Investigating the Development of Mechanistic Models with Reusable Kinetic Parameters using a Synthetic RNA-gene Pulse Generator

Thao T. Nguyen [a, b], Hongchao Zhang [b], Xun Tang [a,*]

[a] Department of Chemical Engineering, Louisiana State University, Baton Rouge LA 70803, United States
[b] Department of Mathematics, Louisiana State University, Baton Rouge, LA 70803, United States
[*]corresponding author

Abstract

Mechanistic modeling is a powerful tool to study the properties of synthetic gene circuits. However, one challenge with modeling is the reusability of the model parameters fitted from one system to another. This can be contributed by factors including variations in the cellular context, model structure and complexity, as well as the parameterization approach. In this study, we focus on a pulse generator that leverages the temporal disparity in small transcription activation RNA (STAR) and CRISPRi-mediated genetic regulations to produce a pulse in the target gene expression, to investigate the impact of model structure, model complexity, parameterization approaches, and the amount of training data on the reusability of kinetic parameters fitted from individual gene circuit parts to the connected comprehensive circuit. Specifically, we developed and parameterized STAR and CRISPRI regulatory pathway separately with different parameterization approaches, model complexity, and amount of cell-free experimental data, and evaluate the applicability of the separately fitted parameters in predicting the comprehensive pulse generator dynamics.

Our results showed that model with the most detailed GFP expression mechanism achieved the lowest mean square error with Genetic Algorithm, in predicting the behavior of the pulse generator, as compared to other simplified model and/with other parameterization algorithms. Interestingly, MATLAB *fmincon* demonstrated remarkable robustness in predicting the pulse generator behavior, even in the presence of irregular noise within the experimental data. Although our cell-free experimental data are inherently time-dependent, the systematic subsampling approach at 10-20% effectively preserves the underlying temporal pattern of GFP concentration, suggesting a sparse sampling would suffice for certain mechanistic model parameterization. Notably, analysis of the parameter variations on the top 30% best predicting parameter sets demonstrated that some parameters show a wide range of variation, indicating the existence of multiple optimal solutions, and further suggesting additional experiments are needed for improved accuracy.

Overall, this study showcases a comprehensive approach to explore the development of mechanistic model from subunit gene circuits for the comprehensive circuit dynamics simulation, we expect findings to benefit future modeling studies for genetic circuits design.

1. Introduction

Harnessing the capability of regulating gene expression and reprogramming metabolic pathways with synthetic gene devices has been a longstanding pursuit in the field of synthetic biology. over the past decades, synthetic biologists have devised a wide range of gene circuits from logic gates [xx], to oscillators [xxx], to biological controllers [xx], and to advanced nucleic acid-based biocomputing circuits [xxx]. Traditionally, extensive empirical testing is required to fine-tune genetic circuits. Researchers embarked on a trial-and-error journey, adjusting parameters, testing hypotheses, and iterating until the desired behavior was achieved. While this approach has yielded valuable insights, it is time-consuming, resource-intensive, and, at times, may not uncover the full range of possibilities within a particular system. The fusion of mathematical modeling and experimental validation introduces a systematic and data-driven approach to genetic circuit design. Instead of relying solely on experimentation, researchers can use mathematical models to simulate the behavior of a genetic network under various conditions. This enables the exploration of a vast design space, predicting how different components and parameters will interact before they are physically implemented in laboratories. Moreover, the process is not confined to a single round of experimentation. Models can be refined, and experiments can be precisely tailored based on model predictions. This iterative cycle accelerates the design-build-test-learn loop inherent in synthetic biology, significantly reducing the time and resources required to engineer novel genetic circuits.

While the prevalent method to designing a novel gene circuit is still via experimental design-validation cycles, model-guided exploration and design of gene circuits has received acclaimed success for both bacteria and mammalian cells applications [29–33]. Modeling gene circuits is currently achieved with: 1) phenomenological or data-driven modeling; 2) deterministic mass action expression (mechanistic modeling) based on first-principles, typically via a set of ordinary differential equations (ODEs); and 3) stochastic modeling, where the occurrence of the reactions are simulated with transition probabilities to capture the inherent stochasticity in biological systems. The deterministic mass action expression-based ODE models represent a bottom-up approach to describe the dynamics on an averaged basis, and holds promise for general applicability. Given its relatively simpler design and intuitive implementation, mass action expression-based ODE models have received a wide popularity [xxx].

Despite its effectiveness, one biggest challenge associated with using mass action expression-based ODE models in simulating and guiding novel gene circuits design is the applicability of the kinetic parameters fitted for one system to other different yet related systems. Several reasons would contribute to this phenomenon. First, the circuit itself is known to be context dependent, which means the behavior of the same circuit would differ in different environment, due to variations in the host system biological contents. Second, overfitting of a parametric model would cause the fitted parameters to not be applicable for other systems. Third, the complexity of the model, the availability of the experimental measurements for model fitting, and the selected optimization approach, could all results in local minimal parametrization that the fitted parameters would not predict accurate for the same design in a different setting. In this study, we investigate how the model complexity, the performance of widely used parameterization methods, and the amount of experimental measurements would affect the accuracy of the fitted parameters for sub-system gene circuits, in predicting the connected circuits. Specifically, we focus on an RNA-based pulse generator, with experimental measurement collected from the cell-free system, expecting that the cell-free testbed would provide a mitigated variation in resource as compared to *in vivo* experiments.

The paper is organized as following. In xx, we describe the gene pulse generator, and the comprehensive base models for the circuit; in xxx, we describe the xxx; in xxx, we xxx.

System of Interest and Methods

The RNA-based pulse generator we focus on in this study is the same as in xx, which is composed by a Small Transcription Activation RNAs (STARs)-mediated activation pathway, and a CRISPRi/Cas9-mediated repression pathway, as in Figure 1A.

