An enhanced transcription factor repressilator that buffers stochasticity and entrains to an erratic external circadian signal

Steven A. Frank[1]

How do cellular regulatory networks solve the challenges of life? This article presents computer software to study that question, focusing on how transcription factor networks transform internal and external inputs into cellular response outputs. The example challenge concerns maintaining a circadian rhythm of molecular concentrations. The system must buffer intrinsic stochastic fluctuations in molecular concentrations and entrain to an external circadian signal that appears and disappears randomly. The software optimizes a stochastic differential equation of transcription factor protein dynamics and the associated mRNAs that produce those transcription factors. The cellular network takes as inputs the concentrations of the transcription factors and produces as outputs the transcription rates of the mRNAs that make the transcription factors. An artificial neural network encodes the cellular input-output function, allowing efficient search for solutions to the complex stochastic challenge. Several good solutions are discovered. The solutions differ significantly from each other, showing that overparameterized cellular networks may solve a given challenge in a variety of ways. The article concludes by drawing an analogy between overparameterized cellular networks and the dense and deeply connected overparameterized artificial neural networks that have succeeded so well in deep learning. Understanding how overparameterized networks solve challenges may provide insight into the evolutionary design of cellular regulation.

Keywords: genetic regulatory networks; transcription factors; optimization; differential equation models; artificial neural networks; automatic differentiation; Julia programming language

Introduction

Transcription factor (TF) networks control cellular response. The concentrations of the TF proteins feed into the TF network, which then stimulates or represses the expression of various genes. The TF network computes an input-output function: TF concentrations in, gene expression out.

A key puzzle of cellular design concerns how TF input-output functions solve the challenges of life. The puzzle poses two questions. First, what functional relation between inputs and outputs solves the biological challenge? Second, how is the functional relation encoded molecularly?

This article focuses on the first problem, the functional solution to the biological challenge. Following a prior article, the challenge concerns maintaining a circadian rhythm of molecular concentrations.[1] The rhythm must buffer stochastic fluctuations in molecular concentrations.

An internal rhythm perturbed by stochasticity cannot retain perfect periodicity without an external signal. In this case, an external circadian signal appears and disappears randomly. That external signal provides an occasional opportunity for entrainment. However, the erratic external signal does not provide sufficient information for the cell to achieve good periodicity by simply mirroring the signal. Instead, the TF network must entrain to the external signal when available and otherwise buffer internal molecular stochasticity to maintain its internal rhythm in the absence of external information.

I use an artificial neural network to search for a TF input-output function that solves this circadian challenge. I embed the TF network in a stochastic differential equation that tracks the concentrations of

[1]Department of Ecology and Evolutionary Biology, University of California, Irvine, CA 92697–2525, USA

web: https://stevefrank.org

mRNAs transcribed by gene expression and the associated TF proteins translated from the mRNAs. The TF network takes the TF protein concentrations as inputs and produces as outputs the mRNA transcription rates for the various TF genes.

I studied the same circadian challenge in the prior article.[1] That prior article tried to solve the functional and mechanistic parts of the puzzle simultaneously. Mechanistically, I encoded the TF input-output function using the currently favored thermodynamic model for TF binding to DNA gene promoter regions and the consequences of that binding for mRNA transcription rates.[2,3]

Among many computer runs in that prior study, I found only one solution that provided reasonably good circadian tracking and entrainment based on explicit thermodynamic parameters for the TF network. That solution was very difficult to find computationally and was sensitive to changes in the thermodynamic parameters. Additionally, although the thermodynamic model I used is the most widely favored description for the molecular mechanism, it is very unlikely for that model to be an accurate and complete description of the actual molecular mechanism.

This article gains by focusing solely on the functional challenge of mapping inputs to outputs without concern for the mechanism. In particular, I use an artificial neural network to encode the functional mapping instead of the TF thermodynamic model. Once we have an understanding of the kinds of functional relations that solve the challenge, we can then search for candidate molecular mechanisms that could potentially encode those functional relations.

For example, given a functional solution encoded by a neural network, one could then fit the classic thermodynamic model to that functional solution. That fitting of the molecular mechanism to the functional solution would be useful when attempting to engineer the mechanism in actual cells and measure their molecular dynamics.

Prior modeling approaches searched for TF networks to achieve particular dynamics.[4–8] The approach in this article has several technical advantages. Neural networks are often ideal function approximators.[9] Computationally, they scale very well to higher dimensions, can easily use automatic differentiation for optimization, and can be embedded within stochastic differential equations while maintaining the great computational advantages of automatic differentiation.[10–12]

Without these technical advantages, computational optimization of TF networks is difficult and has not previously been studied in a widely applicable way. The computer code provided with this article can easily be adapted to study other biological challenges. Additional studies will eventually give a sense of the kinds of input-output mappings required of TF networks to solve the demands of life.

As future studies accumulate, we may find that the evolutionary success of TF networks and the computational success of neural networks arise from a common foundation. Both networks may induce essentially the same geometric manifold of evolutionary or learning dynamics on which improving performance plays out.[13]

Materials and methods

The prior article described the methods.[1] This section updates the methods with the artificial neural network that replaces the thermodynamic model for the TF input-output response function.

The deterministic component of the system dynamics is given by the temporal derivatives for numbers of mRNA molecules, x, and the TFs produced by those mRNAs, y, as

$$\begin{aligned} \dot{x}_i &= m_i f_i(\mathbf{y}) - \delta_i x_i \\ \dot{y}_i &= s_i x_i - \gamma_i y_i, \end{aligned} \tag{1}$$

for m_i as the maximum mRNA production rate, δ_i as the mRNA decay rate, s_i as the TF production rate per mRNA, and γ_i as the decay rate of the $i = 1, \ldots, n$ TFs.[14]

The function f_i transforms the numbers of TFs in the vector $\mathbf{y}$ into the production level of the ith mRNA, varying between 0 for complete repression and 1 for maximum production. In this article, each f_i is a separate neural network that takes n inputs as $\log(\mathbf{y})$.

The neural network architecture follows a standard general form, with the following details. The first layer of the network has $5n$ output nodes. Each of those nodes sums an affine transformation of each

git · master@biorxiv_1.0-0::1b91ac0-2022-10-10 (2022-10-10 18:37Z) · safrank

input, z, of the form $\alpha + \beta z$, in which each of those $5n^2$ transformations has its own α and β parameters. The value of each output node for this first layer is transformed by the mish activation function[15]

$$\mathrm{mish}(z) = z \tanh\big(\log(1 + e^z)\big).$$

The $5n$ outputs from the first layer form the inputs for another neural network layer, which leads to the final single-valued output that controls the mRNA transcription rate for the associated TF gene. That output arises by first summing affine transformations from all $5n$ inputs and then applying the sigmoid function to that sum, in which the sigmoid function outputs values between 0 and 1, as

$$\mathrm{sigmoid}(z) = \frac{1}{1 + e^{-z}}.$$

The same architecture is repeated for each of the n TFs, with the single-valued output of each network influencing the transcription rate of the associated TF gene.

