Heterogeneous Graph Mining for Biological Pattern Discovery in Metabolic Pathways

Alexandra Zaharia, Bernard Labedan, Christine Froidevaux, Alain Denise

Extended abstract

SeqBio 2016

Heterogeneous Graph Mining for Biological Pattern Discovery in Metabolic Pathways

Alexandra Zaharia[1]*, Bernard Labedan[1], Christine Froidevaux[1], Alain Denise[1]

[1]*LRI, CNRS UMR 8623, Université Paris Sud, Orsay, France*
***Corresponding author**: zaharia@lri.fr

Abstract
Systems biology studies biological networks and the relations between them. Among the various types of biological networks, we focus on metabolic pathways and gene neighboring networks, respectively modeled by directed and undirected graphs. We attempt to identify maximal sets of consecutive metabolic reactions catalyzed by products of neighboring genes.
The approach proposed here is HNet, a non-exhaustive exact method that is capable to take into account (i) skipped genes and/or reactions and (ii) metabolic pathways containing cycles. HNet relies on a previously described graph reduction method and on trail finding in a directed graph by performing path finding in its line graph. A trail is a path that can contain repeated vertices, but not repeated arcs.
HNet is used to analyze the genomes and metabolic networks of 50 prokaryotic species in order to gain insight into metabolic pathway evolution.

Keywords
Graph mining — Trail finding — Heterogeneous networks — Metabolic pathway — Gene neighboring network

1. Introduction

Networks model complex relationships in telecommunications, social interactions and biological processes, to cite but a few examples. Two networks are said to be heterogeneous if they present different types of information modeling the same entity. Heterogeneous network mining reveals patterns describing distinct aspects of related processes. Examples of heterogeneous networks in systems biology include the gene neighboring network of an organism, its metabolic pathways, its co-expression, co-regulation, and protein/protein interaction networks. The information contained within is complementary for the organism in question.

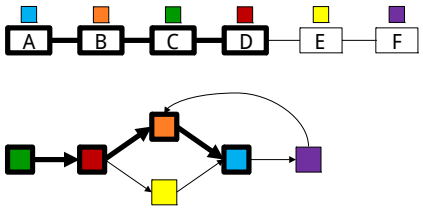

Figure 1. Metabolic clustering. A pattern consisting of four genes (A-D) coding for enzymes that catalyze consecutive reactions (green, red, orange, and blue) is emphasized by thick contours.

The aim of this work is to identify occurrences of metabolic clustering, i.e. consecutive reactions in a metabolic pathway catalyzed by products of neighboring genes (Figure 1). Such patterns can shed light on metabolic pathway evolution.

We first introduce the theoretical concepts (section 2) required to model the biological problem (section 3) that we formally define in section 4. Next, we present the graph reduction method employed (see 5.1) for enabling path finding in the line graph (see 5.2). The exact approach HNET (see 5.3) implements the problem formulation stated in section 4, while also allowing for network gaps (see 5.4). Finally, HNET is applied to biological data on a selected panel of prokaryotes (see 5.5).

2. Preliminaries

Definition 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. If it is not specifically stated that $G = (V, E)$, the notation $V(G)$ can be used to denote the vertex set 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.

Definition 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 $H = (X, F)$ where $F = \{(u, v) \mid u, v \in X \text{ and } (u, v) \in E\}$.

Definition 3 A **strongly connected component (SCC)** of a directed graph D is a maximal subgraph of D such that, for any vertices u and v of D, there is a path from u to v and a path from v to u.

Definition 4 A **walk** in a directed graph $D = (V, A)$ is an ordered sequence of vertices $(v_1,\ v_2,\ \ldots,\ v_k)$ such that $v_i \in V$ for every $i \in \{1,\ \ldots,\ k\}$ and $(v_i,\ v_{i+1}) \in A$ for every $i \in \{1,\ \ldots,\ k-1\}$. A **path** is a walk with no repeated vertices. A **trail** in a directed graph is a walk with no repeated arcs.

Definition 5 Let $D = (V, A)$ be a directed graph. The **line graph** of D is the graph $L(D) = (A, A')$ where: (i) A, the vertex set of $L(D)$, is the set of arcs of D and (ii) A', the set of arcs of $L(D)$, represents adjacencies between arcs of D, i.e. $(x, y) \in A' \Leftrightarrow x \in A,\ y \in A,\ x = (r, s),\ y = (s, t)$, with $r,\ s,\ t \in V$.

