Nikolay *et al. EURASIP Journal on Bioinformatics and Systems Biology* (2017) 2017:10
DOI 10.1186/s13637-017-0063-3

EURASIP Journal on Bioinformatics and Systems Biology

RESEARCH Open Access

Learning directed acyclic graphs from large-scale genomics data

Fabio Nikolay[1*], Marius Pesavento[1], George Kritikos[2] and Nassos Typas[2]

Abstract

In this paper, we consider the problem of learning the genetic interaction map, i.e., the topology of a directed acyclic graph (DAG) of genetic interactions from noisy double-knockout (DK) data. Based on a set of well-established biological interaction models, we detect and classify the interactions between genes. We propose a novel linear integer optimization program called the Genetic-Interactions-Detector (GENIE) to identify the complex biological dependencies among genes and to compute the DAG topology that matches the DK measurements best. Furthermore, we extend the GENIE program by incorporating genetic interaction profile (GI-profile) data to further enhance the detection performance. In addition, we propose a sequential scalability technique for large sets of genes under study, in order to provide statistically significant results for real measurement data. Finally, we show via numeric simulations that the GENIE program and the GI-profile data extended GENIE (GI-GENIE) program clearly outperform the conventional techniques and present real data results for our proposed sequential scalability technique.

Keywords: Genetic interaction analysis, Large-scale gene networks, Discrete optimization, Graph learning, Big data, Multiple hypothesis test

1 Introduction

Genetic interaction analysis aims at uncovering the interactions among a set of genes with respect to a specified cell function of a biological system, e.g., the fitness of a specific bacteria colony. The interactions among the genes under study can be characterized by a directed acyclic graph (DAG) [1] where the hierarchical relationship among two genes of a DAG describes their hierarchical interaction type [2]. However, DAGs cannot be observed directly but only the specified cell function under study which yields observable phenotypes. The term phenotype generally describes the specific manifestation of a biological attribute of an organism which can be observed, e.g., for bacteria, a common biological attribute is the growth measured in colony size, where a specific size of the bacteria colony is a phenotype of this biological attribute.

The role of the studied genes in the cell machinery and the hierarchical interaction types of the genes, as well as the DAG, which describes the latter ones, can only be learned by means of knockout experiments where a gene or a set of genes is functionally switched off and the phenotype is observed. Traditionally, only single-knockout (SK) experiments have been conducted but those mainly provide evidence on the importance of a single gene for the investigated cell process and do not convey much information about the interaction among the genes under study.

Recently, with the technological advances in microarrays and the development of the synthetic genetic array technologies [3], new approaches have been taken that are based on large-scale knockout experiments of pairs of genes. Such double-knockout (DK) experiments are much more powerful for exploring genetic interactions since a DK phenotype of an arbitrary pair of genes generally differs considerably from the superposition of the corresponding SK phenotypes of this pair of genes. According to [2], the gene pairs can be classified into one out of five hierarchical relationship classes based on their SK and DK phenotypes. Further, based on the hierarchical relationship classes, the DAG underlying the observed SK and DK phenotypes can be inferred which directly reflects the genetic interactions among the genes.

*Correspondence: nikolay@nt.tu-darmstadt.de
[1] Communication Systems Group, TU Darmstadt, Merckstr. 25, Darmstadt, Germany
Full list of author information is available at the end of the article

In order to detect the DAG underlying the SK and DK phenotypes, a variety of statistical methods based on scoring the measurements or on thresholding the genetic interaction (GI)-profile data, which is commonly based on Pearson correlation of the SK and DK phenotypes [4–9], respectively, have been developed. However, methods as presented in [4–9] have three considerable disadvantages: (D1) they show poor performance in detecting the DAG underlying the observed SK and DK phenotypes; (D2) they have no ability to combine different types of side information, e.g., GI-profile data with SK and DK phenotypes, to enhance the detection quality; and (D3) they cannot make use of prior knowledge in order to enhance the DAG detection quality. Especially, the ability to overcome the disadvantage in (D2) will become more important in the future, since there is a constantly increasing amount of different data types, e.g., SK and DK phenotypes, Pearson correlation-based GI-profile data, and other types of GI-profile data, freely available. Furthermore, the ability to overcome the deficit in (D3), i.e., to incorporate a priori knowledge about the existing results in genomics research into the DAG detection procedure, is also of great significance, since existing functional relationships among genes are increasingly better understood based on a variety of studies that constantly extend the knowledge on the cell machinery and molecular biology. Although exhibiting the abovementioned disadvantages (D1) to (D3), methods as those presented in [4–9] are the most commonly used methods to detect the DAG underlying the measured SK and DK data. Therefore, we propose the Genetic-Interactions-Detector (GENIE) program, that is an approach based on the biological system model of [2] with which it is possible to overcome the abovementioned shortcomings of the most popular methods as those reported in [4–9]. Since the hierarchical relationship classes are mutually dependent, classifying each pair of genes to a specific hierarchical relationship class corresponds to a multi-hypothesis test. Thus, we formulate this multi-hypothesis test as a linear integer optimization program [10–15] in order to find the set of hierarchical relationship classes, best matching the observed SK and DK phenotypes. Based on the detected set of hierarchical relationship classes, the set of edges of the DAG which reflects the interactions among the genes can be computed. Furthermore, we propose the GI-GENIE program where we advance the proposed GENIE program by incorporating GI-profile data, e.g., GI-profile data based on Pearson correlation of the observed SK and DK phenotypes, into the DAG detection procedure. Due to incomplete knowledge about the true dependencies among the very most sets of genes, i.e., the true DAG of a set of genes with respect to a specific cell function is unknown or only partially known for almost all sets of genes irrespectively of the cell function under study, there is a strong interest in the genomics research community in statistically reliable statements about the topology of the DAGs underlying large sets of genes, i.e., for the empirical probability of a pair of genes to interact with each other. Towards this aim, we propose a sequential technique based on the GENIE/GI-GENIE algorithms that yields statistically significant statements about the interactions among genes from a large set of genes under study.

This paper is organized as follows. We first summarize the biological system model of [2] in Section 2, and then, we present in Section 3 the GENIE program for detecting the set of hierarchical relationship classes, which represents a valid DAG and matches the DK measurements best. In Section 4, we extend the GENIE program with GI-profile data (GI-GENIE). In Section 5, we present our scalability approach in order to obtain statistically significant results for large sets of genes. Following Section 5, we present results for simulated data which demonstrate the performance of the GENIE and the GI-GENIE methods in Section 6. Furthermore, in Section 6, we display real data results for the scalability approach described in Section 5. Finally, we summarize in Section 7 the key parts of this paper and give a brief outlook on future work.

2 System model

In this section, we provide a mathematical description of a DAG as well as its biological implications. Furthermore, we introduce the common biological terms and provide a compact description of the genetic interaction model of [2] including simple explanations on how to read and interpret a DAG.

2.1 Graph properties of a DAG

According to [16], a graph $\mathcal{A} = (\mathrm{V}(\mathcal{A}), \mathrm{E}(\mathcal{A}))$ is well defined by a set of nodes $\mathrm{V}(\mathcal{A}) = \{\mathrm{a}_1, \mathrm{a}_2, ..., \mathrm{a}_A\}$ and a set of edges $\mathrm{E}(\mathcal{A}) = \{\{\mathrm{a}_1, \mathrm{a}_A,\}, \{\mathrm{a}_2, \mathrm{a}_A,\}, ..., \{\mathrm{a}_A, \mathrm{a}_1,\}\}$ where $\left\{\mathrm{a}_i, \mathrm{a}_j,\right\}$ for $\mathrm{a}_i, \mathrm{a}_j \in \mathrm{V}(\mathcal{A})$ denotes a directed edge from a_i to a_j and cardinality $|\mathrm{V}(\mathcal{A})| = A$ denotes the number of elements of set $\mathrm{V}(\mathcal{A})$. The operators $\mathrm{V}(\cdot)$ and $\mathrm{E}(\cdot)$ applied to graph $\mathcal{A}$ yield the set of nodes $\mathrm{V}(\mathcal{A})$ and the set of edges $\mathrm{E}(\mathcal{A})$ respectively. We mostly address sets $\mathrm{V}(\mathcal{A})$ and $\mathrm{E}(\mathcal{A})$ by $\mathcal{G}_{\mathcal{A}}$ and $\mathcal{E}_{\mathcal{A}}$, respectively, for the sake of notational convenience, i.e., $\mathcal{A} = (\mathcal{G}_{\mathcal{A}}, \mathcal{E}_{\mathcal{A}})$. Assume that there is a path P from node $a_i \in \mathcal{G}_{\mathcal{A}}$ to node $a_j \in \mathcal{G}_{\mathcal{A}}$ in graph $\mathcal{A}$, i.e., a directed connection from node $a_i \in \mathcal{G}_{\mathcal{A}}$ to node $a_j \in \mathcal{G}_{\mathcal{A}}$. Then, path P is described by the concatenation of nodes being passed through on the way from node $a_i \in \mathcal{G}_{\mathcal{A}}$ to node $a_j \in \mathcal{G}_{\mathcal{A}}$, i.e., $\mathrm{P} = a_i ... a_j$ and $\mathrm{V}(\mathrm{P}) = \left\{a_i, ..., a_j\right\}$ denotes the set of nodes of path P [16].

The functional dependencies among a set of genes $\mathcal{G} = \{g_1, ..., g_G\}$, with $G = |\mathcal{G}|$ elements, for a given cell process and specie can be characterized by a genetic interaction map (GI map,[17–20]) which is essentially a DAG with a common root node, i.e., the reporter level R, [21]. In particular, an arbitrary DAG $\mathcal{D}$ can be described as a graph $\mathcal{D} = (\mathcal{G}_{\mathcal{D}}, \mathcal{E}_{\mathcal{D}})$ with the set of nodes $\mathcal{G}_{\mathcal{D}} = \{\mathcal{G} \cup R\}$ and the set of directed edges $\mathcal{E}_{\mathcal{D}} = \{\{g_i, g_j\}, ..., \{g_j, g_l\}\}$. As the genetic interactions can only be observed through the reporter, all edges are always orientated in such a way that each path parting from any arbitrary gene $g_i \in \mathcal{G}$ always terminates in the root node R and any gene appears on the path at most once, i.e., there exist no cycles in the graph. Hence, the DAG $\mathcal{D}$ is always connected via its root node R. For the sake of notational convenience, in most cases, we write gene i when addressing gene g_i, [21]. The reporter node R is an artificial node, i.e., not a gene, in the concept of a DAG and represents the measured phenotype of the specific cell process under study.

To provide a better understanding of the information encoded in a DAG, we state a simple example, which is similar to the one in [21], based on DAG $\mathcal{D}_0$ displayed in Fig. 1. In $\mathcal{D}_0$, there exists an direct edge from gene i_0 to gene j_0, i.e., $\{i_0, j_0\} \in \mathcal{E}_{\mathcal{D}_0}$, which indicates that the activity of gene i_0 controls the activity of gene j_0. Hence, gene i_0 only affects the phenotype via gene j_0 and not directly. We emphasize that in this model, the existence of edge $\{i_0, j_0\}$ in the DAG only describes the hierarchical functional dependency between genes i_0 and j_0 and not the quantitative effect of gene i_0 on gene j_0.