I added stochastic perturbations to the molecular abundances in eqn 1, making the system a set of stochastic differential equations. Stochastic fluctuations for a molecule with abundance z follow $\sqrt{z}\mathrm{d}W$ in which W is a standard normal variate with mean 0 and standard deviation 1, thus $\sqrt{z}W$ has a standard deviation of $\sqrt{z}$.

The normalized magnitude of the fluctuations is the ratio of the standard deviation relative to the current abundance, $\sqrt{z}/z = 1/\sqrt{z}$. As z drops, the normalized fluctuations increase. To prevent fluctuations from becoming too large relative to the abundance, which can cause negative abundance values in the numerical analysis, for $z \leq 16$, I replaced $\sqrt{z}$ with $z/4$.

The prior article provides further details on particular calculations and methods.[1] This article emphasizes the much better optimization procedure achieved by using neural networks for the TF network.

Results

Overview

Computational optimization finds parameters for eqn 1 that influence the stochastic dynamics of mRNAs and TF proteins. Different runs of the optimization procedure converged to different parameters. Figure 1 illustrates the results for one run.

In that figure, panel (a) shows the target circadian pattern in gold and the system's circadian pattern in green. The optimization goal is to minimize the total squared distance between those two curves.[1] The blue curve in that panel shows the molecular abundance of the TF 1 protein. The green curve arises by transforming the blue curve by a Hill function, as explained in the figure caption. The x-axis shows time measured in days.

The optimization begins by attempting to fit the parameters to the circadian pattern for part of the first day. Then, the algorithm slowly increases the temporal range in a sequence of steps, for each step allowing the fit to adjust to the additional time range. The final time range for fitting the parameters in these runs was four days. Once the fit is obtained, I studied the match between the target circadian pattern and the system dynamics over a time period greater than the four days used for fitting.

I wrote the code to emphasize the chance of finding a good fit rather than to maximize the speed of the fitting process. A single optimization run takes about ten days when using 12 computational threads on a 2022 Apple Studio Ultra M1 computer. I analyzed eight full runs for this study, which was sufficient to find several good fits and to illustrate a range of outcomes.

Dynamics

In Fig. 1, panel (b) shows the dynamics of the TF 2 protein. The blue curve is the abundance of that TF. The gold curve shows the external circadian entrainment signal, which may be interpreted as the external light intensity experienced by the cell. The entrainment signal comes and goes randomly. For these runs, the signal begins in the off state and then switches on and off randomly with a waiting time drawn from an exponential distribution with a mean of two days. The TF 2 protein increases in abundance in response to the light intensity.[1]

Overall, panels (a-d) show the dynamics of the four TF proteins. For each TF protein, the matching panel to the right shows the dynamics of the mRNAs that encode the TFs.

git · master@biorxiv_1.0-0::1b91ac0-2022-10-10 (2022-10-10 18:37Z) · safrank

The gold curve in panel (i) shows the target circadian pattern over 20 days. Each of the 20 green curves shows the cellular circadian state for a run of the stochastic dynamics. As in panel (a), cellular state is calculated as a Hill transformation of the abundance of TF 1. For each run in panel (i), the external entrainment signal comes on and off randomly with a mean waiting time of two days, as in panel (b).

Panel (j) repeats the setup in panel (i), except that the external signal comes on and off randomly with a mean waiting time of 1000 days. Because the signal starts in the off state, almost always the signal never comes on. The green curves therefore show the ability of the system to maintain its circadian rhythm in the absence of any external signal. Inevitably, the stochastic fluctuations of molecular concentrations cause the cellular rhythm to diverge from the target pattern.

Figures 2 and 3 show the same data for two other optimization runs. The following subsections compare the different optimization runs. For all runs, the settings, optimized parameters, output values, and code for making the figures are included in the online repositories listed in the Data Availability statement at the end of the article.

Deviation from the target pattern

Figure 4a shows the deviations of each optimized system from the target circadian pattern. For a single realization, such as a green curve in panel (i) or (j) of the previous figures, I calculated the deviations as follows. The entry into external daytime occurs at the vertical dotted line in each daily interval, and the gold curve crosses the horizontal dotted line at 10^3. For a cell, the entry into its internal daytime state occurs as the green curve crosses above the horizontal dotted line at abundance 10^3.

For each day, the cellular deviation is the horizontal distance along the dotted line between the green curve and the gold curve. When the green curve is to the left of the gold curve, the deviation is negative. When it is to the right, the deviation is positive. I transformed the deviation into hours, the daily deviation amount divided by 24. For a realization of the dynamics over d days, there are d deviation measures for that realization.

For realizations that ran over d days, I repeated the measurements over 1000 samples for a total of $1000d$ deviation measures. I then calculated percentiles of the deviations. Looking at the far left of Fig. 4a, the two vertical lines and the circle show those percentiles. The lower line ranges from the 5th to the 25th percentile. The circle marks the 50th percentile. The upper line ranges from the 75th to the 95th percentile.

Thus, each vertical set summarizes the distribution of deviations for a sample of 1000 realizations over d days. In the figure, the vertical summary for each distribution occurs within a set of three such distributions. For each set, the left, middle, and right distributions correspond to $d = 10, 20, 30$, respectively, showing the distributions of deviations when realizations run over different numbers of days.

The eight labels below the sets in Fig. 4a provide the names for the optimization runs. I chose for further analysis the two sets on the left and the set on the right. Those three optimization runs had the narrowest distributions for the deviations, in which smaller deviations correspond to better tracking of the target circadian pattern.

The leftmost run, sde–4_1_t4, corresponds to the dynamics in Fig. 1 and to the more detailed summary of deviations in Fig. 4b. The deviation details in Fig. 4b show the distributions induced by different random switching times for the external entrainment signal. Each set of three distributions again corresponds to runs with $d = 10, 20, 30$ days. The different sets correspond to mean waiting times for random exponential switching of 2, 4, 8, 16, and 1000 days.

The mean time of 2 days corresponds to the value used in during optimization and is the default in all other plots unless otherwise noted. The longer mean times illustrate how well the system can hold its internal circadian rhythm when receiving an external entrainment signal less reliably. When the mean is 1000 days, the system almost never receives an external signal. The bigger deviations in that case show the greater difficulty of maintaining a match to the target pattern without the opportunity for entrainment. Although the spreads are greater for larger d, the system nonetheless typically keeps a very good circadian pattern for 10 days and a reasonably good pattern for as along as 30 days.

