Inference of gene regulatory networks with multi-objective cellular genetic algorithm

José García-Nieto[1,*], Antonio J. Nebro, José F. Aldana-Montes

Dept. de Lenguajes y Ciencias de la Computación and Instituto de Investigación Biomédica de Málaga (IBIMA), University of Malaga, ETSI Informática, Campus de Teatinos, Malaga 29071, Spain

Keywords:

Multi-objective optimization
Cellular genetic algorithms
Gene regulatory networks
DREAM challenge

A B S T R A C T

Reverse engineering of biochemical networks remains an important open challenge in computational systems biology. The goal of model inference is to, based on time-series gene expression data, obtain the sparse topological structure and parameters that quantitatively understand and reproduce the dynamics of biological systems. In this paper, we propose a multi-objective approach for the inference of S-System structures for Gene Regulatory Networks (GRNs) based on Pareto dominance and Pareto optimality theoretical concepts instead of the conventional single-objective evaluation of Mean Squared Error (MSE). Our motivation is that, using a multi-objective formulation for the GRN, it is possible to optimize the sparse topology of a given GRN as well as the kinetic order and rate constant parameters in a decoupled S-System, yet avoiding the use of additional penalty weights. A flexible and robust Multi-Objective Cellular Evolutionary Algorithm is adapted to perform the tasks of parameter learning and network topology inference for the proposed approach. The resulting software, called MONET, is evaluated on real-based academic and synthetic time-series of gene expression taken from the DREAM3 challenge and the IRMA in vivo datasets. The ability to reproduce biological behavior and robustness to noise is assessed and compared. The results obtained are competitive and indicate that the proposed approach offers advantages over previously used methods. In addition, MONET is able to provide experts with a set of trade-off solutions involving GRNs with different typologies and MSEs.

1. Introduction

In computational systems biology, the inference of biochemical networks from time-series of gene expression data remains an important open challenge (Iglesias-Martinez et al., 2016). The main goal is to obtain the sparse topological structure and parameters that quantitatively understand and reproduce the dynamics of biological systems. However, although it is possible to computationally predict genetic interaction networks, their precision depends on the characteristics of the model used, as well as the availability and quality of the expression data, in terms of the noise they contain.

In this study, we focus on the S-System model (Savageau, 2010) as it provides a good compromise between biological relevancy and mathematical flexibility. The S-System models the dynamics of a network by means of an Ordinary Differential Equations (ODE) system, which is powerful enough to capture complex dynamics of genetic regulations. In the inference of GRNs modeled by S-System, there are two major challenges that have to be addressed: (1) detecting the sparse topological architecture that is commonly seen in biological networks, and (2) tuning the kinetic order and rate constant parameters from a limited amount of gene expression data that usually show a significant percentage of noise. These make GRN to be a complex global optimization problem (Kikuchi et al., 2003), which requires the use of efficient optimization algorithms to deal with (Hitoshi Iba, 2016; Palafox et al., 2013; Noman et al., 2015).

In this regard, most of existing approaches have applied sums of magnitude of kinetic orders as a penalty term in aggregative evaluation functions based on conventional Mean Squared Error (MSE) (Lee and Hsiao, 2012; Liu and Wang, 2008a; Palafox et al., 2013). However, this requires having to tune at least one suitable penalty weight in the fitness evaluation. To the best of our knowledge, until now no proposals have been published for avoiding penalty weights when inferring GRNs.

With this motivation, in this paper we adopt the following hypothesis: *A Multi-Objective Optimization approach, based on Pareto dominance and Pareto optimality theoretical concepts, may exhibit a successful performance for the inference of S-System structures for GRNs, by exploiting*

* Corresponding author.
E-mail addresses: jnieto@lcc.uma.es (J. García-Nieto), antonio@lcc.uma.es (A.J. Nebro), jfam@lcc.uma.es (J.F. Aldana-Montes).
[1] The authors contributed equally to this work..

time-series of expression data. The inference of GRNs is therefore formulated as a multi-objective optimization problem (MOP) to simultaneously minimize two objectives: (1) the MSE using decoupled S-System, and (2) a Topology Regularization (TR) value. Consequently, it is possible to tune the kinetic order and rate constant parameters (in a decoupled S-System), at the same time that the sparse topology of a given network is obtained, but avoiding the use of penalty weights.

A key contribution of this study is presented as a result of our hypothesis, since we can now take advantage of the different learning procedures induced by the subfamily of multi-objective techniques. Moreover, we are now able to provide experts in biology systems with sets of trade-off solutions, instead of just one, allowing them to decide between a range of network topologies.

To test our hypothesis, a flexible and robust adaption of the Multi-Objective Cellular Genetic Algorithm (MOCell) (Nebro et al., 2007) has been developed to perform the tasks of parameter learning and network topology inference, in the scope of GRNs. The main idea underpinning this algorithm, called MONET, is its ability to perform a wider exploration of the search space, which results in robust solutions modeling precise networks. The performance of MONET is evaluated on time-series of gene expression data from synthetic and benchmarking networks of the DREAM3 challenge (Prill et al., 2010) based on real organisms (*E. coli* and Yeast), as well as the IRMA (Cantone et al., 2009) in vivo datasets. The ability to reproduce biological behavior and robustness to noise are assessed in comparison with other prominent techniques in the current state of the art.

This paper is organized as follows: Section 2 presents a series of related works in the state of the art. In Section 3, background concepts with regards to multi-objective optimization are explained. Models and methods are described in Section 4, where the proposed approach is also detailed. Section 5 reports the experimentation methodology and Section 6 analyzes the results obtained. Finally, Section 7 contains concluding remarks and future lines of research.

2. Related works

Gene regulatory networks have been modeled with different tools, such as: Boolean networks (Akutsu, 2003), by means of which gene states are represented as members of a binomial domain, and Bayesian networks (Friedman et al., 2004), which assign conditional probabilities to the regulation parameters of each gene. Other recent proposals focused on combinations of specific machine learning models, such as: Boosting approaches with kernel-based autoregressive models (Lim et al., 2013), formal on/off with non-parametric models based on decision tress (Huynh-Thu and Sanguinetti, 2015), and recurrent neural network based hybrid models (Khalid Raza and Mansaf Alam, 2016).

In this study, we concentrate on the S-System model (Savageau, 2010) as it provides a good compromise between biological relevancy and mathematical flexibility. The S-System models the dynamics of a network by means of an Ordinary Differential Equations (ODE) system, which is powerful enough to capture complex dynamics of genetic regulations. However, ODE-based models require a large number of parameters to be tuned to properly reconstruct a target network. Therefore, its application is, to date, limited to small/medium sized networks, because of the course of dimensionality (Hitoshi Iba, 2016). Even for small networks, current approaches are able to find correct regulations, but show poor behavior when inferring topological structures.

In order to deal with these issues, recent studies (Lee and Hsiao, 2012; Palafox et al., 2013) have used a decoupled version to disassociate forms that are useful in the reconstruction of small networks. For the inference of correct parameters in an S-System, researchers have traditionally used global optimization search techniques, such as: Genetic Algorithms (GAs) (Kikuchi et al., 2003; Sirbu et al., 2010; Spieth et al., 2005b), Differential Evolution (DE) (Noman and Iba, 2007) and Particle Swarm Optimization (PSO) (Lee and Hsiao, 2012; Palafox et al., 2013; Nobile and Iba, 2015). These approaches have successfully performed in finding correct regulations, but they have also shown a number of false regulations. Therefore, dealing with small/medium sized networks is still an open problem.

Other similar studies employed multi-objective approaches for the inference of GRNs, like those presented in (Chen and Zou, 2016), (Liu and Wang, 2008b) (Spieth et al., 2005a), and (Sirbu et al., 2010). Unfortunately, most of these proposals are preliminary applications of multi-objective Evolutionary Algorithms with different objectives (sometimes using aggregative formulations (Chen and Zou, 2016) (Liu and Wang, 2008b)), but considered neither a decoupled S-System model nor topology regularization. The later approach (Sirbu et al., 2010) used a completely different multi-objective model, as it only considered the MSE of each gene as objective, so (as argued by the authors) only networks with two genes could be tackled.

3. Background concepts

With the aim of making this paper more self-contained, a series of definitions in the context of multi-objective optimization are presented.

