Control strategies for mitigating the effect for external perturbations on gene regulatory networks

Foo, M., Gherman, I., Denby, K. & Bates, D.

DOI 10.1016/j.ifacol.2017.08.2237
ISSN 1474-6670

Publisher: Elsevier

Control Strategies for Mitigating the Effect of External Perturbations on Gene Regulatory Networks

Mathias Foo * **Iulia Gherman** * **Katherine J. Denby** ** **Declan G. Bates** *

* *Warwick Integrative Synthetic Biology Centre, School of Engineering, University of Warwick, Coventry, CV4 7AL, UK. (e-mail: M.Foo@warwick.ac.uk, I.Gherman@warwick.ac.uk, D.Bates@warwick.ac.uk).*
** *Department of Biology, University of York, York YO10 5DD, UK. (e-mail: katherine.denby@york.ac.uk)*

Abstract: External perturbations affecting gene regulatory networks, such as pathogen/virus attacks, can lead to adverse effects on the phenotype of the biological system. In this paper, we propose a systematic approach to mitigate the effect of such perturbations that can be implemented using the tools of synthetic biology. We use system identification techniques to build accurate models of an example gene regulatory network from time-series data, and proceed to identify the kernel architecture of the network, which is defined as the minimal set of interactions needed to reproduce the wild type temporal behaviour. The kernel architecture reveals four key pathways in the network which allow us to investigate a number of different mitigation strategies in the event of external perturbations. We show that while network reoptimisation can reduce the impact of perturbations, combining network rewiring with a synthetic feedback control loop allows the effect of the perturbation to be completely eliminated. The proposed approach highlights the potential of combining feedback control theory with synthetic biology for developing more resilient biological systems.

Keywords: gene regulatory networks, feedback control, network rewiring, mitigation control, synthetic biology, system identification, kernel architecture

1. INTRODUCTION

In any complex network, e.g. the world wide web, power grids, social networks, gene regulatory networks, etc, the presence of external perturbations to the network can potentially result in adverse effects to the entire network operation. These undesirable consequences include the spreading of computer viruses through the internet, major power supply breakdowns to residential and industrial areas, rapid spread of rumours and misinformation through social networks, and the loss of vital functions in biological organisms. As a result, the development of mitigation strategies that can minimise the effect of external perturbations on network functionality is now the subject of intensive research.

Conventionally, work on this problem from the field of network science has looked at ways to mitigate such perturbation effects by focussing on various network related concepts, such as 'hubs', 'components', 'degree of centralities' and so on (see e.g. Newman (2010); Ventresca and Aleman (2013)). Recently, however, the use of strategies based on feedback control theory for mitigating the effect of perturbations on large-scale networks has also started to garner attention (see e.g. Liu et al. (2011, 2013); Vinayagam et al. (2016); Liu and Barabasi (2016)). By treating perturbation as *disturbances* that require appropriate *control actions* to ensure the *output* stays at the desired *reference* value, one can reformulate the problem of perturbation mitigation as a disturbance rejection problem from the perspective of control theory. In view of this, there have been considerable efforts, including those studies mentioned above, dedicated to using tools from control theory to analyse the controllability and stability of complex networks, in order to develop control system design rules to implement appropriate mitigation strategies.

In this paper, we present a systematic approach for perturbation mitigation in the context of gene regulatory networks, starting from the modelling of the dynamics of the network using system identification techniques, to obtaining the kernel architecture of the network, which is then used for further analysis in developing strategies for network mitigation through the combination of synthetic feedback control loops and network rewiring.

The paper is organised the following manner. In Section 2, the description of the example gene regulatory network used to illustrate our approach is presented. The methodologies for system identification and for obtaining the kernel architecture of the network are described in Section 3. In Section 4, we present a number of mitigation strategies and evaluate their capability to robustly reduce the effect of external perturbations on the phenotypic response of the network. We end the paper with some conclusions in Section 5.

2. DREAM GENE REGULATORY NETWORK

The DREAM *in silico* gene regulatory network challenge was recently established to serve as a benchmark for evaluating different proposed methods for inferring the structure of

gene networks from experimental data (Marbach et al. (2009); Stolovitzky et al. (2007, 2009)). In the challenge, temporal data for each gene (node) in an interconnecting network are typically provided, and based on this data, the aim is to decipher the underlying gene regulatory network to obtain information such as the interconnecting edges, the direction of the links connecting the nodes and so on. These gene regulatory networks are part of the actual transcriptional networks in *E. coli* and *S. cerevisiae*, and thus they are representative of real biological systems.