Figure 4c corresponds to sde–4_2_t4, and panel (d)

git · master@biorxiv_1.0-0::1b91ac0-2022-10-10 (2022-10-10 18:37Z) · safrank

corresponds to sde–4_8_t4. The sets do not match exactly between panel (a), which has a mean waiting time of 2 days, and the leftmost sets in panels (b-d), which also have mean waiting times of 2 days. The small mismatches arise because the different panels used different samples of the stochastic trajectories to calculate the distributions.

TF network input-output functions

Figure 5 shows the TF network input-output relations associated with sde–4_1_t4. Each quadrant presents the output as the relative mRNA production rate for a TF gene. For example, the upper-left block describes the output for TF 1. In each plot, the output level varies from off, in dark orange, to maximally on, in dark purple.

The output level depends on four inputs, the abundances of the four TF proteins. The x and y axes of a plot correspond to the abundances of TF 1 and TF 2, respectively, on a $\log 10(1+z)$ scale for abundance z. The abundance of TF 3 rises across the rows of plots, from zero at the bottom of a quadrant to the maximum level at the top. The abundance of TF 4 rises across the columns, from zero at the left to maximal at the right.

The TF network logic can be read from the figure. Consider, for example, the TF 2 block in the upper right that shows the TF 2 mRNA output rate. An increase in TF 3 abundance, going up the rows, associates with repression of TF 2, because increasing orange means lower output. In the TF 3 block (lower left), an increase in TF 4, going right across columns, associates with repression of TF 3. In the TF 4 block (lower right), an increase in TF 2, going right along the x-axis of each plot, associates with repression of TF 4.

Thus, TF 3 represses TF 2, and TF 4 represses TF 3, and TF 2 represses TF 4. That cyclic repression defines the basic repressilator circuit, which can cause periodic oscillation of the associated TF proteins.[16] However, in this case, that core repressilator is embedded within a more complex network of modulating inputs. For example, TF 2 output is stimulated by TF 1 when both TF 2 and TF 3 abundances are low.

In this optimization problem, one major function of the TF network is the entrainment to an external circadian signal when available, illustrated by panels (a) and (b) of Fig. 1. Returning to the upper-left quadrant of Fig. 5, note that increasing TF 2 abundance (y-axis of plots) stimulates TF 1 expression. Thus, when an external light signal increases TF 2, that rise in TF 2 stimulates a rise in TF 1 expression. A decline TF 2 abundance during the night represses TF 1 expression.

Other inputs modulate the relation between TF 2 and TF 1. For example, high TF 3 and TF 4 prevent TF 1 expression, unless both TF 1 and TF 2 are very high. TF 1 stimulates its own expression on the way up and represses its own expression on the way down.

The second major function of the TF network is to maintain the circadian pattern by buffering the intrinsic biochemical stochasticity. That buffering arises from the various interactions of the TF network logic. One aspect may be the tendency to maintain both TF 1 and TF 2 cycling in synchrony with the target rhythm. Joint rhythmicity provides a partially redundant cycle that can be used to smooth out fluctuations in either TF protein.

Figures 6 and 7 show the TF network logic for sde–4_2_t4 and sde–4_8_t4, respectively. Looking at Fig. 6, one can see that the quantitative relations differ significantly from Fig. 5. For example, most effects of changing abundance in Fig. 5 are monotonic. By contrast, in Fig. 6, the upper-left and lower-right quadrants show reversals in the direction of change in outputs with changes in particular inputs.

Overall, Figs. 5–7 demonstrate that TF networks can produce good performance in different ways. Finding the necessary and sufficient characteristics of the TF networks remains an open problem.

Discussion

How do cellular regulatory networks solve the challenges of life? Ultimately, that is an empirical question that must be answered by observational and experimental study. However, it may be a difficult question to answer without some conceptual guidance.

Suppose, for example, that the network connections regulating cellular response are dense, deep, and complex. It might be that the network is overparameterized. In other words, the dimensionality of the molecular parameters controlling cellular response may be greater than the dimensionality of the

git · master@biorxiv_1.0-0::1b91ac0-2022-10-10 (2022-10-10 18:37Z) · safrank

challenges posed by the environment.

Computational deep learning models are typically highly overparameterized artificial neural networks.[17–19] It turns out that overparameterized networks are particularly good at solving complex problems. And, for such overparameterized networks, it is particularly difficult to figure out how the network actually succeeds at solving its challenge.[20]

One might find some network nodes that work together to achieve a particular component of the broader solution. For example, some nodes might together identify an environmental state or trigger a particular response that feeds into other parts of the network. But, overall, the relations between the individual network components and the response of the network to particular inputs typically remains opaque.

In the same way, it may be difficult to build up an understanding of cellular function by starting with small molecular subnetworks and then expanding analysis to increasingly broader subcomponents. If so, then understanding in theory how such networks might solve problems may provide some guidance.

For example, what are the variety of ways in which cells might buffer against stochastic perturbations to circadian periodicity? How might cells entrain their rhythm to erratic external signals? In overparameterized networks, there is rarely just one solution. Theory can help to identify the range of possible solutions, which could then guide where to look within the complexity of actual cellular regulation.

This article provided an early step on the path to developing theoretical approaches. By using artificial neural networks for the transcription factor network, the computational models can explore the variety of potential solutions to life's challenges. This article contributes by showing how relatively simple computer code can discover potential solutions.

Acknowledgments

The Donald Bren Foundation, National Science Foundation grant DEB-1939423, and DoD grant W911NF2010227 support my research.

Data availability statement

All computer code, parameters and output used to generate the figures are available on Zenodo.[21]