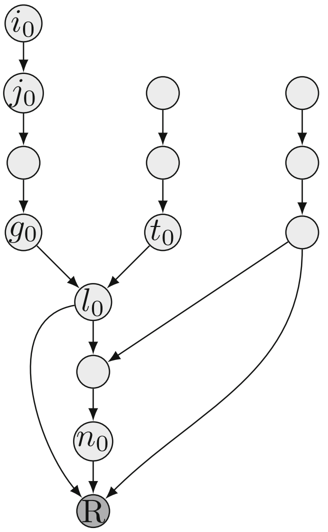

Fig. 1 DAG $\boldsymbol{\mathcal{D}}_0$ of 13 genes and root node R

2.2 Biological interaction model

Let us denote $R(i) \in \mathbb{R}$ as the phenotype for a single gene $i \in \mathcal{G}$ functionally disabled. In the same way, we define the phenotype for the DK of genes $i,j \in \mathcal{G}$ as $R(i,j) \in \mathbb{R}$. Let the datasets $\mathcal{R}_i = \{R(i,1), ..., R(i,G)\}$ and $\mathcal{R}_j = \{R(j,1), ..., R(j,G)\}$ contain all DK phenotypes involving genes $i,j \in \mathcal{G}$. The GI-profile data $\rho(i,j)$ for genes $i,j \in \mathcal{G}$ can be computed as the Pearson correlation between the samples of the datasets $\mathcal{R}_i$ and $\mathcal{R}_j$, respectively. We remark that the GI-profile data $\rho(i,j)$ does not have to be separately computed as the Pearson correlation of $\mathcal{R}_i$ and $\mathcal{R}_j$, respectively. It is commonly extracted from a database where a priori knowledge about the set of genes under study, i.e., $\mathcal{G}$, is stored. Since the gene pairs i,j and j,i are identical, it is sufficient to consider only gene pairs $i,j \in \mathcal{G} : j > i$. Throughout this paper, we mostly omit the specification that j is greater than i for notational convenience. In genomics research, it is a common assumption that if there is an edge between two genes i,j in DAG $\mathcal{D}$, i.e., there is an interaction between genes i,j in DAG $\mathcal{D}$, then the GI-profile $\rho(i,j)$ is very likely to be high. Furthermore, according to [2], we assume that each pair of genes i,j belongs to exactly one out of five hierarchical relationship classes that are characterized in Fig. 2. The hierarchical relationship classes $k \in \mathcal{K} = \{1, ..., 5\}$ are defined according to the model $\mu_k(i,j)$ in which the single-knockout phenotypes $R(i)$ and $R(j)$ are related with the DK phenotype $R(i,j)$. If the gene pair i,j belongs to the hierarchical relationship class k, then the observed DK phenotype $R(i,j)$ is described by the model $\mu_k(i,j)$ provided in Fig. 2. We remark that the five hierarchical dependency graphs in Fig. 2 do not reflect the absolute adjacency relations, but the hierarchical relations between genes i,j in DAG $\mathcal{D}$. Hence, given that two genes i,j of DAG $\mathcal{D}$ are in class k, we cannot conclude that genes i,j are directly arranged in DAG $\mathcal{D}$ as displayed by the depiction of class k in Fig. 2. This follows from the fact that the description of the hierarchical relationship classes provided in Fig. 2 only contains relative topology information about two genes i,j in DAG $\mathcal{D}$. In the following and in addition to [2], we provide, for clarity of presentation, a formal description of the hierarchical relationship classes depicted in Fig. 2 using a graph theoretical representation. Assume that there are I paths $\mathrm{P}_{i,\tau}, \mathrm{for}\tau \in \{1, ..., I\}$, from gene i to the reporter node R in DAG $\mathcal{D}$ and the set $\mathcal{P}_i$ containing all such paths is defined as $\mathcal{P}_i = \{\mathrm{P}_{i,1}, ..., \mathrm{P}_{i,I}\}$. Furthermore, set $\mathcal{P}_j = \{\mathrm{P}_{j,1}, ..., \mathrm{P}_{j,J}\}$ contains all J paths from gene j to the reporter node R in DAG $\mathcal{D}$. Given gene pair i,j in DAG $\mathcal{D}$, then pair i,j belongs to the hierarchical relationship class $k \in \mathcal{K}$ if and only if condition C_k as defined below is satisfied:

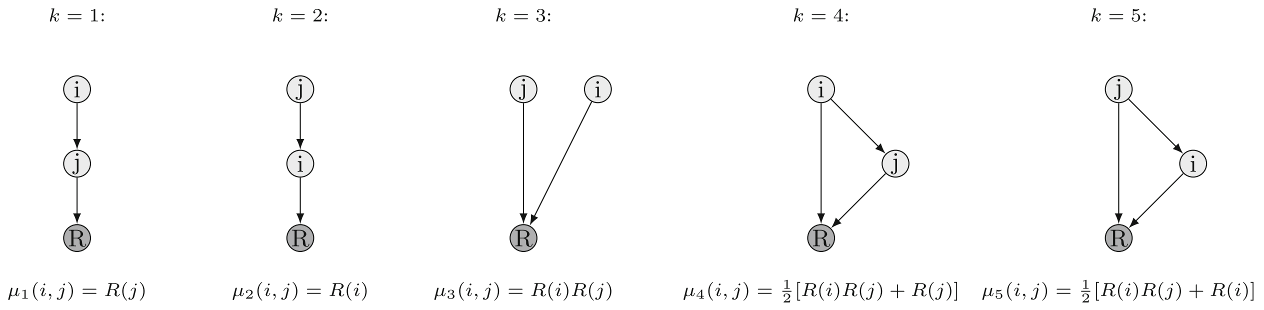

Fig. 2 Possible hierarchical relationship classes between two arbitrary genes *i*, *j* of DAG $\boldsymbol{\mathcal{D}}$ according to [2]

$$\begin{aligned} &\mathrm{C}_1 : \\ &\quad \forall\, \mathrm{P}_{i,\tau} \in \mathcal{P}_i : j \in \mathrm{V}(\mathrm{P}_{i,\tau}) \end{aligned} \tag{1a}$$

$$\begin{aligned} &\mathrm{C}_2 : \\ &\quad \forall\, \mathrm{P}_{j,\tau} \in \mathcal{P}_j : i \in \mathrm{V}(\mathrm{P}_{j,\tau}) \end{aligned} \tag{1b}$$

$$\begin{aligned} &\mathrm{C}_3 : \\ &\quad \big(\forall\, \mathrm{P}_{i,\tau} \in \mathcal{P}_i : j \notin \mathrm{V}(\mathrm{P}_{i,\tau})\big) \bigwedge \\ &\quad \big(\forall\, \mathrm{P}_{j,\tilde{\tau}} \in \mathcal{P}_j : i \notin \mathrm{V}(\mathrm{P}_{j,\tilde{\tau}})\big) \end{aligned} \tag{1c}$$

$$\begin{aligned} &\mathrm{C}_4 : \\ &\quad \big(\exists\, \mathrm{P}_{i,\tau} \in \mathcal{P}_i : j \notin \mathrm{V}(\mathrm{P}_{i,\tau})\big) \bigwedge \\ &\quad \big(\exists\, \mathrm{P}_{i,\tau} \in \mathcal{P}_i, \mathrm{P}_{j,\tilde{\tau}} \in \mathcal{P}_j : \mathrm{V}(\mathrm{P}_{j,\tilde{\tau}}) \subset \mathrm{V}(\mathrm{P}_{i,\tau})\big) \end{aligned} \tag{1d}$$

$$\begin{aligned} &\mathrm{C}_5 : \\ &\quad \big(\exists\, \mathrm{P}_{j,\tilde{\tau}} \in \mathcal{P}_j : i \notin \mathrm{V}(\mathrm{P}_{j,\tilde{\tau}})\big) \bigwedge \\ &\quad \big(\exists\, \mathrm{P}_{i,\tau} \in \mathcal{P}_i, \mathrm{P}_{j,\tilde{\tau}} \in \mathcal{P}_j : \mathrm{V}(\mathrm{P}_{i,\tau}) \subset \mathrm{V}(\mathrm{P}_{j,\tilde{\tau}})\big) \end{aligned} \tag{1e}$$

As stated in condition C_1 in (1a), two genes i, j in DAG $\mathcal{D}$ belong to the hierarchical relationship class $k = 1$, if all paths from gene i to the reporter node R pass through gene j. Hence, gene j is always an element of the set of nodes of each path $\mathrm{P}_{i,\tau} \in \mathcal{P}_i$ from gene i to the reporter node R, i.e., $j \in \mathrm{V}(\mathrm{P}_{i,\tau})$ for all paths $\mathrm{P}_{i,\tau}$ from gene i to the reporter node R. With the same line of argument as used in (1a), two genes i, j in DAG $\mathcal{D}$ belong to the hierarchical relationship class $k = 2$ if condition C_2 in (1b) is satisfied. Two genes i, j in DAG $\mathcal{D}$ belong to the hierarchical relationship class $k = 3$ and are considered to be independent from each other if condition C_3 in (1c) is satisfied which states that there is no path $\mathrm{P}_{i,\tau}$ from gene i to the reporter node R that passes through gene j as well as there is no path $\mathrm{P}_{j,\tilde{\tau}}$ from gene j to the reporter node R that passes through gene i. As stated in (1d), two genes i, j in DAG $\mathcal{D}$ belong to the hierarchical relationship class $k = 4$ if there is at least one path $\mathrm{P}_{i,\tau}$ from gene i to the reporter node R which does not pass through gene j as well as for all paths $\mathrm{P}_{j,\tilde{\tau}} \in \mathcal{P}_j$, there is always a path $\mathrm{P}_{i,\tau} \in \mathcal{P}_i$ that is a super-path of the respective $\mathrm{P}_{j,\tilde{\tau}} \in \mathcal{P}_j$. With the same line of argument as used in (1d), two genes i, j in DAG $\mathcal{D}$ belong to the hierarchical relationship class $k = 5$ if condition C_5 in (1e) is satisfied.

2.3 Class coupling—example

To illustrate this, let us consider the example DAG $\mathcal{D}_0$ of Fig. 1. All paths from gene i_0 to node R pass through gene j_0, i.e., they are in a linear pathway with gene i_0 upwards of gene j_0. Thus, the pair of genes i_0, j_0 belongs to class $k = 1$. Note that with the same line of argument, we conclude that also genes i_0 and l_0 belong to relationship class $k = 1$. Since all paths from gene i_0 to the reporter level R do not pass through gene t_0 and all paths from gene t_0 to the reporter level do not pass through gene i_0, genes i_0 and t_0 belong to the hierarchical relationship class $k = 3$ as given in Fig. 2, which states that genes i_0 and t_0 are independent of each other and the DK phenotype amounts to $R(i_0, t_0) = \mu_3(i_0, t_0)$. Finally, let us investigate the hierarchical relation between genes t_0 and n_0 in DAG $\mathcal{D}_0$. Obviously, gene t_0 has (at least) one path to node R which does not pass through gene n_0, i.e., genes only having paths to R that do not pass through gene n_0 do not affect the activity of gene n_0. Since there is (at least) one other path from gene t_0 to R passing through gene n_0, we can reason that genes t_0 and n_0 belong to class $k = 4$. Generally, there are strong implications among the hierarchical relationship classes of [2], i.e., if some pairs belong to a specific class, then this has strong implications for all other pairs. Let us consider the case that DAG $\mathcal{D}_0$ was not known and only the hierarchical relationship classes for genes i_0 and j_0, i.e., genes i_0 and j_0 belong to class $k = 1$, as well as the hierarchical relationship class for genes i_0 and g_0, i.e., genes i_0 and g_0 belong to class $k = 1$, were available. By definition of the hierarchical dependency graphs in Fig. 2 and the assumptions that genes i_0 and j_0 belong to class $k = 1$ as well as that genes i_0 and g_0 belong to class $k = 1$, we conclude that all paths from gene i_0 to R pass through genes j_0 and g_0. Thus, either all paths from gene g_0 to R pass through gene j_0 or all paths from gene j_0 to R pass through gene g_0. Consequently, genes j_0 and g_0 either belong to the hierarchical relationship class $k = 1$, or $k = 2$.

As we have emphasized by the example above, generally, if the hierarchical relationship class is known for two arbitrary genes i, j as well as for another pair $i, l \in \mathcal{G} : l > i$, then this has strong logical implications on the hierarchical relationship classes genes $j, l \in \mathcal{G} : l \quad > j$ can belong to. Since we can interpret the classification of the pairs of genes i, j, based on their observed SK and DK phenotypes $R(i), R(j)$ and $R(i, j)$, respectively, to exactly one out of the five hierarchical relationship classes as a coupled multi-hypothesis test, we address this problem in Section 3 by a linear integer optimization program. The proposed linear integer optimization program identifies the most consistent set of hierarchical relationship classes, i.e., the set of hierarchical relationship classes that represents a valid DAG and matches best the DK measurements with respect to the logical coupling between the classes. Furthermore, in Section 4, we extend the GENIE program proposed in Section 3 by incorporating GI-profile data in order to jointly detect the most consistent set of hierarchical relationship classes and the corresponding DAG topology.

3 GENIE algorithm

In this section, we formulate the problem of classifying the gene pairs i, j into the classes of hierarchical relationships based on the observed SK and DK phenotype values as a linear integer optimization program. Furthermore, we translate the logical implications among the hierarchical relationship classes into constraints that ensure that the detected set of hierarchical relationship classes represents a valid graph. That is, the detected set of hierarchical relationship classes represents a graph which is a DAG as defined in Section 2. Finally, we propose a policy to derive an estimate $\hat{\mathcal{E}}_{\mathcal{D}}$ of the true set of edges $\mathcal{E}_{\mathcal{D}}$ of DAG $\mathcal{D}$ based on the detected set of hierarchical relationship classes.

