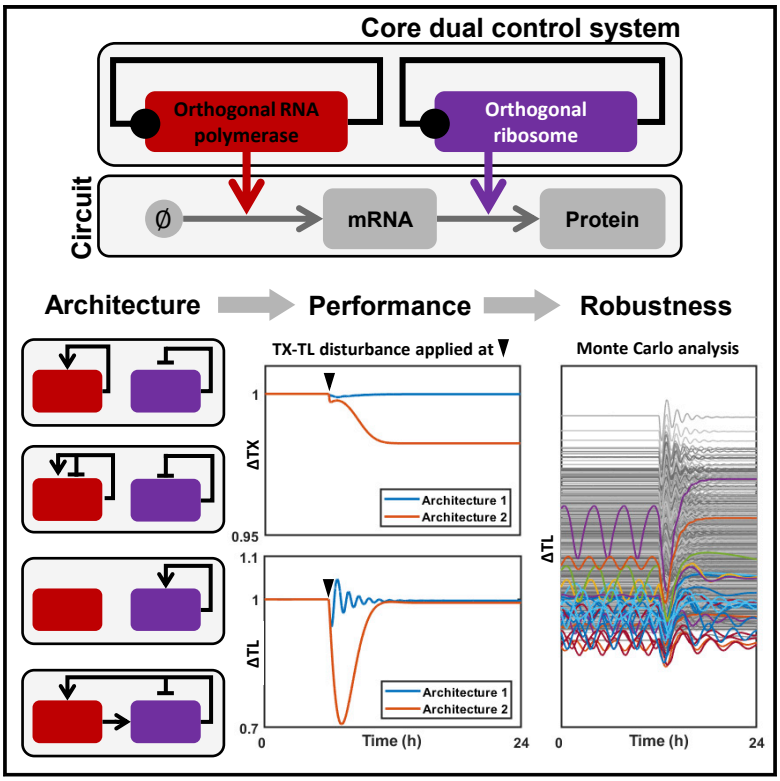
Cell Systems

Report

Architectures for Combined Transcriptional and Translational Resource Allocation Controllers

Graphical Abstract



Highlights

- Competition for gene expression resources can result in the failure of gene circuits
- Performance can be improved by circuit-specific resource allocation controllers
- We analyze potential architectures for dual transcriptional-translational controllers
- A fully orthogonal dual controller shows superior performance and robustness

Authors

Alexander P.S. Darlington, Declan G. Bates

Correspondence

d.bates@warwick.ac.uk

In Brief

Synthetic cellular circuits often fail due to competition for shared gene expression resources. Dynamic allocation of cellular resources using feedback control provides a means to mitigate these effects. Darlington and Bates show how dual transcriptional-translational resource allocation controllers may be designed to improve the performance and robustness of synthetic circuits.

Darlington & Bates, 2020, Cell Systems *11*, 1–11
October 21, 2020
https://doi.org/10.1016/j.cels.2020.08.014

CellPress

Please cite this article in press as: Darlington and Bates, Architectures for Combined Transcriptional and Translational Resource Allocation Controllers, Cell Systems (2020), https://doi.org/10.1016/j.cels.2020.08.014

Cell Systems

CellPress

Report

Architectures for Combined Transcriptional and Translational Resource Allocation Controllers

Alexander P.S. Darlington[1] and Declan G. Bates[1,2,*]
[1]Warwick Integrative Synthetic Biology Centre, School of Engineering, University of Warwick, Coventry, UK
[2]Lead Contact
*Correspondence: d.bates@warwick.ac.uk
https://doi.org/10.1016/j.cels.2020.08.014

SUMMARY

Recent work on engineering synthetic cellular circuitry has shown that non-regulatory interactions mediated by competition for gene expression resources can result in degraded performance or even failure. Transcriptional and translational resource allocation controllers based on orthogonal circuit-specific gene expression machinery have separately been shown to improve modularity and circuit performance. Here, we investigate the potential advantages, challenges, and design trade-offs involved in combining transcriptional and translational controllers into a "dual resource allocation control system." We show that separately functional, translational, and transcriptional controllers cannot generally be combined without extensive redesign. We analyze candidate architectures for direct design of dual resource allocation controllers and propose modifications to improve their performance (in terms of decoupling and expression level) and robustness. We show that dual controllers can be built that are composed only of orthogonal gene expression resources and demonstrate that such designs offer both superior performance and robustness characteristics.

INTRODUCTION

Synthetic gene circuits can be designed to perform complex computations and information processing in living cells with applications in biomedicine, chemistry, and environmental sciences (Brophy and Voigt, 2014). By introducing synthetic gene circuits into microbial hosts, synthetic biologists and biotechnologists are able to control cell function. However, often these initial designs fail due to the effect of unforeseen interactions between the circuit and host cell or due to host constraints (Cardinale and Arkin, 2012; Borkowski et al., 2016). In addition, circuits produced in one strain can behave both quantitatively and qualitatively different when transferred to other strains, due to the subtle impact of changing genetic context (Moser et al., 2012; Vilanova et al., 2015; Ceroni et al., 2015).

A key cause of context-dependent dysfunction of gene circuits is the competition for shared gene expression resources, such as host RNA polymerases and ribosomes (Gyorgy et al., 2015; Ceroni et al., 2015; Carbonell-Ballestero et al., 2016; Gorochowski et al., 2016). Two independently characterized modules, when brought together in a synthetic circuit, interact through the use of common resource pools; e.g., as one gene is induced it indirectly inhibits other genes by sequestering finite cellular resources. RNA encoding genes, such as sRNAs, compete for RNA polymerases. The impact is greater for protein-encoding genes, where the induction of the second gene results in both a transcriptional and translational disturbance as both RNA polymerase molecules are sequestered for mRNA production and ribosomes are sequestered for protein production. RNA encoding genes also impact protein-encoding genes through their competition for RNA polymerases. The competition for ribosomes is a key determinant of the gene expression burden in *E. coli* and is often deemed as the cause of most host circuit coupling (e.g., Ceroni et al. (2015)), however, the transcriptional coupling can have subtle but significant effects (such as changing the slope of a gene expression coupling isocost line (Gyorgy et al., 2015)), and competition for RNA polymerase flux has been blamed for the failure of modularity in logic gate design (Shin et al., 2020).

Circuit-specific "orthogonal" gene expression resources that only transcribe/translate circuit genes (and not host genes) have been proposed to alleviate some aspects of host circuit interactions in *E. coli* (An and Chin, 2009). These systems utilize a bacteriophage RNA polymerase for transcription and synthetic ribosomal RNAs to render host ribosomes orthogonal. These systems reduce host circuit interactions due to reduced competition, as host genes cannot utilize orthogonal resources. However, they do not eliminate coupling between circuit genes by themselves (Darlington et al., 2018a). We propose that synthetic control systems can be employed to manage these orthogonal gene expression resources, creating a cellular "virtual machine" (Liu et al., 2018) that allows decoupling of circuit genes without an extensive redesign (Darlington et al., 2018b; Grunberg and Del Vecchio, 2020). In this paper, we demonstrate the need for, and explore the feasibility of, combining synthetic transcriptional and translational resource allocation systems to function as dual transcription-translation controllers that decouple circuit genes at both levels of expression. We show that interactions

Please cite this article in press as: Darlington and Bates, Architectures for Combined Transcriptional and Translational Resource Allocation Controllers, Cell Systems (2020), https://doi.org/10.1016/j.cels.2020.08.014

between separately functional transcriptional and translational controllers can result in instability when they are implemented simultaneously. We evaluate potential architectures for dual controllers based on experimentally validated transcriptional and translational resource allocation systems and identify potential modifications to improve decoupling performance and robustness. Finally, we propose a dual controller that uses only orthogonal components and verify that it exhibits superior performance and robustness properties compared with the currently available designs.

RESULTS

Modeling Resource-Mediated Coupling at the Transcriptional and Translational Levels Demonstrates the Need for Combined Resource Allocation

Initially, we developed a simple ordinary differential equation model, taking into account the transcription (modeled as RNA polymerase binding/unbinding to/from a promoter and mRNA birth), translation (ribosome binding/unbinding to/from an mRNA and protein birth), and dilution of all species and intermediate complexes. This base model takes into account the usage of the host gene expression resources, and therefore, captures how resource limitations create non-regulatory couplings. For full model details see STAR Methods. We next augmented the base model with the additional orthogonal gene expression resource and their respective control systems, changing the transcriptional or translational apparatus from host to orthogonal, as appropriate.

At the transcriptional level, Kushwaha and Salis have previously designed and implemented a "universal bacterial expression resource" (UBER) to alleviate the impact of the species-context dependency, (Kushwaha and Salis, 2015). This system is based on an RNA polymerase from the T7 bacteriophage, which forms an orthogonal transcriptional resource and is able to transcribe mRNAs in a range of different species (Figure 1A). The orthogonal RNAP (o-RNAP) transcribes its own mRNA creating a positive feedback loop. However, at high concentrations, the T7 RNAP is toxic and results in reduced cell growth. Therefore, the o-RNAP transcription must be tightly controlled using negative feedback, by placing it under the regulation of the repressor TetR, which by itself is transcribed by the o-RNAP. Simulations of the UBER controller decouples co-expressed genes at the transcriptional level with the mRNA of gene 1, m_1, showing no disturbance upon the activation of gene 2 (Figure 1B). However, this does not propagate to the translational level due to the competition for translational resources (Figure 1C). The UBER controller successfully mitigates the impact of a transcriptional-only disturbance (Figures S2A and S2B).

Segall-Shapiro et al. (2014) developed a set of compatible artificial fragments of T7 RNA polymerase. When co-expressed, these fragments bind to form a functional RNA polymerase (Figure 1D). They used these components to develop a "transcriptional resource allocator" (FRAG), which buffers the impact of different gene copy numbers. The core component of RNA polymerase β is constitutively expressed but only becomes functional when bound by the α-fragment that is co-expressed with the circuit genes. Simulations of the FRAG controller show poorer performance with significant coupling remaining when the transcriptional disturbance is applied: the mRNA of gene 1 falls 25% when gene 2 is induced at 12 h (Figure 1E). This still represents an improvement from a 50% fall in gene 1 in the absence of control (Figure S1A). Again, this decoupling does not propagate to the translational level, with protein 1 falling to 50% upon the induction of protein 2 (Figure 1F). The FRAG controller mitigates the impact of a transcriptional-only disturbance (Figures S2C and S2D).

Both controller topologies achieve their transcriptional decoupling action by increasing the effective concentration of the free RNA polymerase, rather than increasing the orthogonal RNA polymerase production to match the demand (Figure S1). In the absence of the control, increasing the concentration of RNA polymerase (either host or orthogonal) *in silico* results in decreased mRNA coupling (Figure S1E), as the concentration of the RNA polymerase significantly exceeds the polymerase-promoter dissociation constant (see Gyorgy et al., 2015). Although, note that increasing host RNA polymerase concentration by genetically engineering the host is largely impracticable as the RNA polymerase is encoded by numerous genes, which are co-regulated by numerous other biological processes. The poorer performance of the Segall-Shapiro et al. controller is due to a fall in the RNA polymerase concentration due to translational coupling between the polymerase expression and circuit genes—this moves the system from a regime of low coupling to moderate coupling (Figure S1E).

