Boolean Network Inference and Optimization for Biological Systems

Hitesh Kandarpa BS22B021[1], Kishore Kumar E CS22B041[2]

1 Biotechnology, IIT Madras, Chennai, Tamil Nadu, India
2 Computer Science & Engineering, IIT Madras, Chennai, Tamil Nadu, India

Abstract

Boolean modeling is a parameter-light approach to capture GRN dynamics but typically relies on expert curation. Automated inference methods utilize high-throughput data but often miss the architectural signatures of real GRNs. Here, we incorporate canalization, redundancy, motif, and criticality measures identified from 129 curated models analyzed by Kadelka et al. to optimize Boolean network inference from binarized expression time series. Our pipeline generates an initial network, scores these signatures, and applies targeted perturbations to optimize these measures while minimizing distance from the data.

1 Introduction

Gene regulatory networks (GRNs) coordinate cellular functions such as proliferation, differentiation, and stress responses via intricate gene–protein interactions [1, 2]. Classic quantitative models, especially ordinary differential equations (ODEs), reproduce these dynamics at a fine-grained level but require large numbers of kinetic parameters (reaction rates, concentration thresholds) that may not be present or are imprecise in vivo [2, 3]. Additionally, ODE simulations become increasingly difficult with growing network size or when exploring vast perturbation spaces [2].

Boolean modeling circumvents these limitations by reducing each gene to a binary condition (ON/OFF) controlled by simple logical rules [4]. This qualitative approach replicates principal dynamical phenomena—stable attractors representing cell-cycle stages or differentiation states [5, 6]—and trajectories of transitions in response to genetic or environmental perturbations, while remaining computationally manageable even for very large networks [4].

Most Boolean GRN models in the literature are hand-curated by researchers who integrate evidence from ChIP assays, gene knock-outs, and transcriptomics into ad hoc logical rules [7]. This process is time-consuming, biased toward well-studied genes, and may miss subtle or indirect regulatory relationships. To address these issues, several network-inference algorithms have been proposed—ranging from mutual-information–based approaches to machine-learning methods such as decision-tree classifiers, random forests, and ensemble methods [8]. However, these methods maximize general statistical criteria (e.g., information gain or classification accuracy) without imposing the design principles typical of biological GRNs [9].

A recent meta-analysis of expert-curated Boolean GRN models aggregated from peer-reviewed literature reported the following five discriminating network measures [9]:

1. **Canalization**: Widespread nested canalizing functions, high canalizing depth, and high input redundancy that impart robustness.
2. **Regulatory redundancy**: Minimal symmetry groups, pointing to redundant regulator roles.
3. **Feed-forward loops (FFLs)**: Substantial enrichment of coherent and incoherent three-node FFL motifs and higher-order FFL clusters for signal filtering and temporal control.
4. **Feedback loops (FBLs)**: Over-representation of auto-regulation and multi-gene loops that support multistability and homeostasis.
5. **Critical dynamicality**: Mean network sensitivity grouped around the critical value, corresponding to nested canalizing layer architectures that place GRNs on the cusp between order and chaos.

In spite of this meta-analytic understanding, current inference techniques do not directly aim at these topological signatures. They tend to produce networks that are consistent with expression data but have unrealistic topology and dynamics. To fill this gap, we introduce a biologically guided inference pipeline that:

1. Derives scoring functions from the complete set of network measures (canalization, redundancy, motif frequencies, and criticality).
2. Combines these scores to optimize the results of proven inference algorithms operating on binarized expression time series.
3. Demonstrates that this biologically inspired optimization greatly enhances the recovery of actual network structure and dynamic attractors on benchmark data.

