amino acid requirements. *Microbiology*, 140(10):2649–2656, 1994. [Cited on page 130.]

Rinke, C., Schwientek, P., Sczyrba, A., Ivanova, N. N., Anderson, I. J., Cheng, J.-F., Darling, A., Malfatti, S., Swan, B. K., Gies, E. A., *et al.* Insights into the phylogeny and coding potential of microbial dark matter. *Nature*, 499(7459):431, 2013. [Cited on page 126.]

Rison, S. C., Teichmann, S. A., and Thornton, J. M. Homology, pathway distance and chromosomal localization of the small molecule metabolism enzymes in *Escherichia coli*. *Journal of Molecular Biology*, 318(3):911–932, 2002. [Cited on page 1.]

Rogozin, I. B., Makarova, K. S., Murvai, J., Czabarka, E., Wolf, Y. I., Tatusov, R. L., Szekely, L. A., and Koonin, E. V. Connected gene neighborhoods in prokaryotic genomes. *Nucleic Acids Research*, 30(10):2212–2223, 2002. [Cited on pages 26 and 137.]

Rogozin, I. B., Managadze, D., Shabalina, S. A., and Koonin, E. V. Gene family level comparative analysis of gene expression in mammals validates the ortholog conjecture. *Genome Biology and Evolution*, 6(4):754–762, 2014. [Cited on page 25.]

Rubinov, M. and Sporns, O. Complex network measures of brain connectivity: uses and interpretations. *Neuroimage*, 52(3):1059–1069, 2010. [Cited on page 34.]

Ruepp, A., Zollner, A., Maier, D., Albermann, K., Hani, J., Mokrejs, M., Tetko, I., Güldener, U., Mannhaupt, G., Münsterkötter, M., *et al.* The FunCat, a functional annotation scheme for systematic classification of proteins from whole genomes. *Nucleic Acids Research*, 32(18):5539–5545, 2004. [Cited on page 27.]

Schnoes, A. M., Brown, S. D., Dodevski, I., and Babbitt, P. C. Annotation error in public databases: misannotation of molecular function in enzyme superfamilies. *PLoS computational biology*, 5(12):e1000605, 2009. [Cited on pages 25 and 66.]

Shu, J.-J. A new integrated symmetrical table for genetic codes. *Biosystems*, 151: 21–26, 2017. [Cited on page 23.]

Sikora, F. *Aspects algorithmiques de la comparaison d'éléments biologiques*. PhD thesis, Université Paris-Est, 2011. [Cited on page 33.]

Singh, S., Kim, Y., Wang, F., Bigelow, L., Endres, M., Kharel, M. K., Babnigg, G., Bingman, C. A., Joachimiak, A., Thorson, J. S., *et al.* Structural characterization of AtmS13, a putative sugar aminotransferase involved in indolocarbazole AT2433 aminopentose biosynthesis. *Proteins: Structure, Function, and Bioinformatics*, 83(8): 1547–1554, 2015. [Cited on page 72.]

Sinha, A. U. and Meller, J. Cinteny: flexible analysis and visualization of synteny and genome rearrangements in multiple organisms. *BMC Bioinformatics*, 8(1):82, 2007. [Cited on page 26.]

Sorokina, M., Stam, M., Médigue, C., Lespinet, O., and Vallenet, D. Profiling the orphan enzymes. *Biology Direct*, 9(1):10, 2014. [Cited on page 10.]

Sorokina, M., Médigue, C., and Vallenet, D. A new network representation of the metabolism to detect chemical transformation modules. *BMC Bioinformatics*, 16 (1):385, 2015. [Cited on pages 35, 153, 166, and 167.]

Spirin, V., Gelfand, M. S., Mironov, A. A., and Mirny, L. A. A metabolic network in the evolutionary context: multiscale structure and modularity. *Proceedings of the National Academy of Sciences*, 103(23):8774–8779, 2006. [Cited on pages 48, 49, 50, and 51.]

Steuer, R. and López, G. Z. Global network properties. In Junker, B. H. and Schreiber, F., editors, *Analysis of biological networks*, chapter 3, pages 31–63. Wiley Online Library, 2008. [Cited on pages 34 and 35.]

Sullivan, D. Google launches knowledge graph to provide answers, not just links. https://searchengineland.com/google-launches-knowledge-graph-121585, 2012. Accessed 18 June 2018. [Cited on page 30.]

Sun, J., Xu, J., Liu, Z., Liu, Q., Zhao, A., Shi, T., and Li, Y. Refined phylogenetic profiles method for predicting protein–protein interactions. *Bioinformatics*, 21(16): 3409–3415, 2005. [Cited on page 26.]

Szklarczyk, D., Franceschini, A., Wyder, S., Forslund, K., Heller, D., Huerta-Cepas, J., Simonovic, M., Roth, A., Santos, A., Tsafou, K. P., *et al.* STRING v10: protein–protein interaction networks, integrated over the tree of life. *Nucleic Acids Research*, 43(D1):D447–D452, 2014. [Cited on pages 25, 139, 145, 146, and 224.]

Taglieber, A., Höbenreich, H., Carballeira, J. D., Mondière, R. J., and Reetz, M. T. Alternate-site enzyme promiscuity. *Angewandte Chemie International Edition*, 46 (45):8597–8600, 2007. [Cited on page 17.]

Tanaka, R. Scale-rich metabolic networks. *Physical Review Letters*, 94(16):168101, 2005. [Cited on page 39.]

Teichmann, S. A., Rison, S. C., Thornton, J. M., Riley, M., Gough, J., and Chothia, C. The evolution and structural anatomy of the small molecule metabolic pathways in *Escherichia coli*. *Journal of Molecular Biology*, 311(4):693–708, 2001. [Cited on page 16.]

Tipton, K. and Boyce, S. History of the enzyme nomenclature system. *Bioinformatics*, 16(1):34–40, 2000. [Cited on page 8.]

Tonon, T., Eveillard, D., Prigent, S., Bourdon, J., Potin, P., Boyen, C., and Siegel, A. Toward systems biology in brown algae to explore acclimation and adaptation to the shore environment. *OMICS: A Journal of Integrative Biology*, 15(12):883–892, 2011. [Cited on page 44.]

Turner, R. D., Vollmer, W., and Foster, S. J. Different walls for rods and balls: the diversity of peptidoglycan. *Molecular Microbiology*, 91(5):862–874, 2014. [Cited on page 139.]

Vakirlis, N., Sarilar, V., Drillon, G., Fleiss, A., Agier, N., Meyniel, J.-P., Blanpain, L., Carbone, A., Devillers, H., Dubois, K., *et al.* Reconstruction of ancestral chromosome architecture and gene repertoire reveals principles of genome evolution in a model yeast genus. *Genome Research*, 26(7):918–932, 2016. [Cited on page 26.]

Vollmer, W., Blanot, D., and De Pedro, M. A. Peptidoglycan structure and architecture. *FEMS Microbiology Reviews*, 32(2):149–167, 2008. [Cited on page 139.]

Wagner, A. and Fell, D. A. The small world inside large metabolic networks. *Proceedings of the Royal Society of London B: Biological Sciences*, 268(1478):1803–1810, 2001. [Cited on page 38.]

Waites, K. B. and Talkington, D. F. *Mycoplasma pneumoniae* and its role as a human pathogen. *Clinical Microbiology Reviews*, 17(4):697–728, 2004. [Cited on page 142.]

Watson, J. D., Laskowski, R. A., and Thornton, J. M. Predicting protein function from sequence and structural data. *Current Opinion in Structural Biology*, 15(3): 275–284, 2005. [Cited on page 27.]

Watts, D. J. and Strogatz, S. H. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440, 1998. [Cited on pages 36 and 39.]

Webb, E. C. *Enzyme nomenclature 1992. Recommendations of the Nomenclature Committee of the International Union of Biochemistry and Molecular Biology on the nomenclature and classification of enzymes*. Academic Press, sixth edition, 1992. [Cited on page 8.]

Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *Journal of Chemical Information and Computer Sciences*, 28(1):31–36, 1988. [Cited on page 152.]

Wek, R. C. and Hatfield, G. W. Transcriptional activation at adjacent operators in the divergent-overlapping *ilvY* and *ilvC* promoters of *Escherichia coli*. *Journal of Molecular Biology*, 203(3):643–663, 1988. [Cited on page 124.]

Wells, J. N., Bergendahl, L. T., and Marsh, J. A. Operon gene order is optimized for ordered protein complex assembly. *Cell Reports*, 14(4):679–685, 2016. [Cited on page 2.]

West, D. B. *Introduction to graph theory*. Pearson, second edition, 2001. [Cited on page 30.]

White, J. G., Southgate, E., Thomson, J. N., and Brenner, S. The structure of the nervous system of the nematode *Caenorhabditis elegans*. *Philosophical Transactions of the Royal Society of London. Series B*, 314(1165):1–340, 1986. [Cited on page 39.]

Willey, J. M., Sherwood, L. M., and Woolverton, C. J. Bacteria: the low G+C Gram positives. In *Prescott, Harley, and Klein's Microbiology, Seventh Edition*, chapter 23, pages 571–588. McGraw-Hill Higher Education, 2008. [Cited on page 143.]

Wu, J., Kasif, S., and DeLisi, C. Identification of functional links between genes using phylogenetic profiles. *Bioinformatics*, 19(12):1524–1530, 2003. [Cited on page 26.]

Wu, Y., Zhu, X., Li, L., Fan, W., Jin, R., and Zhang, X. Mining dual networks: Models, algorithms and applications. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 10(4):40, 2016. [Cited on page 44.]

Xavier, D., Crespo, B., and Fuentes-Fernández, R. A rule-based expert system for inferring functional annotation. *Applied Soft Computing*, 35:373–385, 2015. [Cited on page 28.]

Yamada, T. and Bork, P. Evolution of biomolecular networks — lessons from metabolic and protein interactions. *Nature Reviews Molecular Cell Biology*, 10(11): nrm2787, 2009. [Cited on page 36.]

Yan, X. and Han, J. Discovery of frequent substructures. In Cook, D. J. and Holder, L. B., editors, *Mining Graph Data*, chapter 5, pages 99–115. John Wiley & Sons, 2006. [Cited on page 43.]

Yan, X., Mehan, M. R., Huang, Y., Waterman, M. S., Yu, P. S., and Zhou, X. J. A graph-based approach to systematically reconstruct human transcriptional regulatory modules. *Bioinformatics*, 23(13):i577–i586, 2007. [Cited on page 44.]

Yanai, I., Derti, A., and DeLisi, C. Genes linked by fusion events are generally of the same functional category: a systematic analysis of 30 microbial genomes. *Proceedings of the National Academy of Sciences*, 98(14):7940–7945, 2001. [Cited on page 26.]

Yarza, P., Yilmaz, P., Pruesse, E., Glöckner, F. O., Ludwig, W., Schleifer, K.-H., Whitman, W. B., Euzéby, J., Amann, R., and Rosselló-Móra, R. Uniting the classi-

fication of cultured and uncultured bacteria and archaea using 16S rRNA gene sequences. *Nature Reviews Microbiology*, 12(9):635, 2014. [Cited on page 126.]