To alleviate translational resource competition, we have previously designed and implemented with colleagues a translational resource allocation controller (OR) in *E. coli* (Darlington et al., 2018a) (Figure 1G). The ribosome is a large ribonucleoprotein complex encoded by multiple rRNA and r-protein genes; therefore, unlike in the case of transcription, there does not exist a truly orthogonal ribosome that can be co-opted from another species. However, quasi-orthogonal ribosomes (o-ribosomes) can be created using synthetic 16S rRNAs, which target the ribosome machinery to mRNAs that contain the complementary ribosome binding site (RBS) sequence (e.g., An and Chin, 2009). The translational controller works by regulating the production of the synthetic 16S rRNA. This negative feedback controller takes the form of a repressor protein, which inhibits 16S rRNA production and itself is translated by the o-ribosome pool. As demand for the orthogonal translation increases, the level of the repressor falls due to resource competition and so o-rRNA production (and therefore orthogonal ribosome production) increases to match demand. Previously, we fully analyzed a resource allocation controller at the translational level (in the absence of RNAP competition) to derive design guidelines for potential biological implementations (Darlington et al., 2018b). Applying these findings, we simulated a controller composed of the tightly binding multimeric repressor (here LacI), a high o-rRNA copy number, and a low transcription factor copy number. In the presence of the RNA polymerase competition, this controller is not perfect but still reduces the fall in protein 1 when protein 2 is induced to 20% (Figure 1I). Decoupling is not complete (i.e., no fall in p_1 upon addition of p_2) due to the presence of coupling at the transcriptional level (Figure 1H). Simulating the response of the controller to an RNA-only input (that is a transcriptional

Please cite this article in press as: Darlington and Bates, Architectures for Combined Transcriptional and Translational Resource Allocation Controllers, Cell Systems (2020), https://doi.org/10.1016/j.cels.2020.08.014

CellPress

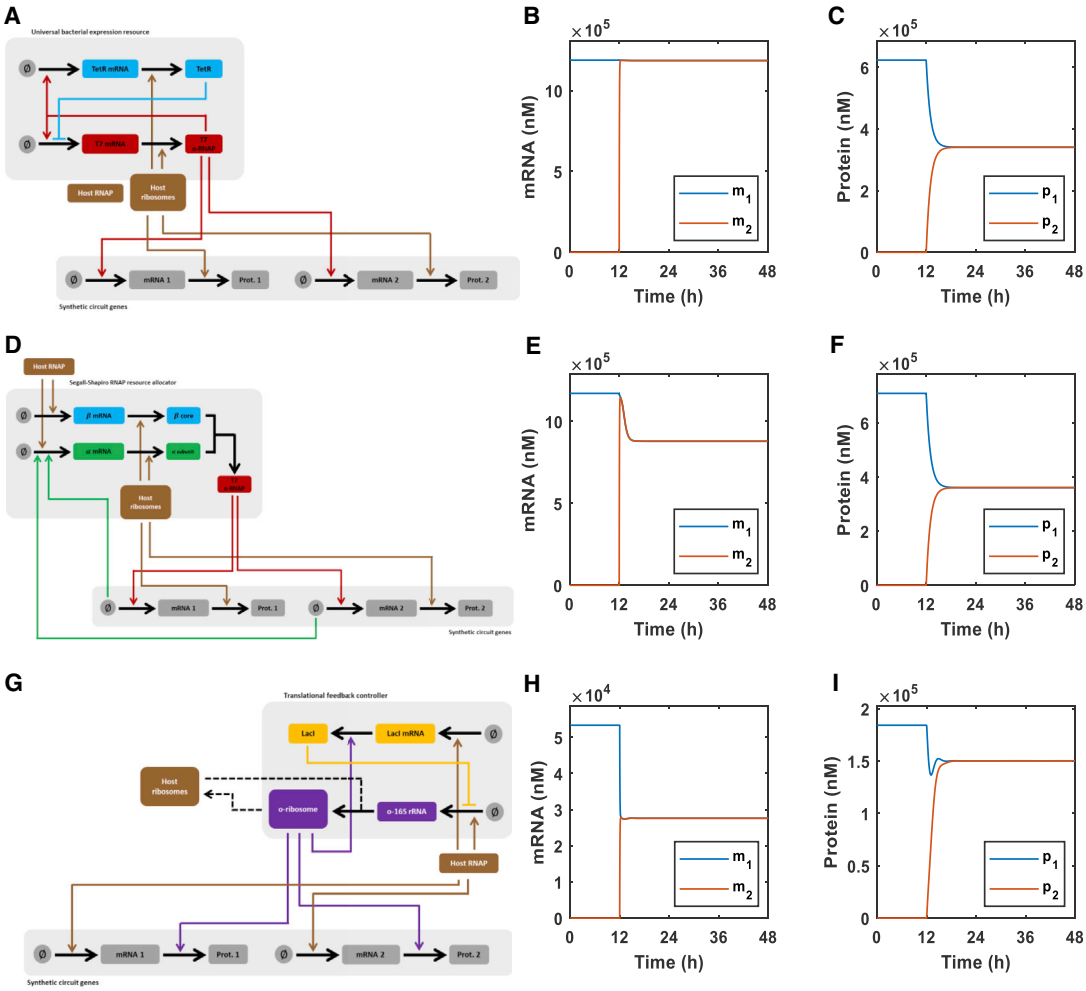


Figure 1. Gene Decoupling by Transcriptional and Translational Controllers

The behavior of candidate transcriptional and translational control systems in isolation. We simulated the ability of each prototype control system to decouple co-expressed genes. We consider the impact on a constitutively expressed gene (g_1) of the induction of a second gene (g_2) at 12 h. This results in a transcriptional disturbance (as promoters compete for RNA polymerase) and a translational disturbance (as mRNAs compete for ribosomes).

(A) The architecture of the universal bacterial expression resource (UBER) that supplies orthogonal RNA polymerases to the circuit.

(B) The UBER controller successfully mitigates the transcriptional disturbance applied at 12 h.

(C) The UBER controller is unable to mitigate the disturbance at the translational level.

(D) Architecture of the fragmented RNA polymerase resource allocation controller (FRAG) that supplies orthogonal RNA polymerases to the circuit.

(E) The fragmented RNA polymerase controller is able to mitigate the transcriptional disturbance at 12 h to some extent with the fall in m_1 only being 25% rather than 50%.

(F) The FRAG controller is unable to mitigate the disturbance at the translational level.

(G) Architecture of the orthogonal ribosome-based translational controller (OR) dynamically supplies translational activity to the circuit.

(H) The translational controller has no impact on the transcriptional disturbance with m_1 falling by 50% upon activation of the second protein-encoding gene.

(I) The translational controller decouples genes at the translational level reducing the fall in p_1 from 50% to only 20%.

but not a translational disturbance) demonstrates that the controller is not able to respond to the transcriptional competition; i.e., there is a 10% decrease in $p_1$even though there is no ribosomal competition, due to the competition for the host RNAP (Figures S2E and S2F).

Combining Transcriptional and Translational Resource Allocation Controllers Necessitates Extensive Redesign

To obtain a resource allocation system that operates simultaneously to decouple circuits at both the transcriptional and translational levels, we combined the experimentally validated transcriptional (UBER or FRAG) and translational (OR) controllers to create two "dual" control systems. This was achieved by setting both gene expression resources for the circuit genes to their orthogonal counterparts (Figures 2A and 2C). We did not change each controller's own gene expression resource usage, and so, each controller's internal topology remained the same. We named these control systems as UBER-OR and FRAG-OR, based on how the orthogonal transcriptional activity was controlled.

Please cite this article in press as: Darlington and Bates, Architectures for Combined Transcriptional and Translational Resource Allocation Controllers, Cell Systems (2020), https://doi.org/10.1016/j.cels.2020.08.014

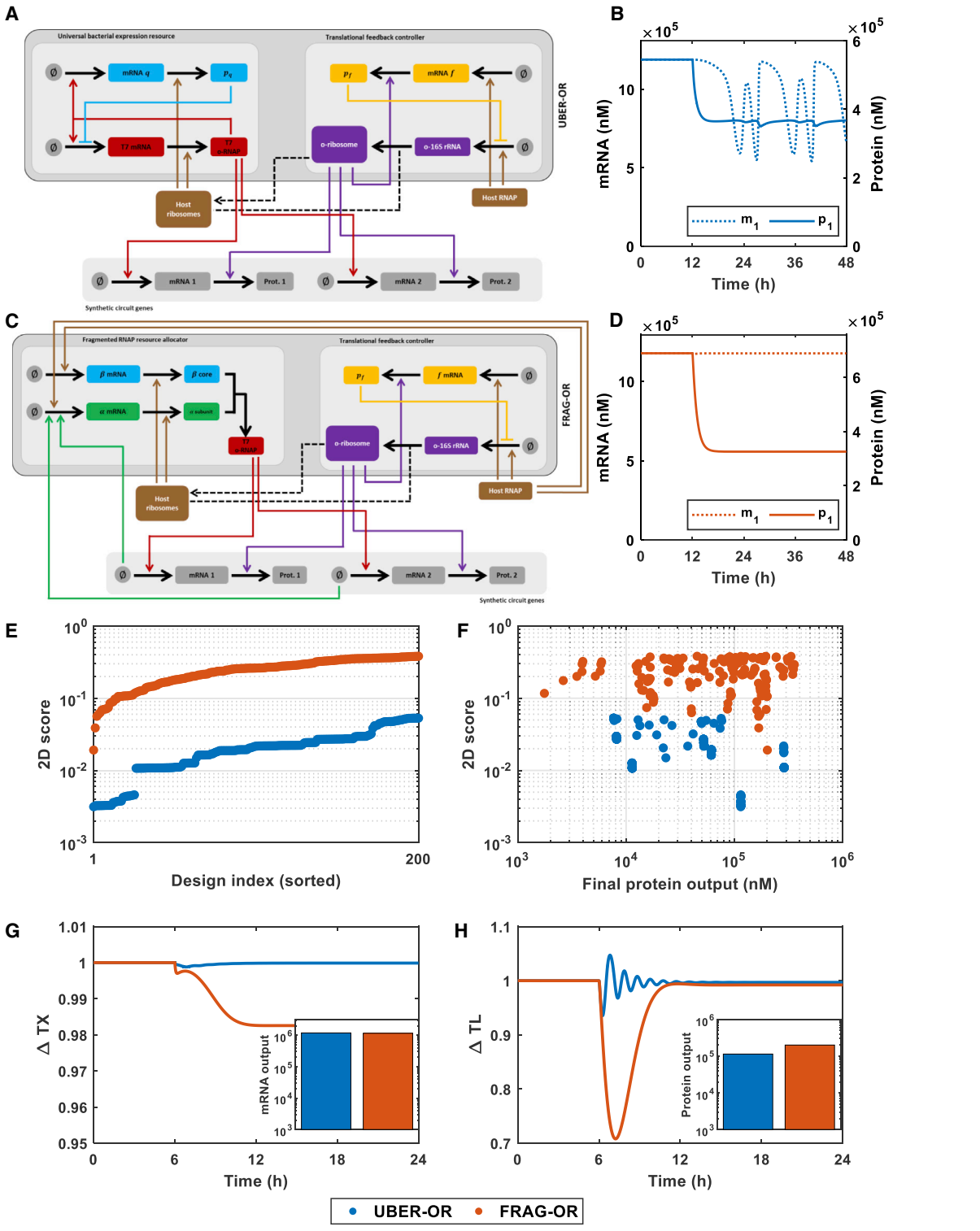


Figure 2. Architecture of the Combined Dual Transcriptional-Translational Controllers

We consider the impact on a constitutively expressed gene (with mRNA m_1 and protein p_1) of the induction of a second gene at 12 h. This results in a transcriptional disturbance (as promoters compete for RNA polymerase) and a translational disturbance (as mRNAs compete for ribosomes). The behavior of the controllers depicted in Figure 1 when combined into dual control systems are shown in (B) and (D). A number of candidate controllers were simulated as described in the STAR Methods. mRNA and protein coupling are calculated as the change in m_1 or p_1 (respectively) upon the induction of the second gene at 12 h. The 2D

(legend continued on next page)

Please cite this article in press as: Darlington and Bates, Architectures for Combined Transcriptional and Translational Resource Allocation Controllers, Cell Systems (2020), https://doi.org/10.1016/j.cels.2020.08.014

Simulating the combination of the existing experimentally validated controllers reveals that the dual system is now unstable, with large oscillations in the transcriptional output, which propagate to the translation level. Translational coupling remains significant (Figure 2B). Visual inspection of Figure 2A shows that the transcription and translational components of the control system interact at the ribosomal proteins. While the transcriptional element utilizes host ribosomes for its own expression and the translational element utilizes orthogonal ribosomes, the latter are produced from the co-option of host ribosomal proteins. To test this hypothesis, we decoupled the processes *in silico* by considering two "virtual" resource pools, one to supply the translational activity to the UBER controller and one to supply ribosomes for the orthogonal translation by the OR controller. Decoupling the controllers in this way restored stability but with poor performance (Figure S3). Combining the experimentally available FRAG and OR controllers resulted in no decoupling at the translational level (Figure 2D). Splitting the resources available into multiple "virtual" pools did not improve performance, suggesting that the failure of FRAG-OR to function was due to a mismatch between the two controllers' input-output functions.