Materials and methods

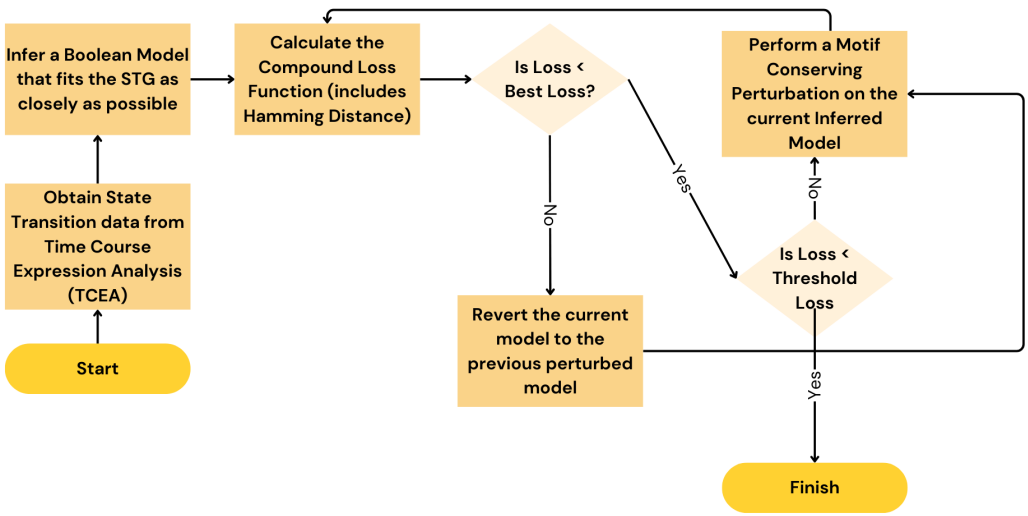

Fig 1. Model Architecture/Pipeline. For the final presented model, MIBNI was utilized as the inference method and simulated annealing was implemented to optimize the loss function through sequential motif conserving perturbations.

Data Generation

We use two 7-node toy networks (TN) to generate synthetic binarized expression data by simulating each network from a set of initial states for 50 synchronous update steps. The second toy network was derived from the first by adding network motifs and enhancing canalization. This simulated data serves as our ground truth (GT).

1.1 Initial Network Inference

As the inferred network provides the starting template for all subsequent optimization steps, the choice of inference method and its output strongly influence the final similarity score. To ensure diversity in the inferred networks (INs), we implemented two complementary inference protocols:

- **Decision-Tree Classifier** [8]: Trains a separate binary decision tree for each target node on the binarized expression profiles, then converts each root-to-leaf path of splits into a compact Boolean update rule.
- **Mutual Information–Based Boolean Network Inference (MIBNI)** [10]: Ranks candidate regulators by their mutual information with the target, then incrementally builds and prunes Boolean functions using the top-scoring inputs to infer sparse network rules.

Benchmarking

We benchmarked each method by simulating its inferred network (IN) alongside the TN and computing the Hamming distance between their state trajectories. We structured our benchmarking into three scenarios:

1. **Single trajectory – single initial state:** A single initial state of the TN generates one trajectory; the INs are simulated from the same state and their trajectory similarity to the GT is measured.
2. **Multiple trajectories – same initial states:** Multiple initial states of the TN generate multiple trajectories; the INs are simulated from those same states and their trajectory similarities are measured.
3. **Multiple trajectories – different initial states:** Multiple initial states of the TN generate multiple trajectories; the INs are simulated from different states than the TN and their trajectory similarities are measured.

We also varied the number of initial states in each scenario to analyze method performance as a function of data availability.

Loss Function

The compound loss function scores a perturbed network on the above measures as well as its similarity to the GT. The following network measures were evaluated:

1. **Average canalization depth**: Measures the average number of hierarchical canalizing layers per node, reflecting how many inputs can "lock in" a fixed output.

$$\overline{d} = \frac{1}{N}\sum_{i=1}^{N} d_i \tag{1}$$

2. **Average input redundancy**: Quantifies the average fraction of inputs that do not influence the node's output.

$$\overline{R} = \frac{1}{N}\sum_{i=1}^{N}\Big(1 - \frac{r_i}{n_i}\Big), \quad n_i = \log_2|f_i| \tag{2}$$

3. **Average sensitivity**: Captures the average probability that a single-bit flip in a node's inputs will change its output.

$$\overline{s} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{n_i\,2^{n_i}}\sum_{x\in\{0,1\}^{n_i}}\sum_{j=1}^{n_i}\big|f_i(x) - f_i(x^{(j)})\big| \tag{3}$$

4. **Number of feed-forward loops (FFLs)**: Counts all three-node motifs where one regulator controls an intermediate, and both jointly regulate a target. Normalization is performed by dividing by the total number of node triplets.

$$\#\text{FFL} = \left|\{(i,k,j) \mid A_{i,j} = A_{i,k} = A_{k,j} = 1\}\right| \tag{4}$$

$$\text{NormFFL} = \frac{\#\text{FFL}}{\binom{N}{3}} \tag{5}$$

5. **Number of feedback loops (FBLs)**: Counts all simple directed cycles longer than one, reflecting the network's feedback complexity. Normalization is performed by dividing by the total possible directed edges among distinct nodes.