3.1. Pareto dominance definition

Optimization problems often need to be addressed by considering two or more objective functions at the same time, being all of them equally important. They are the so called Multi-Objective Optimization Problems (MOP), and a formal definition of them is as follows. Without loss of generality, we assume that minimization is the goal for all the objectives.

Definition 3.1 (*Multi-objective optimization problem*). Find a vector $\vec{x}^* = [x_1^*, x_2^*, ..., x_n^*]$ which satisfies the m inequality constraints $g_i(\vec{x}) \geq 0, i = 1, 2, ..., m$, the p equality constraints $h_i(\vec{x}) = 0, i = 1, 2, ..., p$, and minimizes the vector function $\vec{f}(\vec{x}) = [f_1(\vec{x}), f_2(\vec{x}), ..., f_k(\vec{x})]^T$, where $\vec{x} = [x_1, x_2, ..., x_n]^T$ is the vector of decision variables.

The set of all values satisfying the constraints defines the *feasible region* Ω and any point $\vec{x} \in \Omega$ is a *feasible solution*. We seek the *Pareto optima*.

Definition 3.2 (*Pareto optimality*). for every $\vec{x} \in \Omega$ and $I = \{1, 2, ..., k\}$ either $\forall_{i \in I} (f_i(\vec{x}) = f_i(\vec{x}^*))$ or there is at least one $i \in I$ such that $f_i(\vec{x}) > f_i(\vec{x}^*)$.

This definition states that $\vec{x}^*$ is Pareto optimal if no other feasible vector $\vec{x}$ exists which would improve some criteria without causing a simultaneous worsening in at least one other criterion.

Definition 3.3 (*Pareto dominance*). A vector $\vec{u} = (u_1, ..., u_k)$ is said to dominate $\vec{v} = (v_1, ..., v_k)$ (denoted by $\vec{u} \preccurlyeq \vec{v}$) if and only if $\vec{u}$ is partially less than $\vec{v}$. $\forall\, i \in \{1, ..., k\},\ u_i \leq v_i \ \wedge\ \exists\, i \in \{1, ..., k\}:\ u_i < v_i$.

Definition 3.4 (*Pareto optimal set*). For a given MOP , the Pareto optimal set is defined as $\mathcal{P}^* = \{\vec{x} \in \Omega | \neg\, \exists\, \vec{x}' \in \Omega, \vec{f}(\vec{x}') \preccurlyeq \vec{f}(\vec{x})\}$.

For a given MOP and its Pareto optimal set $\mathcal{P}^*$, the Pareto front is defined as $\mathcal{PF}^* = \{\vec{f}(\vec{x}), \vec{x} \in \mathcal{P}^*\}$.

Obtaining the Pareto front of a MOP is the main goal of multi-objective optimization. The representation of the Pareto set in the objective space is the Pareto front, which in general is presented in a graphic so that the expert in the problem, i.e., the decision maker, can choose the best trade-off solution. In theory, a Pareto front could contain a large number of (or even infinitely many) points. In practice, a usable approximate solution will only contain a limited number of them; thus, an important goal is that they should be as close as possible to the exact Pareto front (*convergence*) and uniformly spread (*diversity*), otherwise, they will not be very useful to the decision maker.

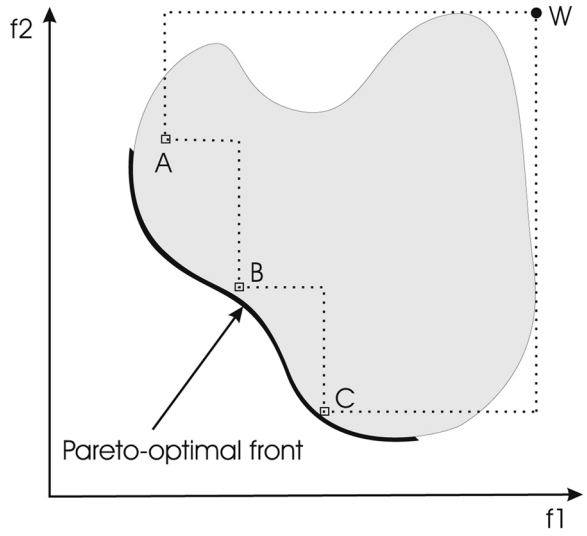


Fig. 1. The hypervolume (delimited by the dotted line) enclosed by the non-dominated solutions.

3.2. Hypervolume

Contrary to single/mono-objective optimization, where assessing the performance of a metaheuristic mainly requires observing the best value yielded by an algorithm (*i.e.*, the lower the better, in the case of minimization), in multi-objective optimization, this is not applicable. Instead, an approximation set to the optimal Pareto front of the problem is computed. Two properties are usually required: convergence and a uniform diversity. A number of quality indicators for measuring these two criteria have been proposed (Deb, 2001).

In this work, we have concentrated on the hypervolume (I_{HV}) quality indicator, which takes into account both convergence and diversity. I_{HV} calculates the n-dimensional space covered by members of a non-dominated set of solutions Q, e.g., the region enclosed by the discontinuous line in Fig. 1, $Q = \{A, B, C\}$, for problems where all objectives are to be minimized. Mathematically, for each solution $i \in Q$, a hypercube v_i is constructed with a reference point W and the solution i as the diagonal corners of the hypercube. The reference point can simply be found by constructing a vector of worst objective function values. Thereafter, a union of all hypercubes is found, and its hypervolume (I_{HV}) is calculated:

$$I_{\mathrm{HV}} = \mathrm{volume}\left(\bigcup_{i=1}^{|Q|} v_i\right) \tag{1}$$

Solution fronts with large values of I_{HV} are desirable.

4. Models and methods

In biological networks inference, the main goal is to capture the dynamics of biological systems from the time-series of gene expression datasets obtained for a given pool of molecular species, in a given time period. Such dynamics can be mathematically modeled by the S-System (Voit, 2000) framework to represent a network as a set of differential equations:

$$f(t, X) = \begin{pmatrix} \alpha_1 \prod_{j=1}^{n+m} X_j^{g_{1j}} - \beta_1 \prod_{j=1}^{n+m} X_j^{h_{1j}} \\ \vdots \\ \alpha_n \prod_{j=1}^{n+m} X_j^{g_{nj}} - \beta_n \prod_{j=1}^{n+m} X_j^{h_{nj}} \end{pmatrix}, X(0) = X_0 \tag{2}$$

where X is a n-dimensional pool of elements and the m-dimensional independent variables are expressed as X_{n+j}, $j = 1, \cdots, m$. That is, X_i is the expression level of the *ith* gene. Parameters $\alpha_i, \beta_i \in \mathbb{R}_+^N$ are rate constants ($N = n + m$), and $g_{ij}, h_{ij} \in \mathbb{R}^{N \times N}$ are kinetic orders that regulate the synthesis and degradation of X_i influenced by X_j.

4.1. Decoupled S-system

For the evaluation of S-System models, numerical methods such as Runge-Kutta have been traditionally applied, since they are highly accurate in finding parameters that lead the model to fit time-series curves of gene expression data values. However, numerical solutions depend on each of the $n + m$ variables to update, which often take a long time to calculate. In addition, evolutionary techniques such as GAs apply learning models based on the evaluation of S-System for all the candidate solutions in a population, throughout multiple iterations. Therefore, in these kinds of approaches, the larger the network, the harder the required computational effort is, since the complexity of Runge–Kutta method is inhered to solve a number of recursive simultaneous equations in each solution evaluation. To cope with this issue, we use a decoupling method based on the data collocation approach proposed by (Tsai and Wang, 2005), by means of which it is possible to calculate equations referring to independent genes, and hence to reduce the computational cost per evaluation. In collocation methods, dynamic variables X in Eq. (2) are spanned by a set of shape functions:

$$X(t) = \sum_{j=0}^{N} x(j)\phi_j(t) \tag{3}$$

with $x(j)$ being an expansion coefficient of $X(t)$ and $\phi_j(t)$ is a set of polynomial shape functions. The collocation method uses a linear Lagrange polynomial by introducing the experimental data (from time-series) for each gene when solving the S-System of the target network. Each new step in the numerical solution of the S-System is formulated as:

$$x_{n+1} = x_n + 0.5\eta(f[x_{n+1,\mathrm{exp}}, \theta] + f[x_n, \theta]) \tag{4}$$

where $x_{n+1,exp}(t)$ are the experimental values in gene expressions dataset at time t, $f[x_n, \theta]$ is the Eq. (2) evaluated for x_n, and θ is the set of tuning parameters $\{g_{ij}, h_{ij}, \alpha_i, \beta_i | i, j = 1 \cdots N\}$ in the S-System. Parameter η in Eq. (4) is a smoothness rate in order to control the approximation overshot.