3.1 Hierarchical relationship class detection

In order to quantify the mismatch between the measured DK phenotypes $R(i, j)$ and the phenotype model $\mu_k(i, j)$ of class $k \in \mathcal{K}$ according to Fig. 2, under the hypothesis that the gene pairs i, j belong to class k given their respective SK values, we propose a simple quadratic score [2, 21], as given in Eq. (2)

$$s_k(i, j) = \left(R(i, j) - \mu_k(i, j)\right)^2, \quad k \in \mathcal{K} \qquad \forall i, j :\in \mathcal{G} : j > i \tag{2}$$

Let us define the following class-selection variables[1]

$$\alpha_k(i, j) = \begin{cases} 1 & \text{if } i, j \text{ are in class } k \\ 0 & \text{else} \end{cases} \qquad k \in \mathcal{K}, \quad \forall i, j :\in \mathcal{G} : j > i \tag{3}$$

We remark that every DAG $\mathcal{D}$ can be represented by a set of hierarchical relationship classes which directly corresponds to a set of class-selection variables $A^{\mathcal{D}} = \bigcup_{\forall i,j \in \mathcal{G}: j > i} \left\{\alpha_1^{\mathcal{D}}(i, j), ..., \alpha_5^{\mathcal{D}}(i, j)\right\}$. The GENIE algorithm of classifying the gene pairs i, j into the set of hierarchical relationship classes that represents a valid DAG and matches the observed SK and DK phenotypes best can be formulated as

$$\mathrm{O}_{\mathrm{GENIE}}:$$

$$\min_{\{\alpha_k(i,j)\}} \quad \sum_{i=1}^{G} \sum_{j=i+1}^{G} \left(\sum_{k=1}^{|\mathcal{K}|} s_k(i, j) \alpha_k(i, j) \right) \tag{4a}$$

$$\text{s. t.} \quad \alpha_k(i, j) \in \{0, 1\} \ \forall k \in \mathcal{K}, \quad \forall i, j \in \mathcal{G} : j > i \tag{4b}$$

$$\sum_{k=1}^{|\mathcal{K}|} \alpha_k(i, j) = 1, \quad \forall i, j \in \mathcal{G} : j > i \tag{4c}$$

$$\mathcal{L} \Longrightarrow \text{additional topology constraints} \tag{4d}$$

where $A^{\mathrm{O}_{\mathrm{GENIE}}} = \bigcup_{\forall i,j \in \mathcal{G}: j > i} \left\{\alpha_1^{\mathrm{O}_{\mathrm{GENIE}}}(i, j), ..., \alpha_5^{\mathrm{O}_{\mathrm{GENIE}}}(i, j)\right\}$ denotes the solution of program $\mathrm{O}_{\mathrm{GENIE}}$ in (4) and the set of best matching selection variables $A^{\mathrm{O}_{\mathrm{GENIE}}}$ corresponds to the most consistent pattern of hierarchical relationship classes. Problem $\mathrm{O}_{\mathrm{GENIE}}$ in (4) is a linear integer program which can be solved efficiently by BB methods [22–29]. The objective of problem $\mathrm{O}_{\mathrm{GENIE}}$ is to minimize the overall mismatch in classifying each gene pair i, j to one out of five hierarchical relationship classes. The constraints in (4b) reflect the binary nature of the class-selection variables $\alpha_k(i, j)$, $\forall k \in \mathcal{K}$, while (4c) represents a multiple choice constraint to enforce that the gene pairs i, j are only classified to one out of the five hierarchical relationship classes. The set $\mathcal{L}$ in (4d) is comprised of additional constraints to ensure that the detected set of selection variables $A^{\mathrm{O}_{\mathrm{GENIE}}}$ always represents a valid graph, i.e., a DAG. In the following, we exemplarily derive topology constraints in set $\mathcal{L}$. In order to identify the numerous logical dependencies among the class-selection variables $\alpha_k(i,j), k \in \mathcal{K}$ for all $i,j \in \mathcal{G} : j > i$, we proceed in the following way. We first fix the assumption that genes i,j belong to class $k = 1$, i.e., $\alpha_1(i,j) = 1$. Further, we assume that genes $i,l \in \mathcal{G} : l > i$ belong to class $k^{'}$, i.e., $\alpha_{k^{'}}(i,l) = 1$. Then, we derive the set of classes $\mathcal{K}^{''}$ that genes $j,l \in \mathcal{G} : l > j$ can belong to under the assumptions made. In the following, we have formulated the logical dependencies among the selection variables for $\alpha_1(i,j) = 1$, i.e., the case that gene i is linearly upstream of gene j, as linear integer inequalities defined in constraints (5a)–(5e) and summarize them as set $\mathcal{L}_1$

$$\mathcal{L}_1 = \Bigg\{$$

$$\begin{aligned}
&\alpha_1(j,l) + \alpha_2(j,l) \geq \alpha_1(i,j) + \alpha_1(i,l) - 1 && \text{(5a)}\\
&\alpha_2(j,l) \geq \alpha_1(i,j) + \alpha_2(i,l) - 1 && \text{(5b)}\\
&\alpha_2(j,l) + \alpha_3(j,l) + \alpha_5(j,l) \geq \\
&\alpha_1(i,j) + \alpha_3(i,l) - 1 && \text{(5c)}\\
&\alpha_2(j,l) + \alpha_4(j,l) \geq \alpha_1(i,j) + \alpha_4(i,l) - 1 && \text{(5d)}\\
&\alpha_5(j,l) + \alpha_2(j,l) \geq \alpha_1(i,j) + \alpha_5(i,l) - 1 && \text{(5e)}
\end{aligned}$$

$$\Bigg\} \forall i,j,l \in \mathcal{G} : l > j > i$$

where constraints (5a)–(5e) are linear after the continuous relaxation of the selection variables $\alpha_k(i,j)$. To explain the origin and the functionality of the constraints in $\mathcal{L}_1$, let us further define a sub-genetic interaction map (SMAP) $\mathcal{S}$, [21], as given in Fig. 3 according to the following definition where we adopt the graph notation of [16]:

Definition 1 *Given a non-empty set of edges $\mathcal{E}_{in}$ and a non-empty set of edges $\mathcal{E}_{out}$, graph $\mathcal{S} = (\mathcal{G}_{\mathcal{S}}, \mathcal{E}_{\mathcal{S}})$, with set of nodes $\mathcal{G}_{\mathcal{S}}$ and set of edges $\mathcal{E}_{\mathcal{S}}$, is a SMAP if the following conditions are fulfilled: (i) the graph $\mathcal{S}$ is acyclic and directed and (ii) there are $\exists e_{in} \in \mathcal{E}_{in}$ and $e_{out} \in \mathcal{E}_{out}$ such that each path P through graph $\mathcal{S}$ incides $\mathcal{S}$ via egde e_{in} and leaves graph $\mathcal{S}$ via edge e_{out}.*

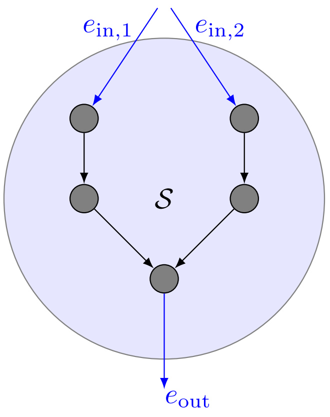

Fig. 3 Example SMAP $\mathcal{S}$

DAG $\mathcal{D}_1$, as displayed in Fig. 4, consists of genes i,j and SMAPs $\mathcal{S}_1$ and $\mathcal{S}_2$. Obviously, genes i,j belong to class $k = 1$, i.e., $\alpha_1(i,j) = 1$. Furthermore, all genes $l \in \mathcal{G}_{\mathcal{D}_1} \setminus \{R\} : l > j > i$ for which $\alpha_1(i,l) = 1$ must be either located in SMAP $\mathcal{S}_1$ or $\mathcal{S}_2$. Thus, it follows from DAG $\mathcal{D}_1$ in Fig. 4 that the gene pair j,l is either in hierarchical relationship class $k = 1$ or $k = 2$, i.e., $\alpha_1(j,l) = 1$ or $\alpha_2(j,l) = 1$.

This logical implication is directly reflected by constraint (5a). Given $\alpha_1(i,j) = 1$ and $\alpha_1(i,l) = 1$, the right-hand side (RHS) of (5a) amounts to 1. In this case also, the left-hand side (LHS) of (5a) becomes 1 to fulfill the inequality (5a). Thus, either $\alpha_1(j,l) = 1$ or $\alpha_2(j,l) = 1$. Reversely, assume that $\alpha_1(i,j) = 1$ and $\alpha_1(i,l) = 1$ does not hold, and then, the RHS of (5a) is less than 1, i.e., 0 or -1, while the LHS of (5a) is always greater than 0. Hence, constraint (5a) is fulfilled irrespectively of the choice of $\alpha_k(j,l)$, i.e., constraint (5a) enforces no logical implications.

Similarly for DAG $\mathcal{D}_2$ in Fig. 4, it is obvious that genes i,j belong to the hierarchical relationship class $k = 1$, i.e., $\alpha_1(i,j) = 1$. All genes $l \in \mathcal{G}_{\mathcal{D}_2} \setminus \{R\} : l > j > i$ which are in a linear pathway upstream of gene i, i.e., $\alpha_2(i,l) = 1$, must be located in SMAP $\mathcal{S}_3$. Hence, it directly follows from DAG $\mathcal{D}_2$ that also, gene l must be in a linear pathway upstream of gene j, i.e., $\alpha_2(j,l) = 1$. This logical implication is compactly represented in constraint (5b). Under the assumption that $\alpha_1(i,j) = 1$ and $\alpha_2(i,l) = 1$, the RHS of (5b) amounts to 1 enforcing $\alpha_2(j,l) = 1$, so that the LHS of (5b) equals the RHS and the inequality in (5b) is fulfilled. Reversely, assume that $\alpha_2(i,l) = 0$, then the RHS of (5b) is less than 1 and hence the LHS of (5b) is always bigger than or equal to the RHS irrespectively of the choice of $\alpha_k(j,l)$, i.e., constraint (5a) enforces no logical implications. We can proceed in the same fashion to explain constraints (5c)–(5e) based on DAGs $\mathcal{D}_3$ to $\mathcal{D}_5$ as given in Fig. 4, respectively. Note that the DAGs $\mathcal{D}_1$ to $\mathcal{D}_5$ are sufficient illustrations of Eqs. (5a)– (5e) in order to derive all logical implications for the case that genes i,j are in class 1, i.e., $\alpha_1(i,j) = 1$. In the case of DAG $\mathcal{D}_1$, for instance, there can be other DAGs than DAG $\mathcal{D}_1$ indeed where genes i,j are in class 1 and genes i,l are in class 1, i.e. $\alpha_1(i,j) = 1$ and $\alpha_1(i,l) = 1$. However, DAG $\mathcal{D}_1$ contains all the necessary information in order to derive the logical implications for gene pair j,l, given that $\alpha_1(i,j) = 1$ and $\alpha_1(i,l) = 1$. The same holds for DAGs $\mathcal{D}_2$ to $\mathcal{D}_5$. Furthermore, with the same line of argument, we can derive the sets $\mathcal{L}_k$ for $k \in \mathcal{K} \setminus 1$ which reflect the logical implications among the selection variables under the assumptions that $\alpha_k(i,j) = 1$

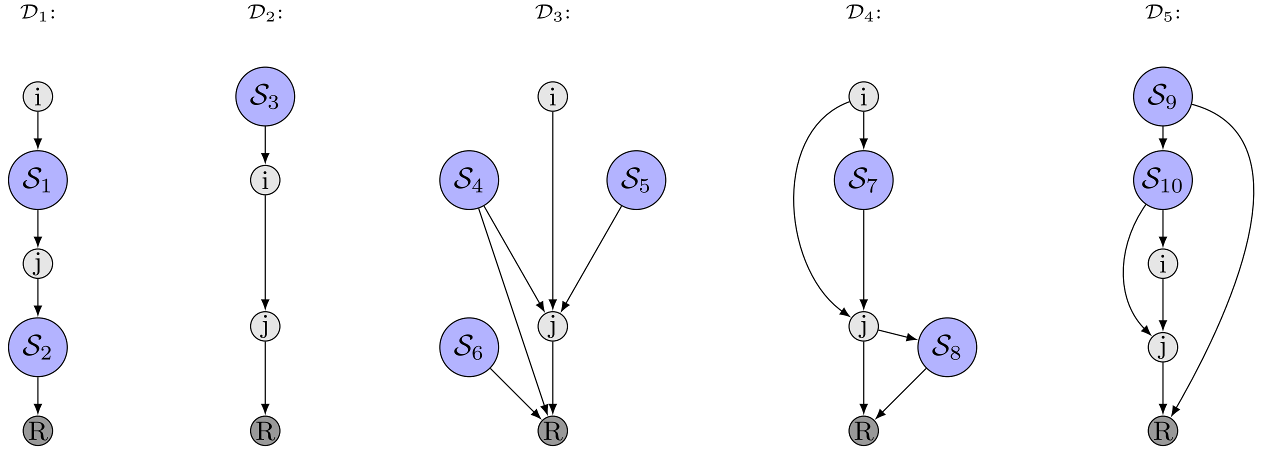

Fig. 4 Schematically reduced DAGs $\boldsymbol{\mathcal{D}}_1$ to $\boldsymbol{\mathcal{D}}_5$ corresponding to Eqs. (5a)–(5e), respectively

for $k \in \mathcal{K} \setminus 1$. However, due to space limitations, we omit the derivation of the full set of logical implications at this point and refer the interested reader to [30] where we will provide the full set of topology constraints $\mathcal{L}$ as well as further supplementary material. The full set of topology constraints $\mathcal{L}$ in (4d) can be computed as

$$\mathcal{L} = \bigcup_{k=1}^{|\mathcal{K}|} \{\mathcal{L}_k\}. \tag{6}$$

Finally, a considerable advantage of the presented algorithm is its ability to incorporate prior knowledge into the classification of the gene pairs i, j to the most consistent hierarchical relationship classes. Assume that it is known from existing experimental results that two specific genes $i_0, j_0 \in \mathcal{G} : j_0 > i_0$ do not interact with each other. Then, we can easily incorporate this prior knowledge into program $\mathrm{O}_{\mathrm{GENIE}}$ in (4) by adding Eq. (7) as defined below

$$\alpha_3(i_0, j_0) = 1 \tag{7}$$

as a topology constraint to program $\mathrm{O}_{\mathrm{GENIE}}$. This property is very valuable since it allows the GENIE algorithm to take advantage of existing results in genetic interaction research to improve the reliability of the classification.