$$\#\text{FBL} = \left|\{\text{directed cycles of length} > 1\}\right| \tag{6}$$

$$\text{NormFBL} = \frac{\#\text{FBL}}{N(N-1)} \tag{7}$$

6. **Criticality (Derrida value at $m = 1$)**: Measures the average Hamming distance between one-step updates of original vs. single-bit-flipped states, indicating the network's dynamical regime.

$$D(1) = \frac{1}{M}\sum_{\ell=1}^{M} d\big(F(X^{\ell}),\, F(X^{\ell} \oplus e)\big) \tag{8}$$

Optimization Framework

Given the compound loss function constructed, we use simulated annealing to optimize the inferred model over random perturbations. At each of 50 steps, one of four perturbation types is chosen uniformly at random:

- **Flip a Literal**: Toggle the presence of a gene variable in a Boolean rule by XOR-ing the expression with that literal (i.e. $x \mapsto x \oplus \ell$).
- **Add a Literal**: Insert a new gene (positive or negated) into the rule via logical OR.
- **Remove a Literal**: Convert the rule to disjunctive normal form (DNF), select one clause, and delete a randomly chosen literal from it.
- **Negate the Entire Expression**: Replace the rule f with its logical complement $\neg f$.

If the perturbed network yields a lower loss, the change is accepted, the model file is updated, and annealing proceeds to the next step.

Motif Conservation

We enforce motif conservation by monitoring the participation of each node in feed-forward loops (FFLs) and feedback loops (FBLs) [1, 11]. For FFLs, their signal-processing roles (e.g. pulse generation, sign-sensitive delay) are well characterized [12]. Nodes appearing in more than a threshold number of motifs—set to 10 FFLs or 11 FBLs for our 7-node toy network—are protected. Because counting may redundantly include some motifs, the threshold is conservatively high. Any perturbation that alters a protected node's participation is rejected, and annealing continues. If all nodes exceed the motif threshold, optimization terminates and the final model is returned.

Results

Behaviour of similarity scores with different no. of initial states

It was observed that when the decision-tree(DT) inferred model was tested on the same initial states as it was trained on, it resulted in a similarity score of 100% (because a decision-tree with sufficient depth is able to memorize the

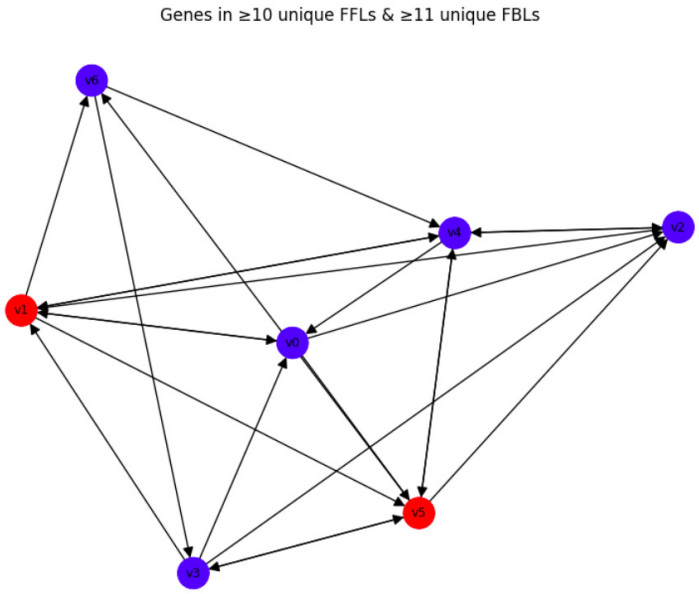

Fig 2. Visualization of motif conservation using networkx. The figure depicts the toy network with 7 genes and the red nodes indicate the genes that are involved in more than the threshold number of motifs.