4.2. Problem formulation

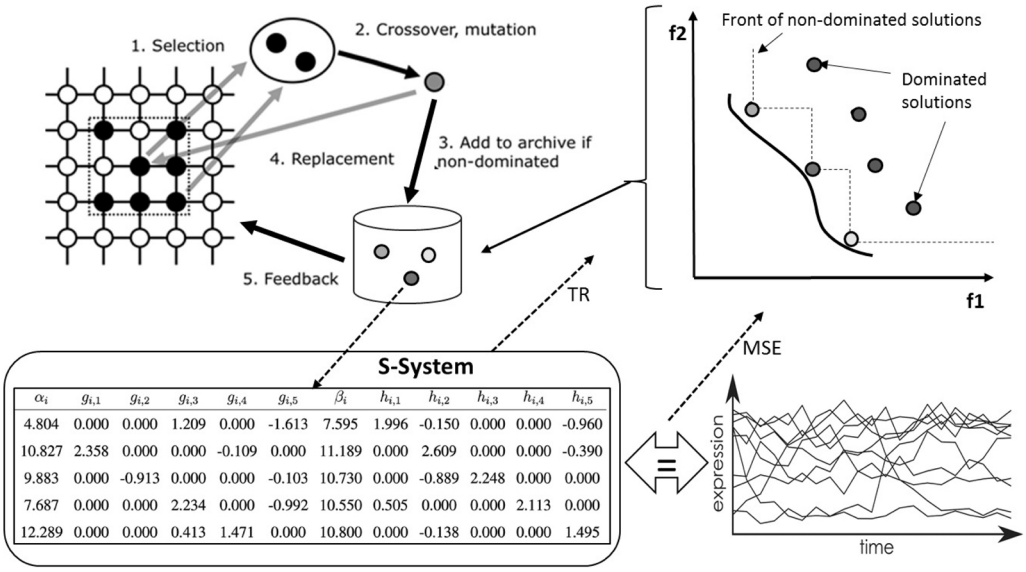
For the solution encoding, each candidate solution generated by MONET is arranged in a vector of real variables representing the tuning parameters: kinetic orders (g_{ij}, h_{ij}) and rate constants (α_i, β_i), in the S-System model. Fig. 2 (bottom-left) represents the structure of tunning parameters encoded within a solution vector.

4.2.1. Time-series estimation error

For the evaluation of candidate solutions, a widely used criterion is the discrepancy between the gene expression levels calculated numerically and those observed from time-series of system dynamics. (Tominaga et al., 2000) standardized the use of the Mean Squared Error (MSE) as the fitness function to evaluate each candidate solution in the S-System. MSE is formulated as a minimization function:

$$f^{\mathrm{MSE}} = \sum_{i=1}^{N} \sum_{k=1}^{M} \sum_{t=1}^{T} \left(\frac{X_{k,i}^{\mathrm{cal}}(t) - X_{k,i}^{\mathrm{exp}}(t)}{X_{k,i}^{\mathrm{exp}}(t)} \right)^2 \tag{5}$$

where $X_{k,i}^{\mathrm{cal}}(t)$ and $X_{k,i}^{\mathrm{exp}}(t)$ are the expression levels of gene i in the *kth* set of time courses at time t in the *calculated* and *experimental* data, respectively. M is the set of time-series considered in the evaluation, whereas T is the number of sampling points in the experimental data (gene expression values). The main goal is to find an optimized set of parameters θ that minimizes f^{MSE}.



α_i	$g_{i,1}$	$g_{i,2}$	$g_{i,3}$	$g_{i,4}$	$g_{i,5}$	β_i	$h_{i,1}$	$h_{i,2}$	$h_{i,3}$	$h_{i,4}$	$h_{i,5}$
4.804	0.000	0.000	1.209	0.000	-1.613	7.595	1.996	-0.150	0.000	0.000	-0.960
10.827	2.358	0.000	0.000	-0.109	0.000	11.189	0.000	2.609	0.000	0.000	-0.390
9.883	0.000	-0.913	0.000	0.000	-0.103	10.730	0.000	-0.889	2.248	0.000	0.000
7.687	0.000	0.000	2.234	0.000	-0.992	10.550	0.505	0.000	0.000	2.113	0.000
12.289	0.000	0.000	0.413	1.471	0.000	10.800	0.000	-0.138	0.000	0.000	1.495

Fig. 2. Graphical representation of the operation of MONET. The MOCell algorithm evolves by applying the reproduction operators: neighborhood selection, crossover, mutation, archive storage, and replacement. The new candidate solutions are used to set the S-System. After the evaluation of solutions, the Pareto front approximation is constituted by using those non-dominated solutions stored in the archive.

4.2.2. Attaining the network topology

MSE is an efficient measure to capture the efficient fitting of time-series curves by means of sampling values in optimization procedures. However, one of the major difficulties in the S-System based inference process lies in obtaining the proper topological structure of gene interactions that generates the observed dynamics. S-System model parameters show a high degree of freedom, which results in a great number of local minima in the search space of solutions that mimics the time courses very closely. Therefore, as experimented in (Kikuchi et al., 2003), a method attempting to reproduce the time dynamics often gets stuck on local optimum solutions and fails to obtain the true topological structure of the network.

In this regard, the use of pruning or penalty terms based on the Laplacian regularization in the basic MSE fitness function, is useful for attaining a sparse network topology in the canonical optimization problem (Kikuchi et al., 2003). However, because of high dimensionality, such fitness functions can be applied to small networks only.

Following the same notion, in (Noman and Iba, 2007) an additional pruning term for the fitness function is proposed, which aims at capturing the topological structure of the network as follows:

$$f_i = f_i^{\text{MSE}} + c \sum_{j=1}^{2N-I} (|K_{i,j}|) \tag{6}$$

where f_i^{MSE} is Eq. (5) but only referring to gene i and $K_{i,j}$ are obtained by sorting kinetic orders g_{ij} and h_{ij} all together in ascending order of their absolute values ($|K_{i,1}| \leq |K_{i,2}| \leq \cdots \leq |K_{i,2N}|$). I is the maximum allowed cardinality degree of the network and c is a penalty constant. This penalty term (second term of Eq. (6)), includes the maximum allowed number of gene interactions (edges) in the network and penalizes only when the number of genes that directly affect the *ith* gene is higher than the maximum cardinality I. As a consequence, this penalization will cause most of the genes to disconnect when their affecting kinetic orders have low values. This will decrease the number of false positives, while keeping the high number of true positives by gradually updating the relative importance of each considered kinetic order g_{ij} and h_{ij}, taking into account the cardinality. The performance of variable selection strongly depends on the regularization parameters of the $|K_{i,1}|$ penalty terms. There are other similar penalty terms applied in the literature in which synthetic and degradative regulations are considered separately (Kimura et al., 2005). However, the penalty term considering both synthetic and degradative regulations together as described in Eq. (6) was found to be more effective in identifying the topological structure and correct parameter values than the one used in (Kimura et al., 2005).

4.2.3. Proposed multi-objective formulation

As mentioned, in single-objective approaches (Noman and Iba, 2007; Liu and Wang, 2008a; Palafox et al., 2013), the common objective is to minimize an error function, often based on MSE, together with an additional penalty term, but following an aggregative formulation, as shown in Eq. (6). This formulation requires additional weight factors to be set to find a good trade-off between the different terms. Nevertheless, this goal can be formulated as a bi-objective optimization problem by considering these two terms separately, although following a Pareto dominance scheme as follows:

- Objective 1: f^{MSE} (Eq. (5)), which aims to estimate the kinetic and order parameters from a limited amount of gene expression data.
- Objective 2: $f^{\text{Topology}} = \sum_{i=1}^{N} \sum_{j=1}^{2N-I} (|K_{i,j}|)$ (second term of Eq. (6)), to detect the sparse topological structure that is most commonly seen in biological networks.

In this way, we can now take advantage of specific learning models of Pareto optimality-based techniques (Deb, 2001) to deal with the inference of gene regulatory networks, which will result in sets of non-dominated solutions with different choices of time course estimations and node topologies. In addition, we avoid the use of additional weight factors that could bias the search procedure to one of the different terms, as usually is seen in single-objective approaches.

