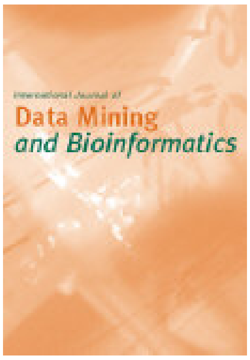


International Journal of Data Mining and Bioinformatics

ISSN online: 1748-5681 - ISSN print: 1748-5673
https://www.inderscience.com/ijdmb

Metaheuristic gene regulatory networks inference using discrete crow search algorithm and quantitative association rules

Makhlouf Ledmi, Mohammed El Habib Souidi, Aboubekeur Hamdi-Cherif, Abdeldjalil Ledmi, Hichem Haouassi, Chafia Kara-Mohamed

DOI: 10.1504/IJDMB.2025.10062651

Article History:

Received:	26 August 2023
Last revised:	11 December 2023
Accepted:	02 January 2024
Published online:	10 July 2025

Metaheuristic gene regulatory networks inference using discrete crow search algorithm and quantitative association rules

Makhlouf Ledmi* and
Mohammed El Habib Souidi

Department of Computer Sciences – ICOSI Lab,
Abbes Laghrour University of Khenchela,
Khenchela 40000, Algeria
Email: ledmi.makhlouf@univ-khenchela.dz
Email: souidi.mohammed@univ-khenchela.dz
*Corresponding author

Aboubekeur Hamdi-Cherif

Department of Computer Science,
Ferhat Abbas University of Setif 1,
Setif, 19000, Algeria
Email: elhamdi62@gmail.com

Abdeldjalil Ledmi and Hichem Haouassi

Department of Computer Sciences – ICOSI Lab,
Abbes Laghrour University of Khenchela,
Khenchela 40000, Algeria
Email: abdeldjalil.ledmi@univ-khenchela.dz
Email: haouassi.hichem@univ-khenchela.dz

Chafia Kara-Mohamed

Department of Computer Science – LRSD Lab,
Ferhat Abbas University of Setif 1,
Setif 19000, Algeria
Email: chafia.kara-mohamed@univ-setif.dz

Abstract: Gene regulatory networks (GRNs) inference appeared as valuable tools for detecting irregularities in cell regulation. Association rule mining (ARM) encompasses specific data mining methods capable of inferring unknown associations between genes. In response to the scarcity of ARM-based GRN inference, a novel metaheuristic algorithm, DCSA-QAR, is presented. This algorithm infers quantitative association rules by discretising the crow search algorithm. A first series of experiments involved comparison with five metaheuristic algorithms on six datasets. The results showed that, for Co-citation and YeastNet datasets, our algorithm was first in precision (100%), specificity (100%) and score (3.75). A second series of experiments involved nine information-theoretic algorithms through the DREAM3 and SOS networks. The average results on DREAM3 datasets are compensated by the SOS real datasets results: the best in accuracy, and true positives. As an overall appraisal, DCSA-QAR can be considered as a good candidate for ARM-based metaheuristic GRNs inference.

Keywords: artificial intelligence; bioinformatics; gene regulatory networks; GRNs; data mining; soft computing; mining association rules.

Reference to this paper should be made as follows: Ledmi, M., Souidi, M.E.H., Hamdi-Cherif, A., Ledmi, A., Haouassi, H. and Kara-Mohamed, C. (2025) 'Metaheuristic gene regulatory networks inference using discrete crow search algorithm and quantitative association rules', *Int. J. Data Mining and Bioinformatics*, Vol. 29, No. 3, pp.278–312.

Biographical notes: Makhlouf Ledmi received his BS from University of Annaba in 1997, MS from Abbes Laghrour University of Khenchela in 2010 and PhD from University of Batna 2, Algeria, in 2020, all in Computer Science. He served as the Head of Mathematics and Computer Science Department at Abbes Laghrour University of Khenchela, between 2015 and 2017. He is currently an Associate Professor at the same department and member of Knowledge Management Group ICOSI Lab. His research interests include data mining, soft computing, and bioinformatics.

Mohammed El Habib Souidi received his BS in Computer Science from Abbes Laghrour University of Khenchela (Algeria) in 2011. He also received his Master's in Computer Science from the same university in 2013, and his PhD in Computer Science from Harbin Institute of Technology (China) in 2017. He is working as an Associate Professor in the Department of Mathematics and Computer Science at Abbes Laghrour University of Khenchela (Algeria). Moreover, he is affiliated as a Researcher in ICOSI Lab at the same University. His research interests include multi-agent task coordination, reinforcement learning, game theory and path planning.

Aboubekeur Hamdi-Cherif received his BSc (Honours) in Electrical Engineering and MSc in Electronic Control Engineering, both from Salford University, England and PhD in Computer Science from Université Pierre & Marie Curie Paris 6, (UPMC), Paris, France. He subsequently worked as Control Engineer with Algerian Petroleum Company SONATRACH, at Ecole Supérieure des Transmissions, Algiers, at Université de Bretagne Occidentale (UBO), France, at Ecole Supérieure Libre des Sciences Commerciales Appliquées (ESLSCA), Paris, France. He was appointed Associate Professor in Computer Science at Ferhat Abbas University, Setif1, (UFAS), Algeria and later Professor in Computer Science and Deputy Head of Computer Science Department at Computer College, Qassim University, Saudi Arabia. His research interests are mainly in soft computing, machine learning, and bioinformatics.

Abdeldjalil Ledmi received his Engineer's degree in Computer Science from Abbes Laghrour University of Khenchela (Algeria) in 2010. He also received his Master's degree in Computer Science from the same university in 2014 and his PhD in Computer Science from Larbi Tebessi University. Tebessa (Algeria) in 2019. He is working as a Lecturer in the Department of

Mathematics and Computer Science at the University of khenchela (Algeria). Moreover, he is affiliated as a Researcher in ICOSI Lab at the same university. His research aims to understand fault tolerance in the distributed system and load balancing in cloud computing.

Hichem Haouassi is an Associate Professor at the Department of Computer sciences, Abbes Laghrour University of Khenchela (Algeria) since 2004. His main research interests are the artificial intelligence, data mining, methaheuristics and swarm-based optimisation, feature selection, and classification.

Chafia Kara-Mohamed received her BSc ('Diplôme d'Ingénieur d'Etat'), MSc ('Magister') and PhD ('Doctorat d'Etat'), all in Computer Science, from Ferhat Abbas University, Setif1, (UFAS), Houari Boumediene University of Science and Technology (USTHB), Algiers, and Ferhat Abbas University, Setif1 (UFAS), Algeria, respectively. She worked as an instructor at USTHB Computer Science Department and Lecturer at UFAS, and as Assistant Professor at Qassim University, Saudi Arabia. She also served as the Deputy Head of Computer Science Department, Female Section, Qassim University, Saudi Arabia. She is now an Associate Professor at UFAS. Her research interests are mainly in data mining, deep learning applications, Arabic text classification, and opinion mining.

1 Introduction

Gene regulatory networks (GRNs) represent a comprehensive framework for modelling the complex regulatory interactions among genes within the living cell and are now considered as an essential tool in bioinformatics and computational biology. Although gene expression is a complex, multifactorial process, it can be simplified and represented as a GRN. In this network, each node corresponds to a gene, and the directed edges signify either the activation or inhibition of a target gene. One of the main issues addressed by computational biology is to explain the dynamic behaviour of genes and how they are functionally related. As a result, GRNs inference, also known as GRNs reconstruction or reverse engineering, is a critical task as it facilitates the understanding of complex regulatory interactions among genes within biological systems. GRN inference can sustain biomedical research in identifying GRNs' behaviour and could assist in discovering irregularities in cell behaviour in addition to finding intricate mechanisms governing various cellular processes, including development, disease, response to environmental conditions, and death. The inference process and its relevant computational methods will help in leading to specific gene-targeting treatment, thus inducing a crucial impact on medicine and pharmacy; and public health at large (Vijesh et al., 2013).

Computationally speaking, GRN inference boils down to developing algorithms that can analyse cells' control and regulation of genes' expressions, on the basis of experimental conditions such as time series or steady-state information. The design of effective algorithms for the study of genetic data has received substantial attention due to the growing volume of biological data, boosted by high-throughput microarray technology, profoundly changing the genes processing methods. This innovative technology has shown its ability of concurrently controlling thousands of genes' expression; thus making GRNs inference from gene expression profiles readily available (Zhao et al., 2021). The principal computational issue resides in first finding the genes that are more relevant and second in identifying the regulatory relationships between them. Additionally, because of our actual lack of knowledge of the complexities of dynamic molecular networks, many GRNs are difficult to be accurately represented by any parsimonious modelling (Chai et al., 2014).

Despite the existence of innumerable algorithms for GRN inference based on many approaches, only very few considered the model-free approach supplemented by metaheuristic methods for GRN inference; hence our contribution. Among the model-free popular approaches that are successfully in use, we find data mining methods which are predominantly attractive because they offer a way to identify regulatory mechanisms directly from the input/output data without any apriori model construction. More specifically, we are interested in finding frequent patterns in genes' behaviour. These patterns play an essential role in many data mining tasks. They are used as a basis for mining interesting relationships in datasets, and consequently applied to association rules with a great degree of success.

Unlike classification, ARM is an unsupervised learning method. Indeed, the ARM approach is essentially characterised by its descriptive nature, as it identifies patterns that elucidate the data. In other words, it is employed to use the attributes of the data itself, as opposed to forecasting the class of unfamiliar data as proposed by the classification approach. The foundation of the ARM methodology primarily stems from the well-known market basket analysis concept. In this scenario, the main objective revolves around establishing patterns hidden within customer profiles regarding the concurrent purchase of products (Srikant and Agrawal, 1996).

Since the ARM approach is a highly abstract model-free technique, it only claims the least amount of data, with an important capability to achieve inferences, leading to GRNs inference. Additionally, the simplicity of the ARM approach allows the inference of large-size models with a higher speed of analysis. In our case, an ARM stands for inferring existing relationships among genes on the basis of a gene experimental database.

Furthermore, given that gene expressions are quantified through numerical values, we rely on a specialised form of association rules, known as quantitative association rule (QAR). In this framework, either the antecedent or the consequent must encompass a numerical attribute, reflecting the inherent nature of gene measurements (Zhu, 2009). The use of QAR as an inference method for GRNs is unfortunately not enough. This is so because as far as GRNs inference is concerned, not only do we need to generate sufficient rules, but more importantly, we need to obtain the best possible results regarding some chosen criteria; hence the use of a metaheuristic approach. This latter is an improved population-based derivative of the genetic algorithms' paradigm founded on randomly generating an initial population of solutions, then using ad hoc operators to curtail it, and finally keeping only the most suitable solutions according to a prescribed fitness function. The process is repeated until a prescribed threshold is attained, such as the number of iterations, or maximum runtime (Liu et al., 2020). Although metaheuristic methods sometimes require time-consuming parameter tuning and do not guarantee finding global optima, they nonetheless represent valuable tools in GRN inference due to their flexibility and ability to tackle complex problems as a whole. As an example of metaheuristic optimisation, we chose the crow search algorithm (CSA) because it has effectively tackled continuous problems and delivered remarkable outcomes. This approach draws inspiration from the behaviour of crows, which stash away surplus food in concealed locations and retrieve it as required. As a constraint and using an intelligent survival tactic, crows have to keep food in safe places in order not to be stolen by other crows (Askarzadeh, 2016).

Our proposal, subsequently called discrete crow search algorithm for mining quantitative association rules (DCSA-QAR), is a discrete version of the existing general purpose continuous CSA algorithm for mining quantitative association rules (Ledmi et al., 2020). The present paper contributes the following:

- The use of a model-free approach offered by ARM and its incorporation within a metaheuristic method for GRN inference.
- The discretisation of the CSA: this process involves employing the confidence-based unsupervised discretisation algorithm (C-BUDA), aiming to achieve the optimal split within a numerical attribute interval. This is accomplished by maximising class prediction.
- The undertaking of a first series of experiments involving the comparison of our method with metaheuristic algorithms using the following datasets and competitors:

  a Expression datasets were sampled and used: (Cho et al., 1998; Someren et al., 2000; Spellman et al., 1999), along with a subset of 20 well-described genes.

  b True networks (Lee et al., 2007; Dwight et al., 2002; Lee et al., 2004).

  c Competitors: comparison of our method is done against five state-of-the-art metaheuristic algorithms including preliminary parameter settings (Soinov et al., 2003; Nepomuceno-Chamorro et al., 2010; Gallo et al., 2011; Martínez-Ballesteros et al., 2014).