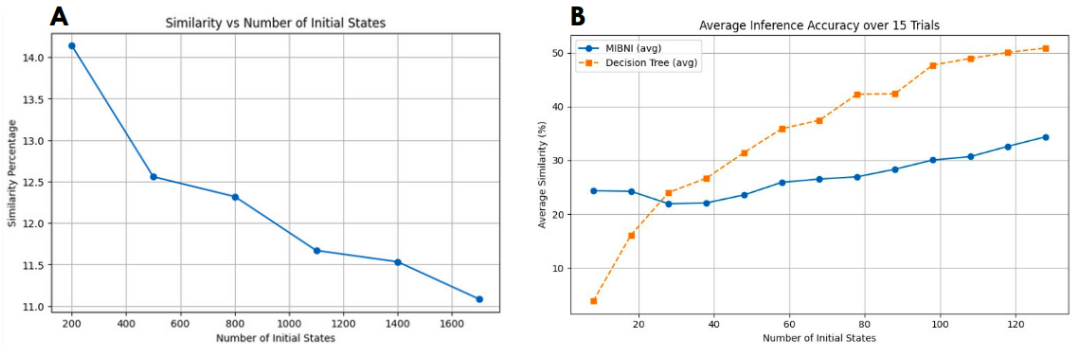

Fig 3. Figure 3A: Behaviour of similarity score over different no. of initial states. Decision-tree method was utilised to infer boolean models for the apoptosis network and a decreasing trend in similarity was observed with increased amount of information about the true network in terms of no. of states. Figure 3B: Benchmarking of Decision Tree and MIBNI on the toy networks with 7 genes and 128 possible initial states. The decision tree reached a higher accuracy of 51% when the complete state space is known but MIBNI performed better when only a few trajectories are known.

state transitions and overfit to the training data) whereas the MIBNI inferred model resulted in a similarity score of 28-36% on different runs. Due to the overfitting nature of DTs, we limited the max-depth parameter to 3 and 4, and observed that the similarity score dropped down to 50-60% on an average.

When the DT-inferred model was tested on different initial states, the similarity score dropped drastically to 10-50% based on the number of initial states it was trained on. We expected the similarity score to increase with the number of initial states expecting the decision-tree to learn more information from a near-complete set of transitions. But as we increased the no. of initial states from 200 to 1800 (on a 39 gene boolean model with 10^{11} possible initial states), the similarity score decreased from 14% to 11%. On the other hand, for a small toy network with 7 genes (128 possible initial states), it was observed that the accuracy of decision tree inference increases consistently from 10% to 50% .

Unexpectedly, MIBNI inference model worked better on predicting the boolean network model on both the extremes (100-128 initial states and 1-20 initial states). It was also observed that the MIBNI inference model had a higher prediction accuracy on different training and testing sets of initial states.

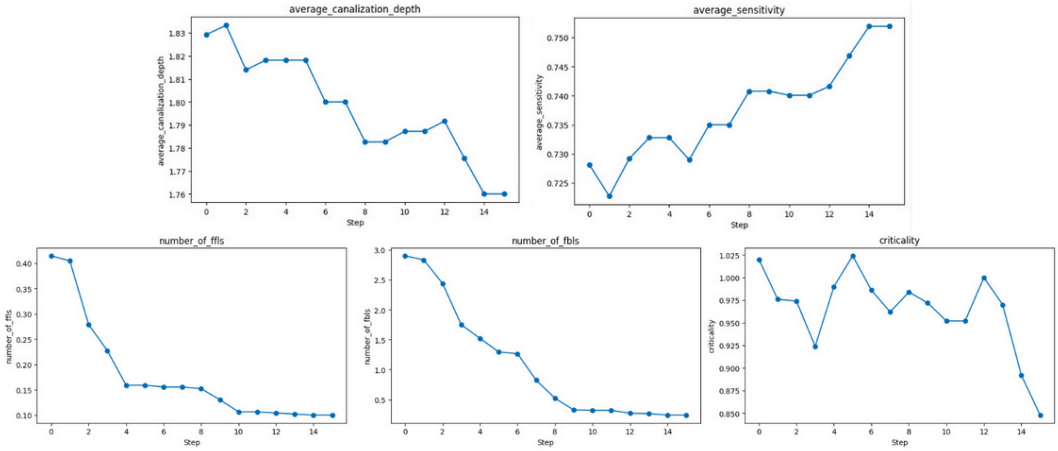

Fig 4. This figure depicts how a model's motif parameters progress with sequential random perturbations. As desired, average sensitivity has increased and criticality has converged at 1. But, no. of FFLs and FBLs and average canalizing depth has decreased over perturbations. Note that the absence of any local trend in the updates is due to the absence of gradient descent like methods to determine direction of optimization.