Yoon, J. Phylogenetic studies on the bacterial phylum Verrucomicrobia. *Microbiology and Culture Collections*, 27:61–65, 2011. [Cited on page 147.]

Zaharia, A., Labedan, B., Froidevaux, C., and Denise, A. CoMetGeNe: mining conserved neighborhood patterns in metabolic and genomic contexts. *BMC Bioinformatics*, in press, 2018. [Cited on page 173.]

Zaslaver, A., Mayo, A., Ronen, M., and Alon, U. Optimal gene partition into operons correlates with gene functional order. *Physical Biology*, 3(3):183, 2006. [Cited on page 2.]

Zhang, J. Evolution by gene duplication: an update. *Trends in Ecology & Evolution*, 18(6):292–298, 2003. [Cited on page 16.]

Zhao, S., Kumar, R., Sakai, A., Vetting, M. W., Wood, B. M., Brown, S., Bonanno, J. B., Hillerich, B. S., Seidel, R. D., Babbitt, P. C., *et al.* Discovery of new enzymes and metabolic pathways by using structure and genome context. *Nature*, 502 (7473):698, 2013. [Cited on page 27.]

Zheng, Y., Szustakowski, J. D., Fortnow, L., Roberts, R. J., and Kasif, S. Computational identification of operons in microbial genomes. *Genome Research*, 12(8): 1221–1230, 2002. [Cited on pages 47 and 49.]

# Appendices

# A

# Appendices for Chapter III

## 1 Disconnected reactions in KEGG ORTHOLOGY maps

This appendix corresponds to section III.3.1 and lists all occurrences of disconnected reactions in KEGG ORTHOLOGY (KO) maps, retrieved via the KEGG FTP in June 2018. Each line displays the name of the KO map, followed by the R number(s) corresponding to a reaction that is not linked in the given map. KO maps are listed in ascending order of their identifiers.

```
ko00073.xml R09448 R09455 R09456
ko00073.xml R09451 R09452 R09454
---
ko00130.xml R04985 R04986
ko00130.xml R05000 R05615
ko00130.xml R05000 R05616 R07273
ko00130.xml R08768 R04987
ko00130.xml R08769 R04988
ko00130.xml R08771 R04711
ko00130.xml R08773 R04982
ko00130.xml R08773 R04989
ko00130.xml R08774 R04983
ko00130.xml R08774 R04990
ko00130.xml R08775 R04984
ko00130.xml R08775 R06146
ko00130.xml R08781 R02175 R07235
ko00130.xml R08781 R05614
---
ko00271.xml R00177
ko00271.xml R00178
ko00271.xml R00179
```

```
ko00271.xml R00192
ko00271.xml R00194
ko00271.xml R00648
ko00271.xml R00650
ko00271.xml R00653
ko00271.xml R00654
ko00271.xml R00946
ko00271.xml R00999
ko00271.xml R01001
ko00271.xml R01286
ko00271.xml R01287 R02026
ko00271.xml R01288
ko00271.xml R01290
ko00271.xml R01291
ko00271.xml R01401
ko00271.xml R01402
ko00271.xml R01776
ko00271.xml R01777
ko00271.xml R01920
ko00271.xml R02025
ko00271.xml R02026
ko00271.xml R02821
ko00271.xml R03217
ko00271.xml R03260
ko00271.xml R03659
ko00271.xml R03940
ko00271.xml R04143
ko00271.xml R04405
ko00271.xml R04420
ko00271.xml R04858
ko00271.xml R07214
ko00271.xml R07363
ko00271.xml R07364
ko00271.xml R07392
ko00271.xml R07393
ko00271.xml R07394
ko00271.xml R07396
---
ko00281.xml R08087 R08096
ko00281.xml R10125 R10126
---
ko00333.xml R11662
ko00333.xml R11673
---
ko00460.xml R11639 R11640
ko00460.xml R11641 R11642
```

```
---
ko00512.xml R05908
ko00512.xml R05912
ko00512.xml R05913
---
ko00513.xml R05970
ko00513.xml R05972
ko00513.xml R05973
ko00513.xml R05976
ko00513.xml R05987
ko00513.xml R06127
ko00513.xml R06128
ko00513.xml R06238
ko00513.xml R06258
ko00513.xml R06260
ko00513.xml R06261
ko00513.xml R06722
ko00513.xml R11316
---
ko00514.xml R03380
ko00514.xml R04491
ko00514.xml R07620
ko00514.xml R09290
ko00514.xml R09295
ko00514.xml R09296
ko00514.xml R09297 R09298
ko00514.xml R09299
ko00514.xml R09300
ko00514.xml R09301
ko00514.xml R09302
ko00514.xml R09303
ko00514.xml R09304
ko00514.xml R09315
ko00514.xml R09316
---
ko00522.xml R06467 R06465
ko00522.xml R06470 R06468
ko00522.xml R06473 R06472
ko00522.xml R06475 R06474
ko00522.xml R06477 R06476
ko00522.xml R06479 R06478
ko00522.xml R06481 R06480
ko00522.xml R06483 R06482
ko00522.xml R06488 R06484
ko00522.xml R06489 R06485
ko00522.xml R06490 R06486
```

```
ko00522.xml R06491 R06487
ko00522.xml R06496 R06492
ko00522.xml R06497 R06493
ko00522.xml R06498 R06494
ko00522.xml R06499 R06495
ko00522.xml R06503 R06504 R06505
---
ko00534.xml R05925
ko00534.xml R05926
ko00534.xml R05927
ko00534.xml R05928
ko00534.xml R10138
ko00534.xml R10139
---
ko00540.xml R01994
ko00540.xml R01996
ko00540.xml R01997
---
ko00550.xml R01150
---
ko00563.xml R05924
ko00563.xml R07129
---
ko00592.xml R07893 R07894
---
ko00601.xml R05978 R06029
ko00601.xml R06024 R06027
ko00601.xml R06025 R06075 R06095 R06230 R06221 R06224 R06227
ko00601.xml R06035 R06085 R06090
ko00601.xml R06037 R06232
ko00601.xml R06038 R06222 R06076
ko00601.xml R06085 R06086 R06089 R06090
ko00601.xml R06086 R06089
ko00601.xml R06156 R06170
ko00601.xml R06165 R06155 R06164 R06163 R06162
ko00601.xml R06169 R06168
---
ko00604.xml R05938 R05939 R05946 R05952
ko00604.xml R05956 R05941 R05948 R05953
ko00604.xml R05957 R05942 R05949 R05954
ko00604.xml R05958 R05943 R05950 R05955
ko00604.xml R05959 R05945 R05951
---
ko00627.xml R04489 R04278 R04279
---
ko00710.xml R00341
```

```
ko00710.xml R00343
---
ko00830.xml R02366 R02367
---
ko00860.xml R09063
ko00860.xml R09065
ko00860.xml R11510
ko00860.xml R11511
ko00860.xml R11512
ko00860.xml R11513
---
ko00906.xml R04218 R07270
ko00906.xml R09653
ko00906.xml R09692
---
ko00910.xml R00025
---
ko00920.xml R00295
ko00920.xml R11487
ko00920.xml R11546
---
ko00942.xml R06544 R07928
ko00942.xml R07880 R07930
ko00942.xml R07882 R07931
ko00942.xml R07883 R07944
ko00942.xml R07927 R07873
ko00942.xml R07929 R07926
---
ko00980.xml R07004 R07003
ko00980.xml R07013 R07014
ko00980.xml R07024 R07023
ko00980.xml R07025 R07026
ko00980.xml R07066 R07068
ko00980.xml R07071 R07072
ko00980.xml R09412 R09413
ko00980.xml R09414 R09415
ko00980.xml R09426 R09427
---
ko00982.xml R08286 R08287
ko00982.xml R08324 R08325 R08326
---
ko00983.xml R08256 R08257
---
ko01040.xml R02222 R03370
ko01040.xml R07758 R07762
ko01040.xml R07759 R07763
```

```
ko01040.xml R07760 R07764
ko01040.xml R07761 R07765
ko01040.xml R07934
ko01040.xml R07935
ko01040.xml R07936
ko01040.xml R07937
ko01040.xml R07950
ko01040.xml R07951
ko01040.xml R07952
ko01040.xml R07953
ko01040.xml R11043
---
ko01056.xml R06643 R06644 R06645
ko01056.xml R09258 R06635 R06637
ko01056.xml R09259
ko01056.xml R09263 R09264
ko01056.xml R09266 R09267
ko01056.xml R09268
ko01056.xml R09269
ko01056.xml R11516
---
ko01100.xml R00926 R01354 R00920 R00928
ko01100.xml R01701 R07599 R07600
ko01100.xml R01702 R07601 R07602
ko01100.xml R02736 R02035
ko01100.xml R03098 R04863 R04390 R03102 R03103
ko01100.xml R03968 R04001
ko01100.xml R04225 R07603 R07604
ko01100.xml R04440 R05071
ko01100.xml R05068 R05069
ko01100.xml R05269 R05267
ko01100.xml R05369 R05370
ko01100.xml R05386 R05387
ko01100.xml R05972
ko01100.xml R06268 R06269 R06270 R06265 R06266 R06267
ko01100.xml R06291 R06292 R06293
ko01100.xml R06294 R06295 R06297
ko01100.xml R06322 R06323 R06326
ko01100.xml R07558 R07559
ko01100.xml R07889 R07890
ko01100.xml R07893 R07894
ko01100.xml R07897 R07898
ko01100.xml R08549 R01700 R02570 R07618 R01197
ko01100.xml R09883
---
ko01110.xml R00014 R03270 R02569 R07618
```

```
ko01110.xml R07603 R07604 R03174 R07618
ko01110.xml R08549 R01700 R02570 R07618
ko01110.xml R09625
ko01110.xml R10671 R10672 R08660 R08661 R08662
ko01110.xml R11510
ko01110.xml R11511
ko01110.xml R11512
ko01110.xml R11513
---
ko01120.xml R00014 R03270 R02569 R07618 R01196 R10866 R00212
ko01120.xml R00295
ko01120.xml R00621 R03316 R02570 R07618 R01197
ko01120.xml R00787 R00789 R00790 R05712
ko01120.xml R02073 R04779 R04780 R09084
ko01120.xml R04198 R04199
ko01120.xml R09883
---
ko01130.xml R00014 R03270 R02569 R07618 R01196 R01197
ko01130.xml R00768
ko01130.xml R06696 R09314 R05705
ko01130.xml R06747
ko01130.xml R07603 R07604 R03174 R07618
ko01130.xml R08549 R01700 R02570 R07618
ko01130.xml R08851 R08853
ko01130.xml R08852 R08854
ko01130.xml R09313 R05705
ko01130.xml R10937
---
ko01200.xml R00475 R00009
ko01200.xml R00756 R04779 R09084
ko01200.xml R00762 R04780
ko01200.xml R01520 R01521 R07147
ko01200.xml R05339 R09780
---
ko01212.xml R02222 R03370
ko01212.xml R11043
---
ko01230.xml R03896 R03898
ko01230.xml R05069 R05068
ko01230.xml R05071 R04440
```

## 2 Inconsistent reactions between pathway maps

This appendix corresponds to section III.3.2. It lists the first occurrence for every reaction found to be inconsistent according to definitions III.2 and III.3 at least

once among two different organism-specific KEGG pathway maps belonging to a given species. For each occurrence, the three- or four-letter organism code is displayed, as well as the complete list of pathway maps where the reaction is present, respectively absent, along with all associated EC numbers and K numbers. If no EC number is indicated, it is either unknown or a partial EC number that is only present in the pathway map drawing, but not through the KGML files, nor through the KEGG REST API. Note that global and overview maps (i.e., maps whose identifiers are greater than or equal to `01100`) are excluded from this analysis.