The use of Pareto-based algorithms has the advantage of providing a set of trade-off (i.e. non-dominated) solutions according the two considered objectives. The set of optimal non-dominated solutions is known as the Pareto optimal set, and its representation in the objective space is named the Pareto front. In general, the use of multi-objective optimization metaheuristics, such as evolutionary algorithms, does not guarantee to find the Pareto front but an approximation to it. These techniques incorporate internally strategies and mechanisms aimed at producing the best Pareto front approximation (Deb, 2001).

4.3. The MONET approach

The MONET approach consists of an adaption of MOCell (Nebro et al., 2007), a multi-objective cellular Genetic Algorithm (cGA), to the inference of GRNs by using the proposed multi-objective formulation. The main characteristic of cellular GAs is that each solution belongs to a cell (or neighborhood) and can only be recombined with a reduced

number of solutions (the surrounding cells or neighbors). The main idea behind this limitation is to perform a wider exploration of the search space. In this way, the algorithm is able to avoid premature convergence to local optima, which will result in robust solutions in terms of kinetic order and rate constant parameters of an S-System model when reconstructing precise networks.

In MOCell, as well as in cGA, the population is structured in a regular grid of d dimensions ($d = 1, 2, 3$), and a neighborhood is defined on it (see the 2D grid representation in Fig. 2-left). The pseudocode of MOCell can be observed in Algorithm 3.5. This algorithm iteratively considers as current each individual (solution vector) in the grid (line 5), which may only interact with other individuals belonging to its neighborhood, so its parents are chosen among from its neighbors (line 6) with a given criterion. Crossover and mutation operators are applied to the individuals in lines 7 and 8, with probabilities p_c and p_m, respectively. Afterwards, the algorithm computes the fitness value of the new offspring individual (or individuals) (line 9), and inserts it (or one of them) into the equivalent place of the current individual in a new auxiliary population following a given replacement policy. After each generation (or loop), the auxiliary population is assumed to be the population for the next generation. This loop is repeated until a stop condition is met (line 3). A maximum number of computed fitness evaluations is considered to be the stop condition.

According to the canonical cGA, in MOCell all the cells can be updated in parallel, yielding the so-called synchronous MOCell. The alternative is the asynchronous version, in which the cells are updated one at a time in sequential order. An asynchronous MOCell can be easily obtained from Algorithm 3.5, assuming that the cells are sequentially updated, so the auxiliary population is not needed in the algorithm. In this study, we have used an asynchronous version of MOCell, called aMOCell4 in (Nebro et al., 2007), in which the cells are updated sequentially (asynchronously).

Algorithm 3.5. Pseudocode of MOCell.

```
1:  pop ⟵ initializePopulation()
2:  pareto_front ⟵ initializeParetoFront() // In archive
3:  while not StopCondition() do
4:    for all individuals in pop do
5:      n_list ⟵ Get_Neighbourhood(position(individual))
6:      parents ⟵ Selection(n_list)
7:      offspring ⟵ Recombination(p_c,parents)
8:      offspring ⟵ Mutation(p_m,offspring)
9:      Evaluate_Fitness(offspring)
10:     Replacement(position(individual),offspring,aux_pop)
11:     Update_Pareto_Front(pareto_front,offspring)
12:   end for
13:   pop ⟵ aux_pop
14:   pop ⟵ Feedback(pareto_front)
15: end while
16: return pareto_front
```

As MOCell is a multi-objective adaption of cGA, it requires the existence of a structure to manage the non-dominated solutions found with the aim of directing the search towards the Pareto optimal set. To this end, an additional population (the external archive) is incorporated to MOCell to gather the non-dominated solutions found throughout the optimization procedure. To this end, MOCell creates an empty Pareto front (line 2) and, after the replacement operation, the generated offspring is included in the external archive (line 11), if appropriate. After each generation, the old population is replaced by the auxiliary one (line 13), and a feedback procedure is invoked to replace a number of randomly chosen individuals with a number of solutions from the archive (line 14). Finally, the "*archived*" Pareto front is returned as algorithm's output (line 16).

It is worth noting that, in the case of cGA, the resulting offspring replaces the individual at the current position if the former is better than the latter. Nevertheless, as is usual in multi-objective optimization, we need to define the concept of "best individual". Our approach is therefore to replace the current individual if it is dominated by the offspring or the two are non-dominated and the current individual has the worst crowding distance (as defined in (Deb, 2001)) in a population composed of the neighborhood plus the offspring. This criterion is also used to decide whether the offspring solutions are added to the external archive (line 11 in Algorithm 3.5), or not. For inserting individuals in the Pareto front, the solutions in the archive are ordered according to the crowding distance. Then, when a new non-dominated solution is inserted, if the Pareto front archive is already full, the solution with the worst (lowest) crowding distance value is removed.

Fig. 2 shows a conceptual overview of the MONET operation. For each new candidate solution, the vector of variables representing the kinetic orders and rate constants are used to set the decoupled S-System, whose outputs are obtained in form of predicted time-series. Therefore, the MSE (objective 1) of the predicted time-series is computed with regards to the gene expression data. In the case of topology regularization (objective 2), it is calculated by just using the kinetic orders, as explained in Section 4.2. In this way, the two objective values are assigned to the evaluated solution.

5. Experiments

For the integration of MONET, we have adapted the implementation of MOCell provided in the jMetal framework (http://jmetal.sourceforge.net/) (Durillo and Nebro, 2011). The parameter settings were performed as recommended in the research study in which MOCell was initially proposed (Nebro et al., 2009), although adapting some of them to the special case of GRNs instances, after a series of preliminary experiments. Specifically, a population size of 100 individuals and a global number of 1,000,000 function evaluations are set as the stopping condition; SBX crossover and polynomial mutation are used as the operators for crossover and mutation, respectively; the distribution indexes for these two operators are $\eta_c = 5$ for crossover, and $\eta_m = 5$ for mutation; the crossover probability is $p_c = 0.9$ and the mutation probability is $p_m = 1/n$, being n the number of decision variables of the tackled problem (i. e., kinetic orders/rate constants).

A series of 25 independent runs were performed in experiments for each GRN problem instance. The computational framework used consisted of a Condor (http://research.cs.wisc.edu/htcondor/) middleware platform, managing a maximum number of 200 cores, which acts as a distributed task scheduler (each task dealing with one independent run). After the experiments, a series of ROC-based performance metrics were computed that will vary depending on the different datasets and comparative analysis carried out. Nevertheless, with the aim of selecting the most accurate Pareto front approximation out of the 25 independent runs, we focused on a quality indicator to reduce the provided scores (two objectives) of set optimal solutions (Pareto front) to one single score. Specifically, the hypervolume indicator (Zitzler et al., 2008) was used to validate the optimization provided by the MONET approach, from the multi-objective point of view. In this way, those resulting Pareto fronts with best hypervolume were selected to provide candidate solutions for the forthcoming analysis and discussions.

6. Results

This section reports the results obtained in the scope of artificial network with noise-free/noisy data, as well as for benchmarking instances from DREAM3 Challenge and in vivo IRMA datasets. Comments with regards to computational effort are also given.

6.1. Inference from artificial noise-free/noisy data

For the first set of experiments, we used an artificial network firstly used by (Hlavacek and Savageau, 1996) that consisted of five elements' interactions. This network has been widely used in early (Kikuchi et al., 2003; Kimura et al., 2005; Liu and Wang, 2008a; Noman and Iba, 2007)

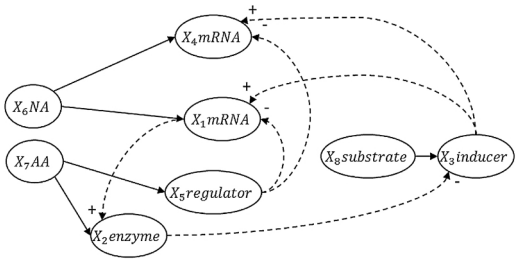
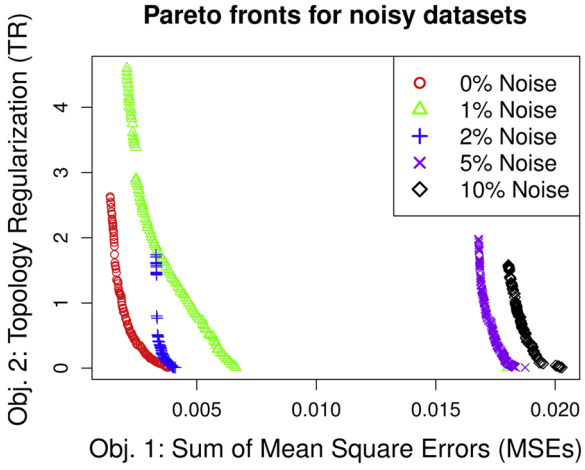