The STAR bind to the target RNAs situated within the 5'-untranslated region (5'-UTR) of the gene of interest to turn on the gene expression. These target RNAs fold into transcriptional terminators, a structural feature that halts transcription by causing the RNA polymerase to disengage from the DNA complex before it reaches the downstream gene. Transcription can proceed only when STARs are present and blind to target RNA. The direct RNA-RNA interaction means the transcription process occurs within minutes (Chappell et al., 2017). On the other hand, Clustered Regularly Interspaced Short Palindromic Repeats Interference (CRISPRi) relies on targeting a catalytically dead Cas9 (dCas9) nuclease. This process allows a guide RNA (gRNA) with a segment that perfectly complements the gene's sequence of interest. When the ribonucleoprotein gRNA-dCas9 complex binds to DNA, it can either block the binding of RNA polymerase if the targeted region is near a promoter or impede transcription elongation if the targeted region is within the gene. This method offers the ability to independently regulate multiple genes or integrate signals for genetic circuits, similar to logic gates. Since CRISPRi requires the formation of RNA-protein repressor complex before binding to DNA for repression, it can take up to hours in order for regulation to occur (Chappell et al., 2017). Due to the timescale difference between the two regulation pathways, a pulse in the output gene production would be observed.

To understand the relationship between model complexity, optimization methods, sampling size for model fitting and the applicability of the fitted parameters for other systems, here we first develop separate models for the STAR and CRISPRi pathways, and perform model parameterization, and then use the fitted parameters to predict the connected pulse generator circuits (Figure 1B). The pulse generator dynamics prediction accuracy is then used as the metric to evaluate the reusability of the parameters, thus guiding the development of the guidelines for model development and parameterization. Specifically, we consider four parameterization approaches, a Python derivative-free gradient descent (BOBYQA), MATLAB fmincon, genetic algorithms, and Bayesian optimization. For each parameterization method, we also varying the amount of experimental data used in fitting the model, as well as varying the complexity of the GFP expression modeling.

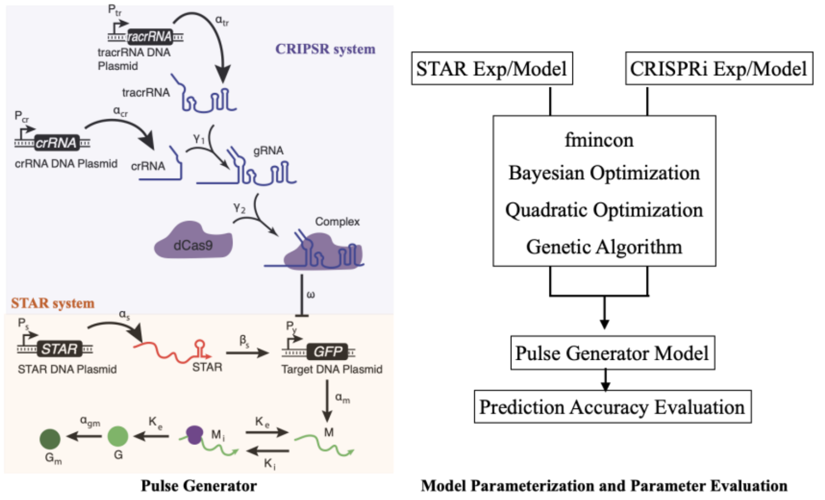


Figure 1. A) pulse generator composed by STAR and CRISPRi pathways. B) Parameters are separately fitted for STAR and CRISPRi models and then used to predict the connected pulse generator.

1.3 Predictive Modeling for Pulse Generator Model

In order to construct an accurate predictive model for the pulse generator model using the STAR and CRISPRi mechanisms, it is essential to optimize and fine-tune the parameters associated with these regulatory components. This process involves identifying the best-fitted parameters that govern the behavior of STARs and CRISPRi within the genetic circuit.

The optimization of STARs model parameter is crucial to capturing its rapid activation kinetics accurately. Time-series measurements of gene expression in response to STARs are collected and utilized as input for optimization algorithms. In our study, we will use Python derivative-free gradient descent (BOBYQA) and MATLAB fmincon, genetic algorithms, and Bayesian optimization to iteratively adjust STARs parameters for the STAR binding affinity, dissociation rates, and RNA-RNA interaction kinetics to minimize the error between model predictions and experimental observations. Once the optimization process converges with the least error between the experimental and predicted values, we can determine the set of best-fitted parameters to model the STARs mechanism. These parameters are now capable of describing how STARs bind to their target RNAs and activate gene expression with a high degree of precision.

Similarly, we optimize the CRISPRi parameters used to model the delayed repression kinetics. The collected experimental data is used to fine-tune parameters such as dCas9 binding affinity, guide RNA (gRNA) efficiency, and the time required for RNA-protein repressor complex formation. These algorithms take into account the complex interactions between CRISPRi components and the target genes to ensure an accurate representation of the regulatory mechanism.

Once we obtain the best-fitted parameters for both STARs and CRISPRi in hand, we can construct a predictive model for the pulse generator. This complex model combines the rapid activation capabilities of STARs with the delayed repression of CRISPRi, as described in the incoherent type 1 feedforward loop (I1-FFL). The predictive model allows researchers to anticipate and understand how the pulse generator behaves under various conditions, enabling them to design genetic circuits with desired pulse characteristics. It also facilitates the exploration of different parameter combinations to optimize the pulse shape and timing for specific applications in synthetic biology, such as precisely timed drug delivery or gene expression control.

2. Mechanistic Models Development

2.1 STAR Model

STARs achieve gene expression activation through its interaction with a sequence-specific target RNA. When STARs are absent, the target RNA sequence resides in the untranslated 5’-UTR region of the green fluorescent protein (GFP) gene. In this state, the target RNA folds into a transcriptional terminator structure, causing RNA polymerase to disengage from the DNA complex prematurely, thereby halting transcription. Conversely, in the presence of STARs, STARs bind to both the linear and 5’ half of the terminator hairpin on the target RNA, preventing terminator formation and allowing downstream transcription to occur.