References

1. Frank, S. A. 2022. Optimization of transcription factor genetic circuits. *Biology* 11, 1294.
2. Bintu, L. et al. 2005. Transcriptional regulation by the numbers: models. *Current Opinion in Genetics & Development* 15, 116–124.
3. Bintu, L. et al. 2005. Transcriptional regulation by the numbers: applications. *Current Opinion in Genetics & Development* 15, 125–135.
4. Bolouri, H. & Davidson, E. H. 2002. Modeling transcriptional regulatory networks. *BioEssays* 24, 1118–1129.
5. Kauffman, S., Peterson, C., Samuelsson, B. & Troein, C. 2003. Random Boolean network models and the yeast transcriptional network. *Proceedings of the National Academy of Sciences USA* 100, 14796–14799.
6. De Jong, H. et al. 2004. Qualitative simulation of genetic regulatory networks using piecewise-linear models. *Bulletin of Mathematical Biology* 66, 301–340.
7. Mishra, B. et al. 2018. Dynamic modeling of transcriptional gene regulatory network uncovers distinct pathways during the onset of Arabidopsis leaf senescence. *NPJ Systems Biology and Applications* 4, 1–4.
8. Patel, N. & Bush, W. S. 2021. Modeling transcriptional regulation using gene regulatory networks based on multi-omics data sources. *BMC Bioinformatics* 22, 1–19.
9. Goodfellow, I., Bengio, Y. & Courville, A. 2016. *Deep Learning*. Cambridge, MA: MIT Press.
10. Baydin, A. G., Pearlmutter, B. A., Radul, A. A. & Siskind, J. M. 2018. Automatic differentiation in machine learning: a survey. *Journal of Machine Learning Research* 18, 1–43.
11. Rackauckas, C. et al. 2020. Universal differential equations for scientific machine learning. *arXiv:2001.04385*.
12. Frank, S. A. 2022. Automatic differentiation and the optimization of differential equation mod-

git · master@biorxiv_1.0-0::1b91ac0-2022-10-10 (2022-10-10 18:37Z) · safrank

els in biology. *arXiv.2207.04487*.

13. Frank, S. A. 2017. Puzzles in modern biology. V. Why are genomes overwired? *F1000Research* 6, 924.
14. Marbach, D. et al. 2010. Revealing strengths and weaknesses of methods for gene network inference. *Proceedings of National Academy of Sciences USA* 107, 6286–6291.
15. Misra, D. 2019. Mish: a self regularized non-monotonic activation function. *arXiv* 1908.08681.
16. Elowitz, M. B. & Leibler, S. 2000. A synthetic oscillatory network of transcriptional regulators. *Nature* 403, 335–338.
17. Radhakrishnan, A., Belkin, M. & Uhler, C. 2020. Overparameterized neural networks implement associative memory. *Proceedings of the National Academy of Sciences USA* 117, 27162–27170.
18. Poggio, T., Banburski, A. & Liao, Q. 2020. Theoretical issues in deep networks. *Proceedings of the National Academy of Sciences USA* 117, 30039–30045.
19. Bartlett, P. L., Long, P. M., Lugosi, G. & Tsigler, A. 2020. Benign overfitting in linear regression. *Proceedings of the National Academy of Sciences USA* 117, 30063–30070.
20. Amey, J. L., Keeley, J., Choudhury, T. & Kuprov, I. 2021. Neural network interpretation using descrambler groups. *Proceedings of the National Academy of Sciences USA* 118, e2016917118.
21. Frank, S. A. *An enhanced transcription factor repressilator that buffers stochasticity and entrains to an erratic external circadian signal: Julia software code*. 2022. DOI: 10.5281/zenodo.7178508.

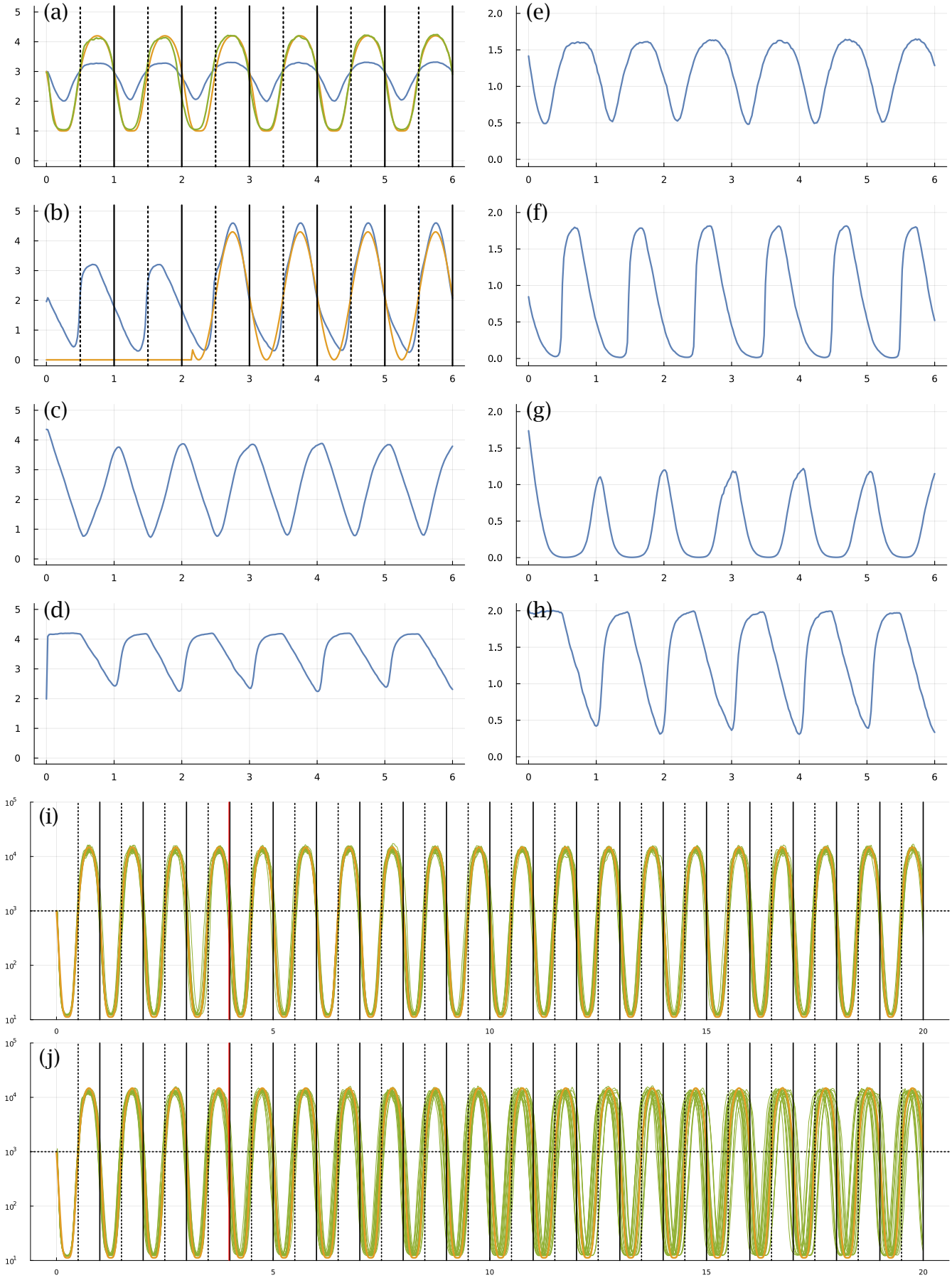

Figure 1: Circadian dynamics of the TF network from run sde-4_1_t4. Caption continues on next page.