## 2.1 Same EC numbers, disjoint K numbers

The first occurrence among the 17 inconsistent reactions according to definition III.2 are listed below.

```
cag: R00237 present in [00720 (4.1.3.25) ({K08691})]
             absent in [00660 (4.1.3.25) ({K18292})]
aay: R00829 present in [00362 (2.3.1.174) ({K07823})]
             absent in [00360 (2.3.1.174) ({K02615})]
aaa: R00982 present in [00627 (6.2.1.32) ({K08295})]
             absent in [00405 (6.2.1.32) ({K18000})]
acj: R01975 present in [00720 (1.1.1.35) ({K15016})]
             absent in [00071 (1.1.1.35) ({K00022, K07516, K10527, K07514, K01825,
                                           K01782}),
                        00380 (1.1.1.35) ({K01825, K01782, K00022, K07514}),
                        00650 (1.1.1.35) ({K00022, K07516, K01825, K01782, K07514})]
aag: R02164 present in [00020 (1.3.5.1, 1.3.5.4) ({K00234, K00235, K00236, K00237,
                                                   K00239, K00240, K00241, K00242,
                                                   K18859, K18860})]
             absent in [00620 (1.3.5.1, 1.3.5.4) ({K00244, K00245, K00246, K00247}),
                        00650 (1.3.5.1, 1.3.5.4) ({K00239, K00240, K00241, K00242,
                                                   K18859, K18860};
                                                  {K00244, K00245, K00246, K00247,
                                                   K00239, K00240, K00241, K00242,
                                                   K18859, K18860})]
ase: R02773 present in [00525 (2.6.1.33) ({K20428})]
             absent in [00523 (2.6.1.33) ({K13308, K21328})]
acj: R03026 present in [00720 (4.2.1.17) ({K15016})]
             absent in [00071 (4.2.1.17) ({K01692, K10527, K01825, K01782, K07511,
                                           K13767, K07514, K07515}),
                        00362 (4.2.1.17) ({K01692, K01782, K01825, K13767}),
                        00380 (4.2.1.17) ({K01692, K01825, K01782, K07511, K07514,
                                           K07515}),
                        00627 (4.2.1.17) ({K01692, K07515, K07514, K07511}),
                        00650 (4.2.1.17) ({K01692, K01825, K01782, K07515, K07514,
```

```
                                          K07511, K01715})]
aai: R05850 present in [00040 (5.1.3.4) ({K01786, K03080})]
             absent in [00053 (5.1.3.4) ({K03077})]
hch: R06746 present in [00333 () ({K21780, K21781})]
             absent in [00401 () ({K12720}; {K12719})]
cfar: R07125 present in [00053 (4.1.1.85) ({K03078})]
              absent in [00040 (4.1.1.85) ({K03081})]
cmo: R07676 present in [00053 (1.1.1.365) ({K19642})]
             absent in [00040 (1.1.1.365) ({K19634})]
actn: R08956 present in [00405 (2.6.1.86) ({K13063})]
              absent in [01059 (2.6.1.86) ({K20159, K21175})]
acid: R09097 present in [00622 (1.2.1.87) ({K18366})]
              absent in [00640 (1.2.1.87) ({K13922})]
aho: R09280 present in [00720 (1.2.1.76) ({K15038, K15017})]
             absent in [00650 (1.2.1.76) ({K18119})]
aho: R09281 present in [00720 () ({K14465})]
             absent in [00650 () ({K18121})]
csy: R09289 present in [00720 () ({K18602})]
             absent in [00240 () ({K16066}; {K09019})]
aac: R10507 present in [00330 () ({K00318})]
             absent in [00332 () ({K18318, K18319})]
```

## 2.2 Disjoint EC numbers and K numbers

The first occurrence among the 41 inconsistent reactions according to definition III.3 are listed below.

```
cuv: R00014 present in [00010 (4.1.1.1) ({K01568})]
             absent in [00020 (1.2.4.1) ({K00163, K00161, K00162}),
                        00620 (1.2.4.1) ({K00163, K00161, K00162})]
acy: R00214 present in [00620 (1.1.1.38) ({K00027})]
             absent in [00710 (1.1.1.39) ({K00028})]
aja: R00230 present in [00620 () ({K04020})]
             absent in [00430 (2.3.1.8) ({K13788, K00625, K15024}),
                        00680 (2.3.1.8) ({K00625, K13788})]
bpg: R00289 present in [00520 (2.7.7.64) ({K12447})]
             absent in [00040 (2.7.7.9) ({K00963}),
                        00052 (2.7.7.9) ({K00963}),
                        00500 (2.7.7.9) ({K00963})]
app: R00485 present in [00250 (3.5.1.38) ({K05597})]
             absent in [00460 (3.5.1.1) ({K01424})]
aaci: R00489 present in [00410 (4.1.1.15) ({K01580})]
              absent in [00770 (4.1.1.11) ({K01579, K18933, K18966})]
abe: R00702 present in [00100 (2.5.1.21) ({K00801}),
                        00909 (2.5.1.21) ({K00801})]
```

```
             absent in [00906 (2.5.1.96) ({K10208})]
aam: R00711 present in [00010 (1.2.1.5) ({K00129})]
             absent in [00620 () ({K00138})]
aac: R00734 present in [00350 (2.6.1.1, 2.6.1.9) ({K14454, K14455, K00811, K00812,
                                                   K00813, K11358, K15849}; {K00817}),
                        00400 (2.6.1.1, 2.6.1.9) ({K14454, K14455, K00811, K00812,
                                                   K00813, K11358, K15849}; {K00817}),
                        00401 (2.6.1.1, 2.6.1.9) ({K00817}; {K00812, K00813, K11358})]
             absent in [00130 (2.6.1.5, 2.6.1.57) ({K00815, K00838})]
aba: R00736 present in [00350 (4.1.1.28) ({K01593})]
             absent in [00680 (4.1.1.25) ({K18933})]
bdi: R00737 present in [00940 (4.3.1.25) ({K13064})]
             absent in [00350 (4.3.1.23) ({K10774})]
aaf: R00801 present in [00500 (3.2.1.26) ({K01193})]
             absent in [00052 (3.2.1.10, 3.2.1.20, 3.2.1.48) ({K12047, K01187, K12316,
                                                               K12317};
                                                              {K01182, K01203})]
aaa: R00829 present in [00362 (2.3.1.16) ({K00632})]
             absent in [00360 (2.3.1.174) ({K02615})]
acm: R00908 present in [00410 (2.6.1.55) ({K15372})]
             absent in [00640 (2.6.1.19) ({K13524, K07250, K00823})]
aho: R00919 present in [00720 (1.3.1.84) ({K14469, K15020})]
             absent in [00640 () ({K19745})]
aac: R01175 present in [00071 (1.3.8.7) ({K00249})]
             absent in [00650 (1.3.8.1) ({K00248})]
bbo: R01177 present in [00071 (2.3.1.9) ({K00626})]
             absent in [00062 (2.3.1.16) ({K07508, K07509})]
aak: R01196 present in [00010 (1.2.7.11) ({K00174, K00175}),
                        00020 (1.2.7.11) ({K00174, K00175}),
                        00620 (1.2.7.11) ({K00174, K00175}),
                        00650 (1.2.7.11) ({K00174, K00175})]
             absent in [00680 (1.2.7.1) ({K00169, K00170, K00171, K00172})]
aad: R01274 present in [01040 () ({K10804})]
             absent in [00062 (3.1.2.22) ({K01074}; {K01074})]
aaci: R01388 present in [00630 (1.1.1.26) ({K00015})]
              absent in [00260 (1.1.1.29) ({K00018, K15893, K15919}),
                         00680 (1.1.1.29) ({K00018})]
aac: R01434 present in [00290 (1.4.1.9) ({K00263})]
             absent in [00280 (1.4.1.23) ({K00271})]
xhr: R01786 present in [00010 (2.7.1.2) ({K12407, K00845}),
                        00052 (2.7.1.2) ({K00845, K12407})]
             absent in [00520 (2.7.1.1) ({K00844}; {K00844})]
aaj: R01914 present in [00410 (1.5.99.6) ({K00316})]
             absent in [00330 (1.5.3.14) ({K13366})]
bvg: R02078 present in [00350 (1.10.3.1) ({K00422})]
             absent in [00965 (1.14.18.1) ({K00505})]
```

```
bvg: R02080 present in [00350 (4.1.1.25) ({K01592}),
                        00950 (4.1.1.25) ({K01592})]
             absent in [00965 (4.1.1.28) ({K01593})]
acx: R02549 present in [00330 () ({K12254})]
             absent in [00410 (1.2.1.19) ({K00137})]
acis: R02739 present in [00010 (5.1.3.15) ({K01792})]
              absent in [00030 (5.3.1.9) ({K01810, K06859, K13810, K15916})]
aaa: R02869 present in [00330 (2.5.1.16) ({K00797}),
                        00410 (2.5.1.16) ({K00797})]
             absent in [00270 (2.5.1.22) ({K00802}),
                        00480 (2.5.1.22) ({K00802})]
afg: R03045 present in [00410 (4.2.1.17) ({K01692, K01825, K01782, K07515, K07514,
                                           K07511}),
                        00640 (4.2.1.17) ({K01692, K01825, K01782, K07515, K07514,
                                           K07511})]
             absent in [00720 (4.2.1.116) ({K14469, K15019})]
adu: R03264 present in [00909 () ({K15891})]
             absent in [00900 (1.1.1.216) ({K15890})]
sco: R04014 present in [00333 () ({K21791})]
             absent in [00061 (3.1.2.14, 3.1.2.21) ({K01071, K10781})]
aad: R04364 present in [00300 (2.3.1.89) ({K05822})]
             absent in [00261 () ({K19107})]
aamy: R05578 present in [00970 (6.1.1.24) ({K09698})]
              absent in [00860 (6.1.1.17) ({K01885, K14163})]
aja: R05705 present in [00740 (1.5.1.36) ({K00484})]
             absent in [01057 () ({K14631}; {K14631})]
hch: R06746 present in [00333 () ({K21780, K21781})]
             absent in [00401 () ({K12720}; {K12719})]
hch: R06747 present in [00333 (1.3.8.14) ({K21782})]
             absent in [00401 () ({K12721})]
aaa: R06941 present in [00360 (1.1.1.157) ({K00074})]
             absent in [00930 (1.1.1.35) ({K00022, K07514, K01825, K01782})]
aaa: R07136 present in [00270 (1.1.1.37) ({K00024, K00025, K00026})]
             absent in [00680 (1.1.1.337) ({K05884})]
aho: R09281 present in [00720 () ({K14465})]
             absent in [00650 () ({K18121})]
aho: R09289 present in [00720 (1.1.1.298) ({K14468, K15039})]
             absent in [00240 () ({K16066}; {K09019})]
aac: R10507 present in [00330 () ({K00318})]
             absent in [00332 () ({K18318, K18319})]
```

# Appendices for Chapter IV

This appendix proves the claim stated in section IV.4.1, namely that it is possible to reduce the input graphs $D$ and $G$ for the MaSST and MaSSCoT problems without loss of solutions. More precisely, MaSST and MaSSCoT yield the same solution when provided with the unreduced graphs $D$ and $G$, as well as when provided with the graphs $D$ and $G$ reduced to their cover set with respect to a path $P$ in $D$.