1. $\frac{dSTAR}{dt} = \alpha_s * P_s - \delta_s * STAR - \beta_s * P_y * STAR$
2. $\frac{dM}{dt} = \alpha_m * \beta_s * P_y * STAR - \delta_m * M - K_i * M + K_e * M_i$
3. $\frac{dM_i}{dt} = K_i * M - K_e * M_i$
4. $\frac{dG}{dt} = K_e * M_i - \alpha_{gm} * G$
5. $\frac{dG_m}{dt} = \alpha_{gm} * G$

In the STAR model, the activation process was simplified and modeled as a one-step reaction, where STAR binds directly to the free promoter P_y, facilitating transcription activation at a rate represented by β_s. To complete the STARs model, a set of eight parameters was estimated through parameterization. These parameters include α_s, δ_s, δ_m, β_s, α_m, K_i, K_e, and α_{gm}, encompassing aspects such as transcription and degradation rates, translation initiation and elongation rates, and GFP maturation respectively.

2.2 CRISPRi Model

The CRISPRi model relies on the formation of an RNA-protein repressor complex before binding to DNA to induce gene repression. This system follows a multi-step process where crRNA, tracrRNA, and dCas9 come together to form the CRISPR complex. The complex, guided by tracrRNA, binds to a specific DNA sequence, ultimately blocking RNA polymerase binding or halting transcription elongation depending on the proximity of the target GFP sequence to the promoter. Additionally, we assumed that the formation of the CRISPR-Cas9 complex and its binding to the free promoter P_y, resulting in the formation of the repressed promoter ${P_y}^{-}$, is irreversible.

1. $\frac{dcrRNA_1}{dt} = \alpha_{cr} * P_{cr} - \delta_{cr} * crRNA - \gamma_1 * crRNA * tracrRNA$
2. $\frac{dtracrRNA_2}{dt} = \alpha_{tr} * P_{tr} - \delta_{tr} * tracrRNA - \gamma_1 * crRNA * tracrRNA$
3. $\frac{dgRNA}{dt} = \gamma_1 * crRNA * tracrR - \gamma_2 * gRNA * dCas9 - \delta_g * gRNA$
4. $\frac{ddCas9}{dt} = \gamma_2 * gRNA * dCas9$
5. $\frac{dComplex}{dt} = \gamma_2 * gRNA * dCas9 - \omega * Complex * P_y$
6. $\frac{dP^{-}{}_{y}}{dt} = \omega * Complex * P_y$
7. $\frac{dM}{dt} = \alpha_m * P_y - \delta_m * M - K_i * M + K_e * M_i$
8. $\frac{dM_i}{dt} = K_i * M - K_e * M_i$
9. $\frac{dG}{dt} = K_e * M_i - \alpha_{gm} * G$
10. $\frac{dG_m}{dt} = \alpha_{gm} * G$
11. $P_y = {P_y}^{tot} - {P_y}^{-}$

The CRISPRi captures this activation process as a two-step reaction process. In the first step, trans-activating crRNA (tracrRNA) binds to CRISPR RNA (crRNA) to form a guide RNA (gRNA) at a rate represented by γ_1. In the second step, dCas9 binds to the guide RNA (gRNA) to complete the formation of the active repressor complex at a rate represented by γ_2. The are a total of 12 parameters were estimated through the parameterization process. These parameters include α_{cr}, α_{tr}, δ_{cr}, δ_{tr}, δ_g, γ_1, γ_2., ω, α_m, K_i, K_e, and α_{gm} used to describe transcription and degradation rates, translation initiation and elongation rates, and GFP maturation respectively.

2.3 IFFL Pulse Generator Model

1. $\frac{dSTAR}{dt} = \alpha_s * P_s - \delta_s * STAR - \beta_s * P_y * STAR$
2. $\frac{dcrRNA_1}{dt} = \alpha_{cr} * P_{cr} - \delta_{cr} * crRNA - \gamma_1 * crRNA * tracrRNA$
3. $\frac{dtracrRNA_2}{dt} = \alpha_{tr} * P_{tr} - \delta_{tr} * tracrRNA - \gamma_1 * crRNA * tracrRNA$

4. $\frac{dgRNA}{dt} = \gamma_1 * crRNA * tracrR - \gamma_2 * gRNA * dCas9 - \delta_g * gRNA$

5. $\frac{ddCas9}{dt} = -\gamma_2 * gRNA * dCas9$

6. $\frac{dComplex}{dt} = \gamma_2 * gRNA * dCas9 - \omega * Complex * P_y$

7. $\frac{dP^{-}y}{dt} = \omega * Complex * P_y$

8. $\frac{dM}{dt} = \alpha_m * \beta_s * P_y * STAR - \delta_m * M - K_i * M + K_e * M_i$