- The undertaking of a second series of experiments involving the comparison with other popular algorithms, such as information-theoretic ones using the following datasets and competitors:

  a Simulated dataset: dialogue for reverse engineering assessments and methods challenge (DREAM), namely DREAM3-10, DREAM3-50 (Marbach et al., 2010), and DREAM3-100 (Margolin et al., 2006)

  b Real dataset: SOS network dataset (Ronen et al., 2002).

  c Competitors: ARACNE (Basso et al., 2005), CLR (Faith et al., 2007), MI3 (Luo et al., 2008), MIDER (Villaverde et al., 2014), MRNET (Meyer et al., 2007), MRMSn (Liu et al., 2016), PCA-CMI (Zhang et al., 2011), RRMRNET (Liu et al., 2017), RWRNET (Liu et al., 2020).

As shown in the experiments, the overall performance of DCSA-QAR yielded encouraging outcomes. The rest of the paper is structured as follows: Section 2 describes related works. Section 3 describes the proposed method. Section 4 reports the materials and methods used. Section 5 describes the experiments with discussion of the results. Lastly, in Section 6, the key conclusions and prospects for future work are summarised.

2 Related work

In this section, we describe key contributions in GRN inference, concentrating on significant methods that carry widespread importance. We distinguish three types of methods: model-based, model-free and hybrid. Model-based methods incorporate Boolean networks, differential equations, and Bayesian networks. Model-free methods, also referred to as data-driven or similarity-based techniques, comprise correlation-based strategies, data mining approaches (such as ARM-based, employed in our method), machine learning techniques (including metaheuristic methods pivotal to our approach, as well as deep learning), and methods based on information theory. In addition to these two, hybrid methods have the objective of integrating multiple data sources by combining different types of omics data, such as gene expression, protein-protein interactions, and transcription factor binding data, to enhance accuracy. The choice of a given method depends on the available data, the size of the network, and the available computational resources. In the context of our work and for immediate relevance, we focus on the first two types of methods in our subsequent description.

2.1 Model-based methods

Among the earliest popular model-based methods, going as far back as to the late 1960s, we find the Boolean methods. In a Boolean network model, genes are designated as either active (1) or inactive (0). This feature allows for assessing the accuracy of inferred networks, especially when the underlying GRN is unknown, and only time series data is provided. Although approaches to Boolean inference typically excel in dynamic accuracy, they tend to exhibit a slightly lower performance in terms of structural correctness (Pušnik et al., 2022). In addition to these limitations, Boolean methods rely on arbitrary discretisations of gene expression values, imposing significant assumptions and constraints on the biological system under consideration. To overcome some of these challenges, Vengateshkumar et al. (2020) introduced a two-phase process called Boolean association rule mining (BAR) for inferring gene rules. In the first phase, BAR generates frequent gene sets by employing logical OR and AND operations. Subsequently, in the second phase, it uncovers Boolean gene association rules using logical AND and XOR operations. This approach aims to enhance the efficiency and effectiveness of gene rule extraction in the context of Boolean association rule mining. The BAR method suffers from the inherent binary representation that potentially overlooks nuanced patterns achievable with more varied representations. Moreover, it is inflexible with continuous data and has poor scalability, with some difficulty in handling imbalanced data and an absence of quantitative measures.

Within the type of model-based methods, an alternative is the continuous network representation provided by the differential equation model. This approach demonstrates proficiency in capturing the intricate dynamics inherent in GRNs. The methodology involves establishing a model that relies on the interconnections among genes and regulatory equations, enabling a precise representation of biological phenomena. In contrast, the Bayesian network model, a prominent probabilistic graphical model, delineates gene dependencies using a directed acyclic graph. Although effective in reducing noise and integrating prior knowledge, the Bayesian network model encounters challenges related to computational complexity. The study in Cantone et al. (2009) concluded that GRN inference, based on both differential equations and Bayesian networks, despite their limitations, are good candidates for correctly inferring regulatory interactions from experimental data.

2.2 Model-free methods

2.2.1 ARM-based GRN inference

As stressed in the introduction, one of the pillars of our method is the ARM approach, a model-free method rooted in data mining. In this domain, the majority of classical techniques rely on the apriori algorithm, supported by its antimonotone property and employing a generate-and-test approach to uncover frequent patterns (Agrawal et al., 1993). While apriori-based methods exhibit commendable performance in scenarios where data is sparse and gene sets are succinct, they tend to be less effective and occasionally unfeasible when dealing with dense datasets like gene expression data. This disadvantage is due to the high computational cost of candidate evaluation and testing. To cope with the exponential expansion of generated frequent gene sets, one strategy is to exclusively consider closed gene sets. In Martínez et al. (2008), the CLOSE algorithm (Pasquier et al., 1999) is introduced. This latter employs a generate-and-test strategy along with a closure mechanism to identify frequent closed gene sets, facilitating the extraction of association rules from highly correlated data. After acquiring all frequent gene sets, the generation of association rules entails partitioning each frequent gene set into two distinct gene sets. For a given frequent gene set I, a total of $2^{|I|} - 2$ association rules can be generated. Hence, confidence plays a role in curbing the number of association rules by favouring those with higher reliability among rules representing a specific level of significance, based on support (Alves et al., 2010). An alternative and effective strategy for managing complexity is to deduce all frequent maximal gene sets and derive a set of association rules from them. This method is particularly useful for handling extensive gene sets within dense domains. Nonetheless, it is unsuitable for rule generation due to the absence of subset counting, as noted by Alagukumar et al. (2020).

2.2.2 Metaheuristic methods for GRN inference

As previously mentioned, the objective is to extract optimal QARs based on selected criteria, and metaheuristic algorithms have demonstrated their efficacy as suitable candidates for this challenging task. Numerous researchers have employed metaheuristic algorithms, including ant colony optimisation (Manju and Kant, 2015), genetic algorithms (Kabir et al., 2017), and particle swarm optimisation (Yan et al., 2019), among others. These algorithms have been utilised to generate sets of association rules, each exhibiting diverse performance characteristics, achieved by employing search algorithms to select the highest-quality rules from the pool of candidate solutions. In Babtie et al. (2021), the authors describe some GRN inference algorithms, comparing their pros and cons, along with other issues faced by these approaches. In Mitra et al. (2021), four meta-heuristic techniques are employed, namely binary particle swarm optimisation (BPSO), binary differential evolution (BDE), simulated annealing and tabu simulated annealing. The training of Bayesian Network-modelled GRN is done on gene expression data. The paper shows the superiority of BDE and BPSO approaches on the basis of F1 score, as compared with other methods. Nevertheless, the approach is susceptible to sensitivity to modelling errors.

In Ponzoni et al. (2007), GRNCOP was introduced as a machine-learning combinatorial optimisation algorithm that avoids making arbitrary or uniform discretisations of gene expression values. The method calculates dynamic thresholds using continuous-valued attribute discretisation techniques, similar to those employed in classification algorithms based on decision trees. A first limitation of GRNCOP is that the discretisation is applied exclusively to regulatory genes, while the thresholds for target genes are determined using mean expression values. A second limitation is that it can only infer rules with a maximum time delay of one unit. These limitations served as an incentive for the development of GRNCOP2, positioned as the successor to its predecessor, GRNCOP (Gallo et al., 2011). The GRNCOP2 aim is to find a set of optimal classifiers that define the potential association rules between a given gene i and other genes, considered as potential regulators. Basically, $\overline{\pi}_i$ is a vector that represents all the potential regulators of the gene i. Then, the classifiers $\overline{\pi}_i$ are calculated using a constructive approach that explores all possible combinations for each element of $\overline{\pi}_i(k)$ by determining the value that maximises the objective function σ, defined by the product between the sensitivity and specificity of the classifier $\overline{\pi}_i$. A common constraint shared by both GRNCOP2 and its precursor, GRNCOP, pertains to the restriction in deducing rules with a maximum time-delay of one unit. Despite this limitation, GRNCOP2 stands out as a state-of-the-art algorithm worthy of competition. Hence, we opted for GRNCOP2 as a competitor to be employed alongside our proposed method.

2.2.3 Information-theoretic methods for GRN inference

Among data-driven methodologies, it is useful to highlight that those based on information theory stand out as particularly powerful. Their effectiveness lies in their capability to apprehend intricate nonlinear regulatory relationships, contributing to their status as a robust approach within the GRN inference domain (Mousavian et al., 2016). The fundamental principle underpinning the majority of the information-theoretic methods revolves around mutual information (MI). Initially employed in information theory for assessing signal similarity, mutual information is later applied in the biological context to quantify regulatory connections among genes.

The most important among these methods are, in chronological order, relevance network (RN), algorithm for the reconstruction of accurate cellular networks (ARACNE), context likelihood of relatedness (CLR), mutual information 3 (MI3), minimum redundancy network (MRNET), path consistency algorithm based on conditional mutual information (PCA-CMI), mutual information distance and entropy reduction (MIDER), maximum-relevance and maximum-significance strategy (MRMSn), redundancy reduction in the MRNET algorithm (RRMRNET), and finally random walk with restart network (RWRNET).

Among the diverse array of methods reported, RN remains notable as it represents one of the pioneering approaches in utilising mutual information for measuring relationships. Along with updates introduced approximately two decades after its initiation, RN maintains its significance in the field, as indicated in Kuzmanovski et al. (2018). This early algorithm was followed by the ARACNE (Basso et al., 2005) which was dedicated to removing indirect associations within cellular networks. Later CLR (Faith et al., 2007) was designed to focus on estimating the likelihood of regulatory relationships between genes, while the MI3 algorithm (Luo et al., 2008) utilises mutual information as a key metric in uncovering associations among genes.

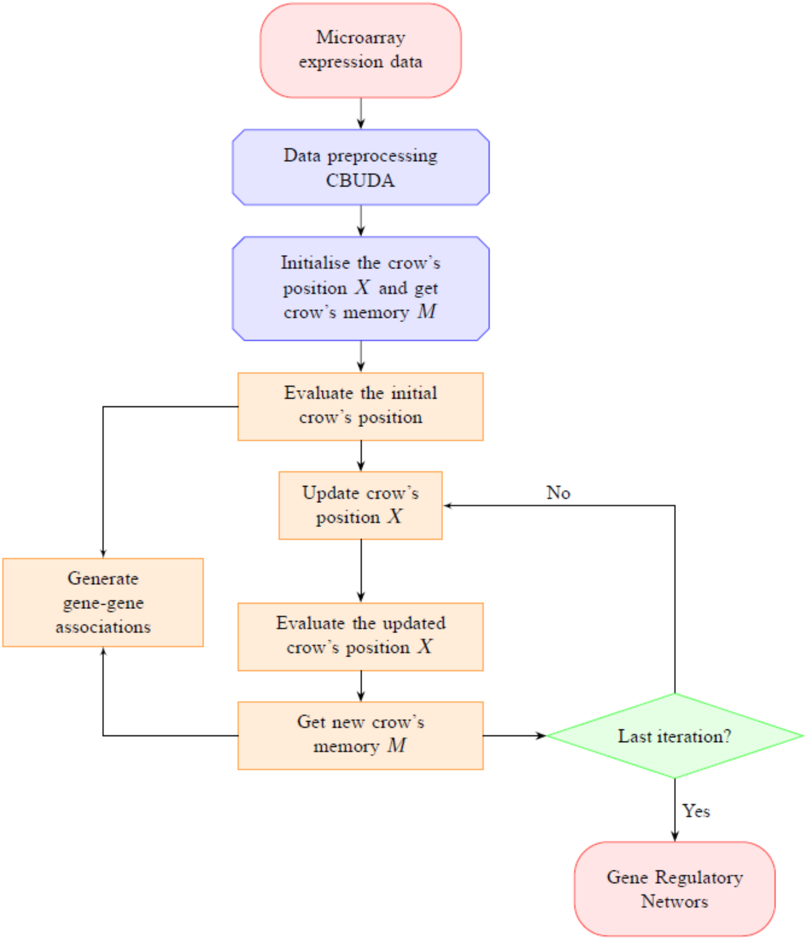
The MRNET (Meyer et al., 2007) identifies relevant genes while concurrently minimising redundancy in the inferred network structure. It operates as a feature selection method, adopting a strategy for the selection of regulatory relationships. Despite mutual information's proficiency in measuring nonlinear regulatory relationships, MRNET encounters some challenges in distinguishing indirect regulatory connections (Margolin et al., 2006). To overcome this limitation, Zhang et al. (2011) proposed PCA-CMI, replacing mutual information with conditional mutual information (CMI). However, CMI tends to underestimate gene relationships, leading to the introduction of conditional mutual inclusive information (CMI2) to rectify the underestimation issue (Zhang et al., 2015). Concurrently, MIDER (Villaverde et al., 2014) was designed as a comprehensive approach, integrating mutual information, distance metrics, and entropy reduction to capture intricate relationships within cellular systems. MRMSn (Liu et al., 2016) was introduced to combine strategies of maximum relevance and maximum significance for enhanced discriminatory power in the identification of gene associations. Still further, in the pursuit of heightened accuracy, Liu et al. (2017) introduced RRMRNET, a refinement of MRNET incorporating two strategies to eliminate redundant regulatory relationships. RRMRNET successfully addressed the challenge of redundancy within the MRNET algorithm, and enhanced the accuracy of the inferred cellular network. Ultimately, the random walk with restart network (RWRNET) (Liu et al., 2020) was introduced to incorporate a random walk strategy with restart mechanisms, offering a dynamic approach to capturing network dynamics over time. Despite their inherent limitations, information-theoretic methods are acknowledged as robust challengers. Hence, we include them for thorough comparative analysis alongside our proposed method.