git · master@biorxiv_1.0-0::1b91ac0-2022-10-10 (2022-10-10 18:37Z) · safrank

Figure 1: Panels (a-d) show the stochastic dynamics of the TF proteins. The dynamics of the matching mRNAs that produce the proteins are shown in panels (e-h). The time period is six days, along the x-axis of each of these panels. The vertical lines in (a,b) show entry into daylight (dotted) and nighttime (solid). The y-axis is $\log 10(1 + y)$ for number of molecules per cell, y. In (a), the optimization procedure attempts to match the number of TF 1 molecules (blue curve) to a circadian rhythm by minimizing a loss value. To calculate the loss value, begin with the number of TF 1 molecules, y, transformed by a Hill function, $\tilde{y} = y^2/(1000^2 + y^2)$, to yield the green curve, which traces $1 + 4\tilde{y}$. The gold curve traces the target circadian pattern. The loss value to be minimized is the sum of the squared deviations between the gold and green curves at 50 equally spaced time points per day. The number of TF 2 proteins in (b) is influenced by the internal cellular dynamics and is also increased in response to an external daylight signal that switches on and off randomly.[1] It is initially off. The average waiting time for a random switch in the presence or absence of the signal is $w = 2$, measured in days. In this example, the signal turns on during the night of the third day and stays on for the remaining days shown. Because the switching is random, daylight can be present or absent for several days in a row, or it can switch on and off several times in one day. In panel (a), stochastic molecular perturbations push the cellular rhythm (green curve) behind the actual circadian pattern (gold curve) during the first few days in this particular realization of the stochastic dynamics. When the daylight signal appears in day 3, the system entrains to the external signal, closely matching the target circadian pattern for the remaining days shown. Panels (i,j) are described in the text. See Frank[1] for additional details.

git · master@biorxiv_1.0-0::1b91ac0-2022-10-10 (2022-10-10 18:37Z) · safrank

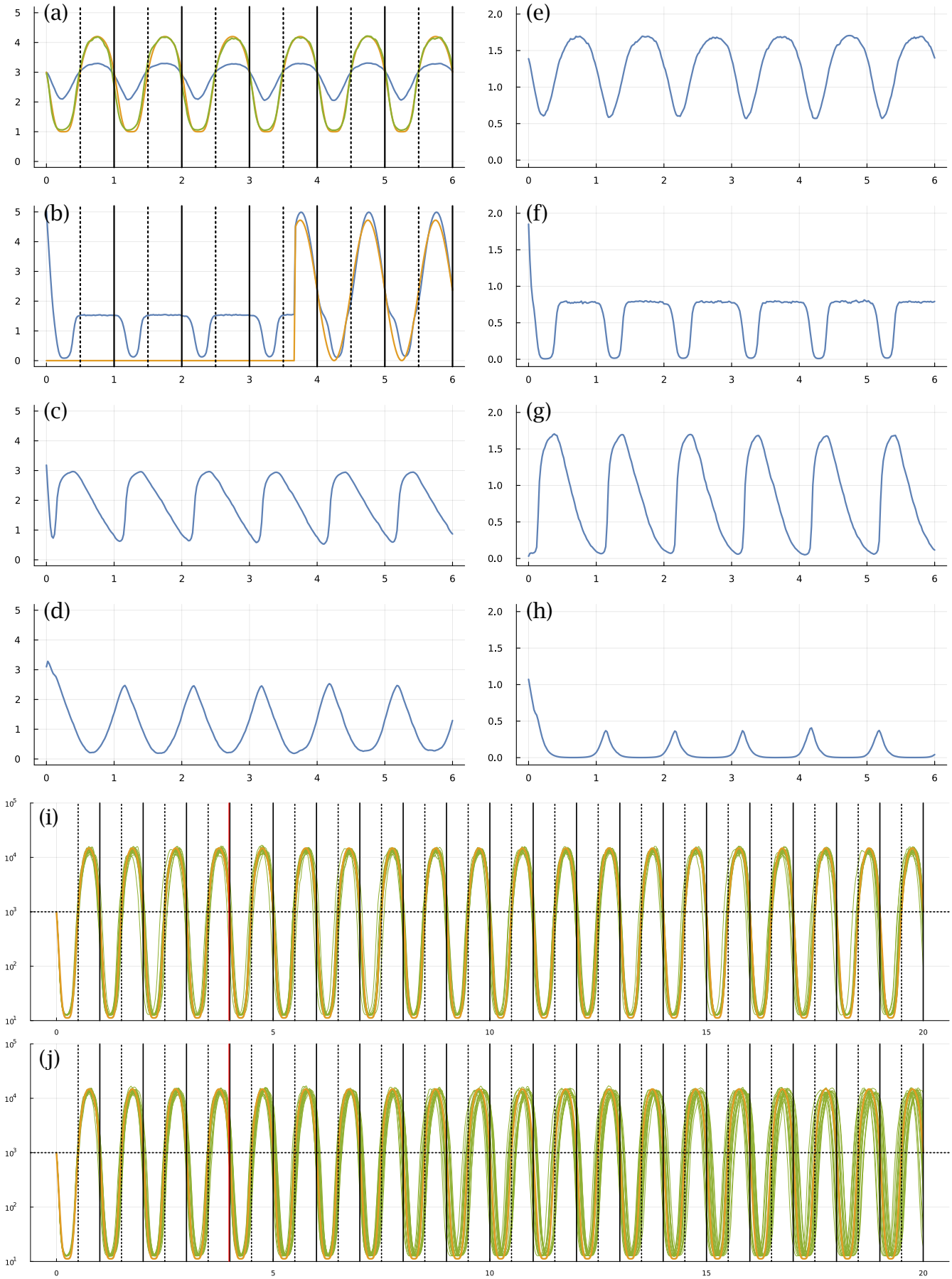

Figure 2: Circadian dynamics for run sde-4_2_t4.