Perturbing without Motif Conservation

When the ground-truth model was directly perturbed (50 perturbation steps) without any kind of motif perturbation, it was observed that criticality converged near 1 (the optimal value), average sensitivity increased as desired, but average canalizing depth, no. of FFLs and FBLs kept decreasing irrespective of the changes that were done to the loss function. Attempts made to add a regularization term to penalize breakage of network motifs did not show promising results either. This implied that there existed a fundamental error in the way models were perturbed so far.

Motif Conservation

By further visual analysis of the generated perturbed models, we concluded that the nature of the perturbations imposed a very high chance of breaking the motifs than constructing new motifs. After going through relevant literature, we discovered two methods which achieves this : Motif-conserving perturbations and Motif Reconstruction.

Motif-conserving perturbations were implemented to preserve integral FFLs and FBLs and as expected, the no. of FFLs and FBLs did not increase in any perturbation (increase in FFLs was observed in just one run). But surprisingly, unlike before, the conservation of these motifs led to a considerable increase in average canalizing depth of the perturbed model.

Integration with MIBNI

After integrating the simulated annealing approach with MIBNI inference as the project pipeline suggests, we observed that the optimization of average canalizing depth, average sensitivity, criticality were intact and the case with number of FFLs and FBLs did not improve as expected.

But more importantly, it was found that the similarity between the ground-truth model's state transitions and the perturbed inference model's state transitions almost always kept reducing. Attempts made to regularize to penalize large deviations from the ground-truth set of state transitions made perturbation updates sparse.

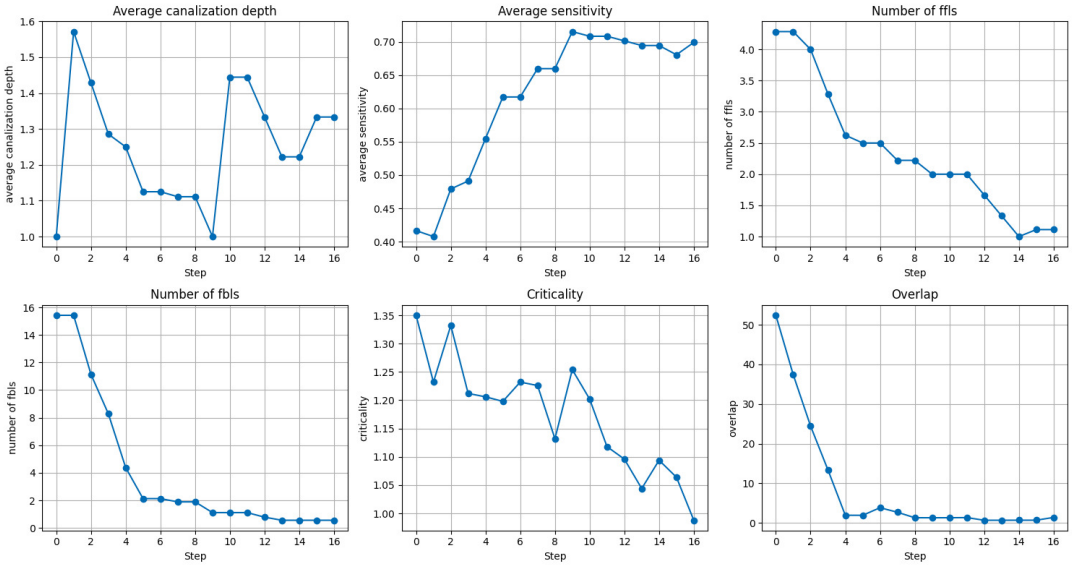

Fig 5. This figure depicts how a model's motif parameters progress with sequential motif conserving perturbations. As desired, average sensitivity and average canalizing depth has increased and criticality has converged at 1. But, no. of FFLs and FBLs has decreased over perturbations and the overlap score fell significantly.

Single trajectory model inference

Instead of training models (both DT and MIBNI) on multiple initial states and computing similarities, we also inferred models from a different toy network using both DT and MIBNI methods; but this time starting from a single chosen initial state. This was performed to simulate the fact that TCEA (Time course expression analysis) experiments usually give a single trajectory of the expression of genes involved.

It was observed that MIBNI resulted in a trajectory similarity of 98% whereas DT resulted in a trajectory similarity of 71%. This clear demarcation explains why MIBNI is the state-of-the-art method for model inference.

Discussion

The poor performance of MIBNI on multiple initial states in comparision to DTs can be attributed to MIBNI's smoothing of rare/contradictory transitions to improve robustness of the inferred model. MIBNI is optimized to perform better when only a few state trajectories are made available. Although the performance trend of MIBNI on different no. of initial states could not be explained.