3 DCSA-QAR design

3.1 Preliminaries

In GRN inference, ARM serves as a valuable tool to uncover meaningful correlations between genes on the basis of their profiles under diverse environmental conditions, as recorded in a gene experimental database. Moreover, ARM finds utility in identifying links between environmental circumstances and gene expression, as well as associations between biological information about genes and gene expression. A general schema of the ARM process is composed of four steps: discretisation, mining of frequent patterns, generation and filtering of association rules, and biological evaluation of the obtained rules (Alves et al., 2010). The starting point of this process is a $n \times m$ matrix of gene expression values with rows representing experimental conditions and columns representing genes. At the discretisation step, the values of the input matrix are transformed into discrete values in order to be more adaptable to the processes of the ARM. Different expression properties could be considered to define the state of a gene as under-expression, over-expression, upregulated, downregulated. The primary goal of the second phase is to identify all frequent gene sets whose level of support surpasses a user-defined threshold. In the following subsections, we present the DCSA-QAR method. The flowchart of DCSA-QAR is given in Figure 1. DCSA-QAR takes as input the gene microarrays expression data that contain N genes and M samples, and returns an inferred gene regulatory network (GRN) as output.

Figure 1 The flowchart of DCSA-QAR (see online version for colours)


Microarray expression data
Data preprocessing CBUDA
Initialise the crow's position X and get crow's memory M
Evaluate the initial crow's position
Update crow's position X
No
Generate gene-gene associations
Evaluate the updated crow's position X
Get new crow's memory M
Last iteration?
Yes
Gene Regulatory Networs

3.2 Data pre-processing

Gene microarrays expression data often consist of continuous numerical values representing the intensity of gene expression levels. These continuous attributes can be extremely high-dimensional, making the discovery of meaningful associations a challenging task. This stage aims to transform these continuous attributes into categorical variables by partitioning the expression levels into discrete intervals or bins. The discretisation is performed through the so-called confidence-based unsupervised discretisation algorithm (C-BUDA): built upon a classification framework, this approach tackles the drawback of conventional discretisation methods, which often overlook interdependencies and correlations between attributes. Within the context of DCSA-QAR, the C-BUDA algorithm strives to optimise the division of a numerical attribute interval, aiming to enhance class prediction performance. For doing so, an equal-width discretisation is used to obtain a class attribute. Next, for every potential point of division, an equi-width discretisation is once again implemented on the target attribute. This process results in the calculation of the average confidence across all rules. Ultimately, the count of intervals that yields the highest average confidence is selected and retained.

3.3 Initialising crow's population

3.3.1 Crow position encoding

During this stage, the DCSA-QAR methodology utilises the discrete encoding that was previously derived to represent association rules. In this encoded representation, the positional information of each crow, also referred to as a particle, is transformed into a vector comprising discrete numerical values, thereby signifying an individual association rule. Within a group of crows, each crow is defined by both its present position and its memorised position. These positions, denoting both current state and past reference, are characterised by a total of $2 \times N$ elements, where N represents the count of attributes present within the dataset. Consequently, each crow is characterised by a dual set of vectors, delineating control and parametric attributes. The control attributes exhibit three potential values: 0, 1, or -1, signifying whether the attribute is absent in the rule, resides within the antecedent, or pertains to the consequent of the rule, respectively. Conversely, the parametric attribute is a whole number that signifies the value associated with the respective attribute. The visualisation of crow positions and memories is presented in Tables 1 and 2, respectively.

Table 1 Crow position structure with $N = 9$

X_i^{it}	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9
C_a	0	-1	1	0	-1	1	1	-1	1
P_a	5	2	3	8	4	2	1	6	3

Table 2 Crow memory structure with $N = 9$

M_i^{it}	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9
C_a	1	-1	1	0	-1	1	-1	1	0
P_a	6	1	2	4	2	5	1	3	4

3.3.2 Flock initialisation

The initialisation of the flock is carried out in a randomised manner. The random initialisation of the flock adheres to predefined criteria, including a minimum attribute count in both the antecedent and consequent, along with a stipulated rule support threshold. The process that generates the initial population is listed below:

- First, the indexes of attributes which can belong to the rule are randomly selected according to a maximum number of attributes given by the user.
- For each selected index, the control attribute is randomly selected from values $1, -1, 0$.
- Then, the parametric attributes are randomly generated for each selected attribute.
- Finally, the position is accepted if it is fully compliant with the user requirements (minimum number of attributes in the antecedent and consequent of the rule, minimum support of the rule).

After generating the initial crow's positions of flock, initial memory position of each crow was assigned to its current position. The initialisation of the flock is carried out in a randomised manner, followed by the evaluation of the fitness function.

3.4 Crow's position

3.4.1 Evaluating crow's position through fitness function

The definition of the fitness function represents one of the most important steps in all metaheuristic methods. The ARM process can be conceptualised as a multi-objective problem wherein the evaluation metrics employed for rule assessment encompass distinct objectives to be concurrently optimised. Nonetheless, a significant portion of the algorithms proposed for ARM, categorised as multi-objective methodologies, converge towards a strategy of incorporating weighted objectives into an aggregated one-objective fitness function. In the context of DCSA-QAR, the objective to be maximised is formulated by equation (1).

$$F(\mathit{Rule}) = w_s \times \mathit{supp} + w_c \times \mathit{conf} + w_l \times lift \tag{1}$$

Here, supp means support, conf represents confidence, and $lift$ stands for lift. The weights w_s, w_c, and w_l are integral components of the fitness function, contributing to the holistic maximisation approach within the DCSA-QAR.

Although machine learning techniques can optimise weights automatically based on training data, manual weight setting remains a valuable and widely used approach when domain expertise and problem-specific customisation are essential considerations in the decision-making process. Manual weight setting allows domain experts to incorporate their knowledge, which enhances interpretability. It also aligns the algorithm with problem-specific needs, provides users with the ability to fine-tune trade-offs, and helps prevent overfitting.

3.4.2 Updating crow's position

In this stage, the search for a new position is undertaken in the search space. At iteration it, the new position of a given crow i depends on its current position X_i^{it} and the displacement step S_i^{it} to displace to the hiding position M_j^{it} of its adversary j. For accurately adapting the algorithm, it was considered appropriate to establish a relationship between the displacement step of a given crow and the hiding position of its adversary. This has been done through a new replacement operator and the so-called non-zero Hamming distance, calculated between the position vector of a given crow and the memory vector of its competitor.

As result, the new position X_i^{it+1} is calculated according to equation (2).

$$X_i^{it+1} = \bowtie (X_i^{it}, M_j^{it}, int(S_i^{it})) \tag{2}$$

where $\bowtie$ means that the crow i execute $int(S_i^{it})$ times the randomly replacement operation (presented in Algorithm 1) between the vectors X_i^{it}, M_j^{it}. The function $int(a)$ calculates the integer value of a given real a. The number S_i^{it} is calculated as in equation (3):

$$S_i^{it} = fl_i^{it} \times nonZHD(M_j^{it}, X_i^{it}) \tag{3}$$

Where, the $nonZHD(M_j^{it}, X_i^{it})$ function calculate the non-zero Hamming distance between two vectors M_j^{it} and X_i^{it}, and fl_i^{it} is the flight length in $[0, 1]$.

Algorithm 1 Replacement operator $\bowtie$

```
Input: $X$, $M$ are vectors
       $S$ is an integer
Output: $Y$ is a vector;
Description:
Y=X;
for ($i = 1$; $i \leq S$ ; $i++$) do
    $j = rand(1; N)$;
    $Y[1, j] = M[1, j]$;
    $Y[2, j] = M[2, j]$;
end
return Y;
```

The non-zero Hamming distance between two position or memory vectors X and Y, denoted by $nonZHD(X, Y)$, is defined as the number of positions where the elements x_i and y_i are different and their control attributes $C_a(x_i)$ and $C_a(y_i)$ are not null at the same time. That is,

$$nonZHD(X, Y) = \sum_{i=1}^{d} \delta(x_i, y_i) \tag{4}$$

where

$$\delta(x_i, y_i) = \begin{cases} 0, & \text{if } \big(x_i = y_i)\big) \\ 0, & \text{if } \big(C_a(x_i) = 0 \wedge C_a(y_i) = 0\big) \\ 1, & \text{otherwise} \end{cases} \tag{5}$$

3.5 Generating gene-gene associations

In this stage, the QARs obtained at each iteration, presented as the best solutions, are used to mine the gene-gene associations as follows:

1 For each obtained QAR:

Separate between antecedent's gene sets and consequent's gene sets.

2 To generate gene-gene associations:

Combine each gene in the antecedent with all genes in the consequent.

3 Return rule.

We note that only QARs verifying certain predefined user thresholds such as min_supp, min_conf and min_acc are qualified to generate gene-gene associations.

Finally, we represent a gene network whereby genes are the graph nodes and gene-gene associations mined from QARs are the graph edges.

For example, from the following obtained QARs:

1 $\texttt{G1} \in \texttt{[-0.33,-0.09]} \Rightarrow \left(\texttt{G2} \in \texttt{[0.24,1.11]} \wedge \texttt{G4} \in \texttt{[-0.49,-0.20]}\right)$

2 $\texttt{G2} \in \texttt{[1.11,1.98]} \wedge \texttt{G3} \in \texttt{[-0.51,0.05]} \Rightarrow \texttt{G5} \in \texttt{[-0.47,-0.06]}$.

We extract the resulting gene-gene associations:

1 G1 ⇒ G2

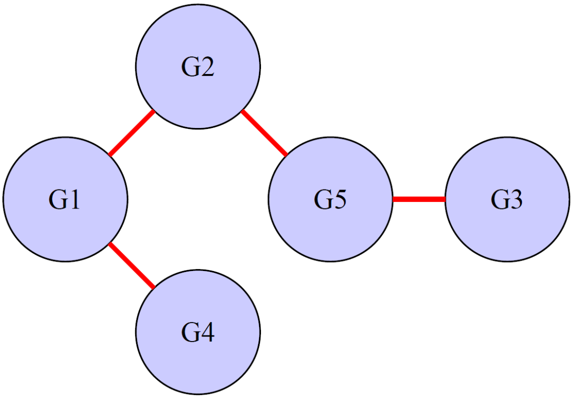
2 G1 ⇒ G4

3 G2 ⇒ G5

4 G3 ⇒ G5.

Figure 2 shows the gene network obtained from resulting gene-gene associations for the example above.