## 1 Notations

The following notations are used for a directed graph $D$:

- $D^*$ is the underlying undirected graph of $D$ obtained by removing arc orientations.
- If $G$ is an undirected graph such that $V(D) = V(G)$ and $P$ is a path in $D$, the notation $CCC(D^*, G, P)$ designates the common connected component of $D^*$ and $G$ containing all vertices in $P$, if such a component exists. If it does not, then $CCC(D^*, G, P) = \varnothing$.
- For a vertex $v$ of $D$, $S_v^+$ designates the descendants of $v$, i.e. the set of vertices in $D$ that are reachable by a path from vertex $v$.
- For a vertex $v$ of $D$, $S_v^-$ designates the ancestors of $v$, i.e. the set of vertices in $D$ reaching vertex $v$ by a path in $D$.

## 2 Cover set definition

The definition of cover set of a path uses the concept of bridge, defined as follows by Fertin *et al.* [2012]:

**Definition B.1.** Let $D = (V, A)$ be a directed graph, $G = (V, E)$ an undirected graph, and $P$ a path in $D$. A vertex $r \in V$ is said to be a *bridge of $P$ with respect to $G$* if there is no common connected component of $D^*[V - \{r\}]$ and $G[V - \{r\}]$ containing all the vertices of $P$ (i.e., $CCC(D^*[V - \{r\}], G[V - \{r\}], P) = \varnothing$).

*Example.* In Figure B.1, vertex 4 is a bridge for path $P = (1, 2, 3)$ with respect to $G$. If this vertex is removed, the vertices of $P$ are found in two distinct connected components of $G$.

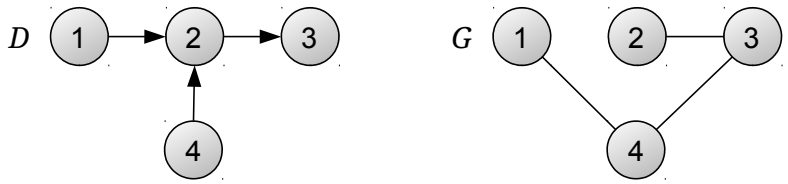



**Figure B.1** The directed graph $D$ and the undirected graph $G$ have the same vertex set.

Fertin *et al.* [2012] define the cover set of a path as follows:

**Definition B.2.** Let $D = (V, A)$ be a directed graph, $G = (V, E)$ an undirected graph, and $P = (v_1, v_2, \ldots, v_k)$ a path in $D$. The *cover set of path $P$ in $D$ with respect to $G$* is the subset $X \subseteq V$ satisfying:

1. $V(P) \subseteq X \subseteq S^-_{v_1} \cup S^+_{v_k} \cup V(P)$.
2. $D^*[X]$ and $G[X]$ are connected.
3. If $r$ is a bridge of $P$ in $D[X]$ with respect to $G[X]$ then $X \subseteq S^-_r \cup S^+_r \cup \{r\}$.
4. $X$ is maximal (with respect to the inclusion order).

*Example.* Figure B.2 shows the cover set of the path $P = (3, 4, 5)$ in $D$ with respect to $G$. Vertices 2, 3, 4, 5, 6, and 8 are bridges of $P$ with respect to $G$ (see Definition B.1).

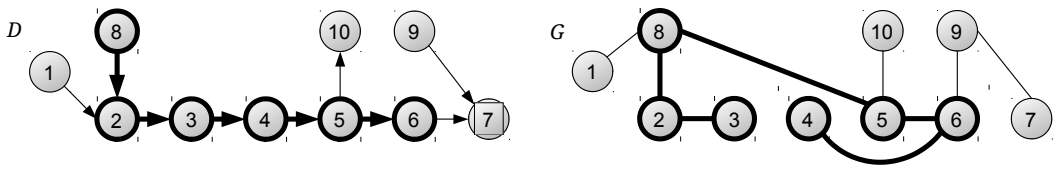



**Figure B.2** Cover set of the path $P = (3, 4, 5)$ in $D$ with respect to $G$. The cover set of $P$ is shown in solid black.

Mohamed-Babou [2012] shows that, if it exists, the cover set of a path is unique.

## 3 Graph reduction

Both MaSST and MaSSCoT take as input a directed graph $D = (V, A)$, an undirected graph $G = (V, E)$, and an arc $(u, v)$ in $D$. Let $S$ be the cover path of arc $(u, v)$ in $D$ with respect to $G$. We prove that $D$ and $G$ can respectively be replaced with $D[S]$ and $G[S]$, yielding the same solutions.

Following is a lemma for which the proof is omitted (as it is straightforward).

**Lemma B.1.** Let $G = (V, E)$ be an undirected graph and let $A$, $B$ and $C$ be three subsets of $V$. If $G[A \cup B]$ and $G[B \cup C]$ are connected, then $G[A \cup B \cup C]$ is connected.

The following definition introduces a shorthand notation for the remainder of this appendix.

**Definition B.3.** Let $D = (V, A)$ be a directed graph, $G = (V, E)$ be an undirected graph and $P$ be a path in $D$. If a trail $T$ in $D$ exists such that (i) $T \supseteq P$, (ii) $G[V(T)]$ is connected and (iii) $\nexists T'$ trail in $D$ such that $T' \supseteq P$, $G[V(T')]$ is connected and $span(T') > span(T)$, then $T$ is said to verify the property of being a *trail of maximum span in $D$ for $P$ with respect to $G$*, which is denoted as $\mathscr{S}_P(D, G)$.

**Lemma B.2.** Let $D = (V, A)$ be a directed graph, $G = (V, E)$ an undirected graph, $P$ a path in $D$, and $S$ the cover set of $P$ in graphs $D$ and $G$. If a trail $T$ in $D$ exists such that $T$ verifies $\mathscr{S}_P(D,\ G)$, then $V(T) \subseteq S$.

*Proof.* $T \supseteq P$ by hypothesis (i) of definition B.3 and $S \supseteq V(P)$ by definition of the cover set. It follows that $V(T) \cap S \neq \varnothing$. Let $I = V(T) \cap S$ be the set of vertices shared by $T$ and $S$. Let $\{t_1, \ldots, t_k\} = V(T) - I$ be the set of vertices of $T$ not shared with $S$.

$G[V(T)]$ is connected by hypothesis (ii) of definition B.3, therefore $G[\{t_1, \ldots, t_k\} \cup I]$ is connected. Since $G[S]$ is connected (by definition of the cover set), it means that $G[\{t_1, \ldots, t_k\} \cup S] = G[\{t_1, \ldots, t_k\} \cup I \cup (S - I)]$ is also connected (by lemma B.1). Also, $D^*[V(T)]$ is connected because $T$ is a trail in $D$, which means that $D^*[\{t_1, \ldots, t_k\} \cup I]$ is connected. Since $D^*[S]$ is connected (by definition of the cover set), it means that $D^*[\{t_1, \ldots, t_k\} \cup S] = D^*[\{t_1, \ldots, t_k\} \cup I \cup (S - I)]$ is also connected (by lemma B.1).

But if $G[\{t_1, \ldots, t_k\} \cup S]$ and $D^*[\{t_1, \ldots, t_k\} \cup S]$ are connected, then property 4 of definition B.2 concerning the maximality of $S$ is contradicted. Therefore $S$ cannot be maximal unless $\{t_1, \ldots, t_k\} = \varnothing$. Hence $V(T) \subseteq S$. □

**Proposition B.1.** Let $D = (V, A)$ be a directed graph, $G = (V, E)$ an undirected graph, $P$ a path in $D$, and $S$ the cover set of $P$ in graphs $D$ and $G$. If a trail $T$ in $D$ exists such that $T$ verifies $\mathscr{S}_P(D,\ G)$, then $T$ is also a trail in $D[S]$ verifying $\mathscr{S}_P(D[S],\ G[S])$, that is, (i) $T \supseteq P$, (ii) $G[S \cap V(T)]$ is connected and (iii) $T$ has maximum span in $D[S]$.

*Proof.* Let $T$ be a trail in $D$ such that $T$ verifies $\mathscr{S}_P(D,\ G)$. By lemma B.2, $V(T) \subseteq S$, therefore $T$ is also a trail in $D[S]$. We now prove that $T$ verifies properties (i)-(iii) for graphs $D[S]$ and $G[S]$.

(i) By hypothesis (i) of definition B.3, $T \supseteq P$.

(ii) Since $V(T) \subseteq S$, $G[S \cap V(T)] = G[V(T)]$. Since $G[V(T)]$ is connected by hypothesis (ii) of definition B.3, it follows that $G[S \cap V(T)]$ is connected.

(iii) Suppose there exists a trail $T'$ in $D[S]$ such that $T' \supseteq P$, $G[S \cap V(T')]$ is connected, and $span(T') > span(T)$. Because $T'$ is a trail in $D[S]$, $V(T') \subseteq S$, which implies that $S \cap V(T') = V(T')$. Since $G[S \cap V(T')]$ is connected, it follows that $G[V(T')]$ is also connected. Moreover, from definition B.2 it follows that $S \subseteq V$. Therefore, $T'$ is also a trail in $D$. Hypotheses (i)-(ii) of definition B.3 are thus fulfilled for $T'$ in $D$, with $T'$ having greater span than $T$. However, this contradicts hypothesis (iii) for $\mathscr{S}_P(D, G)$, this property being satisfied by $T$. Hence, no trail $T'$ can exist in $D[S]$ that includes $P$ such that $G[S \cap V(T')]$ is connected and such that $T'$ has greater span than $T$. $T$ has therefore maximum span in $D[S]$ with respect to properties (i) and (ii).