In this paper, we choose the DREAM4 Size 10 data set (hereafter the term DREAM is used to denote this network), which consists of mRNA data of a network composed of 10 interconnecting genes. Since this dataset does not include separate protein data, in the following we assume that the dynamics of the protein is similar to the dynamics of the mRNA and that the mRNA is linearly translated to protein. With these two assumptions, at steady state, we can lump the protein dynamics with the transcription rate of the mRNA, resulting in a complete network which can be described using only mRNA levels. For this DREAM data set, the information regarding which nodes are connected to which is already provided and the depiction of these interactions is shown in Figure 1(A).

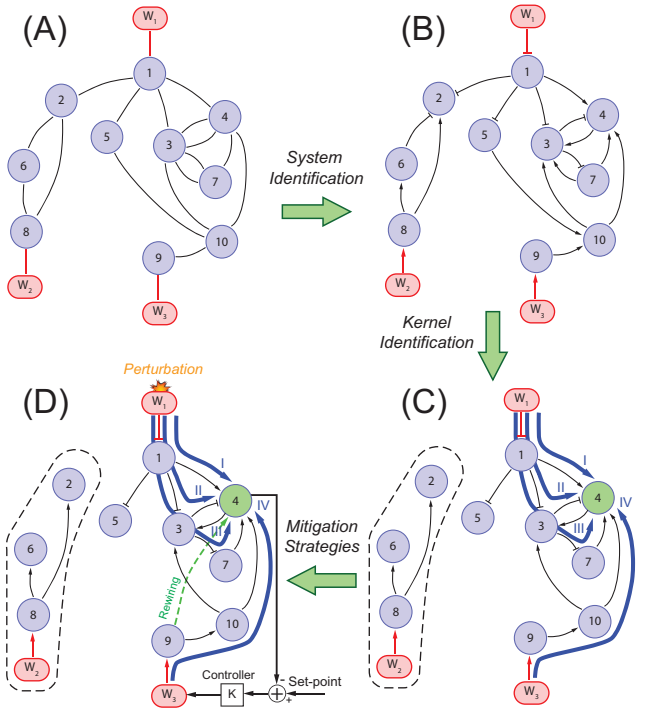

Fig. 1: (A) DREAM gene regulatory network. Purple circles represent genes and red rectangles represent external inputs. (B) Using system identification, the types of regulation in the network are identified. Arrow head indicates activation and Bar head indicates repression. (C) The kernal architecture of the gene regulatory network reveals four main pathways. (D) Control design configuration for mitigation when perturbation occurs.

3. MODEL IDENTIFICATION

3.1 Identification of a linear grey box gene regulatory network model

While the interconnection between different genes in the DREAM gene regulatory network has been established (see Figure 1(A)), the type of regulation between interconnecting genes is not specified, i.e. the regulation could either be an activation or repression. Here, we use system identification techniques (see e.g. Ljung (1999)) to determine the regulation types between interconnecting genes.

The use of system identification in deciphering gene regulatory network has been considered previously in (Gardner et al. (2003); di Bernardo et al. (2005); Bansal et al. (2007)). In those studies, a linear black box network model is considered, and based on the the available gene expression data, the interconnecting genes, the direction of the interconnection, and the types of regulation were identified. Several previous studies have confirmed that the underlying dynamics of gene regulatory network can be accurately identified using linear models (see e.g. Dalchau et al. (2011); Herrero et al. (2012); Foo et al. (2013)).

In this study, given that we have prior knowledge about the interconnections between the genes, we consider a linear *grey box* model and focus on the identification of the types of regulation in this network. Following the standard procedure of system identification, one data set of the DREAM network is used for model identification and another data set is used for model validation. These two data sets correspond to two different scenarios on how perturbations could enter the network.