Figure 2 Gene network from gene-gene associations reported by the example (see online version for colours)



4 Materials and methods

Specific materials (i.e., datasets in our case) and methods (i.e., competing algorithms and performance metrics in our case) are needed in order to undertake experiments. Our objective is to conduct two major types of experiments: one series involving meta-heuristic methods and the other series considering other methods such as information-theoretic methods. For comparison purposes, we rely on different categories of datasets, benchmarks, and metrics for each series of experiments. This comparison is instrumental in gauging the effectiveness of our proposed approach within the field of GRN inference.

4.1 Metaheuristc-based methods

4.1.1 First series competitors

In terms of competitors, we compared our approach against five algorithms:

- a decision-tree-based approach introduced by Soinov et al. (2003)
- RegNet (Nepomuceno-Chamorro et al., 2010), a regression-tree-based technique
- GRNCOP (Gallo et al., 2011) representative of combinatorial optimisation algorithms
- GarNet (Martínez-Ballesteros et al., 2014), a multi-objective evolutionary QARs algorithm, with two variants, GarNet1 and GarNet2, obtained by setting the minimum support to 0.3 and 0.35, respectively.

4.1.2 First series datasets

We assessed the efficacy of DCSA-QAR using microarray data sourced from Spellman et al. (1999) and Cho et al. (1998). These datasets were synchronised through three distinct methodologies: CDC15, CDC28, and alpha-factors. These synchronisation methods are statistically independent and were sampled at intervals of 10 minutes, 10 minutes, and 7 minutes respectively (Someren et al., 2000). We further rely on three authentic datasets as true networks. These datasets encompass three true networks (Lee et al., 2007; Dwight et al., 2002; Lee et al., 2004). For comparative purposes with prior studies, DCSA-QAR was evaluated on a subset of 20 well-defined genes responsible for crucial cell-cycle regulatory proteins. The utilisation of these authentic datasets provides a robust basis for the evaluation of the network quality generated by DCSA-QAR when compared within established competing algorithms. The genes used are outlined in Table 3.

4.1.3 First series performance metrics

The comparison is done using five standard performance metrics, namely precision, sensitivity, specificity, F1-score, and the number of gene-gene associations. Note that we have calculated all the metrics, in the same way as Gallo et al. (2011) and Martínez-Ballesteros et al. (2014), on the basis of the reduced search space defined by the 20 chosen genes, described in Table 4.

Table 3 List of genes used in first series experiments

#	*Gene name*	*Common name*	*Description*
1	YMR199W	CLN1	Cyclin, G1/S-specific
2	YPL256C	CLN2	Cyclin, G1/S-specific
3	YAL040C	CLN3	Cyclin, G1/S-specific
4	YGR108W	CLB1	Cyclin, G2/M-specific
5	YPR119W	CLB2	Cyclin, G2/M-specific
6	YLR210W	CLB4	Cyclin, G2/M-specific
7	YPR120C	CLB5	Cyclin, B-type
8	YGR109C	CLB6	Cyclin, B-type
9	YMR043W	MCM1	Transcription factor of the MADS box family
10	YLR079W	SIC1	Inhibitor of Cdc28p-Clb protein kinase complex
11	YLR182W	SWI6	Transcription factor, subunit of SBF and MBF factors
12	YBR160W	CDC28	Cyclin-dependent protein kinase
13	YDL132W	CDC53	Controls G1/S transition, component of SCF-ubiquitin ligase complexes
14	YDL056W	MBP1	Transcription factor, subunit of the MBF factor
15	YDR054C	CDC34	E2 ubiquitin-conjugating enzyme
16	YDR146C	SWI5	Transcription factor
17	YDR328C	SKP1	Core component of SCF-ubiquitin ligase complexes
18	YER111C	SWI4	Transcription factor, subunit of SBF factor
19	YGL116W	CDC20	Cell division control protein
20	YGL003C	HCT1	Substrate-specific activator of APC-dependent proteolysis

Table 4 Characteristics of potential gene pair-wise associations for first series experiments

YeastNet		*Co-citation*		*GO*	*# of possible associations*
Precision	*Score*	*Precision*	*Score*	*Precision*	
51.58%	1.5303	43.68%	1.3487	45.26%	190

4.2 Information-theoretic methods

We follow the same structure as in Subsection 4.1 above, by reporting the competing algorithms, the datasets and metrics used in this second series of experiments.

4.2.1 Second series algorithms

In terms of competitors, we chose nine information-theoretic algorithms, reported in Section 2.2.3 above. Indeed, these algorithms are among the most performing in the domain (Liu et al., 2020). These competitors are cited below in chronological order.

- ARACNE (Basso et al., 2005) which is based on the removal of indirect associations within cellular networks.
- CLR (Faith et al., 2007) which relies on the likelihood estimation of regulatory relationships between genes.

- MRNET (Meyer et al., 2007) which identifies pertinent genes while simultaneously minimising redundancy within the inferred network structure.
- MI3 (Luo et al., 2008) which employs mutual information as a primary metric to reveal connections among genes.
- PCA-CMI (Zhang et al., 2011) which combines path consistency algorithm with conditional mutual information, offering a nuanced perspective on the relationships among genes.
- MIDER (Villaverde et al., 2014) which is characterised by integrating mutual information, distance metrics, and entropy reduction to capture intricate relationships within cellular systems.
- MRMSn (Liu et al., 2016) which combines both maximum relevance and maximum significance strategies to enhance discriminatory power in identifying gene associations.
- RRMRNET (Liu et al., 2017) which focuses on addressing redundancy within the MRNET algorithm, with the goal of improving the accuracy of the inferred cellular network.
- RWRNET (Liu et al., 2020) which integrates a random walk strategy with restart mechanisms, providing a dynamic approach to capturing network dynamics over time.

4.2.2 Second series datasets

We used two types of datasets:

- SOS network dataset is a real gene expression dataset with experimental verification. It often serves as a standard GRN to test the effectiveness of algorithms. This is a small-scale directed network with nine genes and 24 regulations (Ronen et al., 2002).
- DREAM3 dataset: a simulated dataset with three variants incorporating 10, 50 and 100 genes (Marbach et al., 2010; Margolin et al., 2006). DREAM challenges are typically designed to evaluate and advance the state of the art in various aspects of computational biology, such as network inference, gene expression prediction, and other related tasks. In our context, DREAM3-10, DREAM-50 or DREAM-100 genes refers to a set of genes (with 10, 50 or 100 genes, respectively).

Table 5 describes the relevant the datasets characteristics used in second series experiments.

4.2.3 Second series performance metrics

The standard and well-known performance metrics used in the evaluation phase are extracted from the confusion matrix. TP: true positive; FP: false positive; TPR: true positive rate; FPR: false positive rate; Prec: precision; Acc: accuracy.

Table 5 Descriptions of the second series datasets in our experiments

Datasets	*Variables*	*Samples*	*Type*	*Network nodes*	*Network edges*
DREAM3-10 genes	10	10	Simulated	10	10
DREAM3-50 genes	50	50	Simulated	50	77
DREAM3-50 genes	100	100	Simulated	100	166
SOS	9	9	Real	9	24

5 Experiments and discussion

This section is divided into two complementary sub-sections: first series experiments and second series experiments, in line with the previous section.

5.1 *First series of experiments*

5.1.1 *Parameter settings and performance*

In order to analyse the behaviour of our method for achieving optimal mined solutions, we have to set the main parameters to some tuned values. This step allows us to identify the minimum quality thresholds of QAR; i.e., those that give the best possible values for the quality metrics in gene networks; in a similar standard way as reported in Gallo et al. (2011) and Martínez-Ballesteros et al. (2014). These parameters are used for each analysis carried out for further performance evaluation. Table 6 shows DCSA-QAR main parameters used for the gene expression data experiments.

Table 6 DCSA-QAR parameters used for the gene expression data experiments

Parameter	*Value*
Awareness probability AP	0.1
Flight length f_l	2
Support weight w_s	0.2
Confidence weight w_c	0.4
Lift weight w_l	0.4
Size of the population	150
Number of iterations	100

5.1.2 *Identifying initial minimum thresholds*

We have executed different configurations of DCSA-QAR by modifying the minimum thresholds of performance measures of the QAR obtained in the first phase. To achieve that, we vary the minimum threshold for the confidence and accuracy measures from 0.6 to 0.9 with increments of 0.05. However, we modify the minimum values for the support measure from 0 to 0.3 with increments of 0.05. In this situation, DCSA-QAR runs (7 × 7 × 7), i.e., 343 times in total. Once the minimum thresholds values are obtained, it is then possible to use the algorithm in the best possible conditions, thus allowing a fair comparison with other competitor algorithms.

Table 7 Average performance metrics of gene networks achieved by DCSA-QAR in YeastNet

ID	*Parameters*			*Precision*	*Sensitivity*	*Specificity*	*Score*	*# of*
	Accuracy	*Conf*	*Supp*	*(%)*	*(%)*	*(%)*		*associations*
1	[0.6–0.9]	[0.6–0.9]	0	63.10	51.02	40.50	3.75	102.45
2	[0.6–0.9]	[0.6–0.9]	0.05	64.10	44.02	50.55	3.75	87.16
3	[0.6–0.9]	[0.6–0.9]	0.1	81.70	20.59	88.38	3.83	33.29
4	[0.6–0.9]	[0.6–0.9]	0.15	84.21	7.38	96.20	3.71	11.69
5	[0.6–0.9]	[0.6–0.9]	0.2	86.01	5.13	97.93	3.53	7.76
6	[0.6–0.9]	[0.6–0.9]	0.25	65.33	1.39	97.93	4.80	3.08
7	[0.6–0.9]	[0.6–0.9]	0.3	88.78	1.24	99.50	4.96	1.88
8	[0.6–0.9]	0.6	[0–0.3]	72.21	20.41	78.59	4.01	39.43
9	[0.6–0.9]	0.65	[0–0.3]	73.04	20.18	79.59	3.99	38.49
10	[0.6–0.9]	0.7	[0–0.3]	72.99	19.05	81.41	4.01	35.90
11	[0.6–0.9]	0.75	[0–0.3]	74.23	18.48	82.07	4.11	34.76
12	[0.6–0.9]	0.8	[0–0.3]	79.79	17.80	82.86	4.08	33.39
13	[0.6–0.9]	0.85	[0–0.3]	80.48	17.42	83.23	4.06	32.67
14	[0.6–0.9]	0.9	[0–0.3]	80.48	17.42	83.23	4.06	32.67
15	0.6	[0.6–0.9]	[0–0.3]	72.38	26.64	74.63	3.99	49.80
16	0.65	[0.6–0.9]	[0–0.3]	72.18	25.34	75.29	3.99	47.73
17	0.7	[0.6–0.9]	[0–0.3]	71.41	23.61	75.57	3.98	45.39
18	0.75	[0.6–0.9]	[0–0.3]	64.77	18.60	78.27	4.05	37.37
19	0.8	[0.6–0.9]	[0–0.3]	71.32	15.64	82.17	4.01	31.14
20	0.85	[0.6–0.9]	[0–0.3]	88.83	12.60	89.20	4.07	22.78
21	0.9	[0.6–0.9]	[0–0.3]	92.34	8.33	95.86	4.25	13.10

5.1.3 DCSA-QAR evaluation using varied minimum thresholds

The cumulative outcomes from 343 runs are concisely outlined in Tables 7, 8 and 9. These tables are systematically organised into distinct segments, each associated with a particular test network. Notably, each run signifies the resultant network achieved through the intersection of association pairs extracted from QAR, tailored to individual input datasets that satisfy predefined parameter conditions. Each row within these tables corresponds to 49 distinct cases. The delineation of each section is elucidated as follows:

- First segment (rows 1–7): each row in this segment presents the average performance metrics of the considered gene networks, as obtained from executions. These executions entail maintaining a constant minimum support while variably adjusting the minimum confidence (ranging from 0.6 to 0.9) and minimum accuracy (ranging from 0.6 to 0.9).
- Second segment (rows 8–14): this segment adheres to a similar approach by maintaining a fixed minimum confidence value, while introducing variations in the minimum support (ranging from 0 to 0.3).
- Third segment (rows 15–21): the final segment encompasses executions wherein accuracy parameters remain fixed, while the minimum support ranges from 0 to 0.3, and the minimum confidence varies from 0.6 to 0.9.