These results clearly demonstrate that independently functional transcriptional and translational controllers cannot simply be combined without necessitating their redesign—a common problem in the multivariable control. To investigate the potential for addressing the observed problems, we redesigned both dual controllers' parameters while maintaining their architecture. We created a discrete, rather than a continuous, design space across a range of experimentally tunable parameters representing promoter and ribosome binding site strengths, controller gene copy numbers, and transcription factors. The choice of the transcription factor impacts $\alpha_{r,x}$representing the dissociation constant, $\eta_x$the transcription multimeric state, and the dissociation constant of the target promoter ($\xi_{f,x}$) (where *x* represents the species being regulated). These parameters are not independently designable and so we simulate the action of the common repressors tetR, lacI, and cI. (Note that the UBER-OR controller requires two transcription factors, p_q, which acts as the controller protein for the transcriptional resource, and p_f, which controllers the translational resource. To prevent crosstalk between these two parts of the combined controller we do not allow these proteins to be the same, i.e., $p_q \neq p_f$).

For each controller architecture and numerical design, we discarded controllers that demonstrated instability (e.g., oscillatory behavior) and poor output performance. We define performance in three ways: (1) mRNA coupling (i.e., the change in $m_1$when $g_2$is activated), (2) protein coupling (i.e., the change in $p_1$when $g_2$is induced), and (3) the final protein concentration. We collapse metrics (1) and (2) into one—the 2D score. The ideal dual controller would perfectly decouple genes at the mRNA and protein levels—i.e., $m_1$and $p_1$would not change (equivalent to (0, 0) in a two-dimensional performance space). First, we scaled each coupling metric by the maximum absolute coupling and then considered the Euclidean distance for each pair of (mRNA coupling, protein coupling) values. We identify the best "dual" controllers using this metric (see STAR Methods). Proteins form the main actuators of synthetic gene circuits, and so, we also consider final protein levels as part of our performance assessment.

This analysis reveals that the UBER-OR architecture gives superior decoupling with lower 2D scores than FRAG-OR (Figure 2E). However, the latter architecture tends to produce controllers with higher protein production levels (Figure 2F), indicating a trade-off between these different performance objectives. In general, both controllers give access to the same regions of decoupling performance space. The optimal UBER-OR controller shows the best performance, with no steady-state error at the transcriptional level (Figure 2G) and better dynamic performance at the translational level (H).

We have shown that both controllers can be composed of biologically reasonable parameters corresponding to obtainable promoter copy numbers and transcription factor dynamics. However, at present, the construction of such controllers is complicated by the uncertainty in the kinetic parameters of the available biological "parts," with large potential variations reported for many parts (Tsigkinopoulou et al., 2017). Often, precise measurements of these parameters are only obtainable from *in vitro* measurements, and how these relate to *in vivo* values is usually unknown. Even when a part with a desired set of kinetics exists with a small uncertainty, it is not clear how circuit context effects may impact this level of uncertainty; for example, the surrounding DNA sequence may cause subtle changes in binding rates. The causes of uncertainty in biological circuit design have been reviewed extensively by Cardinale and Arkin (2012) and Zhang et al. (2016). To take account of these biological realities, we assessed the robustness of both dual controllers to parametric uncertainty, focusing on the "designable" parameters governing the production rates of the

score is a one-dimensional representation of the two-dimensional mRNA coupling-protein coupling space and is calculated as outlined in the STAR Methods. The dynamic response of the best two redesigned controllers are shown in (G) and (H).

(A) UBER-OR is based on the universal bacterial expression resource developed in Kushwaha and Salis (2015) and the translational controller developed in Darlington et al. (2018a).

(B) Combining the tetR-based UBER controller from Kushwaha and Salis (2015) with the lacI-based translational controller form Darlington et al. (2018a) results in significant oscillations at the mRNA level which propagates to the translational level. Ignoring this instability, the combined system still shows the translational coupling of 30%.

(C) FRAG-OR is based on the fragmented RNA polymerase resource allocator developed in Segall-Shapiro et al. (2014) and the translational controller developed in Darlington et al. (2018a). (D) The combined FRAG-OR controller decouples genes at the mRNA level but significant coupling at the protein level remains.

(E) The two-dimensional score of the best 200 controllers for each controller architecture.

(F) Two-dimensional score plotted again the final protein concentration for both controller topologies.

(G) The dynamic change in the mRNA m_1 concentration (ΔT_X) in response to the induction of gene 2 of the best design of the UBER-OR and FRAG-OR architectures. The dynamics are normalized by their pre-disturbance levels. The Inset shows the final mRNA concentration (nM).

(H) The dynamic change in the protein p_1 concentration (ΔT_L) in response to the induction of gene 2 of the best design of the UBER-OR and FRAG-OR architectures. The dynamics are normalized by their pre-disturbance levels. The Inset shows the final protein concentration (nM).

Please cite this article in press as: Darlington and Bates, Architectures for Combined Transcriptional and Translational Resource Allocation Controllers, Cell Systems (2020), https://doi.org/10.1016/j.cels.2020.08.014

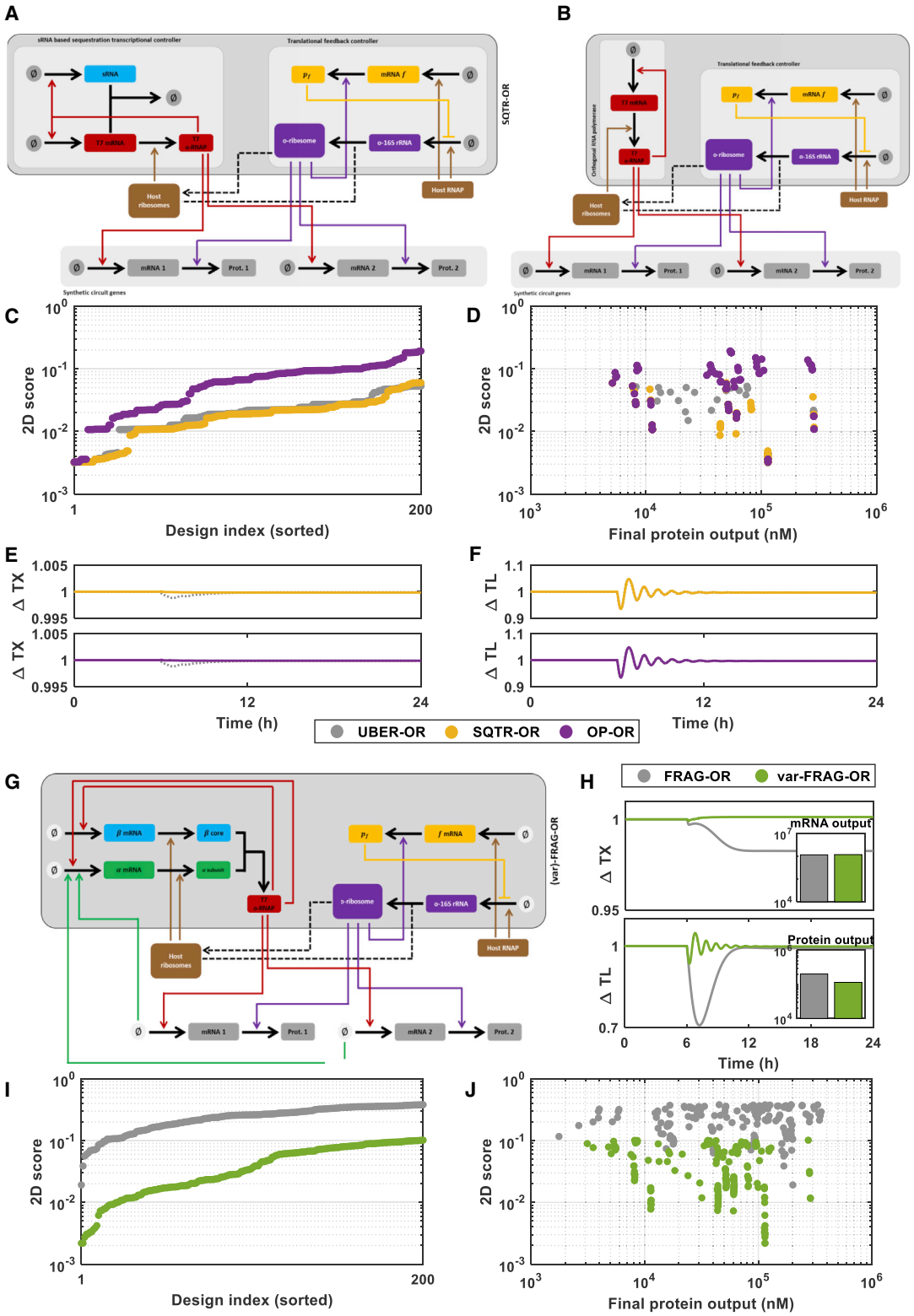


Figure 3. Redesigning UBER-OR and FRAG-OR for Improved Performance or Robustness

The behavior of the topologically redesigned dual controllers. We consider the impact on a constitutively expressed gene (g_1) of the induction of a second gene (g_2) at 12 h. This results in a transcriptional disturbance (as promoters compete for RNA polymerase) and a translational disturbance (as mRNAs compete for ribosomes). A number of biological candidate controllers were simulated as described in the STAR Methods. mRNA and protein coupling are calculated as the

(legend continued on next page)

Please cite this article in press as: Darlington and Bates, Architectures for Combined Transcriptional and Translational Resource Allocation Controllers, Cell Systems (2020), https://doi.org/10.1016/j.cels.2020.08.014

orthogonal RNA polymerase and rRNA, and the production rates and action of the other controller proteins (Table S1).

We assessed each controller design to parametric uncertainty by varying key controller parameters. These perturbed controller parameters were determined by Monte Carlo sampling of 1,000 designs, with each parameter perturbed by up to a set percentage. We carried out this sampling approach for uncertainties of 10% and above, increasing the uncertainty in increments of 10%. This analysis reveals that the UBER-OR controller has a low robustness to parametric uncertainty with perturbations of 20%, typical for a biological part, resulting in the emergence of oscillations (Figure S6). Interestingly, FRAG-OR, which produced lower decoupling performance, showed higher levels of robustness, tolerating variations of up to 60% in key controller parameters (Figure S7). Note that with the FRAG-OR controller, unstable traces emerged in controllers with high resource demand, i.e., high protein production.

Modifying Controller Architectures to Improve Performance/Robustness

Given that UBER-OR produced a better performance, but poorer robustness, than FRAG-OR, we investigated the dual controller architecture for potential sources of fragility. The activation-inhibition motif present in the core UBER architecture is known to be prone to the emergence of oscillations when parameters change slightly (Atkinson et al., 2003; Jayanthi and Del Vecchio, 2012). In the UBER-OR controller, the orthogonal RNA polymerase control system is based on the expression of a protein repressor that acts by reversibly binding the orthogonal RNA polymerase's promoter. These reversible interactions are often highly nonlinear due to the multimeric nature of transcription factor binding. (Although, note that in the best performing UBER-OR controller, this protein is proposed to be tetR, which does not show these highly nonlinear effects, but it still displays reversible binding.) We redesigned the controller to remove this potentially destabilizing aspect of its architecture by replacing the protein-based inhibition with an RNA-based sequestration. This has the additional benefit of reducing competition for ribosomes—reducing the linkages between the two control systems. This sequestration-based controller (SQTR-OR) utilizes the OR translational controller coupled with an RNA polymerase based controller, which is composed of an orthogonal RNA polymerase that transcribes itself and a second small RNA (Figure 3A). The sRNA binds to the polymerase mRNA and targets it for decay in a unidirectional reaction. This architecture shows similar performance to the original UBER-OR controller, with consistently similar 2D scores (Figure 3C) and a slightly improved dynamic performance (Figure 3E). However, the Monte Carlo robustness analysis shows that the modified controller is significantly more robust than the original design and is now able to tolerate levels of parametric uncertainty up to 50% (Figure S9). We removed the activation-inhibition motif entirely by removing the negative feedback loop to produce a transcriptional controller (OP, producing the combined controller OP-OR, Figure 3B). This best parametric design of this controller shows slightly worse performance than both UBER-OR and SQTR-OR (Figure 3C)—although, note that the majority of the scores are below 10^{-2}, which is equivalent to 10% steady-state error at both the transcription and translation levels. While the performance of this controller may be slightly worse, this may be offset by the lack of additional steps or tuning required during *in vivo* construction due to the lack of a feedback controller for the RNA polymerase. The Monte Carlo sampling of the controller parameters showed that the controller is able to tolerate significantly larger levels of parametric uncertainty than UBER-OR—up to 50% (Figure S11).