Definition 6 Let $D = (V, A)$ be a directed graph and $L(D) = (A, A')$ be its line graph. Let $P = (a_1,\ a_2,\ \ldots,\ a_k)$ be a path in $L(D)$, where $a_i = (t_{i-1},\ t_i)\ \forall i \in \{1,\ \ldots,\ k\}$ are arcs in D. The **trail in D corresponding to** P, denoted $L^{-1}(P)$, is the trail $T = (t_0,\ t_1,\ t_2,\ \ldots,\ t_{k-1},\ t_k)$.

Definition 7 Let $T = (v_1,\ \ldots,\ v_k)$ be a trail in a directed graph. The **span** of T is the number of distinct vertices in T.

3. Model

Metabolic pathway A directed graph with vertices representing reactions is used to model a metabolic pathway. There is an arc from reaction R_i to R_j if R_i produces a metabolite that is also a substrate for R_j.

Gene neighboring An undirected graph with vertices representing genes is used to model gene neighboring within the genome. There is an edge between two genes if they are adjacent on the same strand of the same chromosome, irrespective of the intergenic distance.

Correspondence It must be known in advance which gene products are involved in any given reaction. This information can be found in a data source such as KEGG [1] which, for a given species, contains information on its metabolic pathways, the reactions that are known to be catalyzed, and the associated genes.

Graph transformation Once a correspondence between reactions and genes has been established, it is necessary to build an additional undirected graph reflecting gene adjacency in relation to the reactions that the gene products catalyze. As in the case of the directed graph modeling a metabolic pathway, the vertices of this new undirected graph represent reactions. Its construction is thoroughly detailed in [2] and can be briefly described as linking two reactions R_i and R_j with an edge if at least one of the genes coding for an enzyme involved in reaction R_i is adjacent on the chromosome to a gene coding for an enzyme involved in R_j.

4. Problem formulation

Given a metabolic pathway and the gene neighboring network for the same organism, the biological objective is to identify a maximal number of consecutive reactions being catalyzed by products of neighboring genes.

The initial problem formulation as presented in [3], [2] and [4] is Longest Supported Path (LSP), adapted below from [4].

Longest Supported Path (LSP)
Input: A directed graph $D = (V, A)$, an undirected graph $G = (V, E)$.
Solution: A longest path P in D such that $G[V(P)]$ is connected.

Having investigated concrete examples of metabolic pathways and gene neighboring networks, it became apparent that the solution for LSP requiring a path prevented straightforward handling of input pathways containing cycles. Although decompositions into directed acyclic graphs have been proposed [2], repeated vertices could not be integrated into the solution. As the large majority of metabolic pathways exhibit cycles (the simplest examples being reversible reactions), we deemed it necessary to allow solutions to be extended by cycles.

Below we propose two equivalent formulations that address this issue: Supported Trail of Maximum Span (STMS) which is easier to state, and Supported Corresponding Trail of Maximum Span (CTMS) which allows for a more straightforward algorithm. Both problems are NP-hard [5], as well as LSP [4].

Supported Trail of Maximum Span (STMS)
Input: A directed graph $D = (V, A)$, an undirected graph $G = (V, E)$, an arc (u, v) in D.
Solution: A trail of maximum span T in D passing through (u, v) such that $G[V(T)]$ is connected.

Supported Corresponding Trail of Maximum Span (CTMS)
Input: A directed graph $D = (V, A)$, an undirected graph $G = (V, E)$, an arc (u, v) in D.
Solution: 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.

5. Results and discussion

5.1 Graph reduction

Fertin *et al.* [3] 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 V, as well as a path P in D, the cover set of P with respect to D and G is either the empty set, or a maximal subset of V 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. 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 STMS yields the same solution as providing D and G as input [6]. This result also holds true for CTMS.

5.2 Path finding in the line graph

STMS is impractical, since solving it requires brute-force trail enumeration. Instead, we focus on CTMS, which allows for a more refined approach (detailed in 5.3) and opens the possibility of circumventing trail enumeration in the directed graph D by performing path enumeration in its line graph. Since path enumeration in $L(D)$ is unavoidable, we restrict it to a minimum using the following three-step process:

1. The SCCs of $L(D)$ and its condensation graph are computed, where a condensation graph results from replacing every SCC with a single vertex.
2. For every SCC of $L(D)$, vertices acting as entry (exit) points from (to) predecessor (successor) SCCs in the condensation graph are determined.
3. For every SCC X of $L(D)$, path enumeration is performed only between pairs of entry and exit points of feasible pairs of SCCs acting as predecessors/successors of X. These paths are evaluated in terms of the span of their corresponding trails in D and the best candidate paths among them are retained.