On the other hand, theoretically there exists a decision-tree (with a max-depth of 4) than can model the toy networks written with a similarity score of 100%. The low similarity score given by the decision tree method is explained by the greedy behaviour of the decision-tree algorithm. This can be further improved by tree ensemble methods which include bagging, boosting and random forests.

The decrease in similarity score on increase of no. of initial states using DTs can be explained by two phenomena : decision tree overfitting to the set of transitions it is trained on and the criticality in biological networks in general.

The overfitting to rare transitions might have led to predicting rules that don't generalize. But for the toy network we observed the opposite, this is because, for lesser no. of possible initial states, regularization is sufficient to withhold enough information as well as not overfit.

Even after implementing Motif conservation, we did not observe any increase in the no. of FFLs and FBLs, this is because the probability of forming an FFL or FBL through the perturbations is still very small. The increase in canalizing depth as a result of Motif conservation could not be explained.

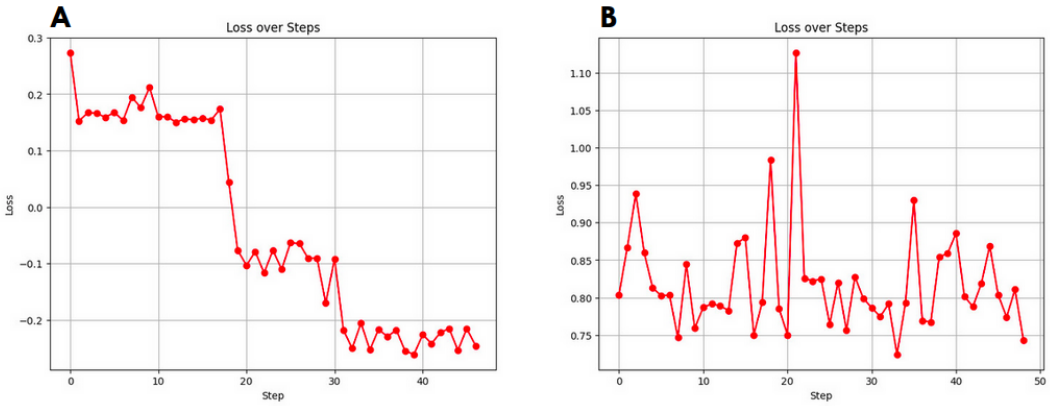

Fig 6. Propagation of Loss over perturbations. Figure 4A depicts the convergence of loss when no regularization was applied. Figure 4B depicts the non-convergence of loss when any kind of regularization was added to the objective function including regularization on FFL deviation, FBL deviation, Overlap/Similarity Deviation

After integration of simulated annealing with MIBNI, the reduce in the overlap/similarity score on perturbations is attributed to the fact that even one wrong perturbation can recursively affect the state transitions. (Example: $G2 = G0 \wedge G1$; $G3 = G2$; $G4 = G3 \vee G0$; a change in the rule for G2 affects all the other non-fixed genes/nodes in the model). Failed regularization to penalize such deviations indicates yet another drawback of randomized perturbations.

The inherent drawback with simulated annealing is that, the probability with which we do not disturb the state similarities is miniscule. Even if regularization is introduced, solving the problem becomes computationally expensive due to the dimensionality of the model and the no. of convex features analyzed. Gradient based methods cannot be applied yet due to the presence of convex features which include number of FFLs and FBLs. This can be mitigated by Nesterov smoothing, which replaces each non-smooth convex feature by a smooth approximation obtained via a strongly convex prox-term in its dual representation.

Most of our testing has been done on only the three mentioned networks out of which one is an expert-written network for apoptosis and the other two are toy-networks generated.The testing can be extended on the available 122 model database.

Conclusion

Our results highlight a clear tradeoff between decision-tree methods and MIBNI. While DTs can memorize transitions and perform well with sufficient training data, they tend to overfit and fail to generalize across different initial states. In contrast, MIBNI, though less accurate with multiple initial states, proves to be robust and significantly more effective when trained on limited trajectories—aligning well with the nature of gene expression data from biological experiments.

Efforts to optimize biologically relevant properties such as criticality, canalizing depth, and motif structures revealed the importance of motif-conserving perturbations in preserving desirable network features. However, integration with perturbation methods like simulated annealing often led to instability in state transitions. Future work could explore smoother optimization techniques such as Nesterov smoothing, ML techniques for weighing individual scores, and a broader benchmarking across more biological networks to improve both robustness and biological fidelity of inferred models.