Fig. 3. Synthetic network used to test noise-free/noisy data.

and in recent studies (Sirbu et al., 2010; Lee and Hsiao, 2012; Palafox et al., 2013) to validate algorithmic proposals, so that we decided to experiment with it, to also be able to compare with previous works. The true system of this network is modeled in the S-System as follows:

$$
\begin{aligned}
X_1 &= \alpha_1 X_3^{g_{13}} X_5^{g_{15}} X_6 - \beta_1 X_1^{h_{11}} \\
X_2 &= \alpha_2 X_1^{g_{21}} X_7 - \beta_2 X_2^{h_{22}} \\
X_3 &= \alpha_3 X_2^{g_{32}} X_8 - \beta_3 X_2^{h_{32}} X_3^{h_{33}} \\
X_4 &= \alpha_4 X_3^{g_{43}} X_5^{g_{45}} X_6 - \beta_3 X_4^{h_{44}} \\
X_5 &= \alpha_5 X_4^{g_{54}} X_7 - \beta_5 X_5^{h_{55}}
\end{aligned} \tag{7}
$$

Fig. 3 illustrates the gene network corresponding to the S-System of Eq. (7). In this network, X_1 is an mRNA produced from gene 1. This gene also produces the enzyme protein X_2, and X_3 is an inducer protein catalyzed by X_2. X_4 is an mRNA produced from gene 4 and X_5 is a regulator protein it produces. The mRNA production processes of genes 1 and 4 assume a positive interaction from the inducer protein X_3 and a negative interaction from the regulator one X_5. Variables X_6, X_7, and X_8 are considered to be independent in the system and represent a pool of nucleic acid, amino acid and substrate, respectively.



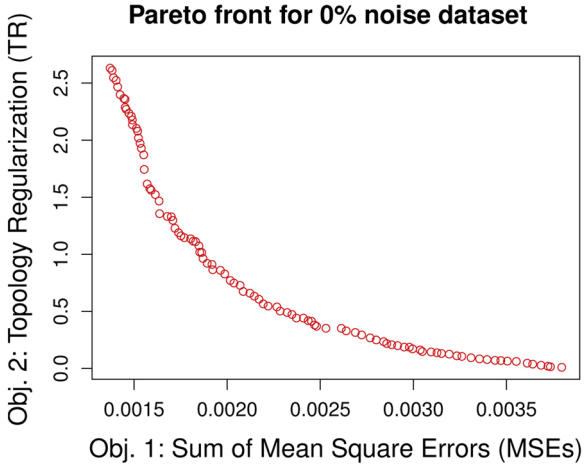


Fig. 4. The Pareto front approximations with best hypervolume obtained by MONET, for datasets with different levels of noise, are plotted at left. Graphic at right contains the approximated Pareto front corresponding to noiseless (0% noisy) dataset in a separate plot.

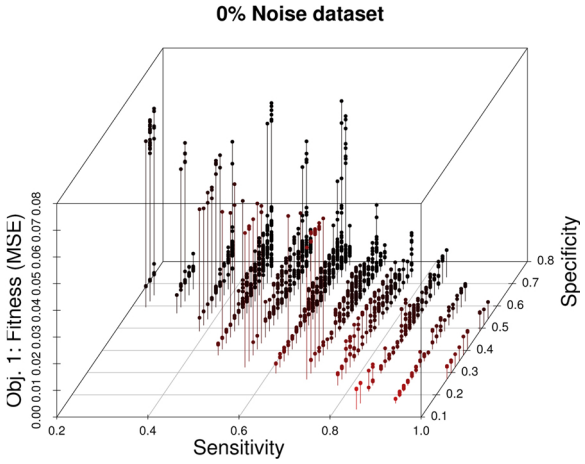


Fig. 5. Sensitivity and specificity with regards to MSEs values (Objective 1) of obtained non-dominated solutions of MONET for noiseless (0% noisy) dataset.

From this network, we generated four new instances with different levels of Gaussian White noise. This was done to evaluate the robustness of the evaluated algorithm on different conditions of noise. Specifically, we added noise to the original data by applying the following procedure:

$$
\begin{aligned}
&X(t)\text{: Data} \\
&\text{Noisy}(X) = X(t) + \epsilon N(0, \sigma) \\
&\epsilon = \text{Mean}[X(t)] \times \rho
\end{aligned} \tag{8}
$$

where ρ is the noise rate applied to the data. Therefore, as ϵ increases, *Noisy*(X) distorts the information of the original data. For these experiments, we set $\sigma = 1$ and ρ = 1%, 2%, 5%, and 10% noise rates, depending on the level of noise in each generated instance. Each dataset comprised 3 time-series with 5×20 time points of gene expression. We followed a similar configuration of search region limits as previously fine-tuned in previous approaches (Palafox et al., 2013; Sirbu et al., 2010; Tsai and Wang, 2005) for the same datasets. We bounded the kinetic orders g_{ij} and h_{ij} to $[-3, 3]$, and the rate constants α and β to [0, 10].

A first observation of the results can be taken from Fig. 4 (left), where the approximated Pareto fronts with best hypervolume computed by MONET for each dataset with different levels of noise are plotted. As mentioned, these Pareto fronts are generated from the archived non-dominated solutions, with respect to the two objectives used: MSE and Topology Regularization (TR). In the case of noiseless (0%) and 1% noisy datasets, the MSE values obtained are close to results in other studies in the state of the art (Sirbu et al., 2010) for the same datasets. In addition, we now gain some quantitative insights of the network structure by examining the topology regularization objective. In this regard, adding 2% noise to the data affects the accuracy of the inference, yet most of the values are kept within an acceptable range. For a 2% noise ratio, the inference is still successful in finding the correct regulations. This means the MONET is robust for this amount of noise. Some of the False Positive (FP) values increased, which is to be expected with increased noise. In the case of 5% and 10% levels of noise, the integrity of the data is significant hampered. In these cases, the number of FPs and FNs (False Negatives) increased, although MONET was still able to recover the correct interactions of the network.

Table 1
Sensitivity (S_n), Specificity (S_p) and Accuracy (Acc) values computed by MONET for the experimented dataset with different levels of noise, in comparison with related approaches in the state-of-the-art.

Noise	0%			1%			2%			5%			10%		
Algorithm	S_n	S_p	Acc	S_n	S_p	Ac.	S_n	S_p	Acc	S_n	S_p	Acc	S_n	S_p	Acc
GA + ANN	0.74	0.81	0.77	1.00	0.89	0.94	0.89	0.78	0.83	0.77	0.78	0.77	0.71	0.72	0.71
GA + ES	0.64	0.89	0.76	0.72	0.93	0.82	0.70	0.89	0.79	0.54	0.83	0.68	0.83	0.75	0.79
DE + AIC	0.58	0.81	0.69	0.63	0.86	0.74	0.68	0.89	0.78	0.69	0.86	0.77	0.54	0.86	0.70
GLSDC	0.72	0.67	0.69	0.74	0.65	0.69	0.73	0.76	0.74	0.74	0.76	0.75	0.65	0.68	0.66
PEACE1	0.55	0.82	0.68	0.36	0.8	0.58	0.36	0.77	0.56	0.44	0.78	0.61	0.96	0.08	0.52
MONET	0.84	0.82	0.83	0.92	0.81	0.86	0.92	0.78	0.85	0.92	0.72	0.82	0.84	0.78	0.81

Continuing with this figure, a second observation concerns the front shapes. Fig. 4 (right) shows a sub-graphic in which the best Pareto front (according to hypervolume) obtained by MONET for the dataset with 0% level of noise is separately plotted. In this front, we can observe that the MSEs obtained are within a short range of values ($[6.4e-4 \cdots 3.8e-3]$), whereas those values of topology regularization (Obj. 2) are within a relatively wider range ($[1.0e-2 \cdots 1.9e+0]$). This led us to check that non-dominated solutions in this front are able to model different network topologies, although showing very close and low square error (MSEs) values when fitting the time-series curves. In fact, this observation is clearer in Fig. 4 (top), where all fronts are plotted within the same range of values.