9. $\frac{dM_i}{dt} = K_i * M - K_e * M_i$

10. $\frac{dG}{dt} = K_e * M_i - \alpha_{gm} * G$

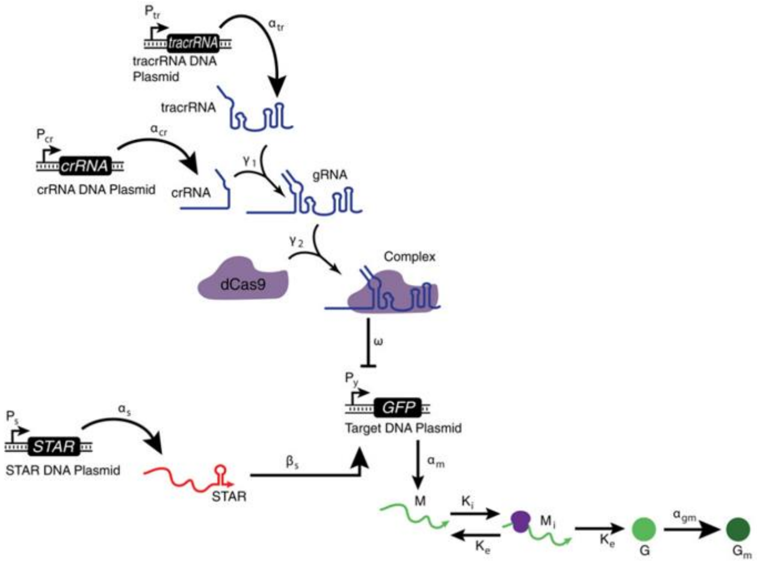
11. $\frac{dG_m}{dt} = \alpha_{gm} * G$

12. $P_y = {P_y}^{tot} - {P_y}^{-}$

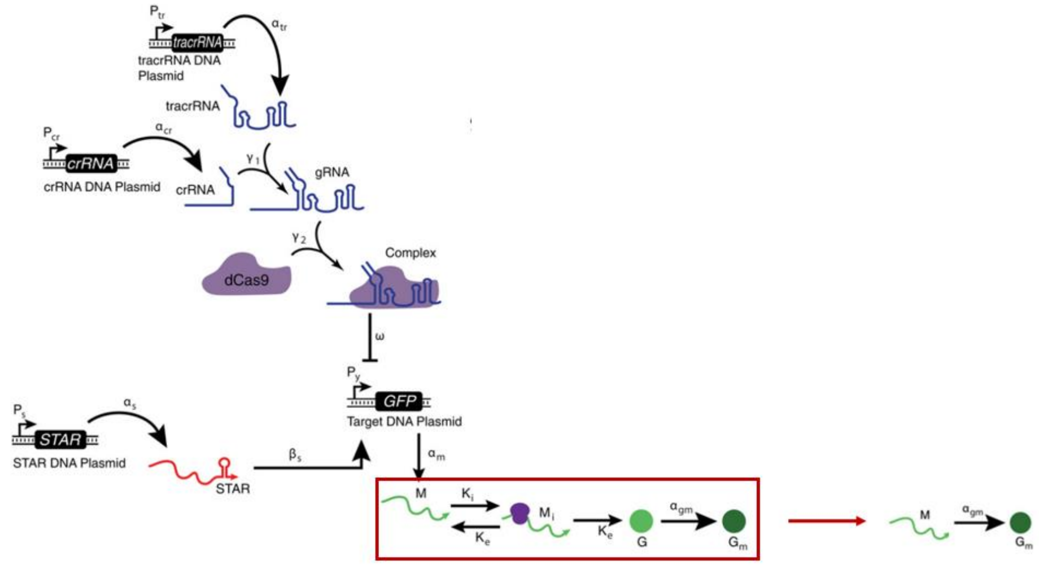
3. Results

3.1 Model Simplification

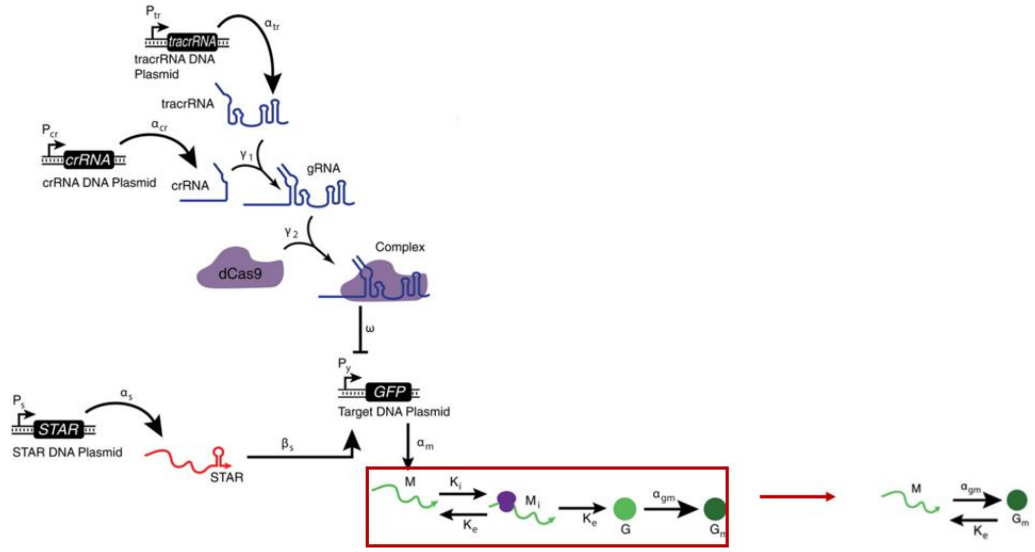
3.1.1 Original Model



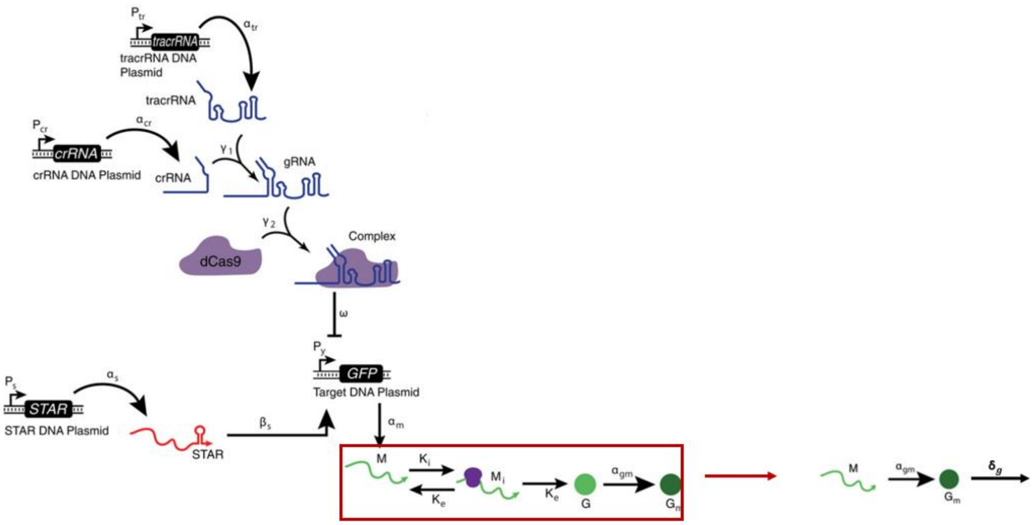
3.1.2 Elementary Model



3.1.3 Reversible Model



3.1.4 Degradation Model



3.2 Parameterization Methods used for Fitting STAR and CRISPRi Model Parameters

We perform parameterization for both the STAR and CRISPRi models to determine the optimal set of parameters that can effectively replicate the TXTL experimental data. To achieve this, we employ various optimization techniques, including MATLAB's *fmincon*, Genetic Algorithm, Bayesian, and Python's Bound-Constrained Optimization by Quadratic Approximation (BOBYQA) optimization.