Acknowledgments

We sincerely thank Associate Professor Manikandan Narayanan, Dept. of Computer Science & Engineering for his continuous guidance throughout the course. We also thank the TAs Saish Jaiswal and Ritviz Kamal for their constructive comments about the project.

We utilized the Canalizing Function Toolbox from the meta-analysis of Boolean networks to compute the components of the designed loss function. The MIBNI algorithm was employed for Boolean network inference. Additionally, one of the Boolean models for apoptosis used in our study was sourced from the curated set of 122 models presented in the meta-analysis paper.

Author Contribution

Hitesh Kandarpa: data collection, defining the loss function, MIBNI inference model, defining overlap and hamming distance-based scores for trajectory similarity, generation of the toy networks with relevant motifs

Kishore Kumar: decision tree inference, perturbation logic, motif conservation, pipeline integration, simulated annealing setup, analysis of varied no. of initial states, model regularization, generation of plots

GitHub Link

Find well-documented codespace of our work here:
`https://github.com/hiteshkandy/Optimised-Boolean-Network-Inference`

Progress After Presentation

1. **Bug fix for increasing number of nodes:** Resolved a formatting issue in the inferred model that previously led to incorrect parsing—Boolean constants and symbols were mistakenly treated as new nodes.
2. **Motif-conserving perturbations implemented:** Designed and integrated mutation operators that preserve local network motifs, allowing biologically plausible model variations.
3. **MIBNI model inference method:** Developed and implemented the Mutual Information-Based Boolean Network Inference (MIBNI) method for rule inference from observed data.
4. **Definition of overlap and Hamming distance score measures:** Established quantitative metrics—overlap and Hamming distance—to evaluate the similarity between inferred and ground-truth Boolean rules.
5. **Integration of MIBNI with simulated annealing and motif-conserving perturbation:** Combined MIBNI inference with simulated annealing optimization using motif-conserving mutations to enhance rule discovery.
6. **Benchmarking of MIBNI and decision-tree method:** Compared the performance of MIBNI with a decision-tree-based inference method across three above specified models.
7. **Bug fix to use inferred model's update rules:** Corrected an error that prevented the simulation engine from applying the newly inferred update rules during dynamic testing.

References

1. Alon U. Network motifs: Theory and experimental approaches. Nature Reviews Genetics. 2007;8(6):450–461.
2. de Jong H. Network modelling of gene regulation. PLoS Computational Biology. 2016;12(1):e1004726.

3. Maraziotis I, Others. A review of modeling techniques for genetic regulatory networks. Mathematical Biosciences and Engineering. 2012;9(2):291–320.

4. Naldi A, Others. Concepts in Boolean network modeling: What do they all mean? Briefings in Bioinformatics. 2020;21(5):1396–1409.

5. Albert R, Othmer HG. The topology of the regulatory interactions predicts the expression pattern of emphDrosophila segment polarity genes. PLoS Computational Biology. 2003;1(5):e54.

6. Li F, Long T, Lu Y, Ouyang Q, Tang CI. The yeast cell-cycle network is robustly designed. Proceedings of the National Academy of Sciences. 2004;101(14):4781–4786.

7. Azócar-Aedo L, Others. Boolean modeling of biological regulatory networks: A methodology tutorial. Journal of Biomedical Informatics. 2012;45:603–610.

8. Meyer PE, Others. Review and assessment of Boolean approaches for inference of gene regulatory networks. BMC Bioinformatics. 2022;23(1):211.

9. Kadelka C, Butrie TM, Others. A meta-analysis of Boolean network models reveals design principles of gene regulatory networks. Science Advances. 2021;7(11):eadj0822.

10. Barman S, Kwon Y. A novel mutual information-based Boolean network inference method from time-series gene expression data. PLoS ONE. 2017;12(2):e0171097. doi:10.1371/journal.pone.0171097.

11. Milo R, Shen-Orr S, Itzkovitz S, Kashtan N, Chklovskii D, Alon U. Network motifs: Simple building blocks of complex networks. Science. 2002;298(5594):824–827.

12. Mangan S, Alon U. Structure and function of the feed-forward loop network motif. Proceedings of the National Academy of Sciences. 2003;100(21):11980–11985.