3.2 Edge computation

Based on the detected set of selection variables $A^{\mathrm{O}_{\mathrm{GENIE}}}$ which corresponds to the most consistent pattern of hierarchical relationship classes given the observed SK and DK phenotypes, an estimate $\mathcal{E}_{\mathrm{GENIE}}$ of the true set of edges $\mathcal{E}_{\mathcal{D}}$ of DAG $\mathcal{D}$ can be computed. It can be theoretically proven that the representation of an arbitrary DAG $\mathcal{D}$ by its corresponding set of hierarchical relationship classes is not unique. $\mathrm{A}^{\mathcal{D}}$ the set of selection variables which directly corresponds to the hierarchical relationship class pattern of DAG $\mathcal{D}$ represents not only the true DAG $\mathcal{D}$ but also a set of similar DAGs which have minorly different sets of edges compared to the true DAG $\mathcal{D}$. Assume we are only given that $\alpha_4^{\mathcal{D}}(i,j) = 1$ for two arbitrary genes i, j of DAG $\mathcal{D}$, then we suffer an information loss on the number of paths from gene i to the reporter node R which are independent of gene j. Similarly, given that $\alpha_5^{\mathcal{D}}(i,j) = 1$ for two arbitrary genes $i, j \in \mathcal{G} : j > i$ of DAG $\mathcal{D}$, we suffer an information loss on the number of paths from gene j to the reporter node R which are independent of gene i. Hence, this information loss yields ambiguities in computing the set of edges $\mathcal{E}_{\mathcal{D}}$ of DAG $\mathcal{D}$ based on the $\mathrm{A}^{\mathcal{D}}$. In order to clarify the origin of the ambiguities further, let us turn to a simple example. Given DAG $\mathcal{D}_a = \left\{\mathcal{G}_{\mathcal{D}_a}, \mathcal{E}_{\mathcal{D}_a}\right\}$ as displayed on the LHS of Fig. 5 and the corresponding set of hierarchical relationship classes represented by the corresponding set of selection variables $\mathrm{A}^{\mathcal{D}_a}$. Note that $\alpha_4^{\mathcal{D}_a}(1,2) = \alpha_4^{\mathcal{D}_a}(1,3) = \alpha_4^{\mathcal{D}_a}(1,4) = 1$ due to edge $e_0 \in \mathcal{E}_{\mathcal{D}_a}$ and $\alpha_1^{\mathcal{D}_a}(1,5) = 1$. Now, assume that we want to compute the topology of DAG $\mathcal{D}_a$, i.e., the set of edges $\mathcal{E}_{\mathcal{D}_a}$, based on $\mathrm{A}^{\mathcal{D}_a}$. DAG $\hat{\mathcal{D}}_a$ displayed on the RHS of Fig. 5 shows the estimated topology of DAG $\mathcal{D}_a$ based on $\mathrm{A}^{\mathcal{D}_a}$. It can be shown that the black edges of DAG $\hat{\mathcal{D}}_a$ are necessary such that DAG $\hat{\mathcal{D}}_a$ is represented by $\mathrm{A}^{\mathcal{D}_a}$. Edges e_1 and e_2 in DAG $\hat{\mathcal{D}}_a$ are optional in a sense that their existence has no effect on set $\mathrm{A}^{\mathcal{D}_a}$. Edges e_1 and e_2 create two paths from gene g_1 to the reporter node R which are independent of gene g_2 and gene g_3, respectively. However, due to edge e_0, gene g_1 already has a path to the reporter node R which is independent of genes g_2 and g_3. Since $\alpha_4^{\mathcal{D}_a}(1,2) = \alpha_4^{\mathcal{D}_a}(1,3) = \alpha_4^{\mathcal{D}_a}(1,4) = 1$ do not contain information on the number of paths from gene g_1 to R that are independent of g_2, g_3 and g_4, edges e_1 and e_2 do not affect the pattern of hierarchical relationship classes representing DAG $\mathcal{D}_a$, i.e., $\mathrm{A}^{\mathcal{D}_a}$, and hence, this yields ambiguities in computing the topology of DAG $\mathcal{D}_a$ based on its corresponding set of selection variables $\mathrm{A}^{\mathcal{D}_a}$.

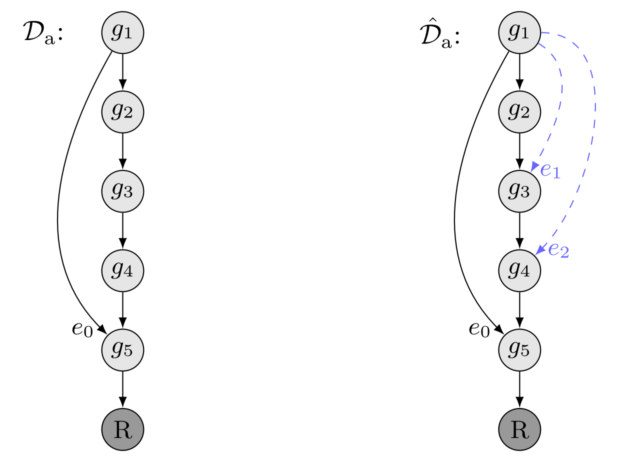

Fig. 5 Left: Original DAG $\boldsymbol{\mathcal{D}}_a$ with corresponding set of hierarchical relationship classes $A^{\boldsymbol{\mathcal{D}}_a}$. Right: Reconstruction $\hat{\boldsymbol{\mathcal{D}}}_a$ of DAG $\boldsymbol{\mathcal{D}}_a$ based on $A^{\boldsymbol{\mathcal{D}}_a}$

Table 1 Proposed sparse edge detection policy

Detection policy: compute the sparsest DAG in line with $\hat{\mathrm{A}}^{\boldsymbol{\mathcal{D}}}$
E_1:
$\left(\hat{\alpha}_1^{\boldsymbol{\mathcal{D}}}(i,j)=1\right)\bigwedge\left(\nexists l\in\boldsymbol{\mathcal{G}}\setminus(i,j):\ \hat{\alpha}_1^{\boldsymbol{\mathcal{D}}}(i,l)=1\bigwedge\hat{\alpha}_2^{\boldsymbol{\mathcal{D}}}(j,l)=1\right)$
• $\Longrightarrow$ there is an edge in DAG $\boldsymbol{\mathcal{D}}$ from gene i to gene j, i.e., $\{i,j\}$
E_2:
$\left(\hat{\alpha}_4^{\boldsymbol{\mathcal{D}}}(i,j)=1\right)\bigwedge\left(\nexists l\in\boldsymbol{\mathcal{G}}\setminus(i,j):\ \left(\hat{\alpha}_1^{\boldsymbol{\mathcal{D}}}(i,l)=1\bigvee\hat{\alpha}_4^{\boldsymbol{\mathcal{D}}}(i,l)=1\right)\bigwedge\left(\hat{\alpha}_2^{\boldsymbol{\mathcal{D}}}(j,l)=1\bigvee\hat{\alpha}_5^{\boldsymbol{\mathcal{D}}}(j,l)=1\right)\right)$
• $\Longrightarrow$ there is an edge in DAG $\boldsymbol{\mathcal{D}}$ from gene i to gene j, i.e., $\{i,j\}$
E_3:
$\left(\hat{\alpha}_2^{\boldsymbol{\mathcal{D}}}(i,j)=1\right)\bigwedge\left(\nexists l\in\boldsymbol{\mathcal{G}}\setminus(i,j):\ \hat{\alpha}_2^{\boldsymbol{\mathcal{D}}}(i,l)=1\bigwedge\hat{\alpha}_1^{\boldsymbol{\mathcal{D}}}(j,l)=1\right)$
• $\Longrightarrow$ there is an edge in DAG $\boldsymbol{\mathcal{D}}$ from gene j to gene i, i.e., $\{j,i\}$
E_4:
$\left(\hat{\alpha}_5^{\boldsymbol{\mathcal{D}}}(i,j)=1\right)\bigwedge\left(\nexists l\in\boldsymbol{\mathcal{G}}\setminus(i,j):\ \left(\hat{\alpha}_2^{\boldsymbol{\mathcal{D}}}(i,l)=1\bigvee\hat{\alpha}_5^{\boldsymbol{\mathcal{D}}}(i,l)=1\right)\bigwedge\left(\hat{\alpha}_1^{\boldsymbol{\mathcal{D}}}(j,l)=1\bigvee\hat{\alpha}_4^{\boldsymbol{\mathcal{D}}}(j,l)=1\right)\right)$
• $\Longrightarrow$ there is an edge in DAG $\boldsymbol{\mathcal{D}}$ from gene j to gene i, i.e.,$\{j,i\}$

Since it is a common assumption in genomics research that GI maps, i.e., DAGs, are not overly dense but rather sparse, we propose a policy which computes the sparsest DAG topology based on the detected pattern of hierarchical relationship classes. Given the detected pattern of hierarchical relationship classes of a DAG $\mathcal{D}$, i.e., $\hat{\mathrm{A}}^{\mathcal{D}} = \bigcup\limits_{\forall i,j\in\mathcal{G}:j>i}\left\{\hat{\alpha}_1^{\mathcal{D}}(i,j),...,\hat{\alpha}_5^{\mathcal{D}}(i,j)\right\}$, we compute an estimate $\hat{\mathcal{E}}_{\mathcal{D}}$ of the true topology set $\mathcal{E}_{\mathcal{D}}$ of DAG $\mathcal{D}$ according to the policy depicted in Table 1 where we make use of the symmetry properties $\hat{\alpha}_1^{\mathcal{D}}(i,j) = \hat{\alpha}_2^{\mathcal{D}}(j,i)$, $\hat{\alpha}_2^{\mathcal{D}}(i,j) = \hat{\alpha}_1^{\mathcal{D}}(j,i)$, $\hat{\alpha}_3^{\mathcal{D}}(i,j) = \hat{\alpha}_3^{\mathcal{D}}(j,i)$, $\hat{\alpha}_4^{\mathcal{D}}(i,j) = \hat{\alpha}_5^{\mathcal{D}}(j,i)$, and $\hat{\alpha}_5^{\mathcal{D}}(i,j) = \hat{\alpha}_4^{\mathcal{D}}(j,i)$. Note that we redundantly expand the set of detected class-selection variables $\hat{\alpha}_k(i,j)$ from all pairs $i,j\in\mathcal{G}:j>i$ to all pairs $i,j\in\mathcal{G}$ in order to obtain a compact formulation of the mutually exclusive conditions E_1 to E_4 as stated in Table 1.

Assume that either condition E_1 or condition E_2 is fulfilled, then we conclude that there is an edge from gene i to gene j in DAG $\mathcal{D}$. Given that either condition E_3 or condition E_4 is fulfilled, we conclude that there exists an edge from gene j to gene i in DAG $\mathcal{D}$. We remark that there cannot be an edge between two genes i,j if they are independent of each other, i.e., $\hat{\alpha}_3^{\mathcal{D}}(i,j) = 1$.

As described by E_1, there is an edge from gene i to gene j in DAG $\mathcal{D}$, if gene i is linearly upstream of gene j, i.e., $\hat{\alpha}_1^{\mathcal{D}}(i,j) = 1$, and there is no gene l in DAG $\mathcal{D}$ that is linearly downstream of gene i, i.e., $\hat{\alpha}_1^{\mathcal{D}}(i,l) = 1$, and linearly upstream of gene j, i.e., $\hat{\alpha}_2^{\mathcal{D}}(j,l) = 1$. According to condition E_2, there is an edge from gene i to gene j in DAG $\mathcal{D}$, if gene i is upstream of gene j with at least one path from gene i to R which is independent of gene j, and furthermore, there is no gene l in DAG $\mathcal{D}$ that is either linearly downstream of gene i or downstream of gene i with gene i having at least one path to R that is independent of l and neither gene l is linearly upstream of gene j nor gene l is upstream of gene j with an independent path to R. In order to elucidate the effect of condition E_2 onto the edge computation, we briefly turn to DAG $\hat{\mathcal{D}}_a$ in Fig. 5. Condition E_2 ensures that the optional edges e_1 and e_2 are not detected but only the necessary edges displayed in black color. We remark that conditions E_3 and E_4 can be elucidated by the same line of argument as used for conditions E_1 and E_2, but due to space limitations, we omit a detailed explanation at this point. Finally, we propose a condition from which all reporter node edges, i.e, all edges that connect gene $i\in\mathcal{G}$ with reporter node R in DAG $\mathcal{D}$, can be computed. Based on the detected set of hierarchical relationship classes, i.e., $\hat{\mathrm{A}}^{\mathcal{D}}$, we follow our policy of computing the necessary edges only. For clarity of presentation and notational compactness, we define set $\mathcal{M}_i$ as

$$\mathcal{M}_i = \left\{l\in\mathcal{G}\setminus i \middle|\ \hat{\alpha}_4^{\mathcal{D}}(i,l) = 1\right\} \qquad i = 1,...,G \tag{8}$$

containing all genes $l\in\mathcal{G}$ which are in class $k = 4$ with gene $i\in\mathcal{G}$, i.e., $\hat{\alpha}_4^{\mathcal{D}}(i,l) = 1$. Furthermore, we define set $\mathcal{M}_i^{'}$ as

$$\mathcal{M}_i^{'} = \left\{ l \in \mathcal{M}_i | \ \exists \tilde{l} \in \mathcal{M}_i \setminus l : \hat{\alpha}_3^{\mathcal{D}}(l, \tilde{l}) = 1 \right\} \qquad i = 1, ..., G \qquad (9)$$