Although evaluating the algorithm's performance based only on fitness functions is helpful (sometimes mandatory) when the inferred model is unknown, it could produce over-fitting in the learning procedure after a large number of iterations. Fig. 5 plots the fitness values in terms of MSE computed by MONET after these experiments with regards to their sensitivity ($S_n = \frac{TP}{TP+FN}$) and specificity ($S_p = \frac{TN}{TN+FP}$) values. This way, a TP (True Positive) denotes the existence of the right interaction between two genes inhibition and activation, whereas a TN (True Negative) indicates the absence of interactions. As we can observe, those solutions with low fitness values usually induce high sensitivities, although with low specificities. That is, these solutions usually select correct interactions (TPs), but also wrong ones (FPs). Therefore, an additional regularization function is required to avoid additional false interactions.

For further validation and comparison of the results, Table 1 shows the best sensitivity and specificity values obtained by MONET predictions for the experimented network with different levels of noise. In this table, the predictions of a series of single-objective inference techniques in the state of the art (Sirbu et al., 2010), when dealing with the same datasets, are also shown. These techniques are: a method using an Artificial Neural Network as a model and GA for parameter inference (GA + ANN), a hybrid GA with an Evolution Strategy (GA + ES), a Differential Evolution with AIC-based fitness (DE + AIC), a method using Genetic Local Search (GLSDC), and an iterative algorithm based on GA (PEACE1). In addition, the Accuracy ($\text{Acc} = \frac{TP+TN}{TP+FP+TN+FN}$) values are also calculated in order to provide overall results, hence to establish as fair as possible comparisons. According to this, MONET obtained the most accurate predictions (shaded in gray) for all datasets with scaling degrees of noise, excepting for 1% of noisy level, for which GA + ANN obtained the best prediction. In general, MONET is competitive and robust against noise. One of the reasons this approximation worked better than the compared single-objective approaches could be due to its topology regularization, which guides the algorithm to avoid large numbers of parameters, thus keeping regulations small and preventing FPs from emerging even with increased noise in the data (Palafox et al., 2013).

6.2. Results on DREAM3 networks

The second set of experiments consisted in evaluating our multi-objective approach in the context of the standard benchmark DREAM3 *in silico challenges*[2] of GRNs inference proposed by (Prill et al., 2010). This benchmark comprises two networks of *E. coli* (*Eschericia coli*) plus three networks of *Yeast* (*Saccharomyces cerevisiae*) with two versions of 10 and 100 sized genes for each of them. Networks of size 10 involve 4 time-series of 21 time points of samples, and networks of 100 genes involve 46 time-series of 21 time points. These data were generated from a thermodynamic model simulation phase for gene expression that also entailed the addition of Gaussian noise. The time-series correspond to different random conditions initially set for the thermodynamical model. The structure of the networks were given from the actual *Eschericia coli* and *Saccharomyces cerevisiae* GRNs, which exhibit heterogeneous patters of sparsity and topology. In the DREAM3 challenge, the target graphs are directed and not labeled with inhibitors or inductors.

To measure the performance of algorithms, we have followed the protocol suggested in this challenge that comprises standard metrics: the Receiver Operating Characteristic curve (ROC), the area under ROC curve (AUROC) and the area under the precision-recall curve (AUPR). The performance of MONET for the inference of network structures sizes 10 and 100 are given in Tables 2 and 3, respectively. In addition, the best predictions obtained separately throughout all executions are displayed in row $\text{MONET}_{BestRuns}$. In these tables, a series of state-of-the-art results are also incorporated, which comprise those of LASSO and the teams that exclusively used the same set of time-series data in the DREAM3 challenge, namely: Teams 236 and 190. The LASSO implements a baseline linear least squares regression: $x_{t+1,i} = \mathbf{x}_t^T \beta_i$, performed on each dimension node $i = 1 \cdots p$ subject to l_i penalty on the β_i parameters. An edge (i, j) is then assigned for each nonzero β_{ij} coefficient.

Additional results obtained from recent studies in the literature have been also incorporated, which follow the same experimental procedure and metrics, although implementing heterogeneous algorithms with different learning procedures. A first set of results are obtained from (Fan et al., 2017), where three variants of Bayesian Network Spline with Nonparametric Regression and Topology information are evaluated in the context of DREAM3 size-10 networks, showing prominent performance. These variants are Bayesian Lasso (BL), Bayesian Group Lasso (BGL) and BGL with spike and slab priors (BGL_pro), and their corresponding AUROC and AUPR are included in 2 . Another interesting results are obtained from citepasoco-grn-2018, where a hybrid multi-agent genetic algorithm with random forests based on fuzzy cognitive maps is proposed and evaluated in the context of DREAM3 Yeast2 network with 10 and 100 genes. In this last approach, an AUROC and AUPR values of 0.509 and 0.352 are obtained (respectively) for Yeast2 10-size, while an AUROC of 0.508 and AUPR of 0.044 are registered for Yeast2 100-size.

The AUROC and AUPR metrics in Table 2 indicate that MONET is highly competitive with regards to the techniques compared in the scope of size 10. Concretely, the proposed approach achieved superior AUROC results for all networks except for Ecoli2 and Yeast2 (AUROC), although with close values to the best ones. In the case of AUPR, MONET obtained outperforming results for Ecoli1, Yeast1, and Yeast2.

[2] Available at URL http://dreamchallenges.org.

Table 2
AUROC and AUPR for MONET, LASSO, Team 236, Team 190 (DREAM3 challenge), BL, BGL, and BGL_prior run on DREAM3 size-10 networks.

	Ecoli1		Ecoli2		Yeast1		Yeast2		Yeast3	
	AUROC	AUPR	AUROC	AUPR	AUROC	AUPR	AUROC	AUPR	AUROC	AUPR
LASSO	0.500	0.119	0.547	0.531	0.528	0.244	0.627	0.305	0.582	0.255
Team 236	0.621	0.197	0.650	0.378	0.646	0.194	0.438	0.236	0.488	0.239
Team 190	0.573	0.152	0.515	0.181	0.631	0.167	0.577	0.371	0.603	0.373
BL	0.494	0.245	0.688	0.532	0.620	0.298	0.441	0.297	0.409	0.201
BGL	0.533	0.207	0.781	0.609	0.552	0.274	0.534	0.287	0.464	0.219
BGL_prior	0.623	0.242	0.787	0.614	0.636	0.279	0.503	0.254	0.498	0.234
MONET	0.647	0.184	0.513	0.200	0.801	0.469	0.522	0.354	0.623	0.324
$MONET_B$	0.761	0.328	0.603	0.322	0.801	0.469	0.595	0.423	0.623	0.324

Table 3
AUROC and AUPR for MONET, LASSO, Team 236 (DREAM3 challenge) run on DREAM3 size-100 networks.

	Ecoli1		Ecoli2		Yeast1		Yeast2		Yeast3	
	AUROC	AUPR	AUROC	AUPR	AUROC	AUPR	AUROC	AUPR	AUROC	AUPR
LASSO	0.519	0.016	0.512	0.057	0.507	0.016	0.530	0.044	0.506	0.044
Team 236	0.527	0.019	0.546	0.042	0.532	0.035	0.508	0.046	0.508	0.065
MONET	0.525	0.014	0.533	0.012	0.522	0.018	0.485	0.038	0.517	0.058
$MONET_B$	0.532	0.034	0.542	0.015	0.522	0.018	0.524	0.047	0.517	0.058