To investigate the potential to improve the performance of the FRAG-OR controller, we introduced a positive feedback loop into the transcriptional element of the controller by placing the α and β subunits under the orthogonal RNA polymerase for transcription (Figure 3G). As before, we then carried out an extensive search of potential biological implementations to identify the optimal design. This variant (var-FRAG-OR) topology shows a significantly improved decoupling performance without a reduction of expression (Figures 3H–3J). However, this modification significantly reduces the robustness of the controller, with oscillations now emerging at only 30% uncertainty (Figure S13).

A Fully Orthogonal Dual Controller Exhibits Higher Performance and Robustness

All the controllers considered above use host gene expression resources for aspects of their own production, for example, the transcriptional activity required for the OR controller is always provided by the host. Here, we propose a novel controller composed only of orthogonal parts, creating a parallel set of

change in m_1 or p_1 (respectively) upon the induction of the second gene at 12 h. The 2D score is a one dimensional representation of the two dimensional mRNA coupling-protein coupling space and is calculated as outlined in the STAR Methods.

(A) The SQTR-OR controller architecture where the inhibitory action of the protein p_q in the UBER-OR controller is replaced with a sRNA-mediated sequestration.

(B) The OP-OR controller architecture where the inhibitory action of the protein P_q in the UBER-OR controller is removed.

(C) The two-dimensional score of the best 200 controllers for the controller architectures based on UBER-OR (namely SQTR-OR and OP-OR).

(D) Two-dimensional score plotted again the final protein concentration for SQTR-OR and OP-OR.

(E) The dynamic change in the mRNA m_1 concentration (ΔT_X) in response to the induction of gene 2 of the best design of the SQTR-OR and OP-OR architectures. The dynamics are normalized by their pre-disturbance levels. The function of the UBER-OR is shown a black dotted line.

(F) The dynamic change in the protein p_1 concentration (ΔT_L) in response to the induction of gene 2 of the best design of the SQTR-OR and OP-OR architectures. The dynamics are normalized by their pre-disturbance levels. The function of the UBER-OR is shown a black dotted line.

(G) The topology of the var-FRAG-OR controller where the RNA polymerase transcribes the mRNAs of its own subunits.

(H) The dynamics of var-FRAG-OR. The change in the mRNA m_1 concentration (ΔT_X) in response to the induction of gene 2 of the best design of the FRAG-OR and var-FRAG-OR architectures. The dynamics are normalized by their pre-disturbance levels. The Inset shows the final mRNA concentration (nM). The corresponding protein dynamics for protein p_1 are shown in the lower panel. The dynamics are normalized by their pre-disturbance levels. The Inset shows the final protein concentration (nM).

(I) The two-dimensional score of the best 200 var-FRAG-OR controllers (note that FRAG-OR are also shown in gray for ease of comparison).

(J) Two-dimensional scores achieved by the var-FRAG-OR controllers plotted again the final protein concentration.

Please cite this article in press as: Darlington and Bates, Architectures for Combined Transcriptional and Translational Resource Allocation Controllers, Cell Systems (2020), https://doi.org/10.1016/j.cels.2020.08.014

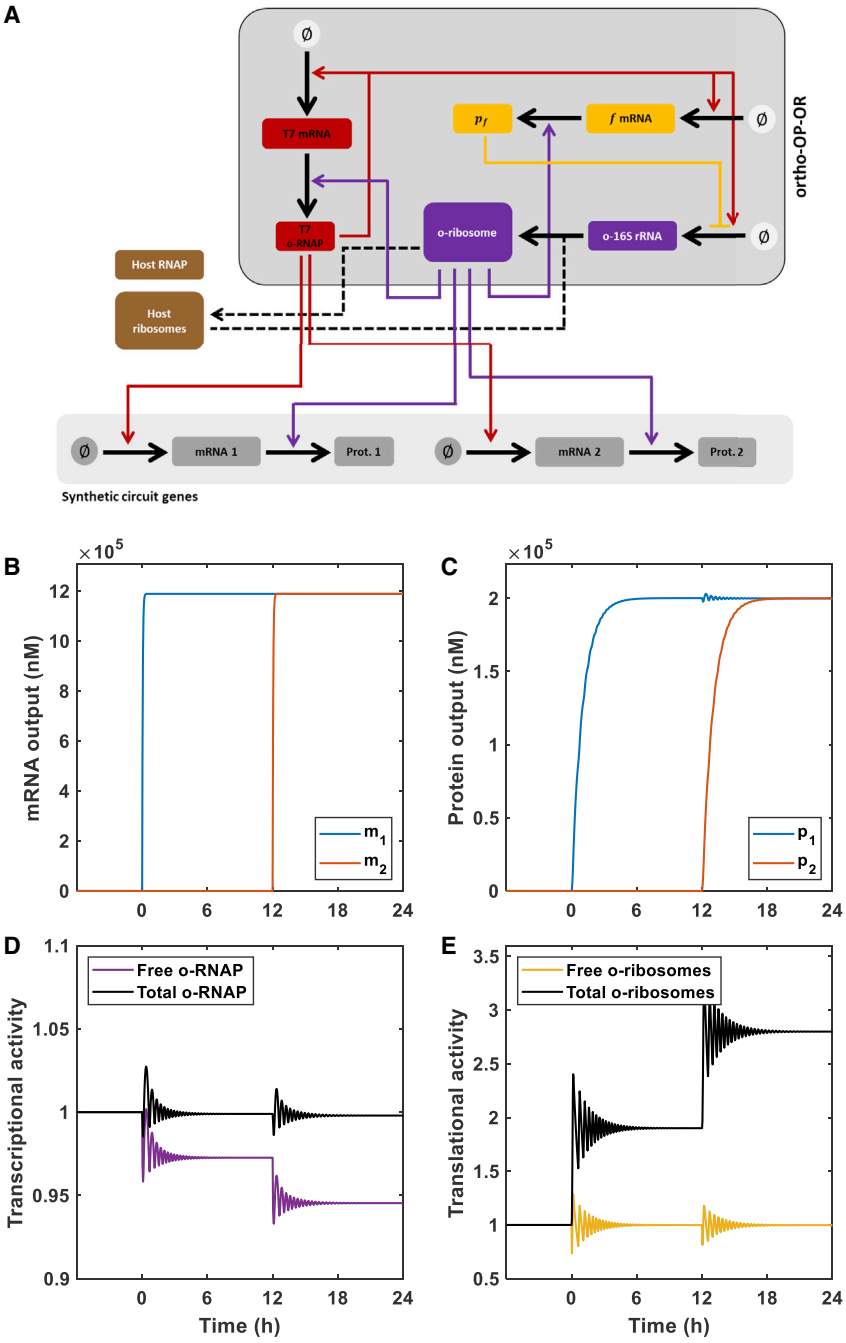


Figure 4. An Entirely Orthogonal Control System Achieves Superior Performance and Robustness

Various potential biological implements of the controller architecture where simulated as described in the STAR Methods. The performance of the controllers was assessed by simulating the production of two genes. The induction of the second gene (g_2) at 12 h results in a transcriptional disturbance (as promoters compete for RNA polymerase) and a translational disturbance (as mRNAs compete for ribosomes) to the first gene g_1.

(A) Controller architecture. The controller is composed of an orthogonal RNA polymerases and orthogonal ribosomes. The activity of the controller is set by the controller protein p_f. The controller utilizes only orthogonal gene expression machinery for its own expression.

(B) The changing mRNA concentration of the two circuit genes over time.

(C) The changing protein concentration of the two circuit genes over time.

(D) The changing levels of the transcriptional activity shown in terms of both free and total orthogonal RNA polymerase (o-RNAP) concentration, normalized by their values before circuit gene induction at 0 h.

(E) The changing levels of the translational activity shown in terms of both free and total orthogonal ribosomes (o-ribosomes) concentration, normalized by their values before circuit gene induction at 0 h.

gene expression machineries with their own homeostatic-like control systems. Note that, as discussed previously, the orthogonal ribosomes still require host ribosomal proteins; so, this interaction between host and circuit is unavoidable—in much the same way that these synthetic gene circuits still utilize host metabolites.

Our analysis shows that a simple controller, based only on o-RNAP, o-ribosomes, and a single controller protein is sufficient to achieve a near-perfect decoupling at the transcriptional and translational levels (Figure 4A). This ortho-OP-OR controller utilizes only orthogonal resources for its own expression. The orthogonal transcriptional activity is again made up by a whole T7 RNA polymerase, which transcribes its own mRNA, the mRNA of the controller protein, and the synthetic ribosomal RNA. The orthogonal ribosomes formed translate the T7 RNA polymerase mRNA and the controller protein. The controller protein inhibits rRNA production and acts a global regulator, setting the orthogonal ribosome level and therefore—indirectly—setting the transcriptional activity.

We designed a number of biological implementations. A number of these implementations showed a near-perfect transcriptional and translation decoupling (Figure S14). We show the performance of the best performing controller in Figure 4. The controller successfully decouples mRNAs with no steady-state error at the transcriptional level (Figure 4B). At the translational level, the controller shows high levels of expression (greater than that seen in FRAG-OR) and an improved dynamic performance with a small overshoot and shorter settling time (compared with UBER-OR). Analyzing the internal dynamics of the controller species, we see how this topology functions to decouple co-expressed genes. At the transcriptional level, the orthogonal RNA polymerase is expressed at a sufficient level to reduce coupling. The combined controller acts to maintain the total orthogonal transcriptional activity in response to the

Please cite this article in press as: Darlington and Bates, Architectures for Combined Transcriptional and Translational Resource Allocation Controllers, Cell Systems (2020), https://doi.org/10.1016/j.cels.2020.08.014

disturbances caused by the circuit gene expression (Figure 4D). At the translational level, the controller co-opts sufficient ribosomes from the host pool to maintain free orthogonal ribosome levels; this effectively insulates each gene of the circuit from the others, as a change in one gene expression level does not change the free ribosome level and so the genes are effectively uncoupled.

Analyzing the best decoupling (i.e., lowest 2D score) designs for these ortho-OP-OR shows that the controller should be implemented on a high copy plasmid, which carries all three genes (Figure S15). The orthogonal RNA polymerase gene requires a strong ribosome binding site (low $\beta_{f,f}$) with the controller protein having a medium ribosome binding site. The master regulator (p_f) needs to take the form of a multimeric strongly binding repressor (here lacI). Alternatively, the three genes can be carried on a medium copy plasmid but not a low copy plasmid. This design has a slightly worse decoupling performance (as measured by the 2D score), but this may be an acceptable trade-off in some settings if high copy plasmids impact growth.