The linear model to describe the dynamics of the DREAM network is given by

$$\begin{aligned}
\frac{dN_1}{dt} &= \theta_1 W_1 + \theta_2 N_1 + \theta_3 \\
\frac{dN_2}{dt} &= \theta_4 N_1 + \theta_5 N_6 + \theta_6 N_8 + \theta_7 N_2 \\
\frac{dN_3}{dt} &= \theta_8 N_1 + \theta_9 N_4 + \theta_{10} N_7 + \theta_{11} N_{10} + \theta_{12} N_3 \\
\frac{dN_4}{dt} &= \theta_{13} N_1 + \theta_{14} N_3 + \theta_{15} N_7 + \theta_{16} N_{10} + \theta_{17} N_4 \\
\frac{dN_5}{dt} &= \theta_{18} N_1 + \theta_{19} N_5 + \theta_{20} \\
\frac{dN_6}{dt} &= \theta_{21} N_8 + \theta_{22} N_6 \\
\frac{dN_7}{dt} &= \theta_{23} N_3 + \theta_{24} N_7 + \theta_{25} \\
\frac{dN_8}{dt} &= \theta_{26} W_2 + \theta_{27} N_8 \\
\frac{dN_9}{dt} &= \theta_{28} W_3 + \theta_{29} N_9 \\
\frac{dN_{10}}{dt} &= \theta_{30} N_5 + \theta_{31} N_9 + \theta_{32} N_{10}
\end{aligned} \tag{1}$$

where W_1 to W_3 are input signals. θ are unknown parameters, which are to be estimated using prediction error method with a quadratic criterion, i.e.

$$\hat{\theta} = \text{argmin}_\theta \frac{1}{N_T} \sum_{i}^{10} \sum_{t=1}^{N_T} [N_i(t) - \hat{N}_i(t,\theta)]^2 \tag{2}$$

where N_T is the length of data, $\hat{N}_i$ is the simulated data from the model and N_i is the real data. The estimated model parameters for the linear model are given in Table 1.

From these estimated parameters, we are able to determine the types of regulation where the activation and repression interac-

Gene	Values
N_1	θ_1 = -0.2467, θ_2 = -0.3913, θ_3 = 0.2732
N_2	θ_4 = -0.0032, θ_5 = -0.0863, θ_6 = 0.3166, θ_7 = -1.8488
N_3	θ_8 = -0.3842, θ_9 = 0.8466, θ_{10} = 0.0165, θ_{11} = 0.6961, θ_{12} = -2.1654
N_4	θ_{13} = 0.7125, θ_{14} = -2.5214, θ_{15} = 3.0607, θ_{16} = 0.8051, θ_{17} = -2.3235
N_5	θ_{18} = -0.6526, θ_{19} = -0.6319, θ_{20} = 0.4780
N_6	θ_{21} = 0.1993, θ_{22} = -0.4197
N_7	θ_{23} = -0.4502, θ_{24} = -0.1303, θ_{25} = 0.2443
N_8	θ_{26} = 0.8956, θ_{27} = -1.3160
N_9	θ_{28} = 0.6712, θ_{29} = -0.9731
N_{10}	θ_{30} = 0.0054, θ_{31} = 1.2359, θ_{32} = -1.2308

Table 1: Esimated parameters for the linear model.

tions are denoted by the respective positive and negative signs (Gardner et al. (2003)). Note that all the estimated parameters for the degradation terms have negative values, which makes sense from a biological point of view.

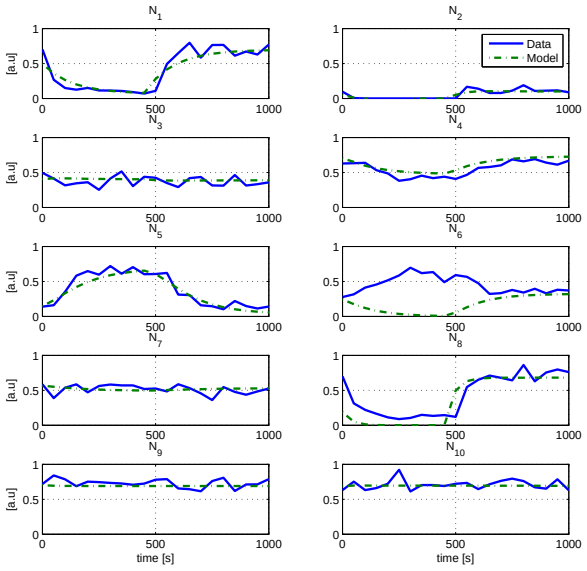

Fig. 2: Comparison between linear model and DREAM data on the validation data set.