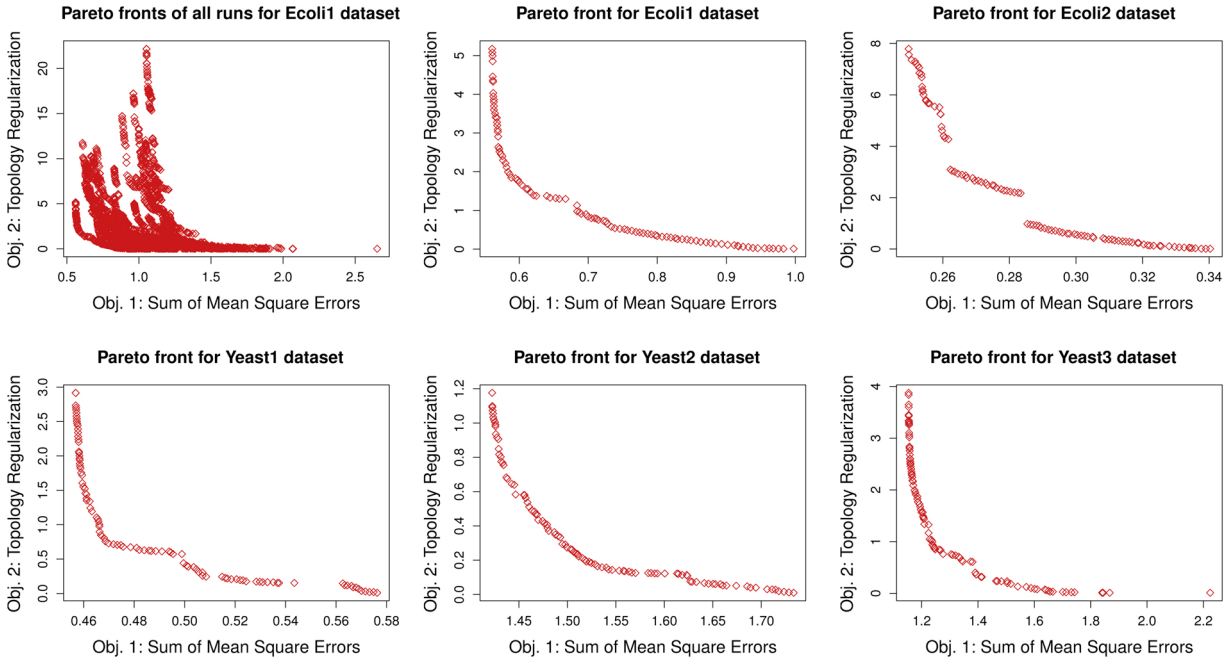


Fig. 6. The Pareto front approximations with best hypervolume obtained by MONET, for the datasets of the DREAM3 challenge, are plotted. Graphic at top-left contains the approximated Pareto fronts corresponding to Ecoli1 dataset obtained by MONET in all the independent runs.

We especially note that our multi-objective approach exhibited excellent AUROC and AUPR values for Ecoli1 and Yeast1. It is worth mentioning that other approaches in the literature used additional knowledge concerning the network structure to generate guiding operators, hence enhancing the inference process. For instance, the Ecoli2 network shows a star topology with several central hubs that regulate many genes (Prill et al., 2010), so this information is used to establish thresholds and cluster strategies that could lead inference methods to perform successfully, but only in the scope of this benchmarking network. In contrast, in the case of MONET only objective functions are used to numerically discern the network topology and the interactions strength according to S-System parameters in non-dominated solutions. This is a key advantage for actual experiments, where the network structure is commonly unknown and only gene expression data are available to train inference methods.

From a graphical perspective, the Pareto fronts with best hypervolume obtained by MONET for the datasets of the DREAM3 challenge are plotted in Fig. 6. In this figure, the plot at top-left contains the Pareto fronts corresponding to the Ecoli1 dataset obtained by MONET in all independent runs. These last plots are given just to show that obtained fronts are in close ranges and no extreme outliers were obtained for these datasets, although this observation is generalized for all the datasets used in this study.

In the case of the DREAM3 size-100 networks, Table 3 shows that MONET is again highly competitive with regards to the approach of Team 236, which was the only team that used time-series datasets for size-100 challenge. Specifically, MONET obtained outperforming AUROC values for networks Ecoli1, Yeast2 and Yeast3. Team 190 did not submit predictions for the size-100 networks. An interesting observation in this table is that, for all the algorithms, the resulting

Table 4
AUPR performances on IRMA network.

Algorithm	Switch-on	Switch-off
BL_prior	0.625	–
TIGRESS	0.714	0.452
GENIRF	0.672	0.327
CMI2NI	0.721	0.456
KFLR	0.896	0.721
Jump3	0.685	0.682
BGRMI	0.904	0.574
MONET	0.827	0.734
$MONET_B$	0.901	0.741

AUROC values are kept with a high precision and similar to those ones obtained for size-10 networks, whereas AUPR values declined in general, with percentages lower than 7%. A reason of this deterioration in AUPR may be because of the lower density of size-100 networks, where the number of non-edge elements is larger than the number of edges.

Finally, in comparison with those results of (Liu and Liu, 2018) for Yeas2 100-size network (only network with available results), MONET obtained better AUROC (0.524) and AUPR (0.047) values than the hybrid technique proposed in that work (with AUROC 0.508 and AUPR 0.044). All these results lead us to suggest that MONET shows competitive performance with regards to original works in the context of DREAM3 Challenge, as well as to modern algorithmic proposals in the current literature.

6.3. Results on the IRMA network

To further test MONET we used time course gene expression data from the *In Vivo* IRMA (In vivo Reverse-engineering and Modeling Assessment) network (Cantone et al., 2009), which was synthesized in the yeast Saccharomyces cerevisiae. The network has 5 genes (CBF1, GAL4, SWI5, GAL80, and ASH1) and 6 regulatory interactions and can be switched on or off by culturing cells in galactose or glucose, respectively. The expression levels of the genes in the network were measured using quantitative RT-PCR at different time points in two different sets of experiments. In the first set, cells were stimulated with galactose and the network was switched on, whereas in the second set the network was switched off by adding glucose. The IRMA network is well studied and is a gold standard network.

Table 4 shows the results of the GRNs reconstructed by the best method used for performance comparison of in-silico data (BL_prior), including the most prominent techniques found in the literature for IRMA network. These techniques are BGRMI, Jump3 (both from (Iglesias-Martinez et al., 2016)), KFLR, CMI2NI, and TIGRESS (from (Pirgazi and Khanteymoori, 2018)), for which the AUPR values are used for comparison (AUROC values are not available for these techniques). As shown in this table, BGRMI obtained the highest AUPR for the Switch-On dataset, although with similar results to those of MONET, which performs the second best method. Conversely, in the case of Switch-Off dataset, MONET obtains the best AUPR values, followed by KFLR and Jump3. These results suggest MONET is highly competitive, not only on in silico datasets, but also on in vivo experimental data.

6.4. Computational effort

MONET used, on average, computing times of 1 minute for networks with 5 genes and 60 observations, 3 minutes for networks with 10 genes and 81 observations (DREAM3-10 size), and 30 hours for networks with 100 genes and 210 observations (DREAM3-100 size). These computational times are in the range of other approaches in the state of the art in which model-based inference methods are applied to similar datasets (Huynh-Thu and Sanguinetti, 2015; Palafox et al., 2013; Sirbu et al., 2010). It is worth noting that the extra amount of time required to infer a DREAM3 size-100 network is due to the high number of observations, although such a high number is not usually encountered in real datasets, where the number of observations is typically much lower than the number of genes.

7. Conclusions

Inferring the interaction network of genes is a fundamental step towards understanding how a cell or an organism can respond to its environment. In the last decade, the computational biology community have made reasonable efforts to develop new techniques to solve this problem. However, there is still a demand for unified frameworks to increase the accuracy and decrease the number of false predictions in correct interaction for small networks.

Our hypothesis is that a multi-objective Pareto dominance-based approach would allow us to obtain a low concentration error when fitting genetic time-series, at the same time as a sparse topology of the network is enhanced. As a result of this approach, we are now able to design and use a sub-class of multi-objective evolutionary optimization algorithms, thereby taking advantage of their different learning procedures.

Experiments on simulated and synthetic data show that MONET is always competitive and often outperforms state-of-the-art GRN inference procedures. In noisy time-series, Table 1 shows how MONET outperforms practically all compared algorithms, so it is able to keep an accurate percentage of correct predictions even with a 10% level of noise, with sensitivity 0.84 and specificity 0.78. It has good scalability with respect to the number of genes and maintains its good performance when inferring large networks. In addition, for standard benchmarks in DREAM3 and in vivo IRMA network, MONET also experimented competitive performance in comparison with other current approaches in the state of the art.

As future work, an important direction to take is the adaption of specific algorithmic operators to the special case of GRNs modeling and inference. The design of crossover, mutation and local search operators that include additional knowledge based on complementary data, such as: microRNA expression, chromatin or protein-protein interactions, could guide the search strategy of the algorithmic proposal and enhance its potential for biologically meaningful hypothesis generation on real datasets. In this sense, the use of pruning or regularization mechanisms to prevent small values (of interactions) from increasing could lead the algorithm to decrease the creation of false positives, while promoting a sparse set of parameters. In addition, the use of generalized Boolean models for network inference would enable more sophisticated forms of logical update in transition of elements, hence allowing transitory state datasets.