Table 8 Average performance metrics of gene networks achieved by DCSA-QAR in the Gene Ontology (GO) dataset

ID	*Parameters*			*Precision*	*Sensitivity*	*Specificity*	*# of*
	Accuracy	*Conf*	*Supp*	*(%)*	*(%)*	*(%)*	*associations*
1	[0.6–0.9]	[0.6–0.9]	0	50.38	51.33	43.26	102.45
2	[0.6–0.9]	[0.6–0.9]	0.05	51.35	44.98	53.15	87.16
3	[0.6–0.9]	[0.6–0.9]	0.1	64.50	21.25	86.54	33.29
4	[0.6–0.9]	[0.6–0.9]	0.15	65.73	7.61	95.42	11.69
5	[0.6–0.9]	[0.6–0.9]	0.2	61.64	4.62	96.50	7.76
6	[0.6–0.9]	[0.6–0.9]	0.25	83.07	2.35	99.17	3.08
7	[0.6–0.9]	[0.6–0.9]	0.3	72.45	1.13	99.17	1.88
8	[0.6–0.9]	0.6	[0–0.3]	66.29	21.52	80.09	39.43
9	[0.6–0.9]	0.65	[0–0.3]	66.12	21.01	80.56	38.49
10	[0.6–0.9]	0.7	[0–0.3]	64.81	19.71	81.99	35.90
11	[0.6–0.9]	0.75	[0–0.3]	61.19	18.55	81.99	34.76
12	[0.6–0.9]	0.8	[0–0.3]	64.01	17.73	82.60	33.39
13	[0.6–0.9]	0.85	[0–0.3]	63.35	17.38	83.00	32.67
14	[0.6–0.9]	0.9	[0–0.3]	63.35	17.38	83.00	32.67
15	0.6	[0.6–0–0.9]	[0–0.3]	55.77	27.09	74.75	49.80
16	0.65	[0.6–0.9]	[0–0.3]	55.41	25.56	75.35	47.73
17	0.7	[0.6–0.9]	[0–0.3]	54.33	24.12	76.36	45.39
18	0.75	[0.6–0.9]	[0–0.3]	61.37	19.69	80.35	37.37
19	0.8	[0.6–0.9]	[0–0.3]	68.30	16.33	83.54	31.14
20	0.85	[0.6–0.9]	[0–0.3]	70.17	12.08	88.11	22.78
21	0.9	[0.6–0.9]	[0–0.3]	83.78	8.41	94.75	13.10

Figures 3, 4, 5 and 6, which represent graphic illustrations of results obtained in Tables 7, 8 and 9, show the relationship between different parameters (accuracy, confidence, and support) and GRNs metrics: precision, sensitivity, specificity, and the number of gene-gene associations. It can be observed that DCSA-QAR achieves similar performances among the true networks, except for the precision metric where DCSA-QAR obtains the best values over Co-citation than GO and YeastNet networks.

From Figure 3, it can be concluded that the precision and specificity metrics of GRN increase when the accuracy threshold is higher. The precision varies from 54% to 93%, while the specificity varies from 74% to 96%. However, the sensitivity metric increase when the accuracy threshold is low; it decreases from 27% to 8%. The score metric has a similar behaviour as precision and specificity metrics. For a higher accuracy value, the score metric achieves 4.25 and 3.88 over YeastNet and Co-citation, respectively.

From Figure 5, it can also be concluded that the precision and specificity metrics of GRN increase when the support threshold is higher. The precision varies from 50% to 94%, while the specificity varies from 40% to 99%. However, the sensitivity metric increase when the support threshold is low; it decreases from 51% to 1%. The score metric follows a parallel pattern to precision and specificity metrics. With an elevated support value, it attains scores of 4.96 and 4.10 across YeastNet and Co-citation, respectively. As both accuracy and support thresholds escalate, the network's complexity diminishes, as depicted in Figure 6. This reduction in dimensionality corresponds to an increase in precision, consequently yielding more coherent gene-gene associations. Illustrated in Figure 4, the values of the GRNs metrics exhibit negligible fluctuations. This assertion makes sense since DCSA-QAR obtains association rules with high confidence by including the confidence measure as part of the fitness function.

Table 9 Average performance metrics of gene networks achieved by DCSA-QAR in Co-citation

ID	*Parameters*			*Precision*	*Sensitivity*	*Specificity*	*Score*	*# of*
	Accuracy	*Conf*	*Supp*	*(%)*	*(%)*	*(%)*		*associations*
1	[0.6–0.9]	[0.6–0.9]	0	66.00	51.76	41.52	3.49	102.45
2	[0.6–0.9]	[0.6–0.9]	0.05	66.57	44.30	50.79	3.48	87.16
3	[0.6–0.9]	[0.6–0.9]	0.1	84.06	20.66	89.13	3.58	33.29
4	[0.6–0.9]	[0.6–0.9]	0.15	90.44	7.70	97.12	3.76	11.69
5	[0.6–0.9]	[0.6–0.9]	0.2	94.38	5.52	98.96	3.70	7.76
6	[0.6–0.9]	[0.6–0.9]	0.25	87.15	1.90	98.96	3.97	3.08
7	[0.6–0.9]	[0.6–0.9]	0.3	88.78	1.20	99.46	4.10	1.88
8	[0.6–0.9]	0.6	[0–0.3]	76.83	20.77	79.29	3.71	39.43
9	[0.6–0.9]	0.65	[0–0.3]	77.76	20.55	80.36	3.70	38.49
10	[0.6–0.9]	0.7	[0–0.3]	77.76	19.33	82.03	3.68	35.90
11	[0.6–0.9]	0.75	[0–0.3]	80.42	18.78	82.74	3.76	34.76
12	[0.6–0.9]	0.8	[0–0.3]	87.34	18.11	83.57	3.76	33.39
13	[0.6–0.9]	0.85	[0–0.3]	88.62	17.75	83.97	3.73	32.67
14	[0.6–0.9]	0.9	[0–0.3]	88.62	17.75	83.97	3.73	32.67
15	0.6	[0.6–0.9]	[0–0.3]	79.46	27.07	75.61	3.66	49.80
16	0.65	[0.6–0.9]	[0–0.3]	79.33	25.80	76.31	3.64	47.73
17	0.7	[0.6–0.9]	[0–0.3]	78.75	24.05	76.45	3.64	45.39
18	0.75	[0.6–0.9]	[0–0.3]	74.64	19.08	79.09	3.69	37.37
19	0.8	[0.6–0.9]	[0–0.3]	82.55	16.22	83.24	3.69	31.14
20	0.85	[0.6–0.9]	[0–0.3]	89.49	12.55	89.19	3.88	22.78
21	0.9	[0.6–0.9]	[0–0.3]	93.14	8.29	96.05	3.88	13.10

In summary, the findings presented in Tables 7, 8 and 9, along with Figures 3, 4, 5, 6 and 7 and lead us to the following conclusions. It becomes evident that the outcomes are notably impacted by the manipulation of accuracy and support thresholds, outweighing the influence of the confidence threshold. Notably, the analysis underscores that the most favourable outcomes are observed within the context of the yeast datasets. Additionally, it becomes apparent that, across the majority of cases, the average outcomes stemming from adjustments to the accuracy threshold surpass those achieved through modifications of the support threshold. This highlights the proposition that the extraction of the most optimal rules is achieved when there's an elevation in the minimum accuracy criterion for the mined rules.

Figure 3 Relationship between the accuracy parameter and GRNs metrics (see online version for colours)

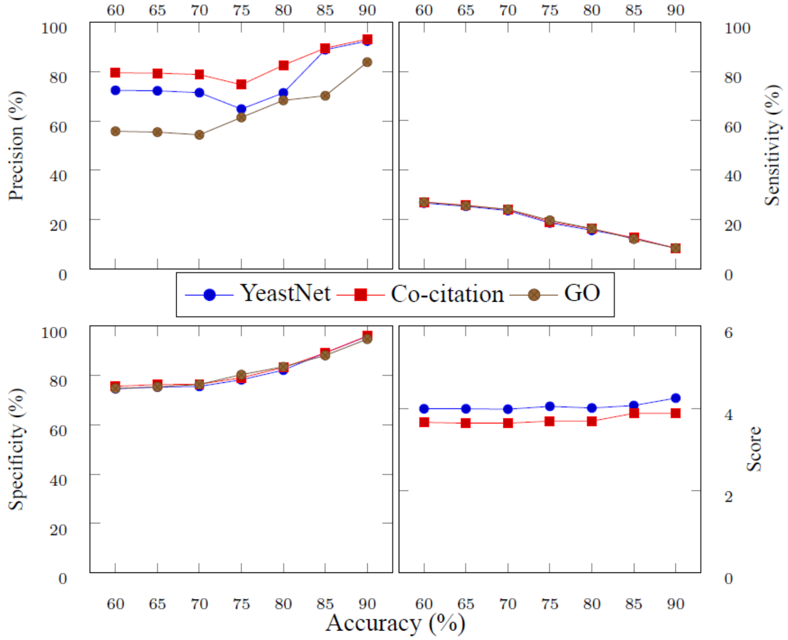


Figure 4 Relationship between the confidence parameter and GRNs metrics (see online version for colours)

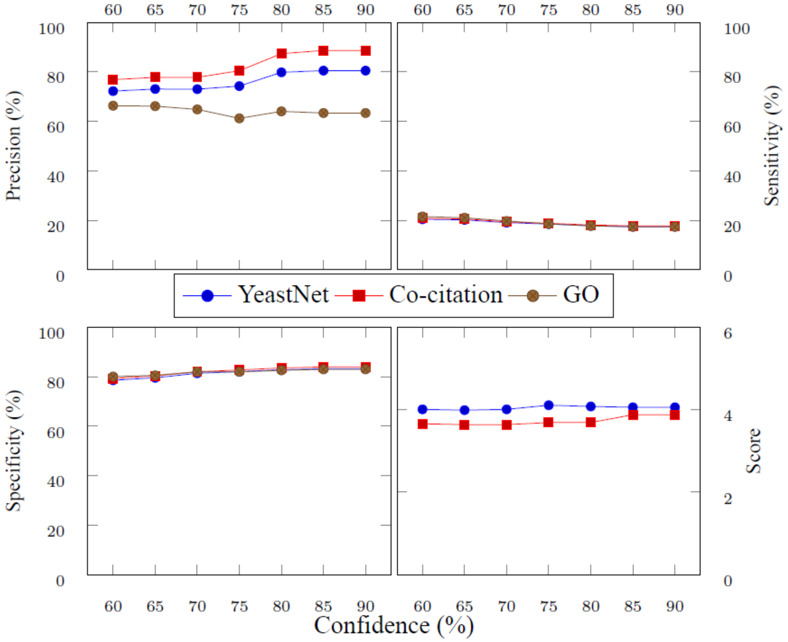


Figure 5 Relationship between the support parameter and GRNs metrics (see online version for colours)