We have thus proven that, if a trail $T$ in $D$ exists such that $T$ verifies $\mathscr{S}_P(D,\ G)$, then $T$ is also a trail in $D[S]$ that verifies $\mathscr{S}_P(D[S],\ G[S])$. The converse is also true (see below). □

**Proposition B.2.** Let $D = (V, A)$ be a directed graph, $G = (V, E)$ an undirected graph, $P$ a path in $D$, and $S$ the cover set of $P$ in graphs $D$ and $G$. If a trail $T$ in $D[S]$ exists that verifies $\mathscr{S}_P(D[S],\ G[S])$, then $T$ is also a trail in $D$ verifying $\mathscr{S}_P(D,\ G)$, that is, (i) $T \supseteq P$, (ii) $G[V(T)]$ is connected and (iii) $T$ has maximum span in $D$.

*Proof.* This proposition is the converse of proposition B.1. Let $T$ be a trail in $D[S]$ such that $T$ verifies $\mathscr{S}_P(D[S],\ G[S])$. By definition B.2, $S \subseteq V$, therefore $T$ is also a trail in $D$. We now be prove that $T$ verifies properties (i)-(iii) for graphs $D$ and $G$.

(i) By hypothesis (i) of definition B.3, $T \supseteq P$.

(ii) Since $T$ is a trail in $D[S]$, $V(T) \subseteq S$. Then $S \cap V(T) = V(T)$, and since $G[S \cap V(T)]$ is connected by hypothesis (ii) of definition B.3, it follows that $G[V(T)]$ is connected.

(iii) Suppose there exists a trail $T'$ in $D$ such that $T' \supseteq P$, $G[V(T')]$ is connected, and $span(T') > span(T)$. By lemma B.2, $V(T') \subseteq S$ and therefore $S \cap V(T') = V(T')$. Since $G[V(T')]$ is connected, it follows that $G[S \cap V(T')]$ is also connected. Moreover, $T'$ is also a trail in $D[S]$ (because $V(T') \subseteq S$). Hypotheses (i)-(ii) of definition B.3 are thus fulfilled for $T'$ in $D[S]$, with $T'$ having greater span than $T$. However, this contradicts hypothesis (iii) for $\mathscr{S}_P(D[S], G[S])$, this property being satisfied by $T$. Hence, no trail $T'$ can exist in $D$ that includes $P$ such that $G[V(T')]$ is connected and such that $T'$ has greater span than $T$. $T$ has therefore maximum span in $D$ with respect to properties (i) and (ii).

We have thus proven that, if a trail $T$ in $D[S]$ exists such that $T$ verifies $\mathscr{S}_P(D[S], G[S])$, then $T$ is also a trail in $D$ that verifies $\mathscr{S}_P(D,\ G)$. □

From propositions B.1 and B.2, it follows that, in the context of the MaSST problem formulation (see section IV.3), the same solution is obtained for the input arc $(u, v)$ whether MaSST is ran on the input graphs $D$ and $G$, or on the input graphs $D$ and $G$ reduced to their cover set with respect to the arc $(u, v)$ (i.e. on $D[S]$ and $G[S]$, where $S$ is the cover set of $(u, v)$ in graphs $D$ and $G$).

C

# Appendices for Chapter VII

| *B. subtilis* gene | lmo | sau | lac | snd | cpe | mpn | syn | pma | cau | bbv | cgl | mtv | sco | dra | tth | fgi | amo | tmm | cex | dth | fsu | gau | cph | bfr | rba | cpn | ote | bbn | emi | heo | din | fnu | dap | tid | aae | nde | aca | eco | ype | vco | spc | pae | xfa | rso | nme | afi | ara | rrj | gsu |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *BSU28250* | x | x | . | . | . | . | . | x | x | . | . | x | . | . | . | . | . | x | . | x | . | . | x | x | . | . | . | . | x | . | . | . | . | x | . | . | x | x | x | x | x | . | . | . | . | . | . | . | x |
| *BSU28260* | x | x | . | . | . | . | . | x | x | . | . | x | . | . | . | . | . | x | . | x | . | . | x | x | . | . | . | . | x | . | . | . | . | x | . | . | x | x | x | x | x | . | . | . | . | . | . | . | x |
| *BSU28270* | x | x | . | . | . | . | . | . | . | . | . | x | . | . | . | . | x | x | . | x | . | . | x | x | . | . | . | . | . | . | . | . | . | . | . | . | x | x | x | x | x | . | . | . | . | . | . | . | . |
| *BSU28280* | x | x | . | . | . | . | . | . | x | . | . | . | . | . | x | . | x | x | . | x | . | . | x | x | . | . | . | . | . | . | . | . | . | . | . | . | x | x | x | x | x | . | x | x | . | . | . | . | x |
| *BSU28290* | x | x | . | x | . | . | . | . | x | . | x | x | x | x | x | x | x | x | . | x | . | . | x | . | x | . | x | . | . | . | . | . | x | x | . | x | x | . | . | . | . | x | x | x | x | x | . | . | x |
| *BSU28300* | x | x | . | x | . | . | . | . | x | x | x | x | x | x | x | x | x | x | . | x | x | . | x | . | x | . | x | . | . | . | x | . | x | . | . | x | x | . | . | . | . | x | x | x | x | x | x | . | x |
| *BSU28310* | x | x | . | x | . | . | . | . | x | x | x | x | x | x | x | x | x | x | . | x | x | . | x | . | x | . | x | . | . | . | x | . | x | . | . | x | x | . | . | . | . | x | x | x | x | x | x | . | x |

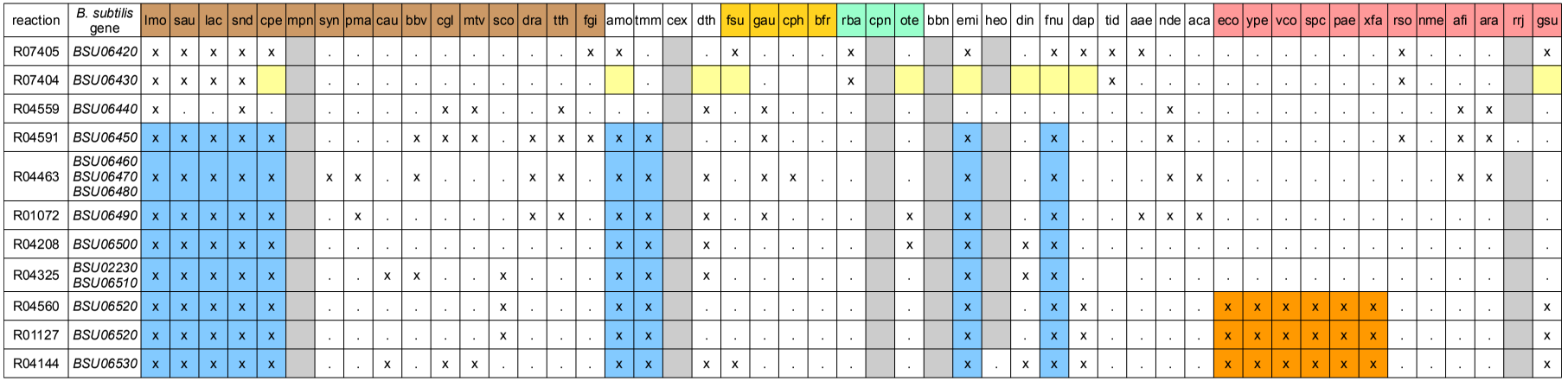
**Figure C.1** Group of genes involved in the `CoMetGeNe` trail in Figure VII.1, obtained for the valine, leucine, and isoleucine biosynthesis pathway in *Bacillus subtilis* (`bsu00290`). This is the complete trail grouping by genes corresponding to Figure VII.2. Columns in purple and gray correspond to species with neighboring functionally similar genes to all, respectively none, of the genes in *B. subtilis* involved in this trail. Cells in light yellow represent species that have neighboring functionally similar genes to at least two *B. subtilis* genes involved in the trail, but not for the gene *BSU28280* involved in the reaction `R01213` (EC 2.3.3.13). Colors in the table header designate the bacterial superphylum (see section VII.1.3 for details).

| reaction | *B. subtilis* gene | lmo | sau | lac | snd | cpe | mpn | syn | pma | cau | bbv | cgl | mtv | sco | dra | tth | fgi | amo | tmm | cex | dth | fsu | gau | cph | bfr | rba | cpn | ote | bbn | emi | heo | din | fnu | dap | tid | aae | nde | aca | eco | ype | vco | spc | pae | xfa | rso | nme | afi | ara | rrj | gsu |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R03968 | *BSU28250 BSU28260* | x | x | | x | | | . | x | x | . | . | x | . | . | . | . | . | x | | x | x | | x | x | . | | . | | x | | x | | . | x | . | . | x | x | x | x | x | . | . | . | . | x | | x |
| R04001 | *BSU28250 BSU28260* | x | x | | x | | | . | x | x | . | . | x | . | . | . | . | . | x | | x | x | | x | x | . | | . | | x | | x | | . | x | . | . | x | x | x | x | x | . | . | . | . | x | | x |
| R04426 | *BSU28270* | x | x | | x | | | . | . | . | . | . | x | . | . | . | . | x | x | | x | . | | x | x | . | | . | | | | . | | . | . | . | x | x | x | x | x | . | . | . | . | x | | . |
| R01213 | *BSU28280* | x | x | | x | | | . | . | x | . | . | . | . | . | x | . | x | x | | x | . | | x | x | . | | . | | | | . | | . | . | . | x | x | x | x | x | . | x | x | . | . | . | | x |
| R05071 | *BSU28290* | x | x | | . | | | . | . | x | x | x | x | x | x | x | x | x | x | | x | . | | x | . | x | | x | | | . | . | | x | x | . | x | x | . | . | . | . | x | x | x | x | . | | x |
| R04440 | *BSU28290* | x | x | | . | | | . | . | x | x | x | x | x | x | x | x | x | x | | x | . | | x | . | x | | x | | | . | . | | x | x | . | x | x | . | . | . | . | x | x | x | x | . | | x |
| R00226 | *BSU28300 BSU28310 BSU36010* | x | x | | . | | | . | . | x | . | x | x | x | x | x | x | x | x | | x | . | | x | . | x | | x | | | . | . | | x | . | . | x | x | . | . | . | . | x | x | x | x | . | | x |

**Figure C.2** Group of reactions defining the `CoMetGeNe` trail in Figure VII.1, obtained for the valine, leucine, and isoleucine biosynthesis pathway in *Bacillus subtilis* (`bsu00290`). This is the complete trail grouping by reactions corresponding to Figure VII.3. The cells in gray correspond to species lacking all or a vast majority of reactions from this trail. Cells in light yellow represent species that have neighboring functionally similar genes to *B. subtilis* genes involved in at least two reactions in the trail, but not in reaction `R01213` (EC 2.3.3.13). Cells in blue and orange correspond to species having neighboring functionally similar genes to *B. subtilis* genes involved in the last and first three reactions in the trail, respectively. Colors in the table header designate the bacterial superphylum (see section VII.1.3 for details).