MATLAB fmincon

We use the MATLAB fmincon optimization process to determine the best parameters for the STAR and CRISPRi model. The fmincon function is part of the MATLAB Optimization Toolbox that uses derivative-free technique to find minimum, or maximum, of a nonlinear objective function subjected to equality or inequality constraints. We use a parameter discretization approach to divide all parameters into 10 evenly spaced intervals within their lower and upper bound. By creating these intervals, the optimization process considers the inherent complexity and variability of biological systems.

Once these intervals are established, the MATLAB function ‘randsample’ randomly selects one value from each of the 10 intervals for every parameter. This random selection introduces an element of exploration into the optimization process. Thus, each time the optimization is initiated, a unique set of parameter values is generated, reflecting a different starting point.

These randomly selected parameter values are then consolidated into an initial parameter vector, denoted as ‘p0’. Subsequently, the MATLAB fmincon function takes in the initial parameter vector ‘p0’ and minimizes the objective function defined by the STAR and CRISPRi model. Throughout this process, fmincon frequently imposes constraints on parameter values to prevent the generation of unrealistic values outside of the lower and upper bounds. The core objective of this optimization process is to identify parameters values that yield model predictions closely aligned with experimental data, effectively minimizing the sum of squared errors between predicted and experimental values.

MATLAB Genetic Algorithm

The MATLAB Genetic Algorithms is a stochastic optimization technique inspired by biological evolution most suitable for solving complex, nonlinear, and multimodal objective functions that does not rely on the derivative of the objective function itself. Unlike MATLAB fmincon which utilized parameter discretization, this method refrains from dividing the model parameters into predefined intervals. Instead, the Genetic Algorithm operates in a more flexible manner, allowing for optimal parameter search across a continuous parameter space. In the context of our model, the predicted parameters can be any real value as long as it is within the specified lower and upper bounds. The Genetic Algorithm begins with a randomly generated population of potential parameter sets which evolves over multiple generations through processes such as mutation, crossover, and selection. By iteratively modifying and evaluating parameter sets, the Genetic Algorithm aims to identify an optimal parameters combination that minimizes the STAR and CRISPRi objective function. We further enhance the algorithm by including 'PlotFcn','gaplotbestf,' option which enables real-time visualization of the optimization process to help track of how the fitted function evolved over multiple generations.

A feature of Genetic Algorithm is its ability to automatically identify the convergence point that closely approximates the best possible fit to the empirical data. When the current best objective function value falls within the convergence threshold set by 'FunctionTolerance' as 10^{-8}, the algorithm concludes that further iterations are unlikely to yield significant improvement to the model fits. Thus, continuous parameter search is terminated, and the set of parameters is saved.

MATLAB Bayesian

The MATLAB Bayesian optimization is also a derivative-free technique. However, the optimization framework uses a surrogate-based Gaussian process to build a probabilistic model of the objective function capable of approximating the behavior of STAR and CRISPRi model. Instead of traversing with the parameter space continuously, like Genetic Algorithm, surrogate models balance exploration by sampling uncertain regions, and exploitation of sampling regions with expected high performance. Bayesian optimizations focus more on local exploration and exploitation, which is advantageous to parameters that precisely fit the STAR and CRISPRi models despite potentially missing the global optimal.

We use the parallel computing option available in MATLAB 'bayesopt' function to expedite the parameter optimization process. Parallel computing involves the simultaneous execution of multiple configuration pools to identify the optimal set of parameters, while minimizing the burden of function evaluations. Increasing the number of parallel configuration pools helps exploration to evaluate more sample points across the optimization space, but it can hinder the optimization process by focusing solely on local regions. Furthermore, a higher number of pools can lead to faster convergence despite the increase in computation. In this study, we used 50 parallel configuration pools. By utilizing parallel computing, the Bayesian optimization algorithm can evaluate multiple parameters sets concurrently, reducing the total number of function evaluations required to reach convergence. This is particularly advantageous when optimizing complex gene circuit models comprised of multiple sets of parameters, where each function evaluation is often computationally expensive.

Python Bound-Constrained Optimization by Quadratic Approximation

The Bound-Constrained Optimization by Quadratic Approximations (BOBYQA) is a Python derivative-free optimizer designed for bound-constrained minimization problems. BOBYQA converges toward the optimal solution by iteratively building and refining quadratic approximations of the STAR and CRISPRi objective function within the specified bound-constrained parameter space. Unlike, MATLAB fmincon optimization which replies on gradients to systematically explore the parameter space, the quadratic approximations capture not only objective function values but gradient-like information without calculating the explicit derivatives.

When the built-in feature ‘set_global_minimum’ is equal to 1, it signifies that BOBYQA has identified a solution that is considered a global minimum, representing the best possible outcome within the constraints and the approximation accuracy. In contrast to other derivative-free methods, both the Genetic Algorithms and Bayesian optimization does not inherently include such a flag to confirm the global minimum. It is important to note that the complexity for modeling RNA-based regulator parameters and error prone experimental values does not guarantee global optimality in all cases.

IFFL Model Prediction

After successfully parameterizing the STAR and CRISPR models, we will investigate the feasibility of reusing these well-fitted parameters when constructing a more comprehensive model, specifically an Incoherent Feed-Forward Loop (IFFL) model. Integrating the IFFL model allows us to evaluate the predictive power and applicability of these parameters within a larger, interconnected gene circuit context. This step is critical to ensure that the comprehensive gene circuit model's predictive capabilities are as accurate as possible. By reusing the parameters obtained from the individual components in the combined IFFL model, our broader objective is to establish a consistent and reliable mathematical model. This model should be capable of simulating and predicting the dynamics of the entire gene circuit accurately.

Systematic Subsampling of Experimental Data

Systematic subsampling is a data reduction technique commonly used to systematically extract a representative subset of data from a larger database. In our study, we decided to perform 10% subsampling, which remove a total of 5 experimental data removes at regular intervals of 10 data points each, and 20% subsampling, which remove a total of 10 experimental data points at regular intervals of 5 data points each. One of the advantages of systematic subsampling is the ability to preserve key characteristics while drastically reducing the sample size of a dataset. Thus, we can significantly reduce the computational efficiency needed to determine the best fitted parameters of the STAR and CRISPRi models compared to analyzing the full 49 EGFP concentrations.