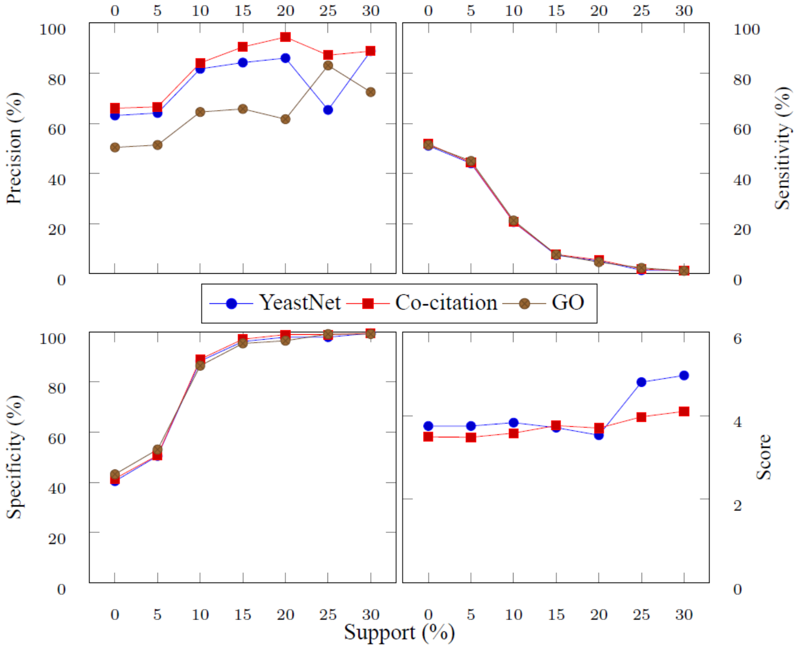
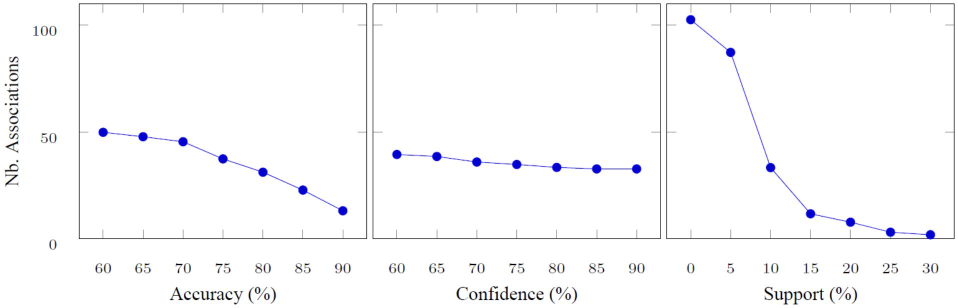


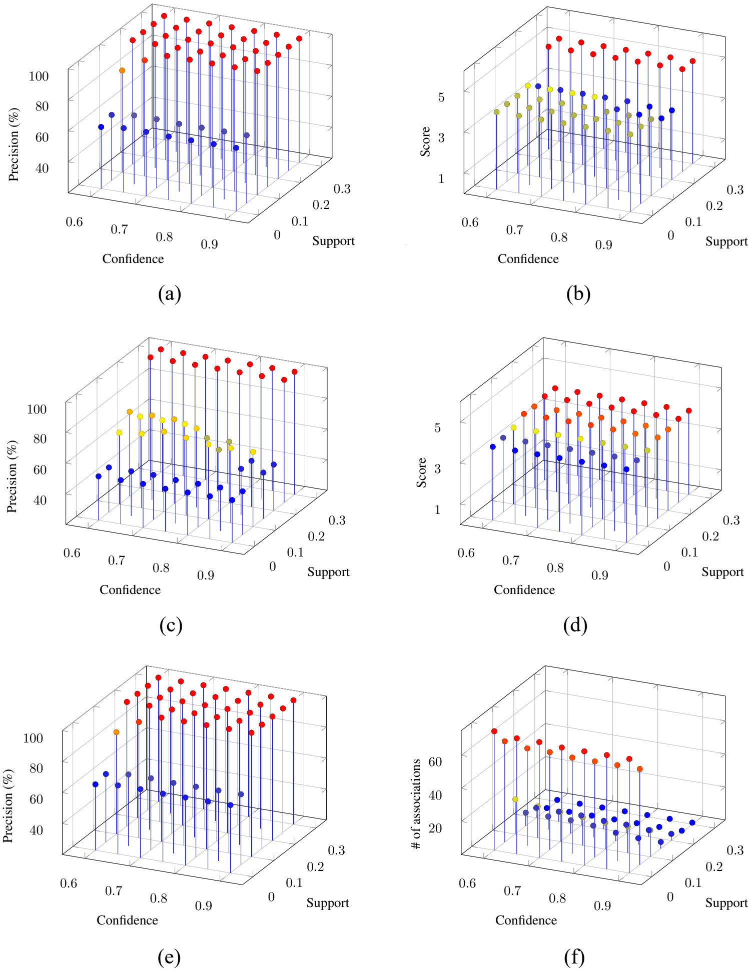
Figure 6 Relationship between the number of associations and GRNs metrics (see online version for colours)



5.1.4 DCSA-QAR gene networks metrics

When the parameters are set to their best values, our algorithm gives the results summarised in Table 10. In order to obtain an overall ranking for all algorithms, for any dataset or across all three datasets we count the total number of best measures obtained by each algorithm. For example, for Co-citation dataset, our algorithm is the best in three measures out of five, followed by RegNet and GarNet$_1$, both achieving the best results in two measures out of five.

Figure 7 Precision and score values achieved by DCSA-QAR with different parameters, (a) YeastNet precision (b) YeastNet score (c) Co-citation precision (d) Co-citation score (e) GO precision (f) Number of associations (see online version for colours)



(a)

(b)

(c)

(d)

(e)

(f)

Note: Values of the precision and score metrics achieved by DCSA-QAR with the support and confidence parameters varying from 0 to 0.3 and from 0.60 to 0.9, respectively, with the accuracy parameter fixed in 0.85. The number of associations is also shown.

5.1.4.1 Co-citation dataset

Compared with competitors' algorithms, our algorithm comes first in the Co-citation dataset, outperforming all competitors since it gives the best precision (100%), the best specificity (100%) and score (3.75). DCSA-QAR is followed by RegNet giving the same precision and specificity as DCSA-QAR while GarNet$_1$ gives the best sensitivity and accuracy. The other three algorithms are lagging and produce no best result in any measure.

Table 10 Values for the gene networks metrics achieved by the best settings of DCSA-QAR against GarNet$_{1,2}$, GRNCOP, RegNet, and Soinov et al.

Networks	*Measures*	*DCSA-QAR*	*GarNet$_1$*	*GarNet$_2$*	*GRNCOP*	*RegNet*	*Soinov et al.*
YeastNet	Precision	**100**	**100**	93.75	93.33	**100**	50.00
	Sensitivity	14.40	**20.40**	15.31	14.29	7.14	3.06
	Specificity	**100**	**100**	98.91	98.91	**100**	96.74
	Accuracy	43.68	**58.94**	55.79	55.27	52.11	48.41
	Score	**3.79**	2.89	2.82	3.04	3.24	1.84
Co-citation	Precision	**100**	95.00	93.75	93.33	**100**	50.00
	Sensitivity	13.95	**22.89**	18.07	16.87	8.13	3.61
	Specificity	**100**	99.07	99.07	99.07	**100**	97.20
	Accuracy	41.58	**65.79**	63.68	63.16	58.42	56.29
	Score	**3.75**	2.92	3.09	3.26	3.51	1.85
GO	Precision	72.22	70.00	**75.00**	73.33	71.43	50.00
	Sensitivity	13.13	**16.28**	13.95	12.79	5.81	3.49
	Specificity	94.51	94.23	96.15	96.15	98.08	**97.12**
	Accuracy	52.11	**58.96**	58.95	58.42	56.32	54.75
Average number of associations		18	**20**	16	15	7	6

5.1.4.2 YeastNet dataset

Our algorithm performs similarly as for the Co-citation dataset and is best in the same three previous measures (i.e., precision, specificity and score). However, DCSA-QAR comes second after GarNet$_2$ which achieves the best results in four out of five measures. RegNet comes after DCSA-QAR and achieves similar results as for the Co-citation dataset. The other tree algorithms produce no best result in any measure.

5.1.4.3 GO dataset

Although our algorithm achieves no best result for GO dataset, it is not far from other algorithms. Indeed, DCSA-QAR comes third after GarNet$_2$ and GRNCOP in precision, and third in sensitivity after GarNet$_1$ and GarNet$_2$. We notice that the favourable outcomes achieved by our method on the Co-citation and YeastNet datasets are counterbalanced by the comparatively modest results observed on the GO dataset. The inherent complexity of the GO dataset is a significant contributing factor to the suboptimal performance of our proposed method. Indeed, unlike the Co-citation and YeastNet datasets, the GO data presents relatively more complexity with its hierarchical and multifaceted relationships between genes. The intricate patterns within the GO dataset may require additional method refinement to effectively capture them, particularly when contrasted with the simpler pairwise relationships found in the other datasets. This can be considered as a limitation of the proposed method.

5.1.4.4 GRN graph and gene networks metrics across all datasets

Table 11 reports an example of gene associations as inferred by different algorithms. The GRN inferred by DCSA-QAR is described in Figure 8, showing 18 associations; and some of these are shared with other algorithms. The number of these shared associations can be read in the second line of Table 11. For example, in comparison with GarNet$_1$, 3/20 means that there are 3 associations (i.e., SWI5-CLB1, CLB1-CLB2, CLN1-CLB1) shared with DCSA-QAR, out of 20 associations generated by GarNet$_1$. Table 12 gives a summary of rankings of all algorithms and the metrics which they perform best with, for each dataset, on the one hand, and across all datasets, on the other hand.

Table 11 Gene–gene associations mined by DCSA-QAR

Id	*DCSA-QAR*	*GarNet$_1$*	*GarNet$_2$*	*GRNCOP*	*RegNet*	*Soinov et al.*
		(3/20)	*(3/16)*	*(3/15)*	*(2/7)*	*(2/6)*
1	CDC28-CLB6					
2	MBP1-SWI6					
3	CDC34-CLB6					
4	SWI5-CLB1	✓	✓	✓	✓	
5	SWI4-SIC1					
6	HCT1-CDC20					
7	CLB1-CLB2	✓	✓	✓	✓	✓
8	SIC1-CLB6					
9	SWI6-CLB6					
10	SWI6-CLB5					
11	CLB4-CLB1					
12	CLB4-CLB2					
13	CLN1-CLB1	✓				
14	CLN1-CLB5					
15	CLN2-CDC28					
16	CLB5-CDC53		✓			
17	CLB5-SWI4					
18	CLB5-CLN2			✓		✓

Figure 8 Gene association network obtained by DCSA-QAR (see online version for colours)

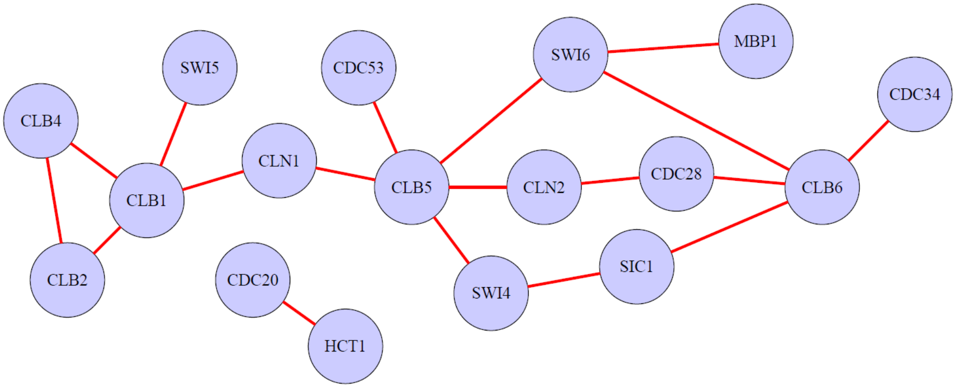


Table 12 Algorithms overall ranking for all datasets

a Ranking of algorithms when using Co-citation dataset

Ranking	1	2		4–6		
Algorithm	DCSA-QAR	GarNet$_1$	RegNet	GRNCOP	Soinov et al.	GarNet$_2$
Best metric achieved	Precision Specificity Score	Sensitivity Accuracy	Precision Specificity	None		

b Ranking of algorithms when using YeastNet dataset

Ranking	1	2	3	4–6		
Algorithm	GarNet$_1$	DCSA-QAR	RegNet	GRNCOP	Soinov et al.	GarNet$_2$
Best metric achieved	Precision Specificity Sensitivity Precision	Precision Specificity Score	Precision Specificity	None		

c Ranking of algorithms when using GO dataset

Ranking	1	2		4–6		
Algorithm	GarNet$_1$	Soinov et al.	GarNet$_2$	DCSA-QAR	GRNCOP	RegNet
Best metric achieved	Sensitivity Accuracy	Specificity	Precision	None		

d Ranking of algorithm by average number of mined associations

Ranking	1	2	3	4		6
Algorithm	GarNet$_1$	DCSA-QAR	RegNet	GarNet$_2$	GRNCOP	Soinov et al.
Number of times where best metrics is achieved	8	6	4	1	1	0

e Ranking of algorithms across all 3 datasets

Ranking	1	2	3	4	5	6
Algorithm	GarNet$_1$	DCSA-QAR	GarNet$_2$	GRNCOP	RegNet	Soinov et al.