which contains all genes l of set $\mathcal{M}_i$ that are independent of at least one other gene of set $\mathcal{M}_i$. Based on sets $\mathcal{M}_i$ and $\mathcal{M}_i^{'}$, we formulate condition E_R as stated in Table 2. We conclude that there is an edge from gene i to reporter node R in DAG $\mathcal{D}$, if condition E_R is fulfilled. Given that gene i is linearly upstream of at least a single gene l, i.e., $\hat{\alpha}_1^{\mathcal{D}}(i,l) = 1$, there cannot exist an edge from gene i to reporter node R in DAG $\mathcal{D}$, since all paths from gene i to R pass through at least one other gene l. Conversely, if there is no such gene l that $\hat{\alpha}_1^{\mathcal{D}}(i,l) = 1$, then the LHS of E_R as given in Table 2 is fulfilled. The RHS of E_R accounts for our policy of detecting sparse DAGs only and is fulfilled if either $\mathcal{M}_i$, $\mathcal{M}_i^{'}$, or $\mathcal{M}_i$ and $\mathcal{M}_i^{'}$ are empty. Note that given $\mathcal{M}_i = \emptyset$, it follows that $\mathcal{M}_i^{'} = \emptyset$ as well, whereas the opposite is not true. In order to explain the effect of the RHS of condition E_R in an intuitive manner, let us turn to DAG $\mathcal{D}_{\mathrm{R}}$ as displayed in Fig. 6. Assume that we are given the pattern of hierarchical relationship classes that corresponds to DAG $\mathcal{D}_{\mathrm{R}}$, i.e., $A^{\mathcal{D}_{\mathrm{R}}}$, and we want to compute all reporter node edges based on $A^{\mathcal{D}_{\mathrm{R}}}$, i.e., all edges that directly connect a gene in DAG $\mathcal{D}_{\mathrm{R}}$ with the reporter node R. Note that $\alpha_1^{\mathcal{D}_{\mathrm{R}}}(2,3) = 1$, $\alpha_5^{\mathcal{D}_{\mathrm{R}}}(2,4) = 1$, $\alpha_4^{\mathcal{D}_{\mathrm{R}}}(3,4) = 1$, and $\alpha_4^{\mathcal{D}_{\mathrm{R}}}(1,l) = 1 \ \forall l \in \mathcal{G}_{\mathcal{D}_R} \setminus \{g_1, R\}$. It can be shown that gene g_3 fulfills the LHS of E_R, i.e., there is no gene which is linearly downstream of g_3, and furthermore, $\mathcal{M}_3 = \{g_4\}$ and $\mathcal{M}_3^{'} = \emptyset$. Hence, condition E_R is fulfilled and edge e_n connecting g_3 and R is computed in DAG $\hat{\mathcal{D}}_{\mathrm{R}}$ that is the reconstruction of DAG $\mathcal{D}_{\mathrm{R}}$ based on $A^{\mathcal{D}_{\mathrm{R}}}$. Furthermore, for set $A^{\mathcal{D}_{\mathrm{R}}}$, the edge e_n in the reconstructed DAG $\hat{\mathcal{D}}_{\mathrm{R}}$ is necessary, since $\alpha_1^{\mathcal{D}_{\mathrm{R}}}(2,3) = 1$, $\alpha_5^{\mathcal{D}_{\mathrm{R}}}(2,4) = 1$, and $\alpha_4^{\mathcal{D}_{\mathrm{R}}}(3,4) = 1$. In contrast to e_n, edge e_o is not necessary for $A^{\mathcal{D}_{\mathrm{R}}}$ to represent $\hat{\mathcal{D}}_{\mathrm{R}}$, since $\alpha_4^{\mathcal{D}_{\mathrm{R}}}(1,l) = 1 \ \forall l \in \mathcal{G}_{\mathcal{D}_R} \setminus \{g_1, R\}$ irrespectively of edge e_o. Hence, e_o is not detected, since $\mathcal{M}_1 \neq \emptyset$ and $\mathcal{M}_1^{'} \neq \emptyset$.

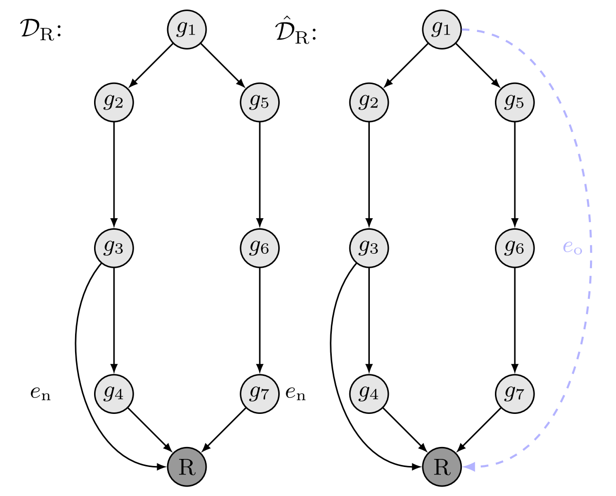

Fig. 6 Example DAG $\boldsymbol{\mathcal{D}}_{\mathrm{R}}$ to elucidate the functionality of the RHS of condition E_R of Table 2

We obtain an estimate $\mathcal{E}_{\text{GENIE}}$ of the true set of edges $\mathcal{E}_{\mathcal{D}}$ of DAG $\mathcal{D}$ by setting $\hat{\mathrm{A}}^{\mathcal{D}} = A^{\mathrm{O}_{\text{GENIE}}}$ and evaluating conditions E_1 to E_4 and condition E_R as stated in Tables 1 and 2, respectively.

4 GI-GENIE algorithm

In this section, we present the proposed GI-GENIE algorithm which jointly formulates the gene pair classification and the corresponding DAG topology estimation. Let us define the following edge-selection variables

$$\beta(i,j) = \begin{cases} 1 & \exists \text{ edge between } i,j \\ 0 & \text{no edge} \end{cases} \qquad \forall i,j \in \mathcal{G} : j > i \qquad (10)$$

where $\beta(i,j) = 1$ denotes that there is an edge between genes i,j in DAG $\mathcal{D}$ and $\beta(i,j) = 0$ denotes that there exists no edge between genes i and j. Note that unlike $\alpha_k(i,j) = 1$ for $k \in \mathcal{K}$, $\beta(i,j) = 1$ does not capture directionality information about the graph topology, i.e., $\beta(i,j) = 1$ states that there is an edge between genes i,j in DAG $\mathcal{D}$ without specifying the hierarchy among both genes. The topology $\mathcal{E}_{\mathcal{D}}$ of any DAG $\mathcal{D}$ can be represented by the corresponding set of class-selection variables $A^{\mathcal{D}} = \bigcup_{i,j} \left\{ \alpha_1^{\mathcal{D}}(i,j), ..., \alpha_5^{\mathcal{D}}(i,j) \right\}$ together with the corresponding set of undirected edges $\{\beta(i,j)\}$ for all $i,j \ :\in \mathcal{G} \ : j > i$. The set $\{\beta(i,j)\}$ can be viewed as the undirected "skeleton" of the DAG that is represented by its corresponding set of class-selection variables $A^{\mathcal{D}}$. The GI-GENIE algorithm yields an estimate $\mathcal{E}_{\text{GI}}$ of the true DAG topology $\mathcal{E}_{\mathcal{D}}$ by computing sets $\mathrm{A}^{\mathrm{O}_{\text{GI-GENIE}}}$ and $\left\{ \hat{\beta}(i,j) \right\}$ which are estimates of the true set of class-selection variables and edge-selection variables, $A^{\mathcal{D}}$ and $\{\beta(i,j)\}$, respectively. Based on SK, DK, and GI-profile data, the proposed GI-GENIE algorithm is formulated as the following LIP:

Table 2 Proposed reporter node edge detection policy

E_R:
$\underbrace{\left(\nexists l : \hat{\alpha}_1^{\boldsymbol{\mathcal{D}}}(i,l) = 1\right)}_{=\text{LHS}} \bigwedge \underbrace{\left(\boldsymbol{\mathcal{M}}_i = \emptyset \bigvee \boldsymbol{\mathcal{M}}_i^{'} = \emptyset\right)}_{=\text{RHS}}$

$$\begin{aligned}
\mathrm{O}_{\text{GI-GENIE}}: \\
\min_{\{\alpha_k(i,j),\beta(i,j),z_l(i,j)\}} \quad & \lambda_d \sum_{i=1}^{G}\sum_{j=i+1}^{G}\left(\sum_{k=1}^{|\mathcal{K}|} s_k(i,j)\alpha_k(i,j)\right) \\
& - \lambda_c \sum_{i=1}^{G}\sum_{j=i+1}^{G} \rho(i,j)\beta(i,j) \\
& + \lambda_p \sum_{i=1}^{G}\sum_{j=i+1}^{G} \beta(i,j) && \text{(11a)} \\
\text{s. t.: } & \text{Eqs. (4b)} - \text{(4d)} && \text{(11b)} \\
& \beta(i,j) \in \{0,1\} \\
& \forall i,j \in \mathcal{G} : j > i && \text{(11c)} \\
& z_l(i,j) \in \{0,1\}\ \forall l \in \mathcal{G} \setminus \left\{i,j\right\}, \\
& \qquad \forall i,j \in \mathcal{G} : j > i && \text{(11d)} \\
& 1 - \alpha_3(i,j) \geq \beta(i,j) \\
& \forall i,j \in \mathcal{G} : j > i && \text{(11e)} \\
& \mathcal{L}_c \Longrightarrow \text{ additional topology} \\
& \text{constraints} && \text{(11f)} \\
& |\mathcal{G}| - 2 + \beta(i,j) \geq \\
& 1 + \sum_{l \in \mathcal{G}\setminus\{i,j\}} z_l(i,j) && \text{(11g)} \\
& \forall i,j \in \mathcal{G} : j > i
\end{aligned}$$

where the scalars $\lambda_d, \lambda_s, \lambda_c,$ and λ_p are non-negative weighting constants to balance the impact of the SK and DK measurements and the GI-profile data, respectively, on the estimates. In particular, the parameter λ_d is used for dual purpose: (i) to scale the domain of the knockout scores $s_k(i,j)$ to the range $[0,...,1]$ which is comparable to range of the correlation data $\rho(i,j)$ and (ii) to trade-off the impact of the knockout scores $s_k(i,j)$ on the estimation outcome. The parameters λ_c and λ_p are in the interval $[0,1]$ where $\lambda_c \geq \lambda_p$. The GI-profile (GIP) term is given by

$$-\lambda_c \sum_{i=1}^{G}\sum_{j=i+1}^{G} \rho(i,j)\beta(i,j) + \lambda_p \sum_{i=1}^{G}\sum_{j=i+1}^{G} \beta(i,j). \qquad (12)$$

The quotient of $\frac{\lambda_c}{\lambda_p}$ defines the threshold for reward of the GI-profile (GIP) term in Eq. (12), where setting the edge selection variable $\beta(i,j) = 1$ is rewarded if the corresponding GI-profile measurement $\rho(i,j)$ is above the quotient $\frac{\lambda_c}{\lambda_p}$.

The auxiliary variables $z_l(i,j) \forall i,j,l \in \mathcal{G} : j > i, l \neq i, l \neq j$ are generally necessary to ensure that the information about the topology of DAG $\mathcal{D}$, which is encoded in the pattern of selection variables $\mathrm{A}^{\mathrm{O}_{\text{GI-GENIE}}}$ detected by program $\mathrm{O}_{\text{GI-GENIE}}$, is not contradicting with the set of edge selection variables $\left\{\hat{\beta}(i,j)\right\} \forall i,j \in \mathcal{G} : j > i$ detected by program $\mathrm{O}_{\text{GI-GENIE}}$. In particular, given that the detected pattern of selection variables $\mathrm{A}^{\mathrm{O}_{\text{GI-GENIE}}}$ enforces that there is an edge between genes i,j in DAG $\mathcal{D}$, then the auxiliary variables ensure that the corresponding edge selection variable indicates that there is an edge between genes i,j, i.e., $\hat{\beta}(i,j) = 1$. Furthermore, given that the detected pattern of selection variables $\mathrm{A}^{\mathrm{O}_{\text{GI-GENIE}}}$ enforces that there is no edge between genes i,j in DAG $\mathcal{D}$, then the auxiliary variables ensure that the corresponding edge selection variable indicates that there is no edge between genes i,j, i.e., $\hat{\beta}(i,j) = 0$. On the contrary, assume that the detected edge selection variables enforce that there is an edge between genes i,j in DAG $\mathcal{D}$, i.e., $\hat{\beta}(i,j) = 1$, then the $z_l(i,j)$ ensure that the detected pattern of selection variables $\mathrm{A}^{\mathrm{O}_{\text{GI-GENIE}}}$ must fulfill one of the conditions stated in Table 1. Consequently, given that the detected edge selection variables enforce that there is no edge between genes i,j in DAG $\mathcal{D}$, i.e., $\hat{\beta}(i,j) = 0$, then the $z_l(i,j)$ ensure that the detected pattern of selection variables $\mathrm{A}^{\mathrm{O}_{\text{GI-GENIE}}}$ does not fulfill any of the conditions stated in Table 1.

Furthermore, let the auxiliary parameters

$$q(i,j) = \begin{cases} 1 & \rho(i,j) \geq \frac{\lambda_c}{\lambda_p} \\ 0 & \rho(i,j) < \frac{\lambda_c}{\lambda_p} \end{cases} \qquad \forall\, i,j \in \mathcal{G} : j > i \qquad (13)$$

describe the detection of the edges of DAG $\mathcal{D}$ based on GI-profile data only, where $q(i,j) = 1$ denotes that there is an edge between genes i,j and $q(i,j) = 0$ denotes that there is no edge between genes i,j. Since any pattern of hierarchical relationship classes implies a specific set of edges and any set of edges implies a specific pattern of hierarchical relationship classes, there is a strong coupling of constraints, i.e., there are strong logical implications among the selection variables $\alpha_k(i,j)$ and the selection variables $\beta(i,j)$; the constraints in Eqs. (11e) to (11g) ensure that the detected hierarchical relationship classes and the corresponding edges, i.e., the $\alpha_k(i,j)$ and $\beta(i,j)$, are not mutually contradicting. Given that two genes i,j in DAG $\mathcal{D}$ are independent, i.e., $\alpha_3(i,j) = 1$, there cannot exist an edge between those genes in DAG $\mathcal{D}$, i.e., $\beta(i,j) = 0$. This logical implication is reflected by (11e). Set $\mathcal{L}_c$ in (11f) and the linear integer inequalities in (11g) model conditions E_1 to E_4 of our proposed edge detection policy as stated in Table 1. Since we do not want to redundantly expand the set of selection variables $\alpha_k(i,j)$ to all $i,j \in \mathcal{G} : j \neq i$ in order to not increase the complexity of program $\mathrm{O}_{\text{GI-GENIE}}$, we have to consider three cases when formulating conditions E_1 to E_4 of Table 1 as linear integer inequalities, i.e., $i,j,l \in \mathcal{G} : l > j > i$, $i,j,l \in \mathcal{G} : j > i > l$

and $i,j,l \in \mathcal{G} : j > l > i$. Then, the constraints in set $\mathcal{L}_{c,1}$

$$
\mathcal{L}_{c,1} = \Bigg\{
\begin{aligned}
&1 - \beta(i,j) \geq \alpha_1(i,j) + \alpha_1(i,l) + \alpha_2(j,l) - 2 \tag{14a}\\
&\frac{1}{2}\big(\alpha_1(i,l) + \alpha_2(j,l)\big) \geq \alpha_1(i,j) - z_l(i,j) \tag{14b}\\
&1 - \beta(i,j) \geq \alpha_2(i,j) + \alpha_2(i,l) + \alpha_1(j,l) - 2 \notag\\
&\frac{1}{2}\big(\alpha_2(i,l) + \alpha_1(j,l)\big) \geq \alpha_2(i,j) - z_l(i,j) \tag{14c}\\
&1 - \beta(i,j) + q(i,j) \geq \alpha_4(i,j) + \alpha_1(i,l) + \notag\\
&\alpha_4(i,l) + \alpha_2(j,l) + \alpha_5(j,l) - 2 \tag{14d}\\
&\frac{1}{2}\big(\alpha_1(i,l) + \alpha_4(i,l) + \alpha_2(j,l) + \alpha_5(j,l)\big) \geq \notag\\
&\alpha_4(i,j) - z_l(i,j) - q(i,j) \tag{14e}\\
&2 - \beta(i,j) - q(i,j) \geq \alpha_4(i,j) + \alpha_1(i,l) \notag\\
&+ \alpha_2(j,l) + \alpha_5(j,l) - 2 \tag{14f}\\
&\frac{1}{2}\big(\alpha_1(i,l) + \alpha_2(j,l)\big) \geq \notag\\
&\alpha_4(i,j) - z_l(i,j) - 1 + q(i,j) \tag{14g}\\
&1 - \beta(i,j) + q(i,j) \geq \alpha_5(i,j) + \alpha_2(i,l) + \notag\\
&\alpha_5(i,l) + \alpha_1(j,l) + \alpha_4(j,l) - 2 \notag\\
&\frac{1}{2}\big(\alpha_2(i,l) + \alpha_5(i,l) + \alpha_1(j,l) + \alpha_4(j,l)\big) \geq \notag\\
&\alpha_5(i,j) - z_l(i,j) - q(i,j) \tag{14h}\\
&2 - \beta(i,j) - q(i,j) \geq \alpha_5(i,j) + \alpha_2(i,l) \notag\\
&+ \alpha_5(i,l) + \alpha_1(j,l) - 2 \notag\\
&\frac{1}{2}\big(\alpha_2(i,l) + \alpha_1(j,l)\big) \geq \notag\\
&\alpha_5(i,j) - z_l(i,j) - 1 + q(i,j) \tag{14i}
\end{aligned}
\Bigg\} \forall i,j,l \in \mathcal{G} : l > j > i
$$

model the logical implications among the selection variables $\alpha_k(i,j), \alpha_{k'}(i,l), \alpha_{k''}(j,l)$, and $\beta(i,j)$ for $k, k', k'' \in \mathcal{K}, \forall i,j,l \in \mathcal{G} : l > j > i$. Together with (11g), constraints (14a)–(14b) model condition E_1 of our detection policy taking into account the GI-profile information $\rho(i,j)$ via selection variables $\beta(i,j)$. Assume that based on the SK and DK phenotypes, it is most consistent that $\alpha_1(i,j) = \alpha_1(i,l) = \alpha_2(j,l) = 1$ for at least one gene l in DAG $\mathcal{D}$ which corresponds to condition E_1 being violated. Hence, there cannot exist an edge between genes i and j in DAG $\mathcal{D}$. In this case, the RHS of (14a) amounts to 1 which enforces the LHS of (14a) to amount to 1 as well, i.e., $\beta(i,j) = 0$. Note that for $\alpha_1(i,j) = \alpha_1(i,l) = \alpha_2(j,l) = 1$, (14b) makes no restrictions on $z_l(i,j)$. Furthermore, assume that for genes i,j, based on the SK and DK phenotypes, it is most consistent that $\alpha_1(i,j) = 1$, but $\alpha_1(i,l)$ and $\alpha_2(j,l)$ are not jointly 1 for all other genes $l \in \mathcal{G} : l > j > i$, i.e., $\alpha_1(i,l) + \alpha_1(j,l) < 2$, then there is an edge between genes i,j in DAG $\mathcal{D}$ according to condition E_1. In this case, it is obvious that (14a) is always fulfilled, i.e., there are no restrictions on $\beta(i,j)$ by (14a). Since $\alpha_1(i,j) = 1$ and $\alpha_1(i,l) + \alpha_2(j,l) \leq 1$ for all $l \in \mathcal{G} : l > j > i$, constraint (14b) can only be fulfilled if $z_l(i,j) = 1 \; \forall l \in \mathcal{G} : l > j > i$. Hence, this enforces $\beta(i,j) = 1$ due to constraint (11g). In this case, constraint (14b) forces $z_l(i,j) = 1 \; \forall l \in \mathcal{G} : l > j > i$. Hence, given that $z_l(i,j) = 1 \; \forall l \in \mathcal{G} : l > j > i$, constraint (11g) sets $\beta(i,j) = 1$.

Given that the GI-profile data strongly supports that there is no edge between genes i,j in DAG $\mathcal{D}$, i.e., $\beta(i,j) = 0$, and $\alpha_1(i,j) = 1$ is most consistent based on the SK and DK phenotypes measured, then it follows from (11g) that there must be at least one $l \in \mathcal{G} : l > j > i$ for which $z_l(i,j) = 0$. In this case, with $\beta(i,j) = 0$, $\alpha_1(i,j) = 1$, and $z_l(i,j) = 0$, the RHS of (14b) amounts to 1, forcing the LHS of (14b) to amount to 1 as well, i.e., $\alpha_1(i,l) = 1$ and $\alpha_2(j,l) = 1$, which is together with the assumption of $\alpha_1(i,j) = 1$ a combination that violates the existence of a direct edge between genes i and j. Furthermore, note that (14a) does not have any implications on the selection variables $\alpha_1(i,j), \alpha_1(i,l)$, and $\alpha_2(j,l)$ for the case that $\beta(i,j) = 0$ and $z_l(i,j) = 0$.

Assume that the GI-profile data strongly supports that there is an edge between genes i,j in DAG $\mathcal{D}$, i.e., $\beta(i,j) = 1$, and $\alpha_1(i,j) = 1$ is most consistent based on the SK and DK phenotypes measured, then according to (14a), there cannot be any gene $l \in \mathcal{G} : l > j > i$ for which $\alpha_1(i,l) = 1$ and $\alpha_2(j,l) = 1$. Hence, Eq. (14b) can only be fulfilled if $z_l(i,j) = 1 \; \forall l \in \mathcal{G} : l > j > i$. Thus, (11g) is fulfilled with equality. We remark that given $\alpha_1(i,j) = 1$, constraints (14c) to (14i) are always fulfilled, i.e., they do not pose any implications among the selection variables $\alpha_k(i,j)$ and $\beta(i,j)$. Together with (11g), the two inequalities in (14c) model condition E_3 where we can elucidate their functionality in the same fashion as before. Constraints (14d) to (14g) along with (11g) model a minor modification of condition E_2 where we detect not only all necessary edges but also optional edges given that their existence is strongly supported by the GI-profile. Given that the existence of an edge between genes i,j in DAG $\mathcal{D}$ is not strongly supported by the GI-profile, i.e., $q(i,j) = 0$, constraints (14d) to (14e) along with (11g) model condition

E_2 which only allows necessary edges to be detected and we can elucidate their functionality in the same fashion as in (14a) to (14b). Note that (14f) to (14g) are always fulfilled for $q(i,j) = 0$, i.e., no implications among the selection variables $\alpha_k(i,j)$ and $\beta(i,j)$ are posed. Assuming that the existence of an edge between genes i,j in DAG $\mathcal{D}$ is strongly supported by the GI-profile, i.e., $q(i,j) = 1$, then the constraints in (14d) and (14e) are always fulfilled, i.e., no implications among the selection variables $\alpha_k(i,j)$ and $\beta(i,j)$ are posed by (14d) and (14e). However, constraints (14f) and (14g) pose relaxed logical implications among the selection variables $\alpha_k(i,j)$ and $\beta(i,j)$ compared to constraints (14d) to (14e). Hence, given that $q(i,j) = 1$ and $\alpha_4(i,j) = 1$, an edge between genes i,j in DAG $\mathcal{D}$ is detected if it is allowed by the pattern of hierarchical relationship classes. Constraints (14h) to (14i) along with (11g) model a minor modification of condition E_4 where we detect not only all necessary edges but also optional edges given that their existence is strongly supported by the GI-profile. Furthermore, the functionality of constraints (14h) to (14i) can be explained with the same line of argument as used to elucidate constraints (14d) to (14g).

Denote $\mathcal{L}_{c,2}$ and $\mathcal{L}_{c,3}$ as the sets of topology constraints that model the logical coupling among the selection variables $\alpha_k(i,j), \alpha_{k'}(i,l), \alpha_{k''}(j,l)$, and $\beta(i,j)$ for $k,k',k'' \in \mathcal{K}$ and $i,j,l \in \mathcal{G} : j > i > l$ and $i,j,l \in \mathcal{G} : j > l > i$, respectively. Then, the full set of coupled constraints of the selection variables $\alpha_k(i,j)$ and $\beta(i,j)$ is given by

$$\mathcal{L}_c = \bigcup_{\kappa=1}^{3} \left\{\mathcal{L}_{c,\kappa}\right\} \tag{15}$$

where we again refer the interested reader to [30] for a detailed description of $\mathcal{L}_c$. We obtain an estimate $\mathcal{E}_{\mathrm{GI}}$ of the true set of edges $\mathcal{E}_{\mathcal{D}}$ of DAG $\mathcal{D}$ based on the computed set of edge selection variables $\left\{\hat{\beta}(i,j)\right\}$ of program $\mathrm{O}_{\text{GI-GENIE}}$ where we infer the directionality of the edges according to $\mathrm{A}^{\mathrm{O}_{\text{GI-GENIE}}}$. Note that all reporter node edges are computed according to our proposed reporter node edge detection policy as given in Table 2. Since the reporter node is an artificial node in the concept of a DAG, there is no GI-profile data $\rho(i,R)\,\forall i \in \mathcal{G}$ and thus, no edge selection variable $\beta(i,R)\,\forall i \in \mathcal{G}$ according to (10).

5 Sequential scalability technique

Due to the combinatorial nature of problems $\mathrm{O}_{\mathrm{GENIE}}$ and $\mathrm{O}_{\text{GI-GENIE}}$, the GENIE algorithm and GI-GENIE algorithm, respectively, cannot be applied to the data of large sets of genes $\mathcal{G}$, since the number of candidate solutions to problems $\mathrm{O}_{\mathrm{GENIE}}$ and $\mathrm{O}_{\text{GI-GENIE}}$, respectively, grows exponentially with the number of genes. In order to nevertheless obtain statistically significant statements about the interactions among genes in a large set of genes $\mathcal{G}$, we propose the sequential scalability (SEQSCA) technique which is based on the GENIE algorithm and the GI-GENIE algorithm, respectively.

In particular, we create a sequence of S subsets $\{\mathcal{G}_s\}_1^S$ of the full set of genes $\mathcal{G}$, i.e., $\mathcal{G}_s \subset \mathcal{G}$, and $\forall s \in \{1,...,S\}$, where we estimate the topology $\mathcal{E}_{\mathcal{D},s}$ of each DAG $\mathcal{D}_s$, underlying the data of the subset of genes $\mathcal{G}_s$, by the GENIE or GI-GENIE algorithm, respectively, in order to translate the estimated topology $\mathcal{E}_{\mathcal{D},s}$ of DAG $\mathcal{D}_s$ into the corresponding adjacency matrix $\boldsymbol{M}_s$ for each $s \in \{1,...,S\}$. Based on the computed sequence of adjacency matrices $\{\boldsymbol{M}_s\}_1^S$, we iteratively compute the reliability matrix $\boldsymbol{M} \in [0,1]^{N \times N}$ of the full set of genes $\mathcal{G}$ in such a way that each entry $[\boldsymbol{M}]_{i,j\in\mathcal{G}}$ describes the empirical probability of an edge to exist between genes $i,j \in \mathcal{G}$, i.e., the empirical probability that genes $i,j \in \mathcal{G}$ directly interact with each other, where a value of 0 means that there is an interaction between the considered pair of genes with probability 0 and a value of 1 means that the considered pair of genes interacts with probability 1.