| *B. subtilis* gene | lmo | sau | lac | snd | cpe | mpn | syn | pma | cau | bbv | cgl | mtv | sco | dra | tth | fgi | amo | tmm | cex | dth | fsu | gau | cph | bfr | rba | cpn | ote | bbn | emi | heo | din | fnu | dap | tid | aae | nde | aca | eco | ype | vco | spc | pae | xfa | rso | nme | afi | ara | rrj | gsu |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *BSU06420* | x | x | x | x | x | . | . | . | . | . | . | . | . | . | . | x | x | . | . | . | x | . | . | . | x | . | . | . | x | . | . | x | . | x | x | . | . | . | . | . | . | . | . | x | . | . | . | . | . |
| *BSU06430* | x | x | x | x | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | x | . | . | . | . | . | . | . | . | x | . | . | . | . | . | . | . | . | . | . | . | x | . | . | . | . | . |
| *BSU06440* | x | . | . | x | . | . | . | . | . | . | . | . | . | . | x | . | . | . | . | x | . | x | . | . | . | . | . | . | . | . | . | . | . | . | . | x | . | . | . | . | . | . | . | . | . | . | x | x | . | x |
| *BSU06450* | x | x | x | x | x | . | . | . | . | x | . | . | x | x | x | x | x | x | . | . | . | x | . | . | . | . | . | . | x | . | x | x | . | . | . | x | . | . | . | . | . | . | . | x | . | x | x | . | . |
| *BSU06460* | x | x | x | x | x | . | x | x | . | x | x | x | x | x | x | . | x | x | . | x | . | x | x | . | . | . | . | . | x | . | x | x | x | . | . | x | x | . | . | . | . | . | . | . | . | x | x | . | x |
| *BSU06470* | x | x | x | x | x | . | x | x | . | x | x | x | x | x | x | . | x | x | . | x | . | x | x | . | . | . | . | . | x | . | x | x | x | . | . | x | x | . | . | . | . | . | . | . | . | x | x | . | x |
| *BSU06480* | x | x | x | x | x | . | x | x | . | x | x | x | x | x | x | . | x | x | . | x | . | x | x | . | . | . | . | . | x | . | x | x | x | . | . | x | x | . | . | . | . | . | . | . | . | x | x | . | x |
| *BSU06490* | x | x | x | x | x | . | . | x | . | . | x | x | . | x | x | . | x | x | . | x | . | x | . | . | . | . | x | . | x | . | . | x | x | . | x | x | x | . | . | . | . | . | . | . | . | . | . | . | x |
| *BSU06500* | x | x | x | x | x | . | . | . | . | . | x | x | . | . | . | . | x | x | . | x | . | . | . | . | . | . | x | . | x | . | . | x | . | . | . | . | . | x | x | x | x | x | x | . | . | . | . | . | . |
| *BSU06510* | x | x | x | x | x | . | . | . | x | x | . | . | . | . | . | . | x | x | . | x | . | . | . | . | . | . | . | . | x | . | . | x | . | . | . | . | . | x | x | x | x | x | x | . | . | . | . | . | . |
| *BSU06520* | x | x | x | x | x | . | . | . | . | . | . | . | . | . | . | . | x | x | . | . | . | . | . | . | . | . | . | . | x | . | . | x | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . |
| *BSU06530* | x | x | x | x | x | . | . | . | x | . | . | . | x | . | . | . | x | x | . | x | x | . | . | . | . | . | . | . | x | . | . | x | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . |

**Figure C.3** Group of genes involved in the `CoMetGeNe` trail in Figure VII.4, obtained for the purine metabolism pathway in *Bacillus subtilis* (`bsu00230`). This is the complete trail grouping by genes corresponding to Figure VII.5. Colors in the table header designate the bacterial superphylum (see section VII.1.3 for details).



| reaction | *B. subtilis* gene | lmo | sau | lac | snd | cpe | mpn | syn | pma | cau | bbv | cgl | mtv | sco | dra | tth | fgi | amo | tmm | cex | dth | fsu | gau | cph | bfr | rba | cpn | ote | bbn | emi | heo | din | fnu | dap | tid | aae | nde | aca | eco | ype | vco | spc | pae | xfa | rso | nme | afi | ara | rrj | gsu |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R07405 | *BSU06420* | x | x | x | x | x | | . | . | . | . | . | . | . | . | . | x | x | . | | . | x | . | . | . | x | | . | | x | | . | x | x | x | x | . | . | . | . | . | . | . | . | x | . | . | . | | x |
| R07404 | *BSU06430* | x | x | x | x | | | . | . | . | . | . | . | . | . | . | . | | . | | | | | . | | | x | | | | | | | | | x | . | . | . | . | . | . | . | . | . | x | . | . | . | | |
| R04559 | *BSU06440* | x | . | . | x | . | | . | . | . | . | x | x | . | . | x | . | . | . | | x | . | x | . | . | . | | . | | . | | . | . | . | . | x | . | . | . | . | . | . | . | . | . | x | x | | . |
| R04591 | *BSU06450* | x | x | x | x | x | | . | . | . | x | x | x | . | x | x | x | x | x | | . | . | x | . | . | . | | . | | x | | . | x | . | . | . | x | . | . | . | . | . | . | x | . | x | x | . | . |
| R04463 | *BSU06460 BSU06470 BSU06480* | x | x | x | x | x | | x | x | . | x | . | . | . | x | x | . | x | x | | x | . | x | x | . | . | | . | | x | | . | x | . | . | . | x | x | . | . | . | . | . | . | . | x | x | | . |
| R01072 | *BSU06490* | x | x | x | x | x | | . | x | . | . | . | . | . | x | x | . | x | x | | x | . | x | . | . | . | | x | | x | | . | x | . | . | x | x | x | . | . | . | . | . | . | . | . | . | | . |
| R04208 | *BSU06500* | x | x | x | x | x | | . | . | . | . | . | . | . | . | . | . | x | x | | x | . | . | . | . | . | | x | | x | | x | x | . | . | . | . | . | . | . | . | . | . | . | . | . | . | | . |
| R04325 | *BSU02230 BSU06510* | x | x | x | x | x | | . | . | x | x | . | . | x | . | . | . | x | x | | x | . | . | . | . | . | | . | | x | | x | x | . | | . | . | . | . | . | . | . | . | . | . | . | . | | . |
| R04560 | *BSU06520* | x | x | x | x | x | | . | . | . | . | . | . | x | . | . | . | x | x | | . | . | . | . | . | . | | . | | x | | . | x | x | . | . | . | . | x | x | x | x | x | x | . | . | . | . | | x |
| R01127 | *BSU06520* | x | x | x | x | x | | . | . | . | . | . | . | x | . | . | . | x | x | | . | . | . | . | . | . | | . | | x | | . | x | x | . | . | . | . | x | x | x | x | x | x | . | . | . | . | | x |
| R04144 | *BSU06530* | x | x | x | x | x | | . | . | x | . | x | x | . | . | . | . | x | x | | x | x | . | . | . | . | | . | | x | . | x | x | x | . | . | . | . | x | x | x | x | x | x | . | . | . | . | | x |

**Figure C.4** Group of reactions defining the `CoMetGeNe` trail in Figure VII.4, obtained for the purine metabolism pathway in *Bacillus subtilis* (`bsu00230`). This is the complete trail grouping by reactions corresponding to Figure VII.6. Cells in gray correspond to species lacking all or a vast majority of reactions from this trail. Cells in light yellow correspond to species that do not perform the reaction `R07404`, if these species possess neighboring functionally similar genes for at least two reactions in the trail. Cells in blue correspond to the maximum set of reactions among the reactions in the trail that are common to different species and performed by neighboring functionally similar genes in these species. Cells in orange correspond to reactions performed by products of neighboring genes in Gammaproteobacteria. Colors in the table header designate the bacterial superphylum (see section VII.1.3 for details).

| *E. coli* gene | ype | vco | spc | pae | xfa | rso | nme | afi | ara | rrj | gsu | nde | aca | din | fnu | dap | tid | aae | bsu | lmo | sau | lac | snd | cpe | mpn | syn | pma | cau | bbv | cgl | mtv | sco | dra | tth | fgi | amo | tmm | cex | dth | fsu | gau | cph | bfr | rba | cpn | ote | bbn | emi | heo |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *b0002* | x | x | x | x | x | x | . | x | . | . | . | x | . | x | . | x | . | . | x | x | x | x | x | . | . | . | . | . | x | x | x | x | . | x | . | x | x | x | x | . | x | . | x | . | . | x | . | x | . |
| *b0003* | x | x | x | . | x | . | . | . | . | . | . | . | x | . | . | . | . | . | x | x | x | . | x | . | . | . | . | . | x | x | x | x | . | . | x | x | x | . | x | . | . | x | . | . | . | . | . | x | . |
| *b0004* | x | x | x | x | x | x | . | x | . | . | . | x | x | x | . | . | . | . | x | x | x | x | . | . | . | . | . | . | . | . | x | x | . | x | x | x | x | x | x | . | x | x | x | . | . | x | . | x | . |

**Figure C.5** Group of genes involved in the `CoMetGeNe` trail in Figure VII.9, obtained for the glycine, serine, and threonine metabolism pathway in *Escherichia coli* (`eco00260`). This is the complete trail grouping by genes corresponding to Figure VII.10. Eleven of the species in the data set (highlighted in light yellow) either do not have functionally similar genes to *b0003*, or are not contiguous with genes functionally similar to *b0002* and *b0004*. Colors in the table header designate the bacterial superphylum (see section VII.1.3 for details).

| reaction | *E. coli* gene | ype | vco | spc | pae | xfa | rso | nme | afi | ara | rrj | gsu | nde | aca | din | fnu | dap | tid | aae | bsu | lmo | sau | lac | snd | cpe | mpn | syn | pma | cau | bbv | cgl | mtv | sco | dra | tth | fgi | amo | tmm | cex | dth | fsu | gau | cph | bfr | rba | cpn | ote | bbn | emi | heo |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R00480 | *b0002 b3940 b4024* | x | x | x | . | x | . | . | . | . | . | . | x | . | x | | x | . | . | . | . | . | . | . | . | . | | | . | . | . | | . | . | . | . | x | x | x | x | . | x | . | x | . | . | x | | . | . |
| {R01773, R01775} | *b0002 b3940* | x | x | x | x | x | x | . | x | . | | . | x | . | x | | x | . | . | x | x | x | x | x | | | | . | . | . | x | x | x | x | . | x | | x | | | | | | | | | | | | | x | . |
| R01771 | *b0003* | x | x | x | . | x | . | . | . | . | | | . | x | . | . | | | . | x | x | x | x | x | | | | . | . | . | x | x | x | x | . | . | x | x | x | | x | . | | x | | | | | | x | . |
| R01466 | *b0004* | x | x | x | x | x | x | . | x | . | | . | x | x | x | | . | . | . | x | x | x | . | . | | | | . | . | . | . | . | x | x | . | x | x | x | x | x | x | . | x | x | x | . | | x | | x | . |

**Figure C.6** Group of reactions defining the `CoMetGeNe` trail in Figure VII.9, obtained for the glycine, serine, and threonine metabolism pathway in *Escherichia coli* (`eco00260`). This is the complete trail grouping by reactions corresponding to Figure VII.11. Colors in the table header designate the bacterial superphylum (see section VII.1.3 for details).