5.2 Second series experiments

In this section, we present the datasets and evaluation metrics employed to assess the performance of DCSA-QAR. In this second series of experiments, we conducted a comparative analysis with various information-theoretic methods, presented in Section 2.2.3, including ARACNE, CLR, MI3, MIDER, MRMSn, MRNET, PCA-CMI, RRMRNET, and RWRNET. The datasets are the DREAM3 dataset of simulated type of network and SOS network as a real type. Table 5 shows the details of datasets used in the second series of experiments.

5.2.1 Results with DREAM3 dataset

Tables 13, 14 and 15 summarise the results given by all competitors on DREAM3-10, DREAM3-50 and DREAM3-100 gene dataset, respectively.

Table 13 Comparison of the different methods' performances in the Dream3-10 gene dataset

	TP	*FP*	*TPR*	*FPR*	*Precision*	*Accuracy*
CLR	6	10	0.6	0.286	0.375	0.689
ARACNE	6	6	0.6	0.171	0.5	0.778
MRNET	6	12	0.6	0.343	0.333	0.644
MI3	8	6	0.8	0.171	0.571	0.822
MIDER	-	-	-	-	-	-
MRMSn	9	1	0.9	0.029	0.9	0.956
RRMRNET	**10**	**0**	**1**	**0**	**1**	**1**
PCA-CMI	9	1	0.9	0.029	0.9	0.956
RWRNET	8	1	0.8	0.029	0.889	0.933
DCSA-QAR	6	2	0.6	0.057	0.75	0.867

Table 14 Comparison of the different methods' performances in the Dream3-50 gene dataset

	TP	*FP*	*TPR*	*FPR*	*Precision*	*Accuracy*
CLR	19	165	0.247	0.144	0.103	0.818
ARACNE	13	125	0.169	0.109	0.094	0.846
MRNET	21	215	0.273	0.187	0.089	0.779
MI3	21	68	0.273	0.059	0.236	0.899
MIDER	4	79	0.052	0.069	0.048	0.876
MRMSn	21	**17**	0.273	0.015	0.553	0.94
RRMRNET	**38**	56	**0.494**	0.049	0.404	0.922
PCA-CMI	25	19	0.325	0.017	**0.568**	0.942
RWRNET	29	16	0.377	**0.014**	0.644	**0.948**
DCSA-QAR	10	65	0.130	0.057	0.133	0.892

Table 15 Comparison of the different methods' performances in the Dream3-100 gene dataset

	TP	*FP*	*TPR*	*FPR*	*Precision*	*Accuracy*
CLR	39	713	0.235	0.149	0.052	0.830
ARACNE	20	417	0.121	0.087	0.046	0.886
MRNET	49	984	0.295	0.206	0.047	0.778
MI3	27	165	0.163	0.035	0.141	0.939
MIDER	13	80	0.078	0.017	0.140	0.953
MRMSn	-	-	-	-	-	-
RRMRNET	**92**	238	**0.554**	0.05	0.28	0.937
PCA-CMI	70	64	0.422	0.013	0.522	0.968
RWRNET	65	**50**	0.392	**0.01**	**0.565**	**0.969**
DCSA-QAR	5	64	0.030	0.013	0.072	0.955

5.2.1.1 Discussion

- DREAM3-10 gene dataset: In terms of TP, and TPR, DCSA-QAR archives the same results as CLR, ARACNE and MRNET. In FP, FPR, precision, and accuracy, DCSA-QAR is better than CLR, ARACNE, MRNET and MI3.

- DREAM3-50 gene dataset: In terms of TP and TPR, DCSA-QAR archives better results than MIDER. In FP and FPR, DCSA-QAR achieves better results than CLR, ARACNE, MRNET, MI3 and MIDER. In precision, DCSA-QAR achieves better results than CLR, ARACNE, MRNET and MIDER . In accuracy, DCSA-QAR is third, just after RWRNET and PCA-CMI.
- DREAM3-100 gene dataset: The worst results obtained here by DCSA-QAR in TP and TPR are compensated by the second best results in FP and FPR, just after RWRNET. In precision, DCSA-QAR is better than CLR, ARACNE, MRNET. Furthermore, DCSA-QAR is third in accuracy, just after RWRNET and PCA-CMI.

5.2.2 Results with SOS dataset

Table 16 gives the performance analysis of all competing methods, on SOS dataset. The acceptable results obtained by DCSA-QAR on the DREAM3 dataset above are compensated by better results on the SOS dataset. The proposed approach outperforms other methods in terms of accuracy, yielding the best results only seconded by PCA-CMI. Additionally, our method exhibited the highest performance in terms of TPR, suggesting that our approach infers more true positive associations. Additionally, DCSA-QAR demonstrated the second best performance in terms of true positives, after PCA-CMI.

Real networks typically possess intricate structures and closely interconnected regulatory relationships, making their inference challenging. Despite these complexities, the proposed method performed well in the SOS network, particularly in identifying true regulatory relationships. This outcome shows the effectiveness of our method, making it well-suited for inferring real networks better than simulated ones.

Table 16 Comparison of the different methods' performances in the SOS dataset

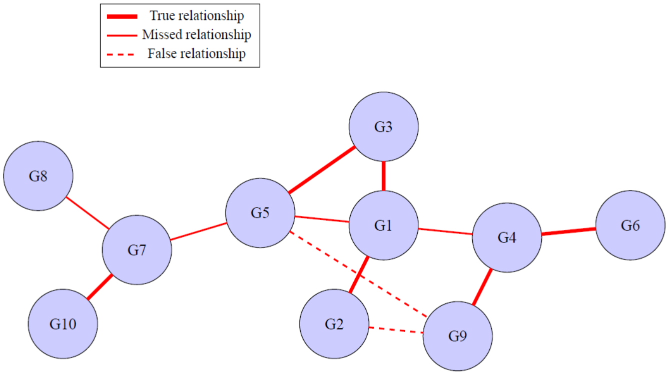
	TP	*FP*	*TPR*	*FPR*	*Precision*	*Accuracy*
CLR	12	5	0.5	0.417	0.706	0.528
ARACNE	7	3	0.292	0.25	0.7	0.444
MRNET	17	6	0.708	0.5	0.739	0.639
MI3	9	5	0.375	0.417	0.643	0.444
MIDER	-	-	-	-	-	-
MRMSn	10	2	0.417	0.167	0.833	0.556
RRMRNET	10	2	0.417	0.167	0.833	0.556
PCA-CMI	19	3	**0.920**	0.250	0.84	**0.778**
RWRNET	15	**1**	0.625	0.083	**0.938**	0.722
DCSA-QAR	**21**	5	0.875	0.417	0.808	**0.778**

5.2.3 Examples of GRN inference with DCSA-QAR

5.2.3.1 GRN obtained for DREAM3-10 gene dataset

Figure 9 shows the GRN obtained with DREAM3-10 genes. It shows the inference of 6 true associations out of 10 with 2 false ones. DCSA-QAR discovers 60% of the true associations.

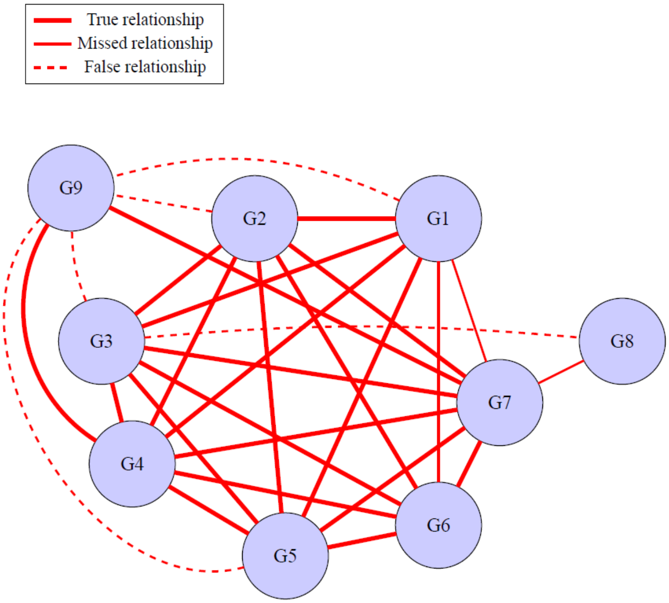
Figure 9 Gene association network obtained by DCSA-QAR from Dream-10 gene dataset (see online version for colours)



5.2.3.2 GRN obtained for SOS dataset

Figure 10 shows the GRN obtained with the SOS dataset. It shows the inference of 19 true associations out of 22 with only 5 false ones. DCSA-QAR discovers 86% of the true associations.

Figure 10 Gene association network obtained by DCSA-QAR from SOS dataset (see online version for colours)



5.3 Discussion of dataset influence on GRN inference results

Our current understanding suggests that there is no existing theory that can reliably forecast how a particular GRN inference algorithm will behave on a given dataset.

In this context, tangible insights and results are achievable solely through empirical experiments and practical testing. From the results reported above, we concluded that our method gave mitigated results: excellent on some datasets (Co-citation, YeastNet, SOS), and average on others (GO and DREAM3). Here, we outline some factors that could influence the performance of GRN inference algorithms, as far as datasets are concerned.

1 On the algorithm side

   a Parameter tuning and optimisation parameters: as our method involves parameter tuning, the optimal parameter settings may vary for different datasets. To maintain a fair and equitable comparison with other competitors, and in order to promote transparency and impartiality in our evaluation, we have employed uniform parameters consistently across all datasets. The experimental findings indicate that while these parameters are appropriate for specific datasets (Co-citation, YeastNet, SOS), their efficacy is diminished when applied to others (GO and DREAM3), leading to suboptimal performance.

   b Algorithm sensitivity: GRN inference methods are based on different underlying assumptions. The appropriateness of these assumptions can vary across datasets, affecting the performance of the method. For example, some methods assume linear relationships between genes, while others account for nonlinear interactions, undirected vs. directed edges, pairwise vs. multiple genes, among others (Kontio et al., 2020).

2 On the data side

   a Sample size: GRN inference methods might face challenges with small datasets, as they may not accurately capture the complexity and diversity of gene expression patterns. Additionally, small sample sizes can result in overfitting, where the model excels on the training data but struggles to generalise to new datasets.

   b Batch effects: variations in experiments or technical biases observed across diverse datasets, commonly referred to as batch effects, have the potential to introduce systematic errors, influencing the efficacy of GRN inference methods.

As a result, developing a profound understanding of a dataset's characteristics, the biological context, and the strengths and limitations of various GRN inference methods is crucial. Such knowledge is essential for making informed decisions in selecting an appropriate approach and for effectively interpreting the outcomes. Given the absence of a universal method that excels across all datasets, it is advisable to undertake an evaluation of multiple methods. Additionally, considering ensemble approaches is recommended to further strengthen the robustness of GRN inference methods when dealing with datasets of diverse nature. This approach not only acknowledges the complexity of biological data but also contributes to the advancement of reliable and applicable GRN inference practices not only within the field of GRN inference methods but in data mining and bioinformatics at large.

6 Conclusions

To address the limited availability of ARM-based algorithms in GRN inference, a new metaheuristic algorithm is introduced: DCSA-QAR. It highlights the utilisation of the model-free approach provided by the ARM, integrated into a metaheuristic method. Specifically, the metaheuristic CSA is discretised using the C-BUDA. This latter aims to optimally split numerical attribute intervals by maximising class prediction. Starting with the formulation of the fitness function, we subsequently fine-tune the algorithm's initial parameter settings to ascertain the minimal thresholds necessary for the extracted QARs. Thereafter, the application of our algorithm is extended to prominent datasets encompassing gene information derived from microarray data. Two series of experiments are undertaken involving the comparison with metaheuristic methods, on the one hand, and information-theoretic methods, on the other hand. In the first series of experiments, DCSA-QAR occupies the following overall ranking positions w.r.t. five state-of-the-art algorithms:

- DSCA-QAR is the best for Co-citation dataset
- second best for YeastNet dataset
- second best across all datasets combined
- second best in average number of mined associations.