Figure 2 shows the comparison between the linear model and the DREAM data on the validation data set. In general, the linear model is able to capture the data well, except for N_6. One possible reason for this could be that the estimation data set does not provide sufficient information on the dynamics of the interconnection between N_6 and N_8.

As a quantitative measure of model performance, the Mean Square Error (MSE) for each gene between the DREAM data and the linear model is computed as follows:

$$\text{MSE} = \frac{1}{N_T}\sum_{t=1}^{N_T}[N_i(t) - \hat{N}_i(t,\theta)]^2 \tag{3}$$

where $i = 1, 2, \ldots, 10$.

The MSEs for both the estimation and validation data sets are given in Table 2. The total MSE, MSE_{total} is computed by summing the MSE of all ten genes. Overall, the values of MSE are small, and similar between the estimation and validation data sets, except for N_6 and N_8, as expected.

Gene	MSE (Estimation)	MSE (Validation)
N_1	0.0052	0.0104
N_2	0.0007	0.0010
N_3	0.0046	0.0054
N_4	0.0019	0.0058
N_5	0.0034	0.0065
N_6	0.0011	0.1332
N_7	0.0066	0.0048
N_8	0.0039	0.0336
N_9	0.0053	0.0049
N_{10}	0.0043	0.0053
MSE_{total}	0.0371	0.2109

Table 2: MSE for both estimation and validation data sets.

To further validate the performance of the linear model, the steady state levels of each gene in the network under knock down and knock out mutations are compared with the steady state levels provided in the DREAM network data for the same mutations. In the knock down mutation, the transcription rate for each genes is halved while for the knock out mutation, the transcription rate for each gene is set to 0. For the resulting 100 steady state values, the model predictions were within $\pm 30\%$ of the corresponding steady state data for 81% of the knock down and 72% of the knock out mutations.

To test the robustness of the linear model fit against parameter uncertainty, a parameter sensitivity analysis of the linear model is carried out where each parameter is multiplied by a constant value ranging from 0 to 2 with 0.2 increment. As a quantitative measure, we compute the relative error between the steady state value for all genes in the model without parameter variation and the steady state value for all genes in the model with parameter variation. The relative steady state error is computed using

$$e_{SS} = \frac{\mathscr{M} - \mathscr{M}_{pt}}{\mathscr{M}}$$

where $\mathscr{M}$ is the steady state value of the model without parameter variation and $\mathscr{M}_{pt}$ is the steady state value of the model with parameter variation. From Figure 3, we see that the model is robust against parameter variation, where most of the values of e_{SS} stay close to zero and within $\pm 10\%$. The regulatory interactions identified by our model are shown in Figure 1(B).

3.2 Kernel architecture of gene regulatory network

In (Foo et al. (2016)), the core genetic circuitry of a plant circadian system is unravelled through identification of its *kernel* architecture. Here *kernel* is defined as the collection of the minimal set of interactions that must be present in the network to generate the temporal behaviour close to wild type. In our context, the kernel should consist of the minimum interactions

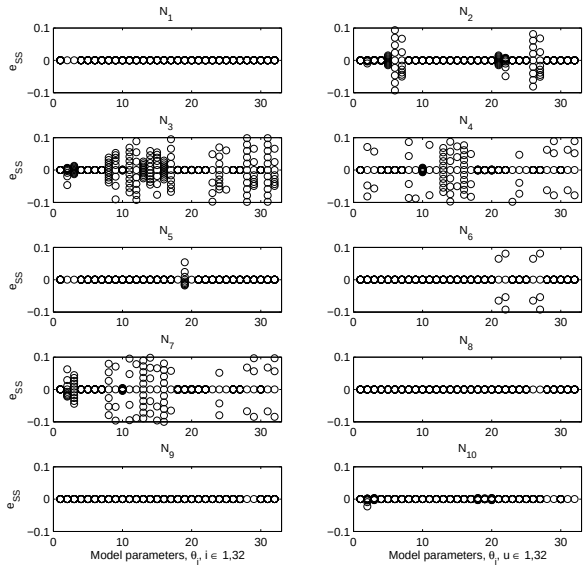

Fig. 3: Parameter sensitivity analysis.

required to reproduce the DREAM network temporal data. The kernel identification procedure suggested in (Foo et al. (2016)) is composed of several steps; the main idea involves removal of the interactions in the network. This is followed by parameter reoptimisation to determine whether the error between model and real data is within a pre-determined threshold. If the model with removed interactions followed by parameter reoptimisation is able to achieve an error within the threshold, those interactions can be removed, otherwise, they are retained.