git · master@biorxiv_1.0-0::1b91ac0-2022-10-10 (2022-10-10 18:37Z) · safrank

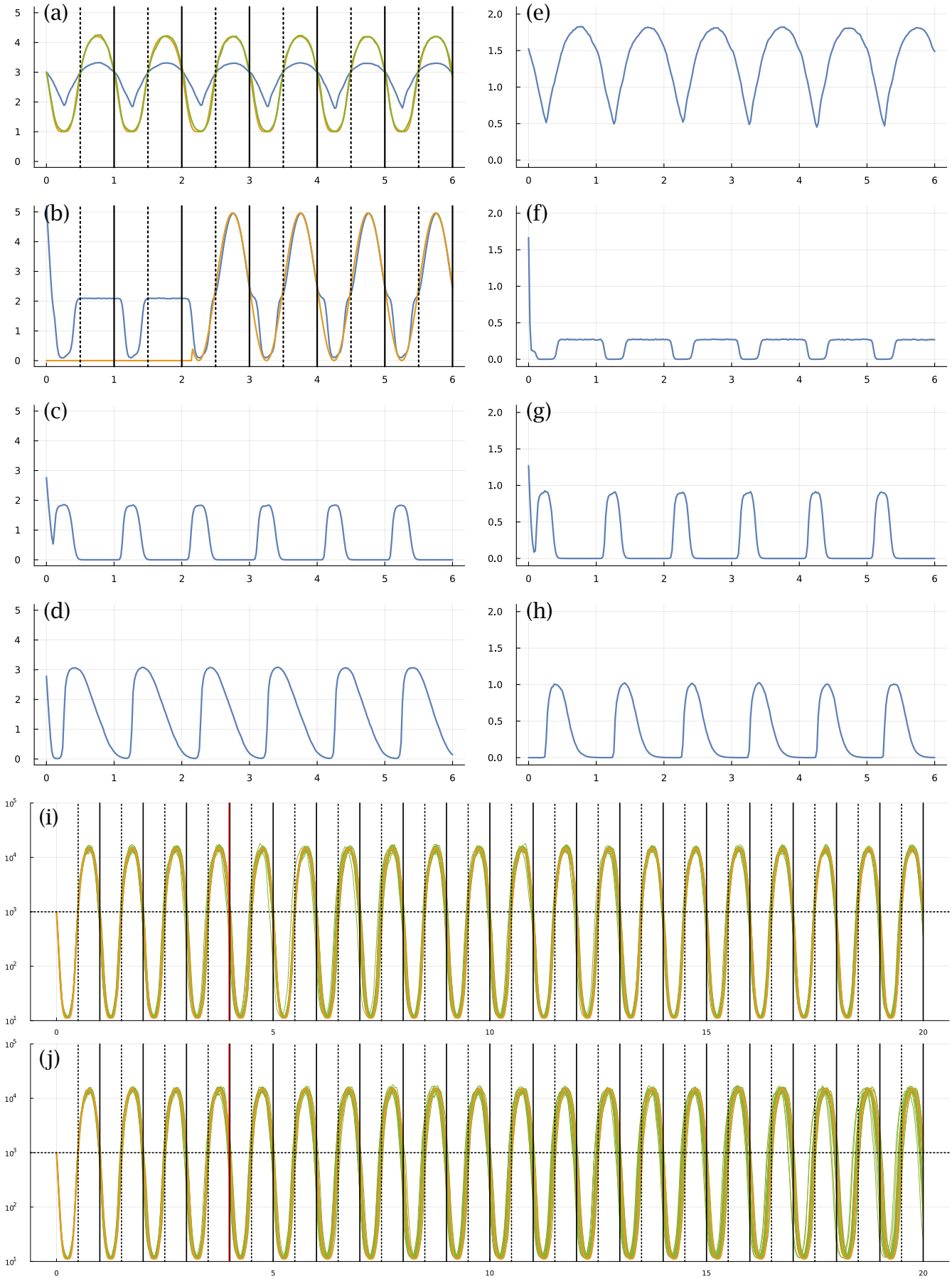

Figure 3: Circadian dynamics for run sde-4_8_t4.