In the second series of experiments, the DCSA-QAR presents the following results:

- average results in DREAM3 datasets
- best results in accuracy and true positives, on SOS real dataset.

It can be concluded that the proposed method yields commendable outcomes across various facets of performance, encompassing widely recognised quality metrics of network assessment, achieving overall good performance while scaling well in GRNs inference in comparison with state-of-the-art algorithms. As a future work, an additional layer incorporating embedded prior knowledge and relevant molecular data might improve the proposed tool.

References

Agrawal, R., Imieliński, T. and Swami, A. (1993) 'Mining association rules between sets of items in large databases', *SIGMOD Rec.*, Vol. 22, No. 2, pp.207–216.

Alagukumar, S., Vanitha, C.D.A. and Lawrance, R. (2020) 'Clustering of association rules on microarray gene expression data', *Advanced Computing and Intelligent Engineering*, pp.85–97, Springer, Singapore.

Alves, R., Rodríguez-Baena, D.S. and Aguilar-Ruiz, J.S. (2010) 'Gene association analysis: a survey of frequent pattern mining from gene expression data', *Briefings in Bioinformatics*, Vol. 11, No. 2, pp.210–224.

Askarzadeh, A. (2016) 'A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm', *Computers & Structures*, Vol. 169, No. 2, pp.1–12.

Babtie, A.C., Stumpf, M.P. and Thorne, T. (2021) 'Gene regulatory network inference', in Wolkenhauer, O. (Ed.): *Systems Medicine*, pp.86–95, Academic Press, Oxford.

Basso, K., Margolin, A.A., Stolovitzky, G., Klein, U., Dalla-Favera, R. and Califano, A. (2005) 'Reverse engineering of regulatory networks in human B cells', *Nature Genetics*, Vol. 37, No. 4, pp.382–390.

Cantone, I., Marucci, L., Iorio, F., Bansal, M., Belcastro, V., Ricci, M., Santini, S., di Bernardo, M., di Bernardo, D. and Cosma, M. (2009) 'A yeast synthetic network for in vivo assessment of reverse-engineering and modeling approaches', *Cell*, Vol. 137, No. 1, pp.172–181.

Chai, L.E., Loh, S.K., Low, S.T., Mohamad, M.S., Deris, S. and Zakaria, Z. (2014) 'A review on the computational approaches for gene regulatory network construction', *Computers in Biology and Medicine*, Vol. 48, No. 1, pp.55–65.

Cho, R.J., Campbell, M.J., Winzeler, E.A., Steinmetz, L., Conway, A., Wodicka, L., Wolfsberg, T.G., Gabrielian, A.E., Landsman, D., Lockhart, D.J. and Davis, R.W. (1998) 'A genome-wide transcriptional analysis of the mitotic cell cycle', *Molecular Cell*, Vol. 2, No. 1, pp.65–73.

Dwight, S.S., Harris, M.A., Dolinski, K., Ball, C.A., Binkley, G., Christie, K.R., Fisk, D.G., Issel-Tarver, L., Schroeder, M., Sherlock, G., Sethuraman, A., Weng, S., Botstein, D. and Cherry, J.M. (2002) 'Saccharomyces genome database (SGD) provides secondary gene annotation using the Gene Ontology (GO)', *Nucleic Acids Research*, Vol. 30, No. 1, pp.69–72.

Faith, J.J., Hayete, B., Thaden, J.T., Mogno, I., Wierzbowski, J., Cottarel, G., Kasif, S., Collins, J.J. and Gardner, T.S. (2007) 'Large-scale mapping and validation of escherichia coli transcriptional regulation from a compendium of expression profiles', *PLOS Biology*, Vol. 5, No. 1, pp.1–13.

Gallo, C., Carballido, J. and Ponzoni, I. (2011) 'Discovering time-lagged rules from microarray data using gene profile classifiers', *BMC Bioinformatics*, Vol. 12, No. 1, p.123.

Kabir, M.M.J., Xu, S., Kang, B.H. and Zhao, Z. (2017) 'A new multiple seeds based genetic algorithm for discovering a set of interesting Boolean association rules', *Expert Systems with Applications*, Vol. 74, No. 1, pp.55–69.

Kontio, J.A.J., Rinta-aho, M.J. and Sillanpää, M.J. (2020) 'Estimating linear and nonlinear gene coexpression networks by semiparametric neighborhood selection', *Genetics*, Vol. 215, No. 3, pp.597–607.

Kuzmanovski, V., Todorovski, L. and Džeroski, S. (2018) 'Extensive evaluation of the generalized relevance network approach to inferring gene regulatory networks', *GigaScience*, Vol. 7, No. 11, p.giy118.

Ledmi, M., Moumen, H., Siam, A., Houassi, H. and Azizi, N. (2020) 'A discrete crow search algorithm for mining quantitative association rules', *International Journal of Swarm Intelligence Research*, Vol. 13, No. 4, pp.101–124.

Lee, I., Date, S.V., Adai, A.T. and Marcotte, E.M. (2004) 'A probabilistic functional network of yeast genes', *Science (New York, N.Y.)*, Vol. 306, No. 5701, pp.1555–1558.

Lee, I., Li, Z. and Marcotte, E.M. (2007) 'An improved, bias-reduced probabilistic functional gene network of baker's yeast, Saccharomyces cerevisiae', *PLOS One*, Vol. 2, No. 10, pp.1–15.

Liu, W., Sun, X., Peng, L., Zhou, L., Lin, H. and Jiang, Y. (2020) 'RWRNET: a gene regulatory network inference algorithm using random walk with restart', *Frontiers in Genetics*, Vol. 11, No. 1.

Liu, W., Zhu, W., Liao, B., Chen, H., Ren, S. and Cai, L. (2017) 'Improving gene regulatory network structure using redundancy reduction in the mrnet algorithm', *RSC Adv.*, Vol. 7, No. 7, pp.23222–23233.

Liu, W., Zhu, W., Liao, B. and Chen, X. (2016) 'Gene regulatory network inferences using a maximum-relevance and maximum-significance strategy', *PLOS One*, Vol. 11, No. 11, pp.1–19.

Luo, W., Hankenson, K.D. and Woolf, P.J. (2008) 'Learning transcriptional regulatory networks from high throughput gene expression data using continuous three-way mutual information', *BMC Bioinformatics*, Vol. 9, No. 1, pp.1–15.

Manju and Kant, C. (2015) 'Mining association rules directly using aco without generating frequent itemsets', *2015 International Conference on Energy Systems and Applications*, pp.390–395.

Marbach, D., Prill, R.J., Schaffter, T., Mattiussi, C., Floreano, D. and Stolovitzky, G. (2010) 'Revealing strengths and weaknesses of methods for gene network inference', *Proceedings of the National Academy of Sciences*, Vol. 107, No. 14, pp.6286–6291.

Margolin, A., Wang, K., Lim, W., Kustagi, M., Nemenman, I. and Califano, A. (2006) 'Reverse engineering cellular networks', *Nature Protocols*, Vol. 1, No. 2, pp.662–671.

Martínez, R., Pasquier, N. and Pasquier, C. (2008) 'GenMiner: mining non-redundant association rules from integrated gene expression data and annotations', *Bioinform.*, Vol. 24, No. 22, pp.2643–2644.

Martínez-Ballesteros, M., Nepomuceno-Chamorro, I.A. and Riquelme, J.C. (2014) 'Discovering gene association networks by multi-objective evolutionary quantitative association rules', *Journal of Computer and System Sciences*, Vol. 80, No. 1, pp.118–136.

Meyer, P.E., Kontos, K., Lafitte, F. and Bontempi, G. (2007) 'Information-theoretic inference of large transcriptional regulatory networks', *EURASIP J. Bioinformatics Syst. Biol.*, Vol. 2007, No. 1, pp.1–9.

Mitra, S., Biswas, S. and Acharyya, S. (2021) 'Application of meta-heuristics on reconstructing gene regulatory network: a Bayesian model approach', *IETE Journal of Research*, Vol. 69, No. 7, pp.4184–4195.

Mousavian, Z., Kavousi, K. and Masoudi-Nejad, A. (2016) 'Information theory in systems biology. Part I: gene regulatory and metabolic networks', *Seminars in Cell & Developmental Biology*, Vol. 51, No. 1, pp.3–13.

Nepomuceno-Chamorro, I.A., Aguilar-Ruiz, J.S. and Riquelme, J.C. (2010) 'Inferring gene regression networks with model trees', *BMC Bioinformatics*, Vol. 11, No. 1, p.517.

Pasquier, N., Bastide, Y., Taouil, R. and Lakhal, L. (1999) 'Efficient mining of association rules using closed itemset lattices', *Inf. Syst.*, Vol. 24, No. 1, pp.25–46.

Ponzoni, I., Azuaje, F., Augusto, J. and Glass, D. (2007) 'Inferring adaptive regulation thresholds and association rules from gene expression data through combinatorial optimization learning', *IEEE/ACM Trans. Comput. Biology Bioinform.*, Vol. 4, No. 4, pp.624–634.

Pušnik, Ž., Mraz, M., Zimic, N. and Moškon, M. (2022) 'Review and assessment of Boolean approaches for inference of gene regulatory networks', *Heliyon*, Vol. 8, No. 8, p.e10222.

Ronen, M., Rosenberg, R., Shraiman, B.I. and Alon, U. (2002) 'Assigning numbers to the arrows: parameterizing a gene regulation network by using accurate expression kinetics', *Proceedings of the National Academy of Sciences*, Vol. 99, No. 16, pp.10555–10560.

Soinov, L., Krestyaninova, M. and Brazma, A. (2003) 'Towards reconstruction of gene networks from expression data by supervised learning', *Genome Biology*, Vol. 4, No. 1, p.R6+.

Someren, E. P. v., Wessels, L. F.A. and Reinders, M. J.T. (2000) 'Linear modeling of genetic networks from experimental data', *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology*, AAAI Press, pp.355–366.

Spellman, P., Sherlock, G., Zhang, M., Iyer, V., Anders, K., Eisen, M., Brown, P., Botstein, D. and Futcher, B. (1999) 'Comprehensive identification of cell cycle-regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization', *Molecular Biology of the Cell*, Vol. 9, No. 12, pp.3273–3297.

Srikant, R. and Agrawal, R. (1996) 'Mining quantitative association rules in large relational tables', *SIGMOD Rec.*, Vol. 25, No. 2, pp.1–12.

Vengateshkumar, R., Alagukumar, S. and Lawrance, R. (2020) 'Boolean association rule mining on microarray gene expression data', *Advanced Computing and Intelligent Engineering*, pp.99–111, Springer, Singapore.

Vijesh, N., Chakrabarti, S.K. and Sreekumar, J. (2013) 'Modeling of gene regulatory networks: a review', *Journal of Biomedical Science and Engineering*, Vol. 6, No. 2, pp.223–231.

Villaverde, A.F., Ross, J., Morán, F. and Banga, J.R. (2014) 'Mider: network inference with mutual information distance and entropy reduction', *PLOS One*, Vol. 9, No. 5, pp.1–15.

Yan, D., Zhao, X., Lin, R. and Bai, D. (2019) 'PPQAR: Parallel PSO for quantitative association rule mining', *Peer-to-Peer Networking and Applications*, Vol. 8, No. 1, p.53.

Zhang, X., Zhao, J., Hao, J-K., Zhao, X-M. and Chen, L. (2015) 'Conditional mutual inclusive information enables accurate quantification of associations in gene regulatory networks', *Nucleic Acids Research*, Vol. 43, No. 5, pp.e31–e31.

Zhang, X., Zhao, X-M., He, K., Lu, L., Cao, Y., Liu, J., Hao, J-K., Liu, Z-P. and Chen, L. (2011) 'Inferring gene regulatory networks from gene expression data by path consistency algorithm based on conditional mutual information', *Bioinformatics*, Vol. 28, No. 1, pp.98–104.

Zhao, M., He, W., Tang, J., Zou, Q. and Guo, F. (2021) 'A comprehensive overview and critical evaluation of gene regulatory network inference technologies', *Briefings in Bioinformatics*, Vol. 22, No. 5, p.bbab009.

Zhu, X. (2009) *Quantitative Association Rules*, pp.2240–2244, Springer US, Boston, MA.