Application of the kernel identification procedure allows us to identify the kernel of the DREAM gene regulatory network as shown in Fig. 1(C). The kernel of the DREAM network reveals several interesting features. Firstly, there is an isolated interaction involving N_2, N_6 and N_8. In other words, any perturbation occurring to either one of those genes would have almost no effect on the whole network. Fortunately, in our case, this finding also negates the poor performance of the model observed in the validation data set between N_6 and N_8 as these genes are not part of the kernel architecture. In addition, we can see that the external input signal W_2 will not be useful for the purposes of perturbation mitigation, as its effect is confined to N_2, N_6 and N_8.

Secondly, the kernel reveals four main pathways (labelled Pathway I, II, III and IV) converging to N_4, making this the most 'vulnerable' gene in the network. Thus, any phenotypes related to or downstream of N_4 would be most significantly affected by perturbations targeting this subnetwork. All these features are revealed through identification of the kernel architecture, which would not have been obvious if the full model were used.

4. RESULTS

4.1 Mitigation strategies based on network reoptimisation and rewiring

Based on our previous analysis of the most vulnerable gene in the network, we assume that an external perturbation enters the network through W_1 and affects the steady state level of N_4. From a control perspective, W_1 and N_4 represent the disturbance and output signal respectively.

The steady state level of N_4 without perturbation simulated using the linear model is 0.732. When a unit step perturbation enters through W_1, the steady state level of N_4 drops to 0.474. From a biological point of view, this can be interpreted as the effect of pathogen/virus attack leading to a drop in the optimal functioning level of N_4. Often in biology, when a perturbation occurs, biological systems react to restore as much as possible their normal functional behaviour. Thus, we reoptimise the network parameters to check whether the steady state level of N_4 can be restored in the presence of perturbation. The reoptimisation exercise is carried out to simulate the ability of the biological system to evolve and adapt in the presence of perturbation. After reoptimisation, we note that the steady state level of N_4 increases to 0.608 as shown by the yellow bar in Fig. 4. Although there is a recovery in the steady state level, it is still unable to regain its initial level of 0.732.

Thus, in the following simulation analyses, in the presence of the perturbation, we remove pathways affecting N_4 individually and jointly to determine whether any other combination of these pathways can be used to restore the steady state level of N_4 to its initial level. Fig. 4 shows the steady state level of N_4 under different investigated scenarios. As a note, we take the average value of the steady state level of N_4 when one to three pathways are removed. Also, we group the cases with and without reoptimisation upon pathway removals with the same color for ease of comparison.

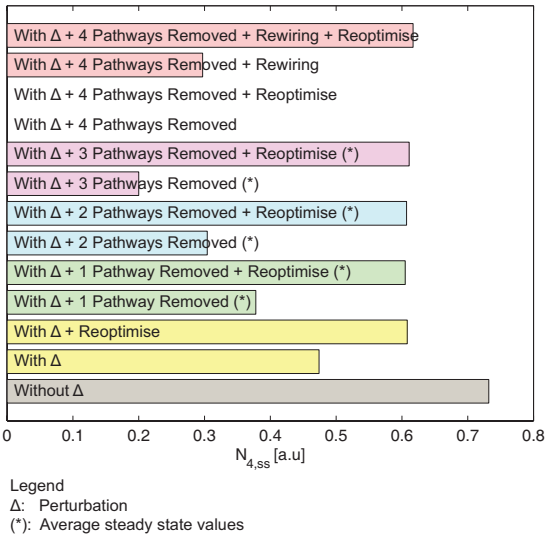

Fig. 4: Steady state level of N_4 under different investigated scenarios. For the case where one to three pathways are removed, the average value of N_4 is presented.