The time-dependent experimental data collected at 5-minute intervals for the Transcription-Translation (TXTL) system presents a unique challenge when considering systematic subsampling. The primary disadvantage of employing this technique lies in its potential to introduce bias into the subset of data, particularly if there exists an inherent temporal pattern within the dataset that coincides with the chosen subsampling interval. In this context, systematic subsampling could inadvertently omit crucial information regarding when the time-dependent green fluorescent protein (EGFP) concentrations ‘spike’ occurs in the gene pulse generator (IFFL).

4. Results

4.1 Error Analysis

We use the mean squared error (MSE) as a metric to directly evaluate the best-fitted parameters obtained from different optimization techniques and the STAR and CRISPRi experimental values, respectively. Furthermore, we want to assess the effectiveness of these parameters in predicting EGFP concentration for complex Incoherent Feed-Forward Loop (IFFL) model across varying amounts of subsampled experimental data.

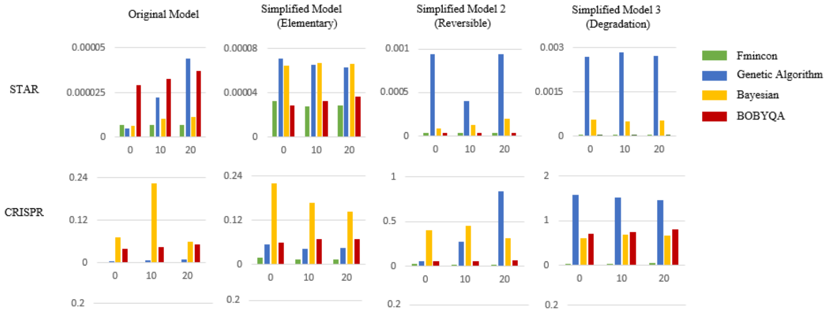


Figure 2: Mean Square Error of STAR and CRISPRi

A major disadvantage to using MSE metric lies in its susceptibility of outliers. If any of the optimization techniques yield parameter values that deviate significantly, it could exert a substantial impact on the prediction of EGFP concentration within the IFFL model, making the MSE results less reliable.

4.2 Best Predictions for Gene Pulse Generator Model (IFFL)

Among the 144 sets of fitted parameters, the original model yielded the lowest mean square error (MSE) values for the top three best IFFL predictions ***(Table 1)***. The smallest prediction error, at 0.2230, was achieved using the Genetic Algorithm optimization technique in MATLAB. In contrast, the remaining best-performing models were obtained through the MATLAB fmincon optimization technique. Interestingly, it seems that the extent of data subsampling does not exhibit a consistent impact, given that the top three prediction models each utilized varying percentage of data reduction.

	Best IFFL Prediction 1	**Best IFFL Prediction 2**	**Best IFFL Prediction 3**
Model	Original		
Optimization Method	Genetic Algorithm	fmincon	fmincon
Systematic Subsampling (%)	10%	20%	0
IFFL Error	0.2230	0.2780	0.3067

Table 1: The 3 IFFL Predictions with the lowest MSE

In ***Figure 2,*** when we plotted the three IFFL predictions with the lowest MSE errors alongside the experimental results, we noticed that all of them accurately predicted the behavior of the pulse generator model (+STAR +crRNA), even when considering the presence of irregular noise in the experimental data.

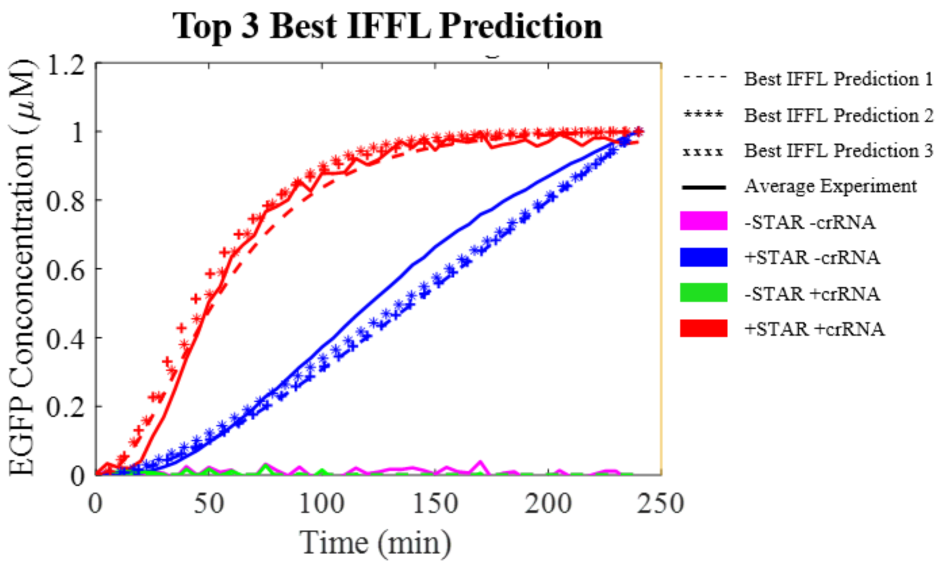


Figure 2: Graph of the 3 IFFL Predictions with lowest MSE

However, under the CRISPRi controlled condition (-STAR +crRNA), these top three predictions consistently underestimated the experimental values and exhibited a linear pattern in their predictions.