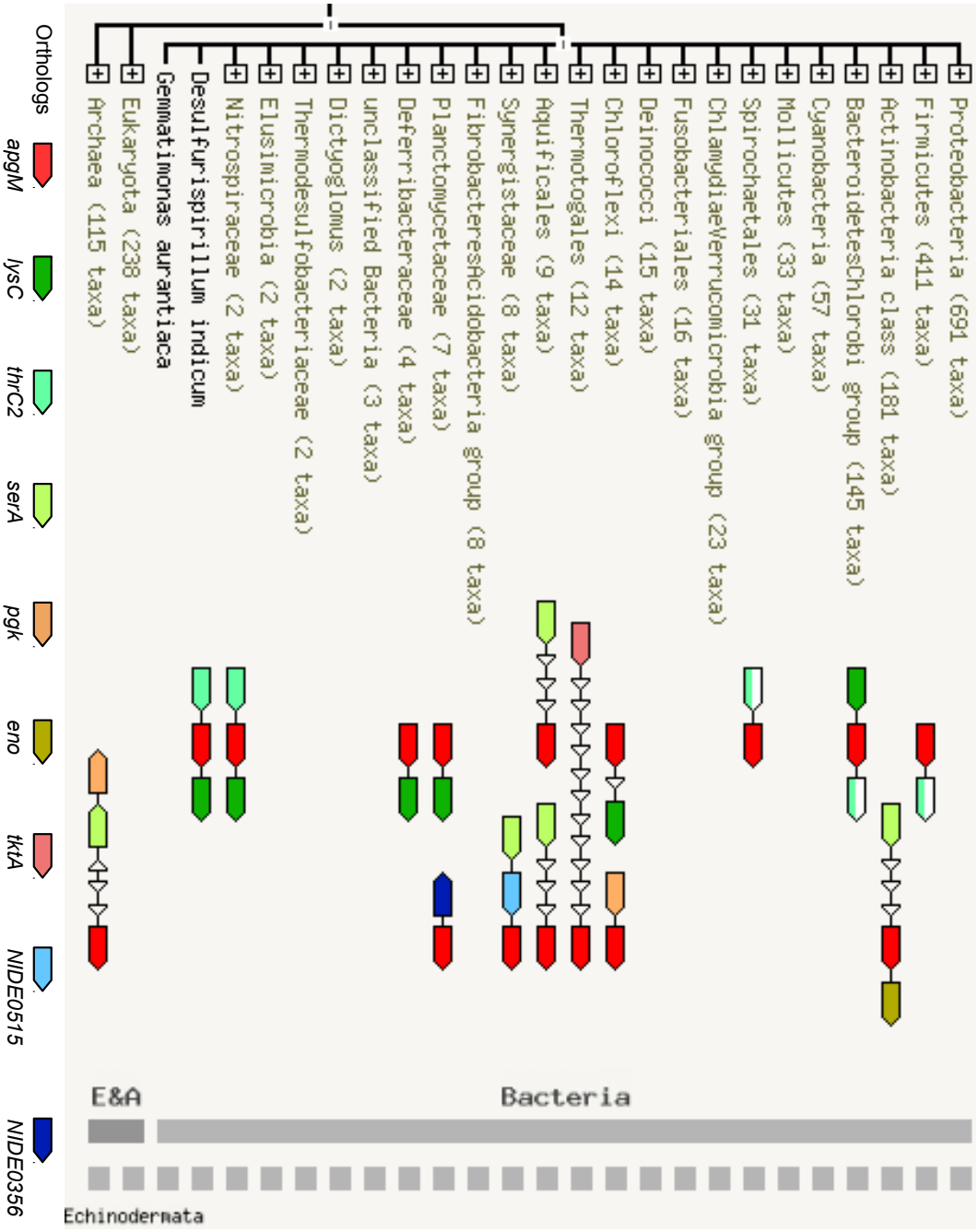



**Figure C.7** Neighborhood evidence from STRING [Szklarczyk *et al.*, 2014] for the query protein ApgM (*NIDE4112*) in *Nitrospira defluvii*. The legend below the figure shows the orthologous sequences corresponding to each gene color. Quoting STRING, *horizontal sections indicate that the orthology relations of the gene are complex. This is either due to gene duplication events (paralogy), or due to technical problems when assigning orthology*.

| *E. coli* gene | ype | vco | spc | pae | xfa | rso | nme | afi | ara | rrj | gsu | nde | aca | din | fnu | dap | tid | aae | bsu | lmo | sau | lac | snd | cpe | mpn | syn | pma | cau | bbv | cgl | mtv | sco | dra | tth | fgi | amo | tmm | cex | dth | fsu | gau | cph | bfr | rba | cpn | ote | bbn | emi | heo |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *b0085* | x | x | x | x | x | x | x | x | x | x | x | x | x | . | . | x | x | . | x | x | . | . | . | x | . | . | . | . | . | x | x | x | . | . | x | x | x | x | x | x | x | x | x | . | . | x | . | . | . |
| *b0086* | x | x | x | x | x | x | x | x | x | x | . | x | x | x | x | x | x | . | . | . | . | . | . | x | . | . | . | x | x | x | x | x | . | x | x | x | x | x | x | x | x | x | . | . | x | x | x | x | . |
| *b0087* | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | . | x | x | x | x | . | x | . | . | . | x | x | x | x | x | . | x | x | x | x | x | x | . | x | x | x | . | x | x | x | x | x |
| *b0088* | x | x | x | x | . | x | x | x | x | . | x | x | x | x | x | x | x | . | x | x | x | x | x | . | . | . | . | x | x | x | x | x | . | x | x | x | x | x | x | . | x | x | x | . | x | x | . | . | x |
| *b0090* | x | x | x | x | x | x | x | x | x | . | x | x | x | x | x | x | x | . | x | x | . | x | x | . | . | . | . | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | . | x | x | . | x | . |
| *b0091* | x | x | x | x | x | x | x | x | x | . | x | x | x | . | x | x | x | . | . | . | . | . | . | . | . | . | . | x | x | x | x | . | x | x | . | x | x | x | x | x | x | x | . | x | . | . | . | . |
| *b0092* | x | . | . | x | x | x | x | x | x | . | x | x | . | . | x | x | . | . | . | . | x | . | x | . | . | . | . | x | . | . | . | . | . | . | . | . | . | x | . | . | . | . | . | . | x | x | x | . | x |

**Figure C.8** Group of genes involved in the `CoMetGeNe` trails in Figure VII.13, obtained for the peptidoglycan biosynthesis pathway in *Escherichia coli* (`eco00550`). This is the complete trail grouping by genes corresponding to Figure VII.15. Colors in the table header designate the bacterial superphylum (see section VII.1.3 for details).

| reaction | *E. coli* gene | ype | vco | spc | pae | xfa | rso | nme | afi | ara | rrj | gsu | nde | aca | din | fnu | dap | tid | aae | bsu | lmo | sau | lac | snd | cpe | mpn | syn | pma | cau | bbv | cgl | mtv | sco | dra | tth | fgi | amo | tmm | cex | dth | fsu | gau | cph | bfr | rba | cpn | ote | bbn | emi | heo |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R02788 | *b0085* | x | x | x | x | x | x | x | x | x | x | x | x | x | . | . | x | x | . | x | x | | . | . | x | | . | . | . | . | x | x | x | . | . | x | x | x | x | x | x | x | x | . | . | x | . | . | . |
| R04617 | *b0086* | x | x | x | x | x | x | x | x | x | x | | x | x | x | x | x | x | . | . | . | . | . | . | x | | . | . | x | x | x | x | x | . | x | x | x | x | x | x | x | x | x | . | | x | x | x | x | . |
| R05630 | *b0087* | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | . | x | x | x | x | . | x | | . | . | x | x | x | x | x | . | x | x | x | x | x | x | . | x | x | x | | x | x | x | x | x |
| R02783 | *b0088* | x | x | x | x | . | x | x | x | x | . | x | x | x | x | x | x | x | . | x | x | x | x | x | . | | . | . | x | x | x | x | x | . | x | x | x | x | x | x | . | x | x | x | | x | x | . | . | x |
| R05032 | *b0090* | x | x | x | x | x | x | x | x | x | . | x | x | x | x | x | x | x | . | x | x | . | x | x | . | | . | . | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | | x | x | . | x | . |
| R03193 | *b0091* | x | x | x | x | x | x | x | x | x | . | x | x | x | . | x | x | x | . | . | . | . | . | . | . | | . | . | x | x | x | x | . | x | x | | x | x | x | x | x | x | x | x | | x | | . | . | . |

**Figure C.9** Group of reactions defining the `CoMetGeNe` trail highlighted in yellow in Figure VII.13, obtained for the peptidoglycan biosynthesis pathway in *Escherichia coli* (`eco00550`). This is the complete trail grouping by reactions corresponding to Figure VII.16. Cells in gray designate missing reactions. Colors in the table header designate the bacterial superphylum (see section VII.1.3 for details).

| reaction | *E. coli* gene | ype | vco | spc | pae | xfa | rso | nme | afi | ara | rrj | gsu | nde | aca | din | fnu | dap | tid | aae | bsu | lmo | sau | lac | snd | cpe | mpn | syn | pma | cau | bbv | cgl | mtv | sco | dra | tth | fgi | amo | tmm | cex | dth | fsu | gau | cph | bfr | rba | cpn | ote | bbn | emi | heo |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R04573 | *b0086* | x | x | x | x | x | x | x | x | x | x | | x | x | x | x | x | x | . | x | x | x | . | x | x | | . | . | x | x | x | x | x | . | x | x | x | x | x | x | x | x | . | | x | x | x | x | x |
| R05629 | *b0087* | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | . | . | . | . | x | . | x | | . | . | x | x | x | x | x | . | x | x | x | x | x | x | . | x | x | x | | x | x | x | x | . |
| R05662 | *b0090* | x | x | x | x | x | x | x | x | x | . | x | x | x | x | x | x | x | . | . | . | . | x | . | . | | . | . | x | x | x | x | x | . | x | x | x | x | x | x | x | x | x | x | | x | x | . | x | . |
| R01150 | *b0092* *b0381* | x | . | . | x | x | x | x | x | x | . | x | x | . | . | x | x | . | . | x | x | x | . | x | . | | . | . | x | . | . | . | . | . | . | x | . | . | x | . | . | . | . | . | . | x | x | . | . | x |

**Figure C.10** Group of reactions defining the `CoMetGeNe` trail highlighted in purple in Figure VII.13, obtained for the peptidoglycan biosynthesis pathway in *Escherichia coli* (`eco00550`). This is the complete trail grouping by reactions corresponding to Figure VII.17. Cells in gray designate missing reactions. Colors in the table header designate the bacterial superphylum (see section VII.1.3 for details).