From Fig. 4, we observe that when one to three pathways are removed, the steady state level of N_4 ranges from 0.200 to 0.378. Through reoptimisation, the steady state levels of N_4 increases to within the range of 0.605 to 0.611 (as indicated by the green, blue and violet color bar graphs), which is greater than 0.474 albeit still less than 0.732. When all four pathways are removed, the steady state level of N_4 cannot be restored even with reoptimisation. Our results suggest that although the steady state level of N_4 can be increased with reoptimisation, no alternative pathway structure can be used to restore the steady

state level of N_4 to its initial value. Therefore, we next consider the approach of network rewiring. A detailed look at Fig. 1(C) reveals that the activation of N_4 by N_9 through W_3 can be a potential candidate for rewiring.

To test this idea, we remove all four pathways and rewire N_9 to activate N_4 directly (see the green dashed line in Fig. 1(D)). The rewiring strategy increases the steady state level of N_4 from 0.00 to 0.297 and with reoptimisation the steady state level of N_4 further increases to 0.617, which is the highest steady state level of N_4 compared to other strategies. Although rewiring is still unable to restore the steady state level of N_4 to its initial value, this finding suggests that there could be advantages to rewiring and it should be considered as part of the control strategy for network mitigation. This is further investigated in the following section.

4.2 Mitigation strategies using synthetic feedback control

In the previous section, we analysed the effect of perturbation on the steady state level of N_4 and note that despite considering several strategies and employing extensive parameter reoptimisation, the steady state value of N_4 is still not restored to its initial level. To address this issue, we propose the following feedback control strategy for network mitigation in our attempt to restore the steady state level of N_4, which is shown in Fig. 1(D), whereby the control action is provided by W_3.

Here, a genetic-based proportional-integral (PI) controller, which has the following form:

$$\frac{dq}{dt} = K_I e$$
$$\frac{dC}{dt} = \gamma(K_P e + q - C)$$

is used, where K_P and K_I are the gains and γ is the production rates of the controller. Biological implementation of this controller structure can be achieved following for e.g. the framework suggested in (Ang et al. (2010); Ang and McMillen (2013)).

The control objective is to maintain the steady state level of N_4 at a given set-point, i.e. 0.732. As shown in Fig. 5(A), when there is no perturbation, the steady state level of N_4 is 0.732. In our simulation, the perturbation enters the network through W_1 at time 4000 s. As expected, without any control or rewiring strategy, the steady state level of N_4 is unable to reach the set-point and the steady state level drops to 0.474 (see Fig. 5(B)). With feedback control, where $K_P = 1$, $K_I = 0.04$ and $\gamma = 0.01$, the level of N_4 is able to recover to within 5% of the set-point at approximately 5500 s (see Fig. 5(C)).

In the previous section, we note that there is an improvement in the steady state value of N_4 when rewiring is carried out. In Fig. 5(D), we can see that the steady state level of N_4 after rewiring is 0.537. We proceed further by reoptimising the network parameters and as shown in Fig. 5(E), this further improves the steady state level of N_4 to 0.618. While rewiring and reoptimisation do help in increasing the steady state value of N_4, there are still not reaching the intended set-point. As such, we introduce feedback control to the rewired network and as shown in Fig. 5(F), the feedback control is able to bring the steady state level of N_4 back to the set-point and with a faster response time compared to the feedback control without rewiring. With a synthetic feedback control loop applied to the rewired network, the level of N_4 is able to reach within 5% of the set-point at approximately 4700 s.

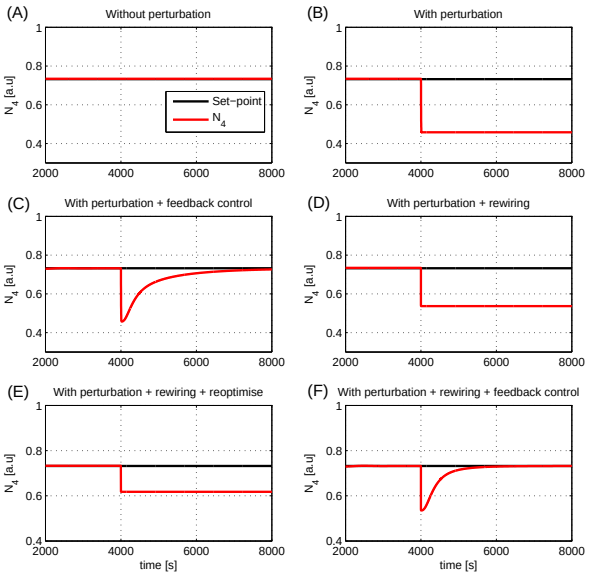

Fig. 5: N_4 set-point regulation.