From another perspective, inferring large scale gene networks from perturbation data is computationally challenging, so few computational tools have been proposed in the literature to deal with this issue (Shojaie et al., 2014). In this sense, the development of archiving strategies in MONET to compute and evaluate overlapping and consensus graphs among non-dominated solutions in the Pareto front approximation, could lead to the generation of competitive techniques for the inference of GRNs, also on gene expression time-series datasets with high degree of perturbation.

Acknowledgements

This work is partially funded by Grants TIN2017-86049-R (Ministerio de Ciencia e Innovación, Spanish Government) and P12-TIC-1519 (Plan Andaluz I + D + I). José García-Nieto is recipient of a Post-Doctoral fellowship of “Captación de Talento para la Investigación” at Universidad de Málaga.

References

Akutsu, T., 2003. Identification of genetic networks by strategic gene disruptions and gene overexpressions under a boolean model. Theor. Comput. Sci. 298 (1), 235–251.

Cantone, I., Marucci, L., Iorio, F., Ricci, M.A., Belcastro, V., Bansal, M., Santini, S., di Bernardo, M., di Bernardo, D., Cosma, M.P., 2009. A yeast synthetic network for in vivo assessment of reverse-engineering and modeling approaches. Cell 137 (1), 172–181.

Chen, Y., Zou, X., 2016. Inferring Gene Regulatory Network Using an Evolutionary Multi-Objective Method. Cornell University Library arXiv:151205055.

Deb, K., 2001. Multi-Objective Optimization Using Evolutionary Algorithms. John Wiley & Sons, Inc, New York, NY, USA.

Durillo, J.J., Nebro, A.J., 2011. jmetal: A java framework for multi-objective optimization. Adv. Eng. Softw. 42, 760–771.

Friedman, N., Linial, M., Nachman, I., er, D.P., 2004. Using Bayesian networks to analyze expression data. J. Comput. Biol. 7 (3–4).

Hitoshi Iba, N.N., 2016. Evolutionary Computation in Gene Regulatory Network Research. Wiley Series in Bioinformatics.

Hlavacek, W.S., Savageau, M.A., 1996. Rules for coupled expression of regulator and effector genes in inducible circuits. J. Mol. Biol. 255 (1), 121–139.

Huynh-Thu, A., Sanguinetti, G., 2015. Combining tree-based and dynamical systems for the inference of gene regulatory networks. Bioinformatics 31 (10), 1614–1622.

Iglesias-Martinez, L.F., Kolch, W., Santra, T., 2016. Bgrmi: a method for inferring gene regulatory networks from time-course gene expression data and its application in breast cancer research. Nat., Sci. Rep. 6 (37140).

Kikuchi, S., Tominaga, D., Arita, M., Takahashi, K., Tomita, M., 2003. Dynamic modeling of genetic networks using genetic algorithm and s-system. Bioinformatics 19 (5), 643–650.

Kimura, S., Ide, K., Kashihara, A., Kano, M., Hatakeyama, M., Masui, R., Nakagawa, N., Yokoyama, S., Kuramitsu, S., Konagaya, A., 2005. Inference of s-system models of genetic networks using a cooperative coevolutionary algorithm. Bioinformatics 21 (7), 1154–1163.

Lee, W.P., Hsiao, Y.T., 2012. Inferring gene regulatory networks using a hybrid GA-PSO approach with numerical constraints and network decomposition. Inf. Sci. 188, 80–99.

Lim, N., Senbabaoglu, Y., Michailidis, G., dAlché Buc, F., 2013. Okvar-boost: a novel boosting algorithm to infer nonlinear dynamics and interactions in gene regulatory networks. Bioinformatics 29 (11), 1416.

Liu, Luowen, Liu, Jing, 2018. Inferring gene regulatory networks with hybrid of multi-agent genetic algorithm and random forests based on fuzzy cognitive maps. Appl. Soft Comput. 69 (18), 585–598.

Liu, P.K., Wang, F.S., 2008a. Inference of biochemical network models in s-system using multiobjective optimization approach. Bioinformatics 24 (8), 1085–1092.

Liu, P.K., Wang, F.S., 2008b. Inference of biochemical network models in s-system using multiobjective optimization approach. Bioinformatics 24 (8), 1085.

Nebro, A.J., Durillo, J., Luna, F., Dorronsoro, B., Alba, E., 2007. Design Issues in a Multiobjective Cellular Genetic Algorithm. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 126–140.

Nebro, A.J., Durillo, J.J., Garcia-Nieto, J., Coello Coello, C.A., Luna, F., Alba, E., 2009. SMPSO: a new PSO-based metaheuristic for multi-objective optimization. IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making 66–73.

Nobile, M.S., Iba, H., 2015. A double swarm methodology for parameter estimation in oscillating gene regulatory networks. 2015 IEEE Congress on Evolutionary Computation (CEC) 2376–2383.

Noman, N., Iba, H., 2007. Inferring gene regulatory networks using differential evolution with local search heuristics. IEEE/ACM Trans. Comput. Biol. Bioinform. 4 (4), 634–647.

Noman, N., Monjo, T., Moscato, P., Iba, H., 2015. Evolving robust gene regulatory networks. PLOS ONE 10 (1), 1–21.

Palafox, L., Noman, N., Iba, H., 2013. Reverse engineering of gene regulatory networks using dissipative particle swarm optimization. IEEE Trans. Evolut. Comput. 17 (4), 577–587.

Pirgazi, J., Khanteymoori, A.R., 2018. A robust gene regulatory network inference method base on Kalman filter and linear regression. PLOS ONE 13 (7), e0200094.

Prill, R.J., Marbach, D., Saez-Rodriguez, J., Sorger, P.K., Alexopoulos, L.G., Xue, X., Clarke, N.D., Altan-Bonnet, G., Stolovitzky, G., 2010. Towards a rigorous assessment of systems biology models: the dream3 challenges. PLoS ONE 5 (2), 1–18.

Khalid Raza, Mansaf Alam, 2016. Recurrent neural network based hybrid model for reconstructing gene regulatory network. Comput. Biol. Chem. 64, 322–334.

Savageau, M., 2010. Biochemical Systems Analysis: A Study of Function and Design in Molecular Biology. Addison-Wesley Educational Publishers Inc.

Sirbu, A., Ruskin, H.J., Crane, M., 2010. Comparison of evolutionary algorithms in gene regulatory network model inference. BMC Bioinform. 11 (1), 59.

Shojaie, A., Jauhiainen, A., Kallitsis, M., Michailidis, G., 2014. Inferring regulatory networks by combining perturbation screens and steady state gene expression profiles. PLOS ONE 9 (2), e82393.

Spieth, C., Streichert, F., Speer, N., Zell, A., 2005a. Multi-objective model optimization for inferring gene regulatory networks. In: Coello Coello, C., Hernández Aguirre, A., Zitzler, E. (Eds.), Evolutionary Multi-Criterion Optimization, Lecture Notes in Computer Science, vol. 3410. Springer Berlin Heidelberg, pp. 607–620.

Spieth, C., Streichert, F., Supper, J., Speer, N., Zell, A., 2005b. Feedback memetic algorithms for modeling gene regulatory networks. Computational Intelligence in Bioinformatics and Computational Biology, 2005. CIBCB '05. Proceedings of the 2005 IEEE Symposium on 1–7.

Tominaga, D., Koga, N., Okamoto, M., 2000. Efficient numerical optimization algorithm based on genetic algorithm for inverse problem. Proceedings of the 2Nd Annual Conference on Genetic and Evolutionary Computation. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, GECCO'00, pp. 251–258.

Tsai, K.Y., Wang, F.S., 2005. Evolutionary optimization with data collocation for reverse engineering of biological networks. Bioinformatics 21 (7), 1180–1188.

Voit, E.O., 2000. Computational Analysis of Biochemical Systems. A Practical Guide for Biochemists and Molecular Biologists. Cambridge University Press.

Fan, Yue, Wang, Xiao, Peng, Qinke, 2017. Inference of gene regulatory networks using Bayesian nonparametric regression and topology information. Comput. Math. Methods Med. 17 (8307530), 1–8.

Zitzler, E., Knowles, J., Thiele, L., 2008. Quality Assessment of Pareto Set Approximations. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 373–404.