4.3 All IFFL Predictions from Original Model

As the three IFFL predictions with the lowest MSE errors are all associated with the original model, we need to broaden our perspective and examine the performance of all 36 sets of best-fitted parameters generated using MATLAB's *fmincon*, Genetic Algorithm, and Bayesian optimization, as well as Python's BOBYQA ***(Figure 3).***

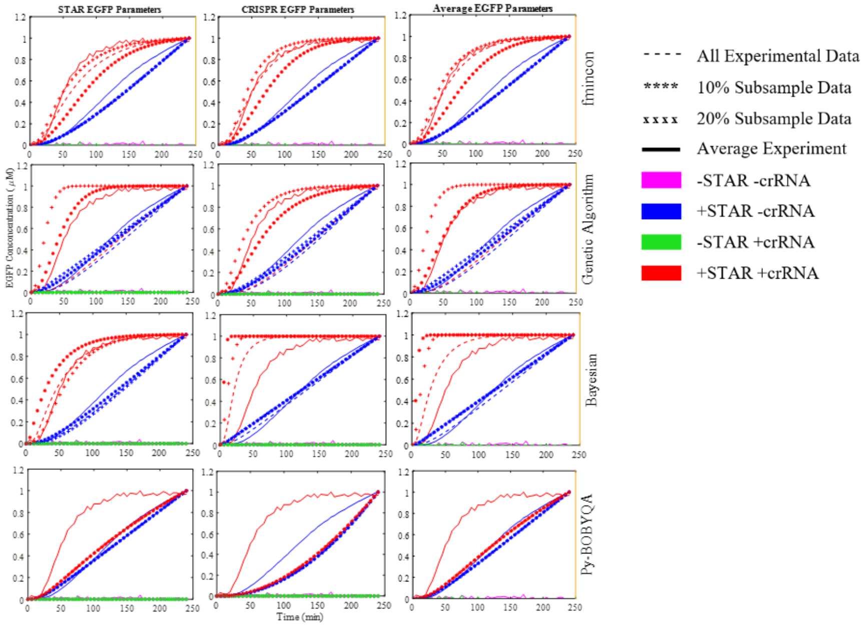


Figure 3: All 36 set of Best-Fitted Parameters used in IFFL Prediction for Original Model

The parameterization process using MATLAB *fmincon* method showed remarkable robustness at predicting the pulse generator behavior, even in the presence of irregular noise within the experimental data of the (+STAR +crRNA). A similar pattern emerges for all IFFL predictions using MATLAB *fmincon* as the top 3 IFFL predictions where the best-fitted parameters underestimated experimental data for the CRISPRi controlled condition (-STAR +crRNA).

In IFFL predictions made by MATLAB Genetic Algorithm, we observed that the set of best-fitted parameters derived from the STAR model's GFP translational initiation, elongation, and maturation parameters only fails to align with experimental data under a 20% reduction of experimental data. It suggests that some of the STAR parameters might be particularly influential in predicting the dynamics of the pulse generator behavior.

Comparatively, the IFFL predictions for MATLAB Bayesian using the set of best-fitted parameters derived from CRISPR model's GFP transcriptions kinetic parameters deviate significantly from the experimental data. The discrepancy indicates that despite parallel computation, the MATLAB Bayesian approach falls short at producing parameters with high predictive accuracy in contrast with the MATLAB fmincon and Genetic Algorithm when there are large model degree of freedom like the CRISPR model.

4.4 Parameters Analysis of the Top 30% IFFL Predictions

Analyzing the range of best-fitted parameters allows us to identify which parameters have the most significant impact on the model's predictions. A wide parameter range indicates a high level of uncertainty surrounding the parameter estimate, while a narrow range signifies a greater degree of confidence in the accuracy of that estimate. Notably, both the CRISPRi-related kinetic parameter ω responsible for binding the ribonucleoprotein complex, and the STAR-related kinetic parameter β_s responsible for binding to the target DNA plasmid, exhibited the highest degree of variability when we conduct an analysis of parameter variations within the top 30% of IFFL predictions with the lowest MSE. This result suggests multiple optimal solutions for binding kinetics parameters, underscoring their crucial role in accurately predicting the gene pulse generator model. Furthermore, any parameters that exhibit a wide range can be particularly sensitive, meaning that even small alterations in their values can trigger significant shifts in the model's outcomes.

The effectiveness of a mathematical model in accurately representing biological processes is contingent upon its capacity to faithfully capture the inherent dynamics governing those processes. The degradation model does not represent the accurate time scale protein degrades over time. The discrepancy can be observed in Figure 5, where all the parameters associated with the degradation model, except for δ_S , the degradation rate of STAR, exhibit a prediction are highly varied across a wide prediction range. Since the range of these predictions are well-defined, we can refine the parameter range to reduce parameter uncertainty and improve the model's accuracy to be within a plausible range. Thus, produce a smaller parameter search range for each optimization process such that the best-fitted parameters are closer to capturing the protein degradation process.