To determine how each parameter impacts the decoupling performance of the controller, and if any two parameters interact, we carried out a two-dimensional sensitivity analysis varying each pair of parameters in turn (Figure S16). This identified the gene copy numbers and ribosome binding site strengths as key interacting parameters. Other parameters, such as the transcription factor dissociation constant and promoter strengths, showed the same impact when varied with other parameters. We found that ω_f and $\beta_{f,f}$ parameters were coupled, and that increasing one can allow for decreases in another, and the level of coupling can be maintained. Varying $\beta_{f,p}$ and $\beta_{f,f}$ demonstrates that translation of the RNA polymerase and controller need to be balanced, with coupling increasing as the $\beta_{f,p}/\beta_{f,f}$ ratio increases. Similarly, if ω_f is small (i.e., less than high copy number) then increasing $\beta_{f,p}$ increases gene coupling. ω_r and $\beta_{f,f}$ also require balancing with a high $\omega_r/\beta_{f,f}$ ratio, also leading to an increased coupling. We also find coupling increases as both ω_r and $\beta_{f,p}$ increases; i.e., increasing the orthogonal ribosome gene copy number (and hence the production rate) while increasing the orthogonal RNA polymerase ribosome binding site (and hence translation rate) also increases coupling between circuit genes.

As described previously, we assessed the robustness of the controller to parametric uncertainty by simulating the impact of perturbations to key controller parameters. We assessed uncertainties in 10% increments from 10% to 90% with 1,000 different sample perturbations and failed to find a single destabilizing parameter set, indicating that this controller is highly robust and that any uncertainties in the dynamics of implemented components will not lead to the emergence of instabilities (Figure S17).

DISCUSSION

Competition for shared cellular resources results in the emergence of non-regulatory interactions between circuit genes. By exploiting recent advances in the creation of orthogonal biological components, RNA polymerase allocation controllers can be used to decouple co-expressed genes at the transcriptional level, and ribosomal allocation controllers can be used to decouple co-expressed genes at the translational level. Here, we investigated the hypothesis that by combining these control systems, we could decouple genes at both the transcriptional and translational levels simultaneously. We found that simply combining separately designed (and functioning) transcriptional and translational controllers can result in an unstable or non-functional dual controller, e.g., implementing the universal bacterial expression resource transcriptional controller with the orthogonal ribosome controller produced sustained oscillations in circuit genes, while combining a translational controller with a fragment RNA polymerase controller abolishes the translational decoupling. From a control engineering viewpoint, this is not unexpected, as it is well known that combining several (separately stable) feedback loops into a single multivariable control system can readily produce instability or significant performance degradation due to interactions between the different controllers (Bates and Postlethwaite, 2002).

The standard approach used by control engineers to deal with this problem is to design and analyze all elements of the overall control system simultaneously using multivariable methods (Cosentino and Bates, 2011), and this is the approach we adopted here, evaluating a number of potential designs based on two alternative (biologically feasible) controller architectures. The controllers were designed assuming that all system parameter values are precisely known. However, in reality, biological parameters are rarely accurately known. Biological measurements are reported within possible error ranges, and, even where these are small, introducing parts into new contexts, either DNA sequence or host genetic background, can cause unpredictable changes in component dynamics. These uncertainties can introduce instabilities such as oscillations in circuit genes. We carried out a rigorous robustness analysis of our nonlinear controllers using standard Monte Carlo sampling techniques and identified that the UBER-OR architecture has poor robustness—becoming unstable if multiple parameters are not precisely tuned. Therefore, we explored potential improvements to the architecture of the controller to improve robustness. We found that replacing protein-based feedback with an sRNA-sequestration motif (i.e., the SQTR-OR controller) or removing the transcriptional feedback entirely (i.e., OP-OR controller) allows the creation of controllers with similar performance to the UBER-OR controller but with significantly improved robustness. While the FRAG-OR controller showed better robustness than UBER-OR, it showed poorer performance and so we tested modifications to the FRAG-OR architecture (i.e., the var-FRAG-OR controller), which improved the performance but unfortunately reduced the robustness.

Previous experimental studies have shown that high levels of orthogonal components may be toxic to cells (e.g., Tan et al. (2009)). We selected our simulation conditions to account for this effect. Our modeling assumes a specific growth rate of 1 per h. Using this growth rate, we set host resource levels, transcription and translation, and component dilution rates (see STAR Methods and citations therein). Our specific control systems are parameterized using biologically reasonable values. Therefore, the levels of components produced within the model within are biologically reasonable bounds. The selected growth rate (and therefore simulation conditions) are similar to those observed in experimental settings where no toxicity effects are identified (Kushwaha and Salis, 2015). Our designs operate in a similar non-toxic range, and therefore, should be biologically feasible.

Motivated by recent progress in developing orthogonal cellular "virtual machines," (Liu et al., 2018; Costello and Badran, 2020), we expanded our analysis to consider the design of a "orthogonal only" controller; i.e., one which does not use host resources for its own expression. This controller is composed of an orthogonal RNA polymerase and an orthogonal ribosome and a single controller protein. All genes within the controller, as well as the circuit, use the orthogonal pools for their expression. We showed that this topology is capable of a near-perfect decoupling within biologically attainable parameter regimes. Our parametric analysis shows that the three controller genes should be carried on a single high copy plasmid and that the controller protein should take the form of a strongly binding multimeric repressor (here lacI). We found that the ribosome binding sites of the RNA polymerase and controller protein need to be finely balanced with the best designs given when the RNA polymerase RBS is stronger than the controller protein. Finally, a Monte Carlo analysis shows that this controller is robust to high levels of parametric uncertainty, making it a viable candidate for future experimental implementation.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- RESOURCE AVAILABILITY
  - Lead Contact
  - Materials Availability
  - Data and Code Availability
- METHOD DETAILS
  - Circuit Model with Competition for Host Cellular Resources
  - UBER Controller Model
  - FRAG Controller Model
  - Translational Resource Allocation Controller Model
  - UBER-OR and FRAG-OR Dual Transcriptional-Translational Controller Models
  - SQTR-OR Dual Transcriptional-translational Controller Model
  - OP-OR Dual Transcriptional-Translational Controller Model
  - var-FRAG-OR Dual Transcriptional-Translational Controller Model
  - An All-othogonal Ortho-OP-OR Dual Transcriptional-translational Controller Model
  - Experimentally Implementable Parameters
  - Controller Design Process
  - Computational Methods

SUPPLEMENTAL INFORMATION

Supplemental Information can be found online at https://doi.org/10.1016/j.cels.2020.08.014.

ACKNOWLEDGMENTS

A.P.S.D. and D.G.B. acknowledge funding from the Leverhulme Trust (grant RPG-2017-284) and the high performance computing facilities provided by Warwick Integrative Synthetic Biology Centre (BBSRC-EPSRC grant BB/M017982/1).

AUTHOR CONTRIBUTIONS

Conceptualization, Methodology, Formal Analysis, and Writing – Original Draft, A.P.S.D.; Supervision and Funding acquisition, D.G.B.; Writing – Review & Editing, A.P.S.D., and D.G.B.

DECLARATION OF INTERESTS

The authors declare no competing interests.

Received: February 13, 2020
Revised: July 23, 2020
Accepted: August 21, 2020
Published: September 15, 2020

REFERENCES

An, W., and Chin, J.W. (2009). Synthesis of orthogonal transcription- translation networks. Proc. Natl. Acad. Sci. USA *106*, 8477–8482.

Atkinson, M.R., Savageau, M.A., Myers, J.T., and Ninfa, A.J. (2003). Development of genetic circuitry exhibiting toggle switch or oscillatory behavior in Escherichia coli. Cell *113*, 597–607.

Bates, D., and Postlethwaite, I. (2002). Robust Multivariable Control of Aerospace Systems (IOS Press).

Borkowski, O., Ceroni, F., Stan, G.B., and Ellis, T. (2016). Overloaded and stressed : whole-cell considerations for bacterial synthetic biology. Curr. Opin. Microbiol. *33*, 123–130.

Brophy, J.A., and Voigt, C.A. (2014). Principles of genetic circuit design. Nat. Methods *11*, 508–520.

Carbonell-Ballestero, M., Garcia-Ramallo, E., Montañez, R., Rodriguez-Caso, C., and Macía, J. (2016). Dealing with the genetic load in bacterial synthetic biology circuits : convergences with the Ohm's law. Nucleic Acids Res *44*, 496–507.

Cardinale, S., and Arkin, A.P. (2012). Contextualizing context for synthetic biology - identifying causes of failure of synthetic biological systems. Biotechnol. J. *7*, 856–866.

Ceroni, F., Algar, R., Stan, G.B., and Ellis, T. (2015). Quantifying cellular capacity identifies gene expression designs with reduced burden. Nat. Methods *12*, 415–418.

Cosentino, C., and Bates, D. (2011). Feedback Control in Systems Biology (CRC Press).

Costello, A., and Badran, A.H. (2020). Synthetic biological circuits within an orthogonal central dogma. Trends Biotechnol. https://doi.org/10.1016/j.tibtech.2020.05.013.

Darlington, A.P.S., Kim, J., Jiménez, J.I., and Bates, D.G. (2018a). Dynamic allocation of orthogonal ribosomes facilitates uncoupling of co-expressed genes. Nat. Commun. *9*, 695.

Darlington, A.P.S., Kim, J., Jiménez, J.I., and Bates, D.G. (2018b). Engineering translational resource allocation controllers: mechanistic models, design guidelines, and potential biological implementations. ACS Synth. Biol. *7*, 2485–2496.

Gorochowski, T.E., Avcilar-Kucukgoze, I., Bovenberg, R.A., Roubos, J.A., and Ignatova, Z. (2016). A minimal model of ribosome allocation dynamics captures trade-offs in expression between endogenous and synthetic genes. ACS Synth. Biol. *5*, 710–720.

Grunberg, T.W., and Del Vecchio, D. (2020). Modular analysis and design of biological circuits. Curr. Opin. Biotechnol. *63*, 41–47.

Gyorgy, A., Jiménez, J.I., Yazbek, J., Huang, H.H., Chung, H., Weiss, R., and Del Vecchio, D. (2015). Isocost lines describe the cellular economy of genetic circuits. Biophys. J. *109*, 639–646.

Hancock, E.J., Stan, G.-B.V., Arpino, J.A.J., and Papachristodoulou, A. (2015). Simplified mechanistic models of gene regulation for analysis and design. J. R. Soc. Interface *12*, 20150312.

Jayanthi, S., and Del Vecchio, D. (2012). Tuning genetic clocks employing DNA binding sites. PLoS One *7*, e41019.

Kushwaha, M., and Salis, H.M. (2015). A portable expression resource for engineering cross-species genetic circuits and pathways. Nat. Commun. *6*, 7832.

Liu, C.C., Jewett, M.C., Chin, J.W., and Voigt, C.A. (2018). Toward an orthogonal central dogma. Nat. Chem. Biol. *14*, 103–106.

Moser, F., Broers, N.J., Hartmans, S., Tamsir, A., Kerkman, R., Roubos, J.A., Bovenberg, R., and Voigt, C.A. (2012). Genetic circuit performance under conditions relevant for industrial bioreactors. ACS Synth. Biol. *1*, 555–564.

Qian, Y., Huang, H.H., Jiménez, J.I., and Del Vecchio, D. (2017). Resource competition shapes the response of genetic circuits. ACS Synth. Biol. *6*, 1263–1272.

Segall-Shapiro, T.H., Meyer, A.J., Ellington, A.D., Sontag, E.D., and Voigt, C.A. (2014). 'A 'resource allocator' for transcription based on a highly fragmented T7 RNA polymerase. Mol. Syst. Biol. *10*, 742.

Shin, J., Zhang, S., Der, B.S., Nielsen, A.A., and Voigt, C.A. (2020). Programming Escherichia coli to function as a digital display. Mol. Syst. Biol. *16*, e9401.

Tan, C., Marguet, P., and You, L. (2009). Emergent bistability by a growth-modulating positive feedback circuit. Nat. Chem. Biol. *5*, 842–848.

Tsigkinopoulou, A., Baker, S.M., and Breitling, R. (2017). Respectful modeling: addressing uncertainty in dynamic system models for molecular biology. Trends Biotechnol *35*, 518–529.

Vilanova, C., Tanner, K., Dorado-Morales, P., Villaescusa, P., Chugani, D., Frías, A., Segredo, E., Molero, X., Fritschi, M., Morales, L., et al. (2015). Standards not that standard. J. Biol. Eng. *9*, 17.