In particular, in each iteration s, we consider a subset $\mathcal{G}_s$ of size $N_S \ll |\mathcal{G}|$ of the full set of genes $\mathcal{G}$, where each gene of $\mathcal{G}_s$ is selected from $\mathcal{G}$ without replacement with equal probability. Based on the selected subset $\mathcal{G}_s$, we compute in each iteration s an estimate $\mathcal{E}_{\mathcal{D},s}$ of the true topology of DAG $\mathcal{D}_s$, underlying the observed data of the genes in subset $\mathcal{G}_s$, by the GENIE or the GI-GENIE algorithm, respectively. Furthermore, the topology estimate $\mathcal{E}_{\mathcal{D},s}$ of DAG $\mathcal{D}_s$ is translated into the corresponding adjacency matrix $\boldsymbol{M}_s$. The update of the reliability matrix for iteration s is computed according to Eq. (16)

$$\left[\boldsymbol{M}^{(s+1)}\right]_{i,j} = \left[\boldsymbol{M}^{(s)}\right]_{i,j} + [\boldsymbol{M}_s]_{\kappa_i,\kappa_j} \quad \forall i,j \in \mathcal{G}_s \tag{16}$$

with $\boldsymbol{M}^{(s)}$ being the $N \times N$ reliability matrix at iteration s, $\kappa_i \in \{1,...,N_S\}\ \forall i \in \mathcal{G}_s, \cup_i \kappa_i = \{1,...,N_S\}$ and $\kappa_i < \kappa_j$ for all $i < j$. Finally, we obtain the reliability matrix $\boldsymbol{M}$ of the full set of genes $\mathcal{G}$ by normalizing each entry $\left[\boldsymbol{M}^{(S)}\right]_{i,j}$ $i,j \in \mathcal{G}$ by $n_{i,j}$ that is the frequency of how often detecting an edge between genes i and j has been considered. Note that the proposed SEQSCA technique does not intend to yield valid DAGs but to provide statistical statements to which empirical probability two genes interact with each other.

In Table 3, we have summarized the SEQSCA technique. Finally, by means of the SEQSCA technique, we are able to provide statistically significant statements about the interactions among the genes from a large set $\mathcal{G}$ by using the GENIE or GI-GENIE algorithm, respectively, in a sequential fashion.

Table 3 Summary of the proposed SEQSCA-algorithm

```
**Initialization:** $\boldsymbol{M}^{(0)} = \mathbf{0}_{N\times N}; \boldsymbol{M}_{s=0} = \mathbf{0}_{N_S\times N_S}$; frequency counter $n_{i,j}^{(0)} = 0$
**Repeat:**
1: Select subset $\boldsymbol{\mathcal{G}}_s$ of size $N_S$ from $\boldsymbol{\mathcal{G}}$; draw each gene from $\boldsymbol{\mathcal{G}}$ with equal probability without replacement
2: Update: $n_{i,j}^{(s+1)} = n_{i,j}^{(s)} + 1$ for all $i,j \in \boldsymbol{\mathcal{G}}_s$
3: Estimate the DAG topology $\boldsymbol{\mathcal{E}}_s$ of set $\boldsymbol{\mathcal{G}}_s$ using GENIE, GI-GENIE, respectively; $\Longrightarrow \boldsymbol{M}_s$
4: Update reliability matrix $\boldsymbol{M}^{(s)}$ according to Eq. (16)
7: Update iteration number: $s \leftarrow s + 1$
**Until:** $s = S$;
Set $[\boldsymbol{M}]_{i,j} = \left[\boldsymbol{M}^{(S)}\right]_{i,j} / n_{i,j}^{(S)}\ \forall i,j \in \boldsymbol{\mathcal{G}}$
```

6 Simulation results

In this section, we first demonstrate the performance of the GENIE algorithm and the GI-GENIE algorithm with respect to conventional techniques for simulated data and further provide statistically significant statements on the interactions among the genes from a large set of genes based on real data using the proposed SEQSCA technique. For the implementation of the proposed algorithms, we used the popular CVX interface [31] along with the well-known MOSEK solver [32].

6.1 Synthetic data results

We have generated the ideal SK phenotypes $R(i) \in \mathbb{R}$ for all $i \in \mathcal{G}$ as well as the ideal DK phenotypes $R(i,j) \in \mathbb{R}$ for all $i,j \in \mathcal{G} : j > i$ according to the model of [2] as displayed in Fig. 2, where we have corrupted the ideal SK and DK phenotypes by independently and identically distributed zero-mean Gaussian noise with variance σ^2. Furthermore, the GI-profile data $\rho(i,j) \forall i,j \in \mathcal{G} : j > i$ has been generated on the basis of the SK and DK phenotypes. We compare both the GENIE algorithm and the GI-GENIE algorithm with the well-known GI-profile approach [2, 33], where the Pearson correlation between the GI-profiles of genes i and j is computed and an edge in the DAG is detected if the Pearson correlation is above a pre-defined threshold t_{corr}, where the directionality is inferred from the selection variable $\alpha_k(i,j)$ corresponding to the least mismatch model $\mu_k(i,j)$. Furthermore, we compare our proposed methods with the solution of program O_{GENIE} without considering set $\mathcal{L}$ as a constraint, which means simply classifying each pair i,j to the least mismatch scoring hierarchical relationship class based on the SK and DK phenotypes $R(i)$ and $R(i,j)$, respectively, without ensuring that the resulting pattern of hierarchical relationship classes represents a valid DAG.

In order to limit the Monte Carlo simulation time, we consider a total of 10 genes amounting to 225 binary variables and 2670 constraints for the GENIE algorithm and 630 binary variables and 9645 constraints for the GI-GENIE algorithm, respectively. For the GENIE method without considering the consistency constraints in $\mathcal{L}$, we have 225 binary variables and 270 constraints. Since we infer the edge orientation for the Pearson correlation-based method from the least mismatch scoring model, i.e., from the GENIE method without considering the consistency constraints in $\mathcal{L}$, we have 270 binary variables and 270 constraints.

In Fig. 7, we display the false detection percentage of edges P_{ed} in the detected DAG normalized to the true number of edges $|\mathcal{E}_\mathcal{D}|$ as defined in Eq. (17) versus the SNR.

$$P_{\text{ed}} = \frac{\left|\left(\mathcal{E}_\mathcal{D} \bigcup \hat{\mathcal{E}}_\mathcal{D}\right) \setminus \mathcal{E}_\mathcal{D}\right|}{|\mathcal{E}_\mathcal{D}|} \tag{17}$$

In Fig. 8, we display the percentage of undetected edges P_{mis} in the detected DAG normalized to the true number of edges $|\mathcal{E}_\mathcal{D}|$ as defined in Eq. (18) versus the SNR, i.e.,

$$P_{\text{mis}} = \frac{\left|\left(\mathcal{E}_\mathcal{D} \bigcup \hat{\mathcal{E}}_\mathcal{D}\right) \setminus \hat{\mathcal{E}}_\mathcal{D}\right|}{|\mathcal{E}_\mathcal{D}|} \tag{18}$$

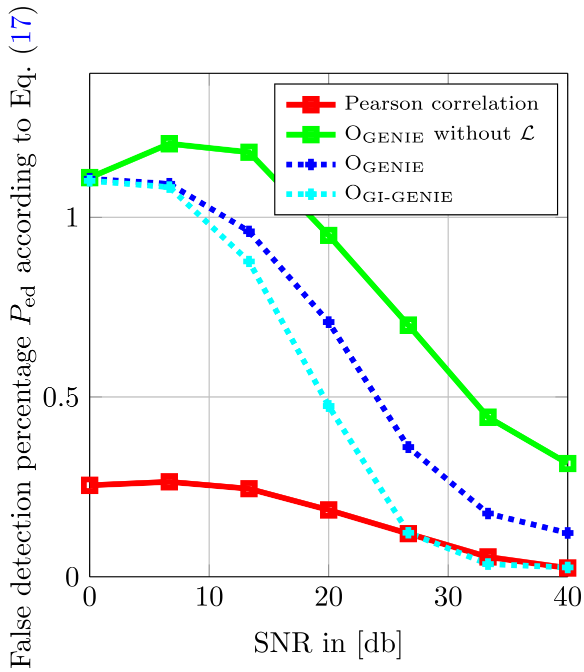

Fig. 7 D_{ed} versus SNR; $t_{\text{corr}} = 0.6$; 200 Monte Carlo runs; $\lambda_d = 0.05$, $\lambda_c = 1$, $\lambda_p = 0.8$

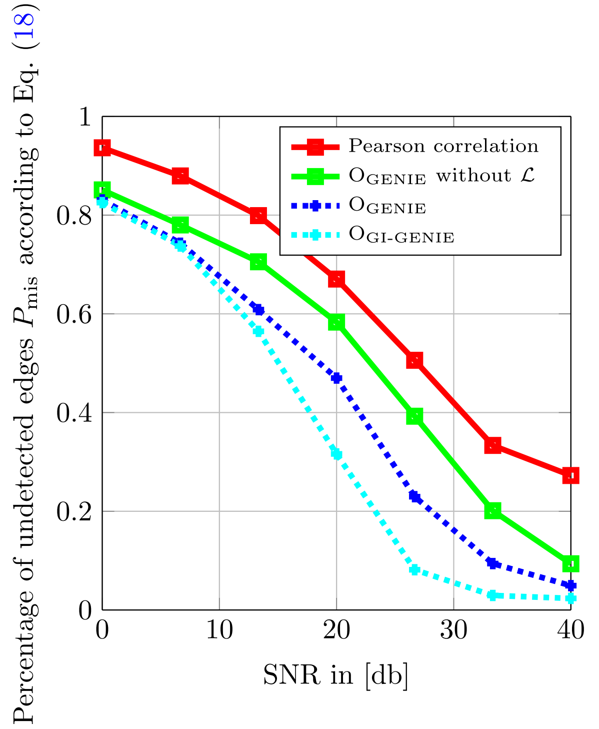

Fig. 8 D_{mis} versus SNR; $t_{\text{corr}} = 0.6$; 200 Monte Carlo runs; $\lambda_d = 0.05$, $\lambda_c = 1$, $\lambda_p = 0.8$

Note that in multi-hypothesis testing problems, it is common to view the diagnostic plots in Figs. 7 and 8 jointly to assess the quality of the proposed algorithms. In Fig. 7, we observe that in the low SNR regime, the Pearson correlation-based method performs best in terms of false detection percentage of edges P_{ed}; however, it fails to improve performance with increasing SNR, because for correct directionality information of the edges, this approach relies on the hierarchical relationship classes detected by method O_{GENIE} without considering $\mathcal{L}$. Especially in the high SNR regime, the proposed GENIE and GI-GENIE methods clearly outperform program O_{GENIE} without the topology rule set $\mathcal{L}$ and approach and respectively reach the performance of the Pearson correlation method. However, the very good performance of the Pearson correlation method in terms of false detection percentage of edges P_{ed} according to Eq. (17) comes at the cost of a rather poor performance in terms of the percentage of undetected edges P_{mis} according to Eq. (18) as can be seen in Fig. 8. In particular, in terms of the percentage of undetected edges P_{mis}, all of the proposed methods outperform the Pearson correlation method. Note that in the high SNR regime, the GI-GENIE of combining SK, DK, and GI-profile data yields the best of both worlds, i.e., it shows an equivalent performance as the Pearson correlation method in terms of false detection percentage of edges P_{ed}, as well as an improvement of the strong performance of the GENIE method in terms of the percentage of undetected edges P_{mis}.