git · master@biorxiv_1.0-0::1b91ac0-2022-10-10 (2022-10-10 18:37Z) · safrank

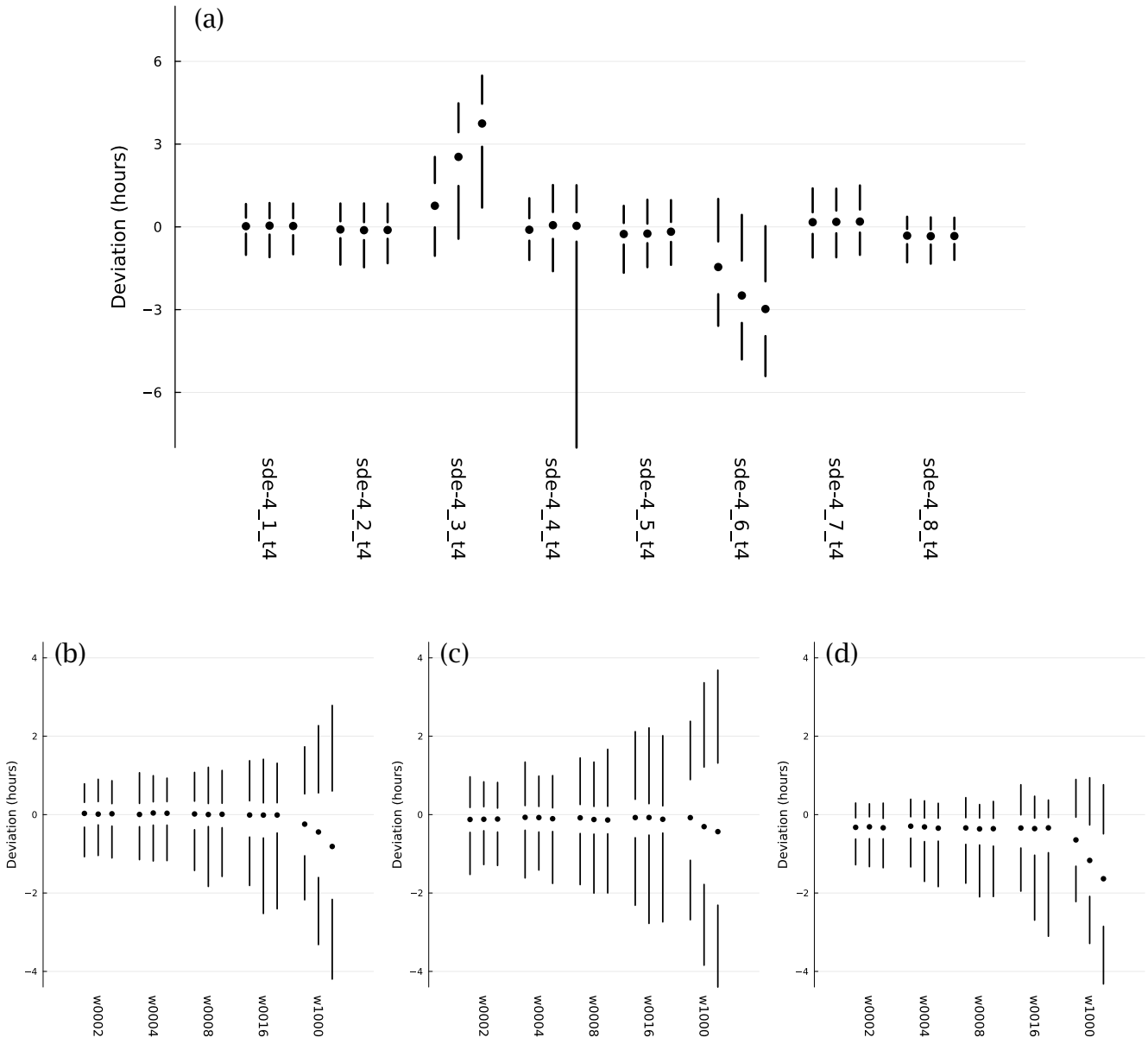

Figure 4: Probability distributions for deviations between the actual circadian pattern of the TF 1 protein and the target circadian pattern. Each set of two vertical lines and a circle describe a distribution, with the 5th to 25th percentiles of the distribution along the lower line, the 50th percentile at the circle, and the 75th to 95th percentiles along the upper line. (a) The overall patterns for the eight independent optimization runs, with the label for each run along the x-axis. (b) The distributions for sde–4_1_t4, for different values of w, the average waiting time for switching of the external circadian light signal. (c) The distributions for sde–4_2_t4. (d) The distributions for sde–4_8_t4. Further details in the text.

git · master@biorxiv_1.0-0::1b91ac0-2022-10-10 (2022-10-10 18:37Z) · safrank

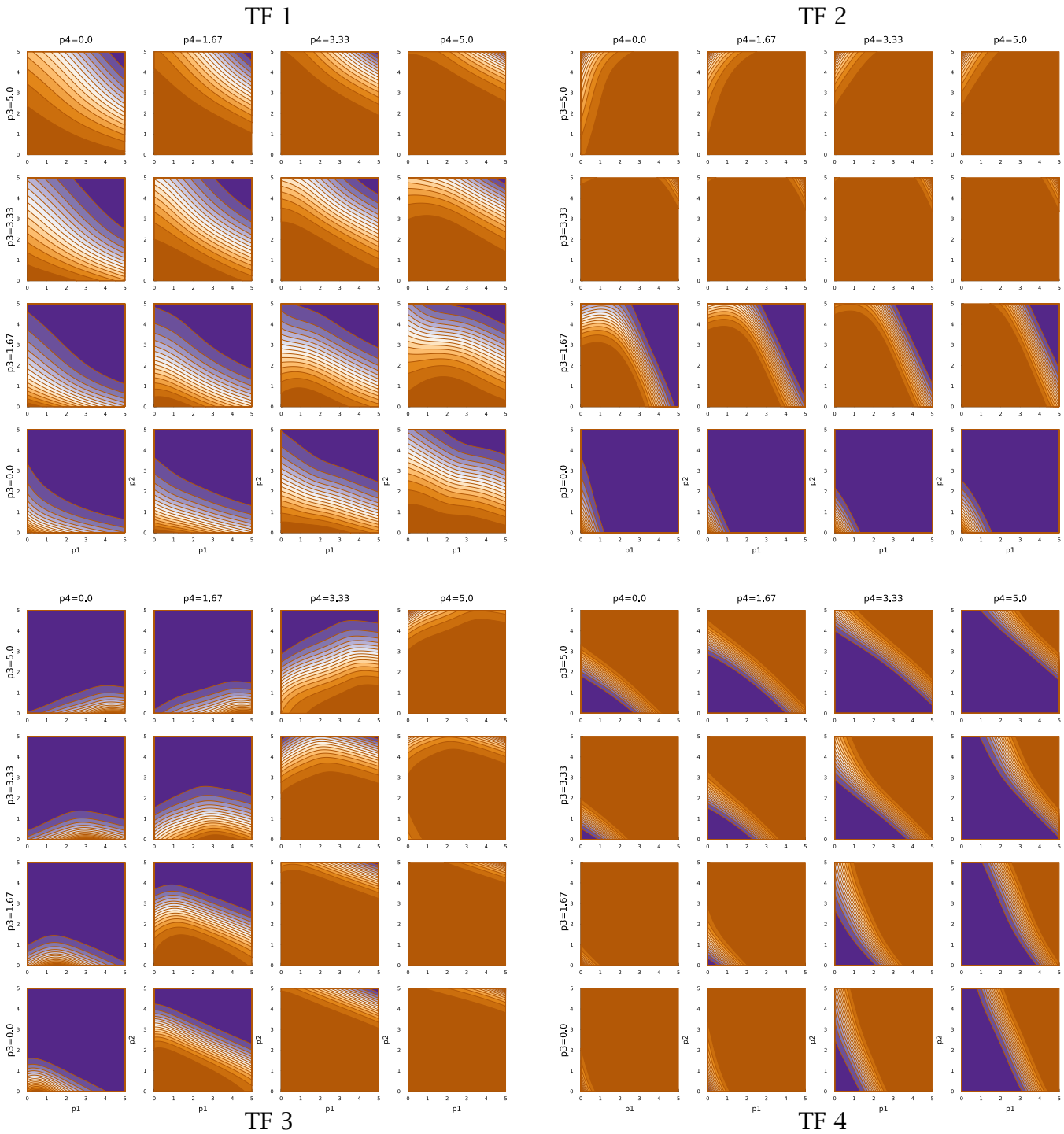

Figure 5: TF network input-output function for run sde-4_1_t4.