Zhang, C., Tsoi, R., and You, L. (2016). Addressing biological uncertainties in engineering gene circuits. Integr. Biol. (Camb) *8*, 456–464.

Please cite this article in press as: Darlington and Bates, Architectures for Combined Transcriptional and Translational Resource Allocation Controllers, Cell Systems (2020), https://doi.org/10.1016/j.cels.2020.08.014

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Software and Algorithms		
MATLAB 2019a	The Math Works, Inc.	https://www.mathworks.com
Parallel Computing Toolbox Ver. 7.0	The Math Works, Inc.	https://www.mathworks.com
ODE models etc.	This work	https://github.com/apsduk/cell-systems-2020

RESOURCE AVAILABILITY

Lead Contact

Further information and requests for resources and reagents should be directed to and will be fulfilled by the Lead Contact, Declan Bates d.bates@warwick.ac.uk.

Materials Availability

This study did not generate new materials.

Data and Code Availability

Original code is freely available for download at https://github.com/apsduk/cell-systems-2020 or archived with the following https://doi.org/10.5281/zenodo.3981244

METHOD DETAILS

Circuit Model with Competition for Host Cellular Resources

The core of the process model represents transcription and translation of a single unregulated gene and takes account of competition for cellular resources. Each circuit gene's promoter, g_i, is reversibly bound by the host's RNA polymerase, P_h, to produce the transcription complex, x_i. This produces the mRNA, m_i, at rate τ_i. Completion of transcription also liberates the free promoter and RNA polymerase:

$$g_i + P_h \underset{\xi_{r,i}}{\overset{\xi_{f,i}}{\rightleftharpoons}} x_i \xrightarrow{\tau_i} g_i + P_h + m_i \qquad \text{(Equation 1)}$$

The mRNA reversibly binds to host ribosomes, R_h, to produce the intermediate translational complex, c_i. This produces protein, p_i, at rate γ_i. Completion of translation also liberates the mRNA and ribosome:

$$m_i + R_h \underset{\beta_{r,i}}{\overset{\beta_{f,i}}{\rightleftharpoons}} c_i \xrightarrow{\gamma_i} m_i + R_h + p_i \qquad \text{(Equation 2)}$$

We assume that all species dilute due to growth while mRNAs also decay Hancock et al. (2015):

$$x_i \xrightarrow{\lambda} \varnothing$$

$$m_i \xrightarrow{\delta_{m,i}+\lambda} \varnothing \qquad \text{(Equation 3)}$$

$$c_i \xrightarrow{\lambda} \varnothing$$

$$p_i \xrightarrow{\lambda} \varnothing$$

By applying the Law of Mass Action we can derive the following dynamics:

Please cite this article in press as: Darlington and Bates, Architectures for Combined Transcriptional and Translational Resource Allocation Controllers, Cell Systems (2020), https://doi.org/10.1016/j.cels.2020.08.014

$$\dot{x_i} = \xi_{f,i} \cdot g_i \cdot P_h - (\xi_{r,i} + \tau_i) \cdot x_i - \lambda \cdot x_i \quad \text{(Equation 4)}$$

$$\dot{m_i} = \tau_i \cdot x_i - \beta_{f,i} \cdot m_i \cdot R_h + (\beta_{r,i} + \gamma_i) \cdot c_i - (\delta_{m,i} + \lambda) \cdot m_i \quad \text{(Equation 5)}$$

$$\dot{c_i} = \beta_{f,i} \cdot m_i \cdot R_h - (\beta_{r,i} + \gamma_i) \cdot c_i - \lambda \cdot c_i \quad \text{(Equation 6)}$$

$$\dot{p_i} = \gamma_i \cdot c_i - \lambda \cdot p_i \quad \text{(Equation 7)}$$

Due to their own control mechanisms the copy number of each plasmid and therefore gene $g_i$remains constant such that a conservation law can be applied. The total number of promoters for each gene is constant $\omega_i = g_i + x_i$. From this we calculate the concentration of the free promoter:

$$g_i = \omega_i - x_i \quad \text{(Equation 8)}$$

Similarly, the host's internal control mechanisms maintain the total number of RNA polymerases and ribosomes for any given growth rate and as such we can calculate the concentration of free resources:

$$P_h = P_{h,T} - \sum_{i=1:N} (x_i) \quad \text{(Equation 9)}$$

$$R_h = R_{h,T} - \sum_{i=1:N} (c_i) \quad \text{(Equation 10)}$$

where $P_{h,T}$ and $R_{h,T}$ are the total concentrations of the host's RNA polymerase and ribosomes respectively.

Induction of each circuit module can be simulated by varying the concentration of each promoter ω_i. Note that this allows us to neglect the regulation of each circuit promoter and so simplify the model.

UBER Controller Model

The universal bacterial expression resource developed by Kushwaha and Salis utilises an orthogonal RNA ploymerase, P_o, as the new circuit specific transcriptional resource. The RNA polymerase transcribes itself and a second repressor protein, p_q. A number η_q of molecules of p_q bind to the o-RNAP promoter, g_p, to produce the sequestered, and therefore transcriptionally inhibited, complex k_p:

$$g_p + \eta_q \cdot p_q \underset{\alpha_{r,q}}{\overset{\alpha_{f,q}}{\rightleftharpoons}} k_p \quad \text{(Equation 11)}$$

This species dilutes due to growth as in Hancock et al. (2015):

$$k_p \xrightarrow{\lambda} \varnothing \quad \text{(Equation 12)}$$

We assume that the production of the orthogonal RNA polymerase is leaky:

$$\varnothing \xrightarrow{\gamma_{p,0}} P_o \quad \text{(Equation 13)}$$

The transcription and translation of P_o and p_q mirror the chemical reaction networks described in "Circuit model with competition for host cellular resources". Applying the Law of Mass Action yields the following dynamics for the production of the o-RNAP:

$$\dot{k_p} = \alpha_{f,q} \cdot p_q^{\eta_q} - \alpha_{r,q} \cdot k_p - \lambda \cdot k_p \quad \text{(Equation 14)}$$

$$\dot{x_p} = \xi_{f,p} \cdot g_p \cdot P_o - (\xi_{r,p} + \tau_p) \cdot x_p - \lambda \cdot x_p \quad \text{(Equation 15)}$$

$$\dot{m_p} = \tau_p \cdot x_p - \beta_{f,p} \cdot m_p \cdot R_h + (\beta_{r,p} + \gamma_p) \cdot c_p - (\delta_{m,p} + \lambda) \cdot m_p \quad \text{(Equation 16)}$$

$$\dot{c_p} = \beta_{f,p} \cdot m_p \cdot R_h - (\beta_{r,p} + \gamma_p) \cdot c_p - \lambda \cdot c_p \quad \text{(Equation 17)}$$

The dynamics of the orthogonal RNA polymerase which transcribes circuit genes are given by:

Please cite this article in press as: Darlington and Bates, Architectures for Combined Transcriptional and Translational Resource Allocation Controllers, Cell Systems (2020), https://doi.org/10.1016/j.cels.2020.08.014

$$\dot{P_o} = \gamma_{p,0} + \gamma_p \cdot c_p - \lambda \cdot P_o \ldots \quad \text{(Equation 18)}$$

$$\ldots - \xi_{f,p} \cdot g_p \cdot P_o + (\xi_{r,p} + \tau_p) \cdot x_p \ldots$$

$$\ldots - \xi_{f,q} \cdot g_q \cdot P_o + (\xi_{r,q} + \tau_q) \cdot x_q \ldots$$

$$\ldots + \sum_{i=1:N} (- \xi_{f,i} \cdot g_i \cdot P_o + (\xi_{r,i} + \tau_i) \cdot x_i)$$

The production of the controller protein, p_q, follows the dynamics for circuit proteins in (4) to (6) except that the host RNAP, P_h, is replaced with the o-RNAP, P_o. The translational machinery remains the host ribosome, R_h:

$$\dot{x_q} = \xi_{f,q} \cdot g_q \cdot P_o - (\xi_{r,q} + \tau_q) \cdot x_q - \lambda \cdot x_q \quad \text{(Equation 19)}$$

$$\dot{m_q} = \tau_q \cdot x_q - \beta_{f,q} \cdot m_p \cdot R_h + (\beta_{r,q} + \gamma_q) \cdot c_q - (\delta_{m,q} + \lambda) \cdot m_q \quad \text{(Equation 20)}$$

$$\dot{c_q} = \beta_{f,q} \cdot m_q \cdot R_h - (\beta_{r,q} + \gamma_q) \cdot c_q - \lambda \cdot c_q \quad \text{(Equation 21)}$$

The dynamics of the final protein, including the g_p-p_q repression interaction, are given by:

$$\dot{p_q} = \gamma_q \cdot c_q - \lambda \cdot p_q - \eta_q \cdot \alpha_{f,q} \cdot p_q^{\eta_q} - \eta_q \cdot \alpha_{r,q} \cdot k_p \quad \text{(Equation 22)}$$

The concentration of free promoters of the orthogonal RNA polymerase and its regulator p_q can be calculated from their respective total ω_p and ω_q:

$$g_p = \omega_p - x_p - k_p \quad \text{(Equation 23)}$$

$$g_q = \omega_q - x_q \quad \text{(Equation 24)}$$

This controller utilises its own RNA polymerase for its own expression and for circuit gene transcription, therefore the host RNA polymerase is not utilised:

$$P_h = P_{h,T} \quad \text{(Equation 25)}$$

The controller utilises the host translation system for its own expression and expression of circuit genes:

$$R_h = R_{h,T} - c_p - c_q - \sum_{i=1:N} (c_i) \quad \text{(Equation 26)}$$

FRAG Controller Model

The fragmented RNA polymerase controller (FRAG) developed by Segall-Shapiro et al. utilises a RNA polymerase which is split into two components, rather than being made of up a single protein. The core β fragment, p_c, is constitutively expressed and sets a 'transcriptional budget' which is then targeted by the expression of a 'synthetic' σ factor, p_a. These two components bind to create the functional circuit-specific orthogonal RNA polymerase:

$$p_a + p_c \underset{\theta_r}{\overset{\theta_f}{\rightleftharpoons}} P_o \quad \text{(Equation 27)}$$

Again, we assume that the production of the orthogonal RNA polymerase is leaky:

$$\varnothing \xrightarrow{\gamma_{p,0}} P_o \quad \text{(Equation 28)}$$

The production of each protein component follows the same dynamics as for circuit proteins. Applying the Law of Mass Action, including the production of RNAP in Equation 27, gives:

$$\dot{x_c} = \xi_{f,c} \cdot g_c \cdot P_h - (\xi_{r,c} + \tau_c) \cdot x_c - \lambda \cdot x_c \quad \text{(Equation 29)}$$

$$\dot{m_c} = \tau_c \cdot x_c - \beta_{f,c} \cdot m_c \cdot R_h + (\beta_{r,c} + \gamma_c) \cdot c_c - (\delta_{m,c} + \lambda) \cdot m_c \quad \text{(Equation 30)}$$

Please cite this article in press as: Darlington and Bates, Architectures for Combined Transcriptional and Translational Resource Allocation Controllers, Cell Systems (2020), https://doi.org/10.1016/j.cels.2020.08.014

$$\dot{c_c} = \beta_{f,c} \cdot m_c \cdot R_h - (\beta_{r,c} + \gamma_c) \cdot c_c - \lambda \cdot c_c \quad \text{(Equation 31)}$$

$$\dot{p_c} = \gamma_c \cdot p_a \cdot p_c - \lambda \cdot p_c - \theta_f \cdot p_a \cdot p_c + \theta_r \cdot P_o \quad \text{(Equation 32)}$$

$$\dot{x_a} = \xi_{f,a} \cdot g_a \cdot P_h - (\xi_{r,a} + \tau_a) \cdot x_a - \lambda \cdot x_a \quad \text{(Equation 33)}$$

$$\dot{m_a} = \tau_a \cdot x_a - \beta_{f,a} \cdot m_a \cdot R_h + (\beta_{r,a} + \gamma_a) \cdot c_a - (\delta_{m,a} + \lambda) \cdot m_a \quad \text{(Equation 34)}$$

$$\dot{c_a} = \beta_{f,a} \cdot m_a \cdot R_h - (\beta_{r,a} + \gamma_a) \cdot c_a - \lambda \cdot c_a \quad \text{(Equation 35)}$$

$$\dot{p_a} = \gamma_a \cdot c_a - \lambda \cdot p_a - \theta_f \cdot p_a \cdot p_c + \theta_r \cdot P_o \quad \text{(Equation 36)}$$