5.3 HNet

The method we propose, HNet, provides a solution to the CTMS problem. Unlike the heuristic solution introduced by Fertin *et al.* [3] to the LSP problem, HNet is an exact method. However, it is non-exhaustive, meaning that if several paths in $L(D)$ have corresponding trails of maximum span in D passing through a given arc (u, v), then only one of them is reported as solution.

HNet starts by determining the cover set S of the arc (u, v) in D with respect to graphs D and G, which are then reduced to their respective subgraphs induced by S (see 5.1). Next, best candidate paths for every SCC of $L(D)$, the line graph of D, are retained (see 5.2). Finally, the solution for CTMS is computed as follows:

1. All possible paths in the condensation graph of $L(D)$ are enumerated.
2. Every path in the condensation graph of $L(D)$ is translated to the corresponding path(s) in $L(D)$ by concatenating best candidate paths determined and stored as described in 5.2.
3. For every such path P in $L(D)$, if it contains vertex (u, v), it is checked whether $G[L^{-1}(P)]$ is connected. P is retained as a candidate solution if $L^{-1}(P)$ has maximum span so far. The initial candidate solution is the empty set.

In terms of computational complexity, HNet is exponential in the size of $L(D)$ but is, nevertheless, quite fast for the biological data that it is applied to (see 5.5).

5.4 Allowing for skipped vertices

The CTMS problem formulation implies that solutions consist of strictly neighboring genes and reactions. Inspired by a previous graph-based approach for the integration of heterogeneous biological data [7], a pre-processing step was added to HNet in order to allow for discontiguous reactions and/or genes. The pre-processing step consists in modifying graphs D (respectively G) by adding arcs (edges) between vertices separated by at most δ_D (δ_G) other reactions (genes).

5.5 Application to biological data

Metabolic pathways and gene neighboring information were extracted from KEGG [1] for a panel of 50 species spanning all major phyla of the bacterial tree of life. HNet was executed for all metabolic pathways of the selected species with gap parameters δ_D and δ_G varying between 0 and 3. The total run time was under four days on a desktop computer. Using a timeout of five minutes in order to stop execution if no solution was produced for a given pathway and set of (δ_D, δ_G) parameters, more than 95% of the dataset has been processed.

As expected for prokaryotes, the results revealed a high degree of correlation between the ordering of genes on the chromosome, their clustering in operons and their involvement in successive steps of a given metabolic pathway. We are currently focusing on studying metabolic pathway variation in related prokaryotic taxa and considering applying HNet to help reconstructing ancestral prokaryotic metabolism. It would also be interesting to employ the same reasoning to fungal species.

References

[1] KEGG Pathway. http://www.kegg.jp/kegg/pathway.html. Version 80.0. Accessed: 2016-10-17.

[2] Hafedh Mohamed-Babou. *Comparaison de réseaux biologiques*. PhD thesis, Université de Nantes, 2012.

[3] Guillaume Fertin, Hafedh Mohamed-Babou, and Irena Rusu. Algorithms for subnetwork mining in heterogeneous networks. In *International Symposium on Experimental Algorithms*, pages 184–194. Springer, 2012.

[4] Guillaume Fertin, Christian Komusiewicz, Hafedh Mohamed-Babou, and Irena Rusu. Finding supported paths in heterogeneous networks. *Algorithms*, 8(4):810–831, 2015.

[5] NP-hardness proof sketch. https://www.lri.fr/~zaharia/np_sketch.pdf. Accessed: 2016-11-07.

[6] Reduction proof. https://www.lri.fr/~zaharia/reduction_proof.pdf. Accessed: 2016-11-07.

[7] Frédéric Boyer, Anne Morgat, Laurent Labarre, Joël Pothier, and Alain Viari. Syntons, metabolons and interactons: an exact graph-theoretical approach for exploring neighbourhood between genomic and functional data. *Bioinformatics*, 21(23):4209–4215, 2005.