4.4.1 Model Complexity

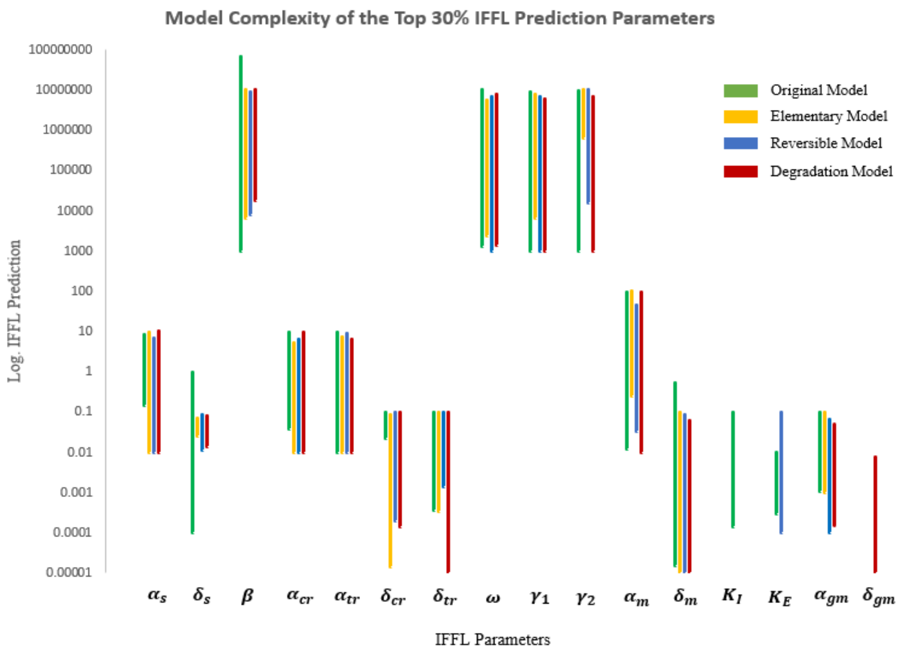


Figure 4: Top 30% IFFL Predictions based on Model Complexity

4.4.2 Parameterization Techniques

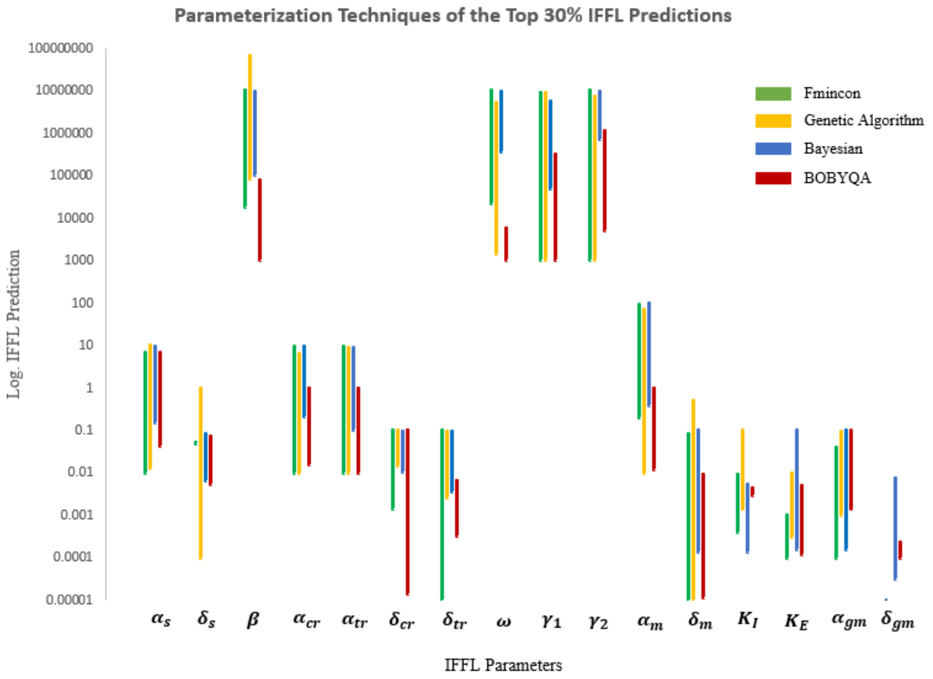


Figure 5: Top 30% IFFL Prediction based on Parameterization Techniques

4.4.3 Subsampling Experimental Data

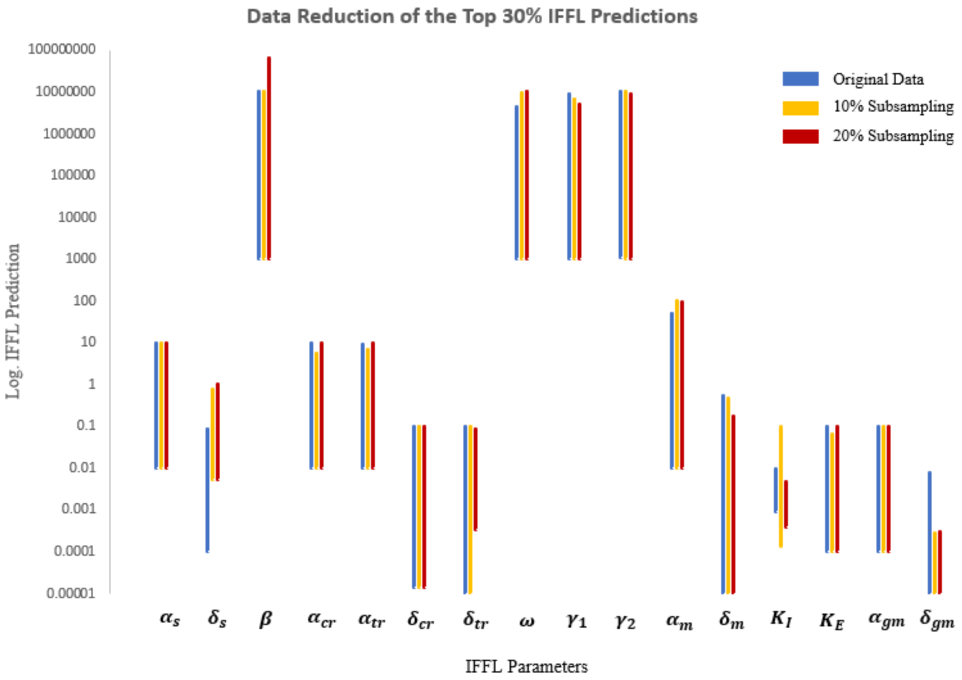


Figure 6: Top 30% IFFL Prediction based on Experimental Data Subsampling

References

1. Agrawal, D. K., Tang, X., Westbrook, A., Marshall, R., Maxwell, C. S., Lucks, J., Noireaux, V., Beisel, C. L., Dunlop, M. J., & Franco, E. (2018). Mathematical modeling of RNA-based architectures for closed loop control of gene expression. *ACS Synthetic Biology*, *7*(5), 1219–1228. https://doi.org/10.1021/acssynbio.8b00040

2. Chappell, J., Westbrook, A., Verosloff, M., & Lucks, J. B. (2017). Computational design of small transcription activating RNAs for versatile and dynamic gene regulation. *Nature Communications, 8*(1), 795. https://doi.org/10.1038/s41467-017-01082-6

3. Alexandra Westbrook, Xun Tang, Ryan Marshall, Colin S. Maxwell, James Chappell, Deepak K. Agrawal, Mary J. Dunlop, Vincent Noireaux, Chase L. Beisel, Julius Lucks, Elisa Franco. Distinct timescales of RNA regulators enable the construction of a genetic pulse generator. *Biotechnology and Bioengineering* 2019, *116* (5), 1139-1151. https://doi.org/10.1002/bit.26918

4. Zhang, H., Conn, A. R., & Scheinberg, K. (2010). A derivative-free algorithm for least-squares minimization. *SIAM Journal on Optimization*, *20*(6), 3555–3576. https://doi.org/10.1137/09075531x