The dynamics of the orthogonal RNA polymerase which transcribes circuit genes are given by:

$$\dot{P_o} = \gamma_{p,0} + \theta_f \cdot p_a \cdot p_c - \theta_r \cdot P_o - \lambda \cdot P_o + \sum_{i=1:N} \left(- \xi_{f,i} \cdot g_i \cdot P_o + (\xi_{r,i} + \tau_i) \cdot x_i \right) \quad \text{(Equation 37)}$$

The concentration of free promoters of the core RNA polymerase and its targeting fragment can be calculated from their respective total ω_c and ω_a:

$$g_c = \omega_c - x_c \quad \text{(Equation 38)}$$

$$g_a = \omega_a - x_a \quad \text{(Equation 39)}$$

As described in Segall-Shapiro et al. (2014), the promoters of the targeting fragment of the orthogonal split RNA polymerase are carried on the circuit plasmids and therefore their total copy number is the sum of the circuit promoters:

$$\omega_a = \sum_{i=1:N} (\omega_i) \quad \text{(Equation 40)}$$

This controller utilises its own RNA polymerase for circuit gene expression but utilises the host's for its own production such that the free host RNA polymerase is given by:

$$P_h = P_{h,T} - x_c - x_a \quad \text{(Equation 41)}$$

The controller utilises the host translation system for its own expression and expression of circuit genes:

$$R_h = R_{h,T} - c_a - c_c - \sum_{i=1:N} (c_i) \quad \text{(Equation 42)}$$

Translational Resource Allocation Controller Model

We model the conversion of ribosomes between the host and orthogonal ribosome pool by considering the one step reaction between an orthogonal rRNA, r, and the host ribosome, R_h, to produce orthogonal ribosomes, R_o:

$$r + R_h \underset{\varrho_r}{\overset{\varrho_f}{\rightleftharpoons}} R_o \quad \text{(Equation 43)}$$

The rRNA and orthogonal ribosome are also subject to dilution:

$$r \xrightarrow{\delta_r + \lambda} \varnothing \qquad R_o \xrightarrow{\lambda} \varnothing \quad \text{(Equation 44)}$$

In Darlington et al. (2018a), we developed a translational controller which dynamically allocates the distribution between host and orthogonal ribosomes by placing the o-rRNA gene, g_r, under the control of a protein, p_f, which itself is translated by the orthogonal ribosome pool. A number η_f of controller p_f proteins sequesters free rRNA promoters to an inactive complex, k_r:

$$g_r + \eta_f \cdot p_f \underset{\alpha_{r,f}}{\overset{\alpha_{f,f}}{\rightleftharpoons}} k_r \quad \text{(Equation 45)}$$

Please cite this article in press as: Darlington and Bates, Architectures for Combined Transcriptional and Translational Resource Allocation Controllers, Cell Systems (2020), https://doi.org/10.1016/j.cels.2020.08.014

As with all other species, this dilutes due to growth as in Hancock et al. (2015):

$$k_r \xrightarrow{\lambda} \varnothing \quad \text{(Equation 46)}$$

Applying the Law of Mass Action yields the following dynamics:

$$\dot{k_r} = \alpha_{f,f} \cdot g_r \cdot p_f^{\eta_f} - \alpha_{r,f} \cdot k_r - \lambda \cdot k_r \quad \text{(Equation 47)}$$

$$\dot{x_r} = \xi_{f,r} \cdot g_r \cdot P_h - (\xi_{r,r} + \tau_r) \cdot x_r - \lambda \cdot x_r \quad \text{(Equation 48)}$$

$$\dot{r} = \tau_r \cdot x_r - \varrho_f \cdot r \cdot R_h + \varrho_r \cdot R_o - (\delta_r + \lambda) \cdot r \quad \text{(Equation 49)}$$

The dynamics of the orthogonal ribosome pool are given by:

$$\dot{R_o} = \varrho_f \cdot r \cdot R_h - \varrho_r \cdot R_o - \lambda \cdot R_o \ldots \quad \text{(Equation 50)}$$

$$\ldots - \beta_{f,f} \cdot m_f \cdot R_o + (\beta_{r,f} + \gamma_f) \cdot c_f \ldots$$

$$\ldots + \sum_{i=1:N} \left(- \beta_{f,i} \cdot m_i \cdot R_o + (\beta_{r,i} + \gamma_i) \cdot c_i \right)$$

The production of the controller protein, p_f, follows the same dynamics of circuit proteins, except the host ribosome, R_h, is replaced with its orthogonal counterpart, R_o:

$$\dot{x_f} = \xi_{f,f} \cdot g_f \cdot P_h - (\xi_{r,f} + \tau_f) \cdot x_f - \lambda \cdot x_f \quad \text{(Equation 51)}$$

$$\dot{m_f} = \tau_f \cdot x_f - \beta_{f,f} \cdot m_f \cdot R_o + (\beta_{r,f} + \gamma_f) \cdot c_f - (\delta_{m,f} + \lambda) \cdot m_f \quad \text{(Equation 52)}$$

$$\dot{c_f} = \beta_{f,f} \cdot m_f \cdot R_o - (\beta_{r,f} + \gamma_f) \cdot c_f - \lambda \cdot c_f \quad \text{(Equation 53)}$$

The dynamics of the inhibitory protein are:

$$\dot{p_f} = \gamma_f \cdot c_f - \lambda \cdot p_f - \eta_f \cdot \alpha_{f,f} \cdot p_f^{\eta_f} - \eta_f \cdot \alpha_{r,f} \cdot k_r \quad \text{(Equation 54)}$$

The concentration of free promoters of the o-rRNA and controller protein can be calculated from their respective total $\omega_r$and ω_f:

$$g_r = \omega_r - x_r - k_r \quad \text{(Equation 55)}$$

$$g_f = \omega_f - x_f \quad \text{(Equation 56)}$$

This controller utilises the host RNA polymerase for all transcription:

$$P_h = P_{h,T} - x_r - x_f - \sum_{i=1:N} (x_i) \quad \text{(Equation 57)}$$

As described above the controller co-opts ribosomes from the host such that the free host ribosome pool is given by:

$$R_h = R_{h,T} - R_o - c_f - \sum_{i=1:N} (c_i) \quad \text{(Equation 58)}$$

UBER-OR and FRAG-OR Dual Transcriptional-Translational Controller Models

We combine the two transcriptional controllers with the translational controller as follows. In each case the host cellular resources in circuit (Equations 4, 5, 6, and 7) are replaced with their orthogonal counterparts ($P_h \Rightarrow P_o$ and $R_h \Rightarrow R_o$ respectively). We also update the equations describing the use of host resources.

Combining the UBER controller with the translational controller results in the following host resource usage:

$$P_h = P_{h,T} - x_r - x_f \quad \text{(Equation 59)}$$

$$R_h = R_{h,T} - R_o - c_p - c_q - c_f - \sum_{i=1:N}(c_i) \quad \text{(Equation 60)}$$

The designable parameters of this controller are: ω_p, $\xi_{f,p}$, $\beta_{f,p}$, ω_r, $\xi_{f,r}$, ω_f, $\xi_{f,q}$, $\beta_{f,q}$, ω_q, $\xi_{f,f}$, $\beta_{f,f}$, $\alpha_{r,q}$, η_q, $\alpha_{r,f}$, η_f.

Combining the FRAG controller with the translational controller results in the following host resource usage:

$$P_h = P_{h,T} - x_r - x_f - x_c - x_a \quad \text{(Equation 61)}$$

$$R_h = R_{h,T} - R_o - c_c - c_a - c_f - \sum_{i=1:N}(c_i) \quad \text{(Equation 62)}$$

The designable parameters of this controller are: ω_c, $\xi_{f,c}$, $\beta_{f,c}$, $\xi_{f,a}$, $\beta_{f,a}$, ω_r, $\xi_{f,r}$, ω_f, $\xi_{f,f}$, $\beta_{f,f}$, $\alpha_{r,f}$, η_f.

The equations needed to simulate the full models of each controller are shown in Table S1.

SQTR-OR Dual Transcriptional-translational Controller Model

The SQTR-OR utilises an orthogonal RNA polymerase, P_o, which transcribes a small RNA, r_s. The sRNA binds to the o-RNAP mRNA, m_p, to form an RNA duplex, d_s, which rapidly decays:

$$m_p + r_s \underset{\alpha_{r,s}}{\overset{\alpha_{f,s}}{\rightleftharpoons}} d_s \quad \text{(Equation 63)}$$

Both the small RNA and RNA duplex dilute and decay:

$$r_s \xrightarrow{\lambda+\delta_{r,s}} \varnothing \qquad d_s \xrightarrow{\lambda+\delta_{d,s}} \varnothing \quad \text{(Equation 64)}$$

The transcription and translation of $P_o$mirror the chemical reactions in "Circuit model with competition for host cellular resources". Applying the Law of Mass Action yields the same dynamics for k_p, $x_p$and $c_p$as shown in Equations 14, 15, and 17. The dynamics of the mRNA $m_p$are updated to include its sRNA-mediated sequestration:

$$\dot{m}_p = \tau_p \cdot x_p - \beta_{f,p} \cdot m_p \cdot R_h + (\beta_{r,p} + \gamma_p) \cdot c_p - (\delta_{m,p} + \lambda) \cdot m_p - \alpha_{f,s} \cdot m_p \cdot r_s + \alpha_{r,s} \cdot d_s \quad \text{(Equation 65)}$$

The dynamics of the orthogonal RNA polymerase which transcribes circuit genes are given by:

$$\dot{P_o} = \gamma_{p,0} + \gamma_p \cdot c_p - \lambda \cdot P_o \ldots \quad \text{(Equation 66)}$$

$$\ldots - \xi_{f,p} \cdot g_p \cdot P_o + (\xi_{r,p} + \tau_p) \cdot x_p \ldots$$

$$\ldots - \xi_{f,s} \cdot g_s \cdot P_o + (\xi_{r,s} + \tau_s) \cdot x_s \ldots$$

$$\ldots + \sum_{i=1:N} \left(- \xi_{f,i} \cdot g_i \cdot P_o + (\xi_{r,i} + \tau_i) \cdot x_i \right)$$

The transcription dynamics of r_q mirror those of mRNA with the RNA polymerase binding the gene g_s to form a transcription complex $x_s$which produces r_s. The dynamics of these reactions are:

$$\dot{x_s} = \xi_{f,s} \cdot g_s \cdot P_o - (\xi_{r,s} + \tau_s) \cdot x_s - \lambda \cdot x_s \quad \text{(Equation 67)}$$

$$\dot{r_s} = \tau_s \cdot x_s - \alpha_{f,s} \cdot m_p \cdot r_s + \alpha_{r,s} \cdot d_s - (\delta_{r,s} + \lambda) \cdot m_p \quad \text{(Equation 68)}$$

$$\dot{d_s} = \alpha_{f,s} \cdot m_p \cdot r_s - \alpha_{r,s} \cdot d_s - (\delta_{r,s} + \lambda) \cdot d_s \quad \text{(Equation 69)}$$

Note that we set $\alpha_{r,s}$ to zero in all simulations, making the sRNA:mRNA reaction unidirectional.

The concentration of the free promoters of the sRNA can be calculated from its total ω_s:

$$g_s = \omega_s - x_s \quad \text{(Equation 70)}$$

The translational elements of the controller replicate the dynamics as described before.