4.3 Robustness analysis of the feedback control system

Here, the robustness of the two feedback control systems with and without rewiring is investigated using Monte Carlo simulations. In the simulation, all the parameters in the linear model are randomly drawn from a uniform distribution and the above simulations are performed 1060 times. The number 1060 is chosen following the Chernoff bound (Vidyasagar (1998)), following the guidelines given in (Williams (2001)): this number of simulations is required to accomplish an accuracy level of 0.05 with confidence level of 99% (Vidyasagar (1998); Menon et al. (2009)). All the parameters are varied within ranges of 20% around their nominal values, i.e. $\phi(1-\delta\Lambda)$, where δ = 0.2 and Λ is a random number from the uniform distribution in [-1,1].

The simulation results are shown in Fig. 6. The shaded regions encompass all responses from 1060 Monte Carlo simulations for randomly perturbed parameters in the range of $\pm\delta$ from the nominal values. Both the controller strategies show good level of robust performance with no stability issues as a result of varying parameters. We repeat our simulations with $\delta = 0.5$ and observe similar results.

5. CONCLUSIONS

In this paper, we use system identification techniques to build a linear model and identify types of regulation in a gene regulatory network. Using this linear model, we proceed to obtain the kernel architecture of the network, which is defined as the collection of minimal interactions that must be present to generate the temporal behaviour close to wild type. From the kernel architecture, we identify four main pathways within the network. Based on these pathways, we suggest simulation scenarios and mitigation strategies in the event of perturbation entering the

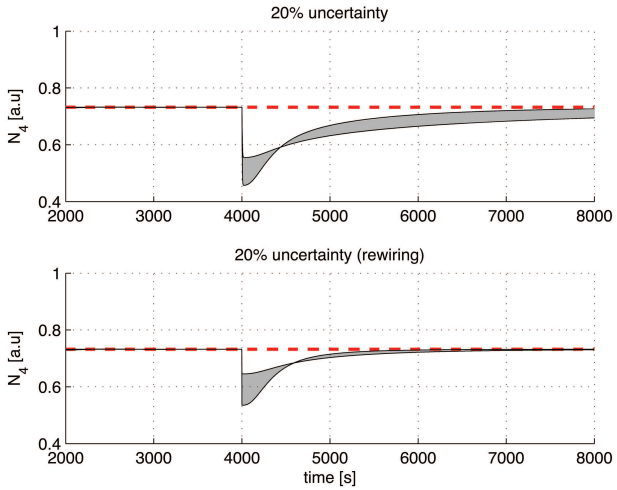

Fig. 6: Robustness analysis of the feedback controller. Dashed-line: set-point.

network. We found that through network reoptimisation, the effect of the perturbation can be reduced, while through feedback control with network rewiring, the effect of the perturbation can be eliminated completely. While many strategies for network rewiring have been proposed in the literature, in many cases their experimental implementation on real biological systems is often not feasible or is strongly constrained by experimental limitations. Our approach of mitigation has taken this factor into consideration, and has great potential for application using the tools of synthetic biology. For instance, through combined use of feedback control and network rewiring strategies, we can reduce the effect of perturbations such as infectious agents in crop plants, or increase the robustness of bacterial networks to the introduction of synthetic circuits.

ACKNOWLEDGEMENTS

The support through EPSRC and BBSRC research grants BB/M017982/1 and the School of Engineering of the University of Warwick are gratefully acknowledged.

REFERENCES

J. Ang, S. Bagh, B.P. Ingalls, and D.R. McMillen. Considerations for using integral feedback control to construct a perfectly adapting synthetic gene network. *Journal of Theoretical Biology*, 266:723–738, 2010.

J. Ang, and D.R. McMillen. Physical constraints on biological integral control design for homeostasis and sensory adaptation. *Biophysical Journal*, 104:505–515, 2013.

M. Bansal, V. Belcastro, A. Ambesi-Impiombato, and D. di Bernando. How to infer gene networks from expression profiles. *Molecular Systems Biology*, 78, 2007.