# D

## Résumé substantiel

La biologie des systèmes est un domaine en continuelle expansion où les nouveaux développements en matière de techniques de biologie moléculaire génèrent des données plus riches ou accélèrent la production de telles données. Ce déluge d'informations génomiques, transcriptomiques, protéomiques, interactomiques et métabolomiques, pour n'en citer que quelques exemples, crée la nécessité de disposer d'algorithmes de traitement et d'analyse de plus en plus spécialisés et efficaces. Un accent particulier est mis sur des approches intégratives capables d'incorporer des données issues de sources hétérogènes afin d'approfondir notre compréhension quand les systèmes cellulaires sont considérés dans leur totalité.

Dans ce contexte, de nombreuses approches destinées aux réseaux biologiques hétérogènes sont modélisées comme des problèmes de théorie des graphes. En général, de telles approches visent soit l'intégration de réseaux hétérogènes, soit l'extraction de motifs à partir de ces réseaux. D'un point de vue algorithmique, les travaux présentés dans cette thèse s'inscrivent dans la dernière catégorie. Le but principal de cette thèse est d'explorer la relation entre le métabolisme et le génome.

Il est communément admis que des étapes enzymatiques successives impliquant des produits de gènes situés à proximité sur le chromosome traduisent un avantage évolutif du maintien de cette relation de voisinage au niveau métabolique ainsi que génomique. En conséquence, nous avons choisi de nous concentrer sur la détection de réactions voisines catalysées par des produits de gènes voisins, où la notion de voisinage peut être modulée en autorisant que certaines réactions et/ou gènes soient omis. Plus spécifiquement, les motifs recherchés sont des *trails* de réactions (c'est-à-dire des séquences de réactions pouvant répéter des réactions, mais pas les liens entre elles) catalysées par des produits de gènes voisins. De tels motifs de voisinage sont appelés des *motifs métaboliques et génomiques*.

Le choix d'extraire des *trails* est motivé par trois aspects. Premièrement, les *trails* sont les seuls motifs capables de capturer des cycles dans des voies métaboliques. Deuxièmement, il s'agit de *trails* dans des graphes orientés, préservant l'orientation des réactions. Enfin, les *trails* garantissent le fait que les motifs extraits correspondent à des routes métaboliques réelles, à la différence des méthodes extrayant des sous-graphes.

En plus de l'identification des motifs métaboliques et génomiques, nous étudions leur degré de *conservation* entre une multitude d'espèces. De façon analogue à la notion de voisinage métabolique et génomique, nous proposons une définition flexible pour la conservation des motifs. Ainsi, en évaluant la conservation d'un motif métabolique et génomique, l'ordre des réactions dans des *trails* ainsi que l'ordre des gènes fonctionnellement similaires au niveau du chromosome peuvent être différents entre les espèces. Par ailleurs, la conservation peut être partielle, traduisant le fait que les contenus des *trails* et des contextes génomiques peuvent varier, avec certaines espèces ayant conservé juste des parties d'un motif métabolique et génomique détecté dans d'autres organismes.

Afin de détecter des motifs métaboliques et génomiques pour une espèces donnée, nous proposons une méthodologie de fouille de graphes hétérogènes dont la modélisation sous-jacente est facilement adaptable à d'autres types de données biologiques. Nous présentons un algorithme exact pour énumérer des *trails* dans une voie métabolique, reposant sur l'énumération des chemins dans le *line graph* associé. Cette opération coûteuse du point de vue computationnel est limitée par une réduction des graphes de départ et par le fait qu'on énumère uniquement des chemins entre sommets pouvant appartenir à la solution finale.

Nous proposons également une méthodologie pour regrouper les *trails* obtenus afin de détecter des motifs métaboliques et génomiques conservés. Pour prendre en compte les variations entre *trails* similaires en termes d'ordre des réactions et/ou de gènes, ainsi qu'en termes de leur présence ou absence, les *trails* sont traduits en ensembles de réactions. Nous décrivons deux approches pour évaluer la conservation des *trails* appartenant à une espèce désignée comme référence: le regroupement par réactions, focalisé sur la conservation des motifs métaboliques, et le regroupement par gènes, focalisé sur la conservation des motifs génomiques. Les deux approches construisent des tables similaires aux profils phylogénétiques pour des ensembles de réactions ou groupes de gènes voisins impliqués dans des *trails* de l'espèce de référence. Conjointement, ces deux approches permettent de comparer le degré de conservation des *trails* parmi les espèces étudiées.

Les méthodologies d'extraction et de regroupement des *trails* sont implémentées dans un pipeline libre appelé `CoMetGeNe` (***Co**nserved **Met**abolic and **Ge**nomic*

***Ne**ighborhoods*). À l'aide de ce logiciel, nous analysons un jeu de données de 50 espèces bactériennes représentant les principaux phylums du domaine bactérien de l'arbre du vivant. Nous montrons que l'extraction des *trails* ainsi que leur regroupement sont des méthodologies exploratoires permettant de découvrir des liens entre contextes métaboliques et génomiques. L'intérêt de notre approche est mis en évidence en montrant que les motifs métaboliques et génomiques identifiés peuvent conduire à des intuitions biologiques, à la formulation d'hypothèses biologiques, ainsi qu'à la découverte de problèmes d'annotation dans des bases de connaissances.

La notion de motif métabolique et génomique est étendue pour prendre en compte la similarité chimique entre *trails*. Ceci nous conduit à identifier des motifs étendus, appelés motifs *chimiques, métaboliques et génomiques*. Ils reflètent le fait que la nature chimique des transformations effectuées est un facteur supplémentaire dans la relation entre le métabolisme et le génome. En utilisant une approche existante de chémoinformatique, on calcule des *signatures* réactionnelles, consistant en une description des atomes et liens atomiques qui diffèrent entre les substrats et les produits d'une réaction donnée. Nous proposons deux approches pour regrouper les *trails* obtenus avec `CoMetGeNe` en fonction de leur similarité chimique, en utilisant les signatures réactionnelles. La première est une approche qualitative consistant à remplacer les ensembles de réactions par des ensembles de signatures réactionnelles. La deuxième approche est quantitative et consiste à remplacer les ensembles de réactions par des clusters de signatures réactionnelles. En général, les études portant sur la modularité du métabolisme définissent les modules comme étant des séquences de transformations enzymatiques similaires du point de vue chimique. Nous montrons que les motifs chimiques, métaboliques et génomiques détectés à l'aide des signatures réactionnelles correspondent à une classe de modules métaboliques ayant la particularité que les gènes encodant les enzymes impliquées sont voisins.

Finalement, une dernière contribution est la détection de problèmes de consistance dans la base de connaissances KEGG. Celle-ci est une ressource de référence en biologie des systèmes, son objectif principal étant de lier les séquences aux fonctions biologiques. L'utilisation intensive de cette ressource durant les travaux de cette thèse nous ont amenée à remarquer plusieurs types d'inconsistances entre les différentes bases de données de KEGG. Nous exposons ici deux types de tels problèmes, en donnant des approches générales pour leur inventaire systématique.

université PARIS-SACLAY
ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

**Titre :** Identification des motifs de voisinage conservés dans des contextes métaboliques et génomiques

**Mots clés :** réseau métabolique, contexte génomique, fouille de graphes, algorithme d'énumération de trails, recherche de motifs, similarité chimique

**Résumé :** Cette thèse s'inscrit dans le cadre de la biologie des systèmes et porte plus particulièrement sur un problème relatif aux réseaux biologiques hétérogènes. Elle se concentre sur les relations entre le métabolisme et le contexte génomique, en utilisant une approche de fouille de graphes.
Il est communément admis que des étapes enzymatiques successives impliquant des produits de gènes situés à proximité sur le chromosome traduisent un avantage évolutif du maintien de cette relation de voisinage au niveau métabolique ainsi que génomique. En conséquence, nous choisissons de nous concentrer sur la détection de réactions voisines catalysées par des produits de gènes voisins, où la notion de voisinage peut être modulée en autorisant que certaines réactions et/ou gènes soient omis. Plus spécifiquement, les motifs recherchés sont des trails de réactions (c'est-à-dire des séquences de réactions pouvant répéter des réactions, mais pas les liens entre elles) catalysées par des produits de gènes voisins. De tels motifs de voisinage sont appelés des motifs métaboliques et génomiques.
De plus, on s'intéresse aux motifs de voisinage métabolique et génomique conservés, c'est-à-dire à des motifs similaires pour plusieurs espèces. Parmi les variations considérées pour un motif conservé, on considère l'absence/présence de réactions et/ou de gènes, ou leur ordre différent.
Dans un premier temps, nous proposons des algorithmes et des méthodes afin d'identifier des motifs de voisinage métabolique et génomique conservés. Ces méthodes sont implémentées dans le pipeline libre CoMetGeNe (COnserved METabolic and GEnomic NEighborhoods). À l'aide de CoMetGeNe, on analyse une sélection de 50 espèces bactériennes, en utilisant des données issues de la base de connaissances KEGG.
Dans un second temps, un développement de la détection de motifs conservés est exploré en prenant en compte la similarité chimique entre réactions. Il permet de mettre en évidence une classe de modules métaboliques conservés, caractérisée par le voisinage des gènes intervenants.

**Title :** Mining conserved neighborhood patterns in metabolic and genomic contexts

**Keywords :** metabolic network, genomic context, graph mining, trail enumeration algorithm, pattern search, chemical similarity

**Abstract :** This thesis fits within the field of systems biology and addresses a problem related to heterogeneous biological networks. It focuses on the relationship between metabolism and genomic context through a graph mining approach.
It is well-known that succeeding enzymatic steps involving products of genes in close proximity on the chromosome translate an evolutionary advantage in maintaining this neighborhood relationship at both the metabolic and genomic levels. We therefore choose to focus on the detection of neighboring reactions being catalyzed by products of neighboring genes, where the notion of neighborhood may be modulated by allowing the omission of several reactions and/or genes. More specifically, the sought motifs are trails of reactions (meaning reaction sequences in which reactions may be repeated, but not the links between them). Such neighborhood motifs are referred to as metabolic and genomic patterns.
In addition, we are also interested in detecting conserved metabolic and genomic patterns, meaning similar patterns across multiple species. Among the possible variations for a conserved pattern, the presence/absence of reactions and/or genes may be considered, or the different order of reactions and/or genes.
A first development proposes algorithms and methods for the identification of conserved metabolic and genomic patterns. These methods are implemented in an open-source pipeline called CoMetGeNe (COnserved METabolic and GEnomic NEighborhoods). By means of this pipeline, we analyze a data set of 50 bacterial species, using data extracted from the KEGG knowledge base.
A second development explores the detection of conserved patterns by taking into account the chemical similarity between reactions. This allows for the detection of a class of conserved metabolic modules in which neighboring genes are involved.

**Université Paris-Saclay**
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France