6.2 Real data results

Since discovering genetic interaction maps, i.e., DAGs, for specific organisms is an ongoing field of research and the knowledge on genetic interactions is far away from being complete, there is generally no *ground truth* to directly compare with, even not for yeast which is one of the best understood organisms in this aspect. Therefore, we base the evaluation of the detection performance of the GENIE and the GI-GENIE methods on the biological knowledge that genetic interactions are generally rare and furthermore on the successful detection of known interactions provided by the well-known *yeast database* of [34]. We remark that to obtain statistically significant statements about large sets of genes, we have applied the proposed GENIE and GI-GENIE algorithms, respectively, along with the SEQSCA technique presented above. To demonstrate the benefit of using multiple data types instead of only one data type, we compare the reliability matrix results for SEQSCA and GI-GENIE with SEQSCA and GENIE which only utilizes SK/DK data. We have applied the abovementioned algorithms to the dataset reported in [35] to obtain the reliability matrices for the GENIE-based SEQSCA as well as for the GI-GENIE-based SEQSCA, $\boldsymbol{M}_{\text{G}}$ and $\boldsymbol{M}_{\text{GI}}$, respectively. The phenotypes reported in [35] are colony size measurements normalized to the wild-type size for each particular SK and DK, respectively. Typically, the colony size serves as a proxy for the fitness of the organism under study, which is the actual cell function of interest that cannot be observed. Therefore, the phenotypes in [35] are non-negative real numbers within the range $[0, C_{\text{max}}]$, where C_{max} denotes the maximum size dictated by the experiment setup. For computational reasons, we only considered the first 200 genes, i.e., $|\mathcal{G}| = 200$, of the *query gene list* of [35]. Figure 9 shows $\boldsymbol{M}_{\text{G}}$ obtained by the GENIE-based SEQSCA. In Fig. 10, we have displayed $\boldsymbol{M}_{\text{GI}}$ obtained by the GI-GENIE-based SEQSCA. For both results, we decomposed $\mathcal{G}$ into a sequence of $S = 5e4$ subsets $\mathcal{G}_s$ of equal size $N_s = 10$. In Fig. 9, 78% of the gene pairs i, j considered by $\boldsymbol{M}_{\text{G}}$ of the GENIE-based SEQSCA interact with each other with an empirical probability of less than 20%, i.e., $[\boldsymbol{M}_{\text{G}}]_{i,j} \leq .2$. Hence, the GENIE-based SEQSCA yields approximately sparse results. This is a good performance in terms of sparsity, since it is known from biology that genetic interactions are generally very rare. However, we can clearly see by the reliability matrix $\boldsymbol{M}_{\text{GI}}$ that the proposed GI-GENIE algorithm predicts genetic interactions with a much lower frequency as compared to the GENIE algorithm, which means a very good performance

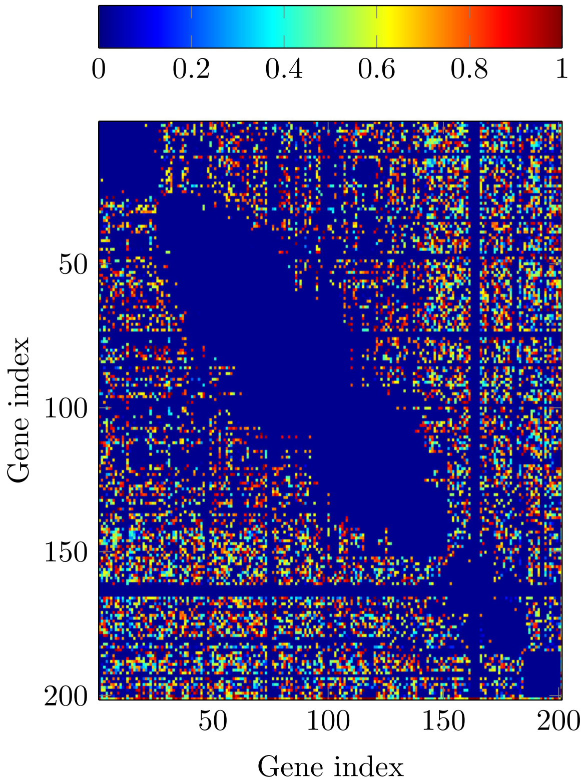

Fig. 9 Reliability matrix $\boldsymbol{M}_{\mathrm{G}}$; $S = 5e^4$ subsets considered; subset size $N_S = 10$

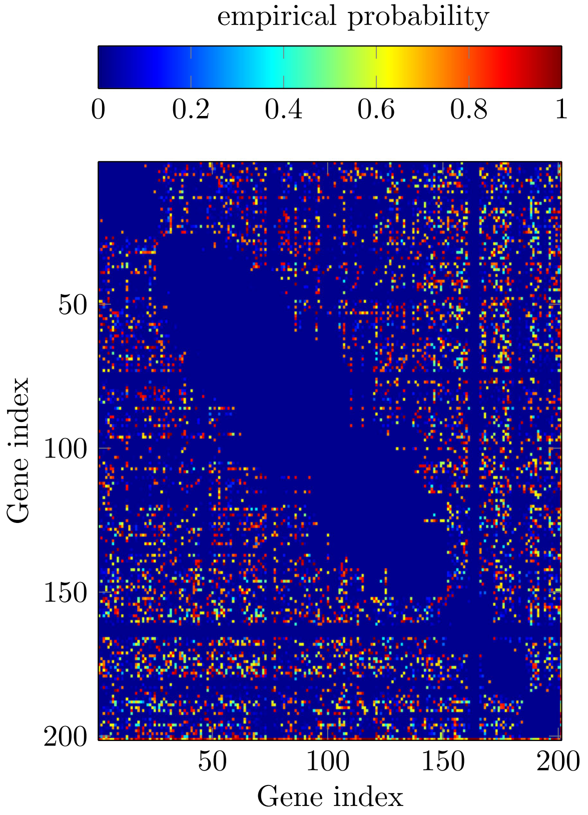

Fig. 10 Reliability matrix $\boldsymbol{M}_{\mathrm{GI}}$; $S = 5e^4$ subsets considered; subset size $N_S = 10$; $\lambda_d = 1e3, \lambda_c = 1, \lambda_p = 0.85$

in terms of sparsity. We have computed the acceptance ratio

$$\Gamma = \frac{N_{\mathrm{r}}}{N_{\mathrm{t}}} \tag{19}$$

where N_{r} is the number of interactions found with high significance ($[\boldsymbol{M}_{\mathrm{G}}]_{i,j}, [\boldsymbol{M}_{\mathrm{GI}}]_{i,j} \geq 1 - \epsilon$) and which are deposited in the database of [34] as well. N_{t} is the total number of highly significant interactions. Given the confirmed interactions at [34] for our set of genes under study, we remark that evaluating the number of confirmed interactions, that we have also found with our proposed method, would not be a reasonable performance metric, since [34] combines knowledge and experimental results of numerous sources. In contrast to that, we only had the dataset of [35] which only considers a particular phenotype, i.e., colony growth. As depicted in Table 4, we have computed Γ for both the GI-GENIE-based SEQSCA and the GENIE-based SEQSCA. It is obvious that the GI-GENIE-based SEQCA outperforms the GENIE-based SEQSCA, since the acceptance ratio for the GI-GENIE-based SEQSCA is significantly higher than the one of the GENIE-based SEQSCA.

7 Conclusions

In this paper, we have considered the problem of learning the interactions between genes in a genetic network. We have proposed the GENIE algorithm and the GI-GENIE algorithm to reconstruct the DAG underlying the observed data. The GENIE method is purely based on SK and DK data whereas the GI-GENIE method combines SK and DK data with GI-profile data in order to compute an estimate of the true DAG topology. In Section 5, we have presented the SEQSCA technique in order to obtain statistically significant statements about the interactions among a large set of genes under study. Furthermore, we have shown by simulations that the GI-GENIE algorithm outperforms the conventional techniques and the GENIE algorithm due to the combination of multiple data types, i.e., SK/DK and GI-profile data. Finally, based on the SEQSCA technique, we have presented real data results for the GENIE and the GI-GENIE algorithm, respectively,

Table 4 Acceptance ratios; $\epsilon = 0.05$

Method:	Γ (%)
SEQSCA and GENIE	53
SEQSCA and GI-GENIE	74

where we have confirmed that the GI-GENIE method outperforms the GENIE method.

Endnote

[1] In a discrete optimization context, the class-selection variables defined in (3) are denoted as SOS-1 type variables. However, for the sake of readability, we will mostly omit this optimization context-based annotation and refer to the variables defined in (3) as class-selection variables.

Authors' contributions

FN and MP contributed to the main idea, designed and implemented the proposed algorithms, carried out the simulations, and analyzed the results. FN wrote the initial draft of the paper. MP revised the paper. GK and NT brought in the biological background. All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

[1]Communication Systems Group, TU Darmstadt, Merckstr. 25, Darmstadt, Germany. [2]European Molecular Biology Laboratory, Heidelberg, Meyerhofstraße 1, 69117 Heidelberg, Germany.

Received: 7 September 2017 Accepted: 8 September 2017
Published online: 20 September 2017

References

1. A Shojaie, G Michailidis, Discovering graphical Granger causality using the truncating lasso penalty. **26 ECCB 2010**, i517–i523 (2010). Department of Statistics, University of Michigan, ECCB, Vol.26
2. A Battle, MC Jonikas, P Walter, JS Weissman, D Koller, Automated identification of pathways from quantitative genetic interaction data. Mol.Syst. Biol. **6**, 379–391 (2010)
3. AHY Tong, et al, Systematic genetic analysis with ordered arrays of yeast deletion mutants. Science. **294**, 2364–2368 (2001)
4. B Snijder, P Liberali, M Frechin, T Stoeger, L Pelkmans, Predicting functional gene interactions with the hierarchical interaction score. Nat. Methods. **10**(11), 1089–1094 (2013)
5. A Baryshinkova, et al, Quantitative analysis of fitness and genetic interactions in yeast on a genome scale. Nat. Methods. **7**, 1017–1024 (2010)
6. SR Collins, A Roguev, NJ Krogan, Quantitative genetic interaction mapping using the E-MAP approach. Methods Enzymol. **470**, 205–231 (2010)
7. RO Linden, VP Eronen, T Aittokallio, Quantitative maps of genetic interactions in yeast—comparative evaluation and integrative analysis. BMC Syst. Biol. **5**, 45–58 (2011)
8. SJ Dixon, M Constanzo, A Baryshinkova, B Andrews, C Boone, Systematic mapping of genetic interaction networks. Annu.Rev. Genet. **43**, 601–625 (2009)
9. GN Brock, et al, Methods for detecting gene gene interaction in multiplex extended pedigrees. BMC Genet. **6**, 144–149 (2005)
10. TC Hu, AB Kahng, *Linear and integer programming in practice*. (Springer International Publishing, Schweiz, 2016). ISBN-10: 3319239996
11. G Sierksma, *Linear and integer programming: theory and practice*, second edition. (CRC Press, Boca Raton, 2001). ISBN-10: 0824706730
12. G Sierksma, Y Zwols, *Linear and integer optimization: theory and practice*, third edition. (CRC Press, Boca Raton, 2015). ISBN-10: 1498710166
13. E Demirel, N Demirel, H Gökcen, A mixed integer linear programming model to optimize reverse logistics activities of end-of-life vehicles in Turkey. J. Clean. Prod. **112**, 1813–2144 (2016)
14. CH Antunes, MJ Alves, J Climaco, *Multiobjective linear and integer programming*. (Springer International Publishing, Schweiz, 2016). ISBN-13: 9783319287447
15. M Diaby, MH Karwan, *Advances in combinatorial optimization*. (World Scientific Publishing Co. Pte. Ltd., Singapore, 2016). ISBN-10: 9814704873
16. R Diestel, *Graphentheorie*. (Springer-Verlag, Heidelberg, 2012). ISBN 978-3-642-14911-5
17. A Jaimovich, et al, Modularity and directionality in genetic interaction maps. Nat. Methods. **26**, 38–45 (2010)
18. A Baryshinkova, M Constanzo, CL Myers, B Andrews, C Boone, Genetic interaction networks: toward an understanding of heritability. Annu.Rev. Genomics Hum. Genet. **14**, 111–133 (2013)
19. A Rogueav, et al, Quantitative genetic-interaction mapping in mammalian cells. Nat. Methods. **10**, 432–437 (2013)
20. M Constanzo, et al, The genetic landscape of a cell. Science. **327**, 425–431 (2010)
21. F Nikolay, M Pesavento, *Learning directed-acyclic-graphs from large-scale double-knockout experiments*. (C, Communications System Group, TU Darmstadt, EUSIPCO, 2016). Budapest, August – September 2016
22. V Balakrishnan, S Boyd, S Balemi, Branch and bound algorithm for computing the minimum stability degree of parameter-dependent linear systems. Int. J. Robust Nonlinear Control. **1**(4), 295–317 (1991)
23. EL Lawler, DE Wood, Branch-and-bound methods: a survey. Oper. Res. **14**, 699–719 (1966)
24. RE Moore, Global optimization to prescribed accuracy. Comput. Math. Appl. **21**(6/7), 25–39 (1991)
25. Y Cheng, M Pesavento, Joint rate adaptation and downlink beamforming using mixed integer conic programming. IEEE Trans. Signal Process. **63**, 1750–1764 (2013)
26. Y Cheng, M Pesavento, An optimal iterative algorithm for codebook-based downlink beamforming. IEEE Signal Process. Lett. **20**, 775–778 (2013)
27. Y Cheng, M Pesavento, Joint optimization of source power allocation and distributed relay beamforming in multiuser peer-to-peer relay networks. IEEE Trans. Signal Process. **60**(6), 2395–2404 (2012)
28. Y Cheng, M Pesavento, A Philipp, Joint network optimization and downlink beamforming for CoMP transmissions using mixed integer conic programming. IEEE Trans. Signal Process. **61**, 3972–3987 (2013)
29. CH Papadimitriou, K Steiglitz, *Combinatorial optimization: algorithms and complexity*. (Dover Publications, Mineola NY, 1998). ISBN 0486402584
30. Supplementary Material. https://www2.spg.tu-darmstadt.de/fnikolay/supp_journal.pdf
31. CVX – A Matlab based convex modeling framework. http://cvxr.com
32. MOSEK Solver. https://www.mosek.com/
33. M Babu, et al, Quantitative genome-wide genetic interaction screens reveal global epistatic relationships of protein complexes in Escherichia coli. PLoS Genet. **10**, 400–414 (2014)
34. SGD - Saccharomyces genome database. http://www.yeastgenome.org
35. M Costanzo, et al, DRYGIN - Data repository of yeast genetic interactions. Terence Donnelly Centre for Cellular and Biochemical Research, University of Toronto. http://drygin.ccbr.utoronto.ca/~costanzo2009/