D. di Bernardo, M.J. Thompson, T.S. Gardner, S.E. Chobot, E.L. Eastwood, A.P. Wojtovich, S.J. Elliot, S.E. Schaus, and J.J. Collins. Chemogenomic profiling on a genome-wide scale using reverse-engineered gene networks. *Nature Biotechnology*, 23:377–383, 2005.

N. Dalchau, S.J. Baek, H.M. Briggs, F.C. Robertson, A.N. Dodd, M.J. Gardner, M.A. Stancombe, M.J. Haydon, G-B. Stan, J.M. Goncalves, and A.A.R. Webb. The circadian oscillator gene *GIGANTEA* mediates a long-term response of the *Arabidopsis thaliana* circadian clock to sucrose. *Proceedings of the National Academy of Sciences, USA*, 108:5104–5109, 2011.

M. Foo, H.Y. Yoo, and P.-J. Kim. System identification of circadian clock in plant *Arabidopsis thaliana*. *Proceedings of International Conference on Control, Automation and Systems*, 241–246, 2013.

M. Foo, D. Somers, and P.-J. Kim. Kernel architecture of the genetic circuitry of the *Arabidopsis* circadian system. *PLoS Computational Biology*, 12:e1004748, 2016.

T.S. Gardner, D. di Bernardo, D. Lorenz, and J.J. Collins. Inferring genetic networks and identifying compound mode of action via expression profiling. *Science*, 301:102–105, 2003.

E. Herrero, E. Kolmos, N. Bujdoso, Y. Yuan, M. Wang, M.C. Berns, H. Uhlworm, G. Coupland, R. Saini, M. Jaskolski, A. Webb, J. Goncalves, and S.J. Davis. EARLY FLOWERING4 recruitment of EARLY FLOWERING3 in the nucleus sustains the *Arabidopsis* circadian clock. *Plant Cell*, 24:428–443, 2012.

Y-Y. Liu, J.J. Slotine, and A-L. Barabasi. Controllability of complex networks. *Nature*, 473:167–173, 2011.

Y-Y. Liu, J.J. Slotine, and A-L. Barabasi. Observability of complex networks. *Proceedings of the National of Academy of Sciences, USA*, 110:2460–2465, 2013.

Y-Y. Liu, and A-L. Barabasi. Control principles of complex systems. *Reviews of Modern Physics*, 88:035006 , 2016.

L. Ljung. *System identification: theory for the user.* 2nd Edition, Prentice-Hall, Upper Saddle River, NJ 1999.

D. Marbach, T. Schaffter, T. Mattiussi, and D. Floreano. Generating realistic *in silico* gene networks for performance assessment of reverse engineering methods. *Journal of Computational Biology* 16:229–239, 2009.

P.P. Menon, I. Postlethwaite, S. Bennani, A. Marcos, and D.G. Bates. Robustness analysis of a reusable launch vehicle flight control law. *Control Engineering Practice* 7:751–765, 2009.

M. Newman. *Networks: an introduction.* Oxford University Press, 2010.

G. Stolovitzky, D. Monroe, and A. Califano. Dialogue on reverse-engineering assessment and methods: the DREAM of high-throughput pathway inference. In G. Stolovitzky, and A. Califano, editor, *Annals of the New York Academy of Science*, 1115:11–22, 2007.

G. Stolovitzky, R.J. Prill, and A. Califano. Lessons from the DREAM2 challenges. In G. Stolovitzky, and A. Califano, editor, *Annals of the New York Academy of Science*, 1158:159–195, 2009.

M. Ventresca, and D. Aleman. Evaluation of strategies to mitigate contagion spread using social network characteristics. *Social Networks*, 35:75–88, 2013.

M. Vidyasagar. Statistical learning theory and randomised algorithm for control. *IEEE Control Systems*, 18:69–85, 1998.

A. Vinayagam, T.E. Gibson, H-J. Lee, B. Yilmazel, C. Roesel, Y. Hu, Y. Kwon, A. Sharma, Y-Y. Liu, N. Perrimon, and A-L. Barabasi. Controllability analysis of the directed human protein interaction network identifies disease genes and drug targets. *Proceedings of the National Academy of Sciences, USA*, 113:4976–4981, 2016.

P.S. Williams. A Monte Carlo dispersion analysis of the x-33 simulation software. *Proceedings of AIAA Conference on Guidance, Navigation and Control*, 69–85, 2001.