The SQTR-OR controller utilises host resources for some of its processes. The concentration of free host resources is calculated from the total:

Please cite this article in press as: Darlington and Bates, Architectures for Combined Transcriptional and Translational Resource Allocation Controllers, Cell Systems (2020), https://doi.org/10.1016/j.cels.2020.08.014

$$P_h = P_{h,T} - x_r - x_f$$
$$R_h = R_{h,T} - c_P - R_o - c_f - \sum_{i=1:N}(c_i) \quad \text{(Equation 71)}$$

The designable parameters of this controller are: ω_p, $\xi_{f,p}$, $\beta_{f,p}$, ω_r, $\xi_{f,r}$, ω_f, $\xi_{f,f}$, $\beta_{f,f}$, ω_q, $\xi_{f,q}$, $\alpha_{r,f}$, η_f.

OP-OR Dual Transcriptional-Translational Controller Model

As described in the main text, the OP-OR controller is similar to the UBER-OR controller but with the $p_q$protein removed. Equation 18 is updated to remove the the transcription of p_qmRNA:

$$\dot{P}_o = \gamma_{p,0} + \gamma_p \cdot c_p - \lambda \cdot P_o \ldots \quad \text{(Equation 72)}$$

$$\ldots - \xi_{f,p} \cdot g_p \cdot P_o + (\xi_{r,p} + \tau_p) \cdot x_p \ldots$$

$$\ldots + \sum_{i=1:N}(-\xi_{f,i} \cdot g_i \cdot P_o + (\xi_{r,i} + \tau_i) \cdot x_i)$$

As p_q no-longer requires translation the number of free host ribosomes' expression also changes:

$$R_h = R_{h,T} - c_P - R_o - c_f - \sum_{i=1:N}(c_i) \quad \text{(Equation 73)}$$

All other equations highlighted in Table S1 for the UBER-OR controller are unchanged.

The designable parameters of this controller are: ω_p, $\xi_{f,p}$, $\beta_{f,p}$, ω_r, $\xi_{f,r}$, ω_f, $\xi_{f,f}$, $\beta_{f,f}$, $\alpha_{r,f}$, η_f.

var-FRAG-OR Dual Transcriptional-Translational Controller Model

The variant var-FRAG-OR controller replaces host RNA polymerase mediated transcription of the alpha and core components with the orthogonal RNA polymerase. This creates a positive feedback loop. To model this we replace P_h with P_o in Equations 29 to 36 and update Equation 37. The dynamics of the orthogonal RNA polymerase is now given by:

$$\dot{P}_o = \gamma_{p,0} + \theta_f \cdot p_a \cdot p_c - \theta_r \cdot P_o - \lambda \cdot P_o \ldots \quad \text{(Equation 74)}$$

$$\ldots - xi_{f,c} \cdot g_c \cdot P_o + (\xi_{r,c} + \tau_c) \cdot x_c \ldots$$

$$\ldots - xi_{f,a} \cdot g_a \cdot P_o + (\xi_{r,c} + \tau_a) \cdot x_a \ldots$$

$$\ldots + \sum_{i=1:N}(-\xi_{f,i} \cdot g_i \cdot P_o + (\xi_{r,i} + \tau_i) \cdot x_i)$$

The new expression for the host RNA polymerase is given by:

$$P_h = P_{h,T} - x_r - x_f \quad \text{(Equation 75)}$$

The dynamics of the translational resource (and corresponding definition of host ribosomes usage) are the same as given in Table S1 for the FRAG-OR dual controller.

The designable parameters of this controller are: ω_c, $\xi_{f,c}$, $\beta_{f,c}$, $\xi_{f,a}$, $\beta_{f,a}$, ω_r, $\xi_{f,r}$, ω_f, $\xi_{f,f}$, $\beta_{f,f}$, $\alpha_{r,f}$, η_f.

An All-othogonal Ortho-OP-OR Dual Transcriptional-translational Controller Model

The ortho-OP-OR controller has a similar structure to the OP-OR controller but utilises only orthogonal resources for its own expression. The dynamics of the intermediate complexes take the same form as those depicted in previous sections but with host RNA polymerase P_h and ribosomes R_h replaced with their orthogonal counterparts $P_o$and R_o.

The dynamics of the orthogonal RNA polymerase are given by:

$$\dot{P}_o = \gamma_{p,0} + \gamma_p \cdot c_p - \lambda \cdot P_o \ldots \quad \text{(Equation 76)}$$

$$\ldots - \xi_{f,p} \cdot g_p \cdot P_o + (\xi_{r,p} + \tau_p) \cdot x_p \ldots$$
$$\ldots - \xi_{f,r} \cdot g_r \cdot P_o + (\xi_{r,r} + \tau_r) \cdot x_R \ldots$$

Please cite this article in press as: Darlington and Bates, Architectures for Combined Transcriptional and Translational Resource Allocation Controllers, Cell Systems (2020), https://doi.org/10.1016/j.cels.2020.08.014

$$\ldots - \xi_{f,f} \cdot g_f \cdot P_o + (\xi_{r,f} + \tau_f) \cdot x_f \ldots$$

$$\ldots + \sum_{i=1:N} \left(- \xi_{f,i} \cdot g_i \cdot P_o + (\xi_{r,i} + \tau_i) \cdot x_i \right)$$

The dynamics of the orthogonal ribosome pool are given by:

$$\dot{R_o} = \varrho_f \cdot r \cdot R_h - \varrho_r \cdot R_o - \lambda \cdot R_o \ldots \quad \text{(Equation 77)}$$

$$\ldots - \beta_{f,p} \cdot m_p \cdot R_o + (\beta_{r,p} + \gamma_p) \cdot c_p \ldots$$

$$\ldots - \beta_{f,f} \cdot m_f \cdot R_o + (\beta_{r,f} + \gamma_f) \cdot c_f \ldots$$

$$\ldots + \sum_{i=1:N} \left(- \beta_{f,i} \cdot m_i \cdot R_o + (\beta_{r,i} + \gamma_i) \cdot c_i \right)$$

Host RNA polymerase are not used while host ribosomes are impacted by the *quasi*-orthogonal nature of the orthogonal ribosome pool as discussed in the main text.

$$\begin{aligned} P_h &= P_{h,T} \\ R_h &= R_{h,T} - R_o - c_p - c_f - \sum_{i=1:N} (c_i) \end{aligned} \quad \text{(Equation 78)}$$

The designable parameters of this controller are: ω_p, $\xi_{f,p}$, $\beta_{f,p}$, ω_r, $\xi_{f,r}$, ω_f, $\xi_{f,f}$, $\beta_{f,f}$, $\alpha_{r,f}$, η_f.

Experimentally Implementable Parameters

As discussed in Darlington et al. (2018b), we set the total concentration of host RNA polymerase and ribosomes to 250 nM and 2,500 nM respectively assuming a constant growth rate of $1h^{-1}$. The model assumes that each gene is bound by only one RNA polymerase and each mRNA is bound by one ribosome. However, *in vivo* each gene is transcribed by multiple RNA polymerase and each mRNA is translated by multiple ribosomes. We account for this in our model by increasing the copy number of each gene and increaseing the mRNA production rate such that RNA polymerase and ribosomes are subject to the the appropriate level of competition. From Darlington et al. (2018b), we take conservative estimates and increase gene copy numbers by 10 and the mRNA production rates by 20 throughout.

The strengths of promoters, RBS sites and protein DNA binding are often quoted as dissociation constants. Following from the definitions in Gyorgy et al. (2015); Qian et al. (2017); Darlington et al. (2018b), the RNAP-promoter, ribosome-RBS and transcription factor dissociation constants are given by k_X, k_L and μ respectively:

$$k_X = (\xi_r + \tau)/\xi_f \qquad k_L = (\beta_r + \gamma)/\beta_f \qquad \mu = \alpha_r/\alpha_f \quad \text{(Equation 79)}$$

As our model contains resource turnover (as a specific consequence of modelling control of resources) we do not make the quasi-steady state assumption as in Gyorgy et al. (2015); Qian et al. (2017); Darlington et al. (2018b) and so the individual binding/unbinding rate constants remain in their original form in the ODEs. These lumped dissociation constants do not appear. Therefore, in order to vary k_X and k_L we vary ξ_f and β_f. From Darlington et al. (2018b), $k_X$ranges from 5 nM to 1,000 nM and assuming that $\beta_r + \gamma \approx 1100$ then ξ_f varies from 11 and 2200. Similarly, k_L ranges from 10^4 to 10^7 nM with $\beta_r + \gamma \approx 10^6$ then β_f varies from 0.1 to 100.

We assume that promoters can be carried on a low (10 nM), medium (100 nM) or high (500 nM) copy number plasmids.

For simplicity we limit the choice of repressors for $p_f$and p_q to the commonly used repressors lacI, cI from bacteriophage λ and tetR. This limits the values of the promoter-RNA polymerase dissocation constant, transcription factor dissociation constant and multimerisation. Note that we assume that the transcription factor binding rate (α_f) is 1 and therefore $\alpha_r$is set to the value of the dissociation constant, μ. The values of these parameters are listed in Table S2. All other parameters are shown in Table S3.

Controller Design Process

We developed a range of experimentally feasible designs based on biologically realisable promoter and RBS strengths, gene copy numbers, and the common repressors tetR, lacI and cI (as discussed above). We assess the performance of these controllers by simulating the response to two sequential large step inputs (i.e. simulating the induction of genes sequentially from a high copy number plasmid). To determine stability (or otherwise) of the controllers, we numerically calculate the jacobian and eigvenvalues in each part of the simulation (i.e. before induction, after induction of the first gene and after the induction of the second gene) and remove controllers which have positive real eigenvalues. We also remove controllers which inhibit the process output sufficiently to reduce process output to near zero.

Please cite this article in press as: Darlington and Bates, Architectures for Combined Transcriptional and Translational Resource Allocation Controllers, Cell Systems (2020), https://doi.org/10.1016/j.cels.2020.08.014

To assess the performance of each controller, we calculate coupling at both the transcriptional and translational levels by assessing the fall in the first circuit gene (mRNA m_1 and protein p_1 respectively) in response to the induction of the second circuit gene by comparing the values at the time of the second gene induction t_{ind} and the end of the simulate t_{end}:

$$\Delta_X = (m_1(t = t_{end}) - m_1(t = t_{ind}))/m_1(t = t_{ind}) \quad \text{(Equation 80)}$$

$$\Delta_L = (p_1(t = t_{end}) - p_1(t = t_{ind}))/p_1(t = t_{ind}) \quad \text{(Equation 81)}$$

To create a single performance metric we consider the behaviour of the controllers in a two dimensional plane whose axes are the scaled values of Δ_X and Δ_L. The origin $(0, 0)$ represents perfect decoupling at both the transcriptional and translational levels. We calculate the Euclidean distance of each point from the origin:

$$r = \sqrt{(\Delta_X^2 + \Delta_L^2)} \quad \text{(Equation 82)}$$

We sort the controller by r and take the top $N = 100$ controllers which represent those which show the best decoupling performance. We call this metric the 2D score. We assess these controllers further in terms of both decoupling and protein output.

Computational Methods

All ordinary differential equation models presented where solved numerically in MATLAB 2019a inbuilt ordinary differential equation solver ode23s in MATLAB 2019a with default tolerances. Models were originally simulated with no-circuit gene expression ($\omega_1 = 0$, $\omega_2 = 0$) to identify the controller species steady states for a time period of 120 minutes. For the coupling assessments ω_1 was increased from 0 to 500 nM and the system simulated for 120 minutes to identify the new steady state – this is the 'constitutive p_1 steady state' before induction of the second gene. The second gene was induced as shown in the Figures. The robustness analysis was carried out by varying key controller parameters by up to $\pm\delta$ (where δ is specified in the main text). Random parameter sets were created by drawing numbers from a uniform distribution in the interval $[-\delta, +\delta]$ using the inbuilt *unifrnd* function. 1,000 sample controllers were assessed for stability Cosentino and Bates (2011). The MATLAB Parallel Computing Toolbox was utilised to reduce computational time when iterating over large number of parameter sets during the design or robustness analysis stages.