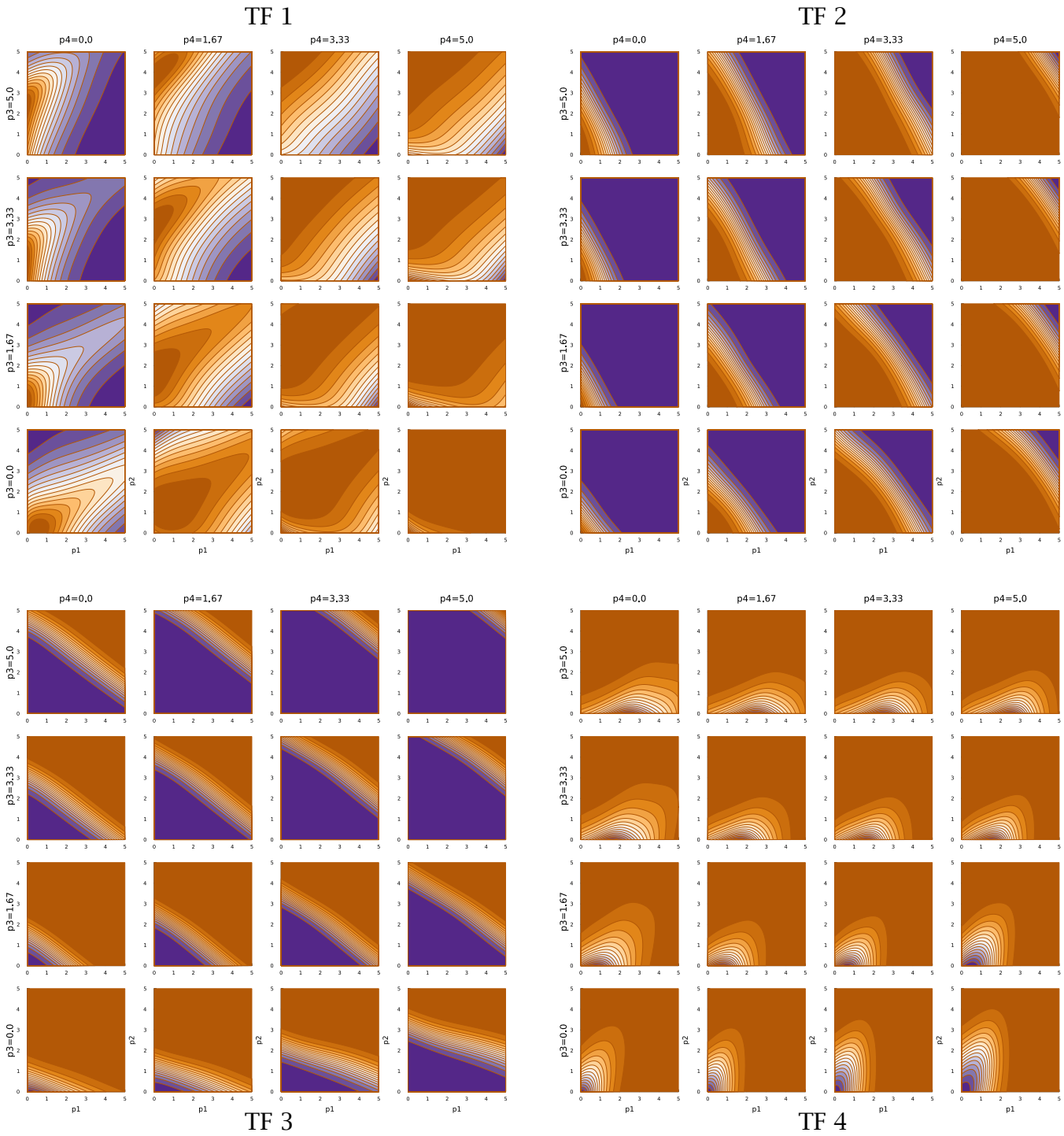

Figure 6: TF network input-output function for run sde-4_2_t4.

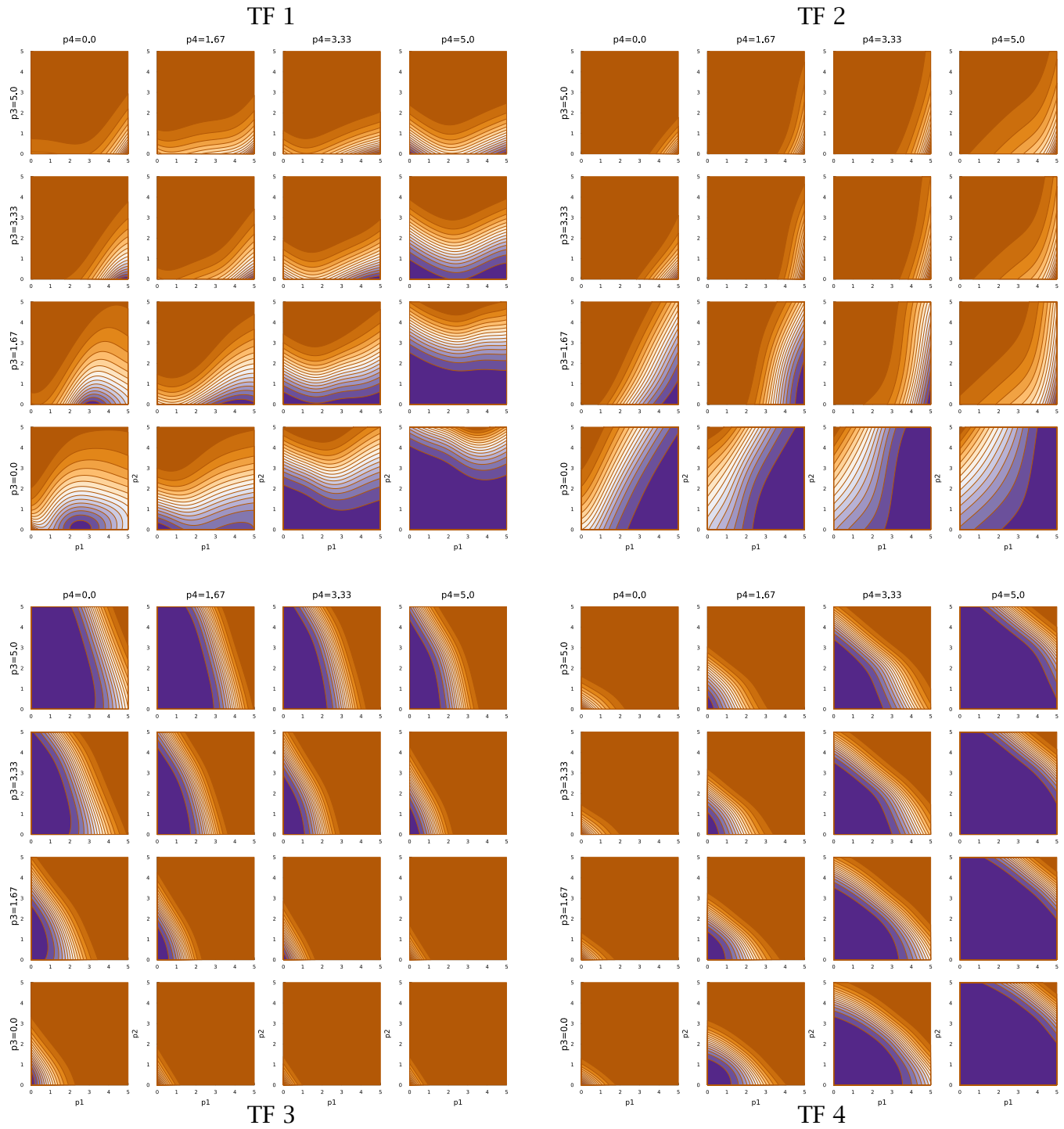

Figure 7: TF network input-output function for run sde-4_8_t4.

git · master@biorxiv_1.0-0::1b91ac0-2022-10-10 (2022-10-10 18:37Z) · safrank