



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ESCUELA TÉCNICA
SUPERIOR INGENIERÍA
INDUSTRIAL VALENCIA

Curso Académico:

Agradecimientos

En primer lugar, a Jesús por la oportunidad de poder trabajar en un proyecto que combine el mundo de la ingeniería tradicional con el de la biología. Espero poder seguir explorando esta combinación en el futuro ya que ofrece un abanico de posibilidades inmenso.

A Alejandro por su tiempo y su paciencia. Han sido meses de trabajo, con sesiones formativas densas y largas, pero siempre has puesto de tu parte para que las cosas pudieran salir adelante y has conseguido despertar en mí un gran interés por esta materia gracias a tu pasión por la misma.

A mis amigos, a los que llevan toda la vida conmigo y a los que han llegado más tarde, pero para quedarse.

Y por último y como parte fundamental, a mi familia. Conseguís aguantarme cuando ni yo lo hago y sois un apoyo imprescindible en mi vida.

Resumen

El seguimiento de señales y el problema relacionado de estimación de la derivada de una señal son piezas clave en el diseño de sistemas de control de retroalimentados. En este trabajo, se planteará el diseño y ajuste óptimo de un circuito seguidor biomolecular que contiene dos motivos antitéticos en retroalimentación a través de un motivo de interruptor push-pull. Se evaluarán modelos realistas de diferentes posibilidades para implementar los dispositivos biomoleculares que componen el circuito seguidor, dando lugar a un conjunto de diseños candidatos. La optimización multiobjetivo se utilizará para diseñar una biblioteca de componentes biológicos óptimos correspondientes a circuitos de seguimiento con diferentes compromisos de desempeño para cada uno de los diseños candidatos. Durante el proceso de diseño, se utilizará un modelo de la célula anfitriona para tener en cuenta las interacciones entre el anfitrión y el circuito que podrían conducir a un funcionamiento incorrecto del último causado por la competencia por los recursos de la célula. Como objetivo secundario, los mejores diseños biológicos serán considerados como candidatos potenciales para implementar rastreadores de señales electrónicos bioinspirados.

Palabras clave: circuito seguidor, circuitos biomoleculares, optimización multi-objetivo, realimentación, estimación de derivadas de una señal

Resum

El seguiment de senyals i el problema relacionat d'estimació de la derivada d'una senyal son peces clau en el disseny de sistemes de control de retroalimentació. En aquest treball, es plantejarà el disseny i ajust òptim d'un circuit seguidor biomolecular que conté dos motius antitètic en retroalimentació per mitjà d'un motiu d'interruptor push-pull. Es evaluaran models realistes de diferents possibilitats per a implementar els dispositius biomoleculars que componen el circuit seguidor, donant lloc a un conjunt de dissenys candidats. La optimització multiobjectiu s'utilitzarà per a dissenyar una biblioteca de components biològics òptims corresponents a circuits de seguiment amb diferents compromisos de desempeñy per a cadascún dels dissenys candidats. Durant el procés de disseny, s'utilitzarà un model de la cèl·lula anfitriona per a tindre en compte les interaccions entre l'anfitrió i el circuit que podrien conduir a un funcionament incorrecte de l'últim causat per la competència per els recursos de la cèl·lula. Com a objectiu secundari, els millors dissenys biològics serán considerats com a candidats potencials per a implementar rastrejadors de senyals electrònics bioinspirats.

Paraules clau: circuit seguidor, circuits biomoleculars, optimització multi-objectiu, realimentació, estimació de derivades d'una senyal

Abstract

Signal tracking and the related problem of signal derivative estimation are key pieces in the engineering of feedback control systems. In this work, the design and optimal tuning a biomolecular circuit that contains two antithetic motifs in feedback via a push-pull switch motif will be considered. Realistic models of different possibilities to implement the biomolecular devices that compose the tracker will be assessed, leading to a set of candidate designs. Multi-objective optimization will be used to design a library of optimal biological components corresponding to tracker circuits with different performance trade-offs for each of the candidate designs. During the design process, a model of the host-cell will be used to account for the interactions between host and circuit that might lead to miss functioning of the last caused by competition for cell resources. As a secondary goal, the best biological designs will be considered as potential candidates to implement electronic bio-inspired signal trackers.

Keywords: follower circuit, biomolecular circuits, multi-objective optimization, feedback, estimation of derivatives of a signal

Índice general

Agradecimientos	3
Resumen	4
Resum	5
Abstract	6
I MEMORIA	14
1. Objetivos	15
2. Introducción.....	15
2.1. Biología sintética	15
2.2. Dogma Central de la Biología Molecular	15
2.3. Circuitos genéticos sintéticos	17
3. Métodos y materiales	19
3.1. Teoría de control en la Biología Sintética.....	19
3.2. Ley de acción de masas.....	21
3.3. Circuito antitético	24
3.4. Optimización multiobjetivo.....	26

3.5. Materiales	27
4. Estudios previos y justificación	28
5. Diseño y resultados	30
5.1. Diseño del modelo	30
5.2. Resultados y discusión	40
II PRESUPUESTO	47
6.1. Introducción y objetivo	48
6.2. Presupuesto	48
Anexo A	52
Bibliografía	58

Índice de figuras

Figura 2.1. Proceso de producción de una proteína. Durante la expresión genética, una ARNp leerá la información codificada en un segmento de ADN para generar un producto intermediario denominado mRNA. Posteriormente un ribosoma se unirá al mRNA para decodificar la información que contiene y generar a partir de él, una cadena polipeptídica formada a partir de aminoácidos. (Boada Acosta, 2018).	16
Figura 2.2 Expresión de una Unidad Transcripcional mediante el SBOL.....	17
Figura 2.3 Simbología y otros elementos del SBOL	18
Figura 3.1 Sistema de control de lazo cerrado con retroalimentación.....	19
Figura 3.2 Representación de las dinámicas de las concentraciones de las sustancias hasta alcanzar el equilibrio químico	23
Figura 3.3 Estrategias de control para redes de circuitos genéticos	23
Figura 3.4 Implementación de control con retroalimentación de células vivas (Hsiao et al., 2018).....	24
Figura 3.5 Descripción de las partes del circuito (Lillacci, Aoki et al., 2017).....	25
Figura 4.1 Representación del circuito biomolecular de seguimiento de señal	29
Figura 4.2 Barrido de simulaciones del modelo de Picó, Vignoni et al., 2019.....	30
Figura 5.1 Descripción del circuito con lenguaje SBOL.	31
Figura 5.2 Simulaciones con un barrido de parámetros. La y es la señal de entrada y x4 la variable que trata de seguirla y la cual estamos tratando de controlar y modelar	39

Figura 5.3 Simulación final del modelo	41
Figura 5.4 Evolución de la concentración de la especie x_1	41
Figura 5.5 Evolución de la concentración de la especie x_2	41
Figura 5.7 Evolución de la concentración de la especie x_1x_2	42
Figura 5.6 Evolución de la concentración de la especie x_3	42
Figura 5.8 Evolución de la concentración de la especie x_4	42
Figura 5.9 Evolución de la concentración de la especie x_5	42
Figura 5.10 Evolución de la concentración de la especie x_4x_5	42
Figura 5.11 Respuesta frente a diferentes valores de la señal de entrada	43
Figura 5.12 Diferentes respuestas frente a oscilaciones o perturbaciones	45

Índice de tablas

Tabla 4.1 Especies del modelo reducido	36
Tabla 4.2 Parámetros del modelo reducido	37
Tabla 4.3 Parámetros iniciales de las simulaciones	38
Tabla 4.4 Variables de decisión.....	40
Tabla 4.5 Valores de los parámetros del modelo reducido	40

Parte I

Memoria

1. Objetivos

El objetivo principal de este proyecto es obtener modelos que consigan realizar el seguimiento de bioseñales por medio de un circuito biomolecular. Otros objetivos secundarios pero que también pueden resultar de interés son:

- Diseñar e implementar diferentes modelos que permitan el control de bioseñales.
- Modelado de circuitos sintéticos que permiten alcanzar objetivos definidos.
- Obtención de sistemas de ecuaciones reducidos que simulen situaciones prácticas reales.
- Desarrollo y aprendizaje del uso de software especializado como Matlab® y, en concreto, de *solvers* de sistemas de ecuaciones diferenciales ordinarias (EDO).
- Aplicación, desarrollo y adaptación de algoritmos de implementación de optimización multiobjetivo.
- Reforzar conceptos del desarrollo de proyectos como la elaboración de documentación técnica o de valoración económica como los presupuestos.

2. Introducción

2.1. Biología sintética

La Biología Sintética es un campo multidisciplinar que combina diferentes áreas de conocimiento como la Biología, la Ingeniería, la Computación, entre otras, y cuyo fin es la obtención de nuevos procesos y sistemas biológicos o la mejora de los ya existentes de forma natural.

Según la **Organización de Innovación Biotecnológica** la Biología Sintética supone “la aplicación de principios de la ingeniería a la biología” y tiene como objetivo “el (re)diseño y fabricación de componentes y sistemas biológicos que todavía no existen en el mundo natural.” Así, la Biología Sintética se aplica en muchas industrias como la biotecnológica, química, energética y de la salud.

Dado que el desarrollo de este Trabajo se enmarca en el contexto descrito por la Biología Sintética, a continuación, se desarrollarán algunos conceptos y procesos básicos de Biología e Ingeniería de Control que serán de interés para la contextualización del mismo.

2.2. Dogma Central de la Biología Molecular

Para poder regular o controlar procesos biológicos es esencial conocer como los microorganismos vivos transfieren su información genética. Tras la definición de la

estructura del ácido desoxirribonucleico (ADN) por Watson y Crick (Watson et al., 1953), este último estableció el proceso de la expresión génica (Figura 2.1) con el conocido como **Dogma Central de la Biología Molecular**. En él se describe con detalle el proceso mediante el cual la información genética es transferida en organismos vivos (Crick, 1970), en concreto, cómo los organismos son capaces de decodificar la información genética contenida en su ADN para la síntesis de proteínas.

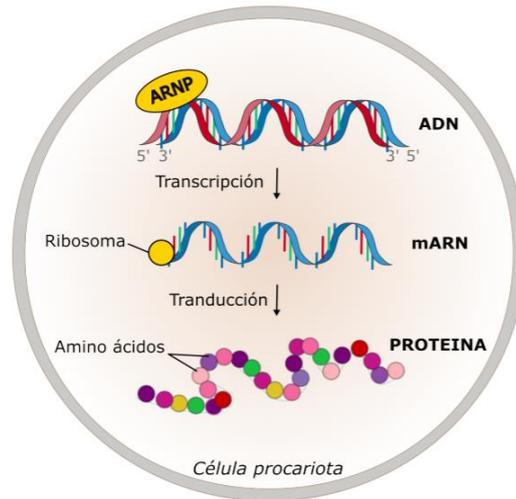


Figura 2.1. Proceso de producción de una proteína. Durante la expresión genética, una ARNp leerá la información codificada en un segmento de ADN para generar un producto intermediario denominado mRNA. Posteriormente un ribosoma se unirá al mRNA para decodificar la información que contiene y generar a partir de él, una cadena polipeptídica formada a partir de aminoácidos. (Boada Acosta, 2018).

Como se puede apreciar en la Figura 2.1 la expresión génica tiene dos procesos principales: **Transcripción** y **Traducción**. Durante la *transcripción* una secuencia de ADN se decodifica y da lugar a ácido ribonucleico mensajero (ARNm), que, a su vez, se traduce en proteína en el proceso de *traducción*. A continuación, se expondrán con mayor detalle ambos procesos.

La Transcripción comienza cuando una enzima llamada ARN polimerasa (ARNp) se une a una secuencia concreta de ADN. Esta secuencia contiene la información necesaria para generar una proteína específica y la ARNp la transcribe a una molécula de ARNm. Esta secuencia concreta de ADN se conoce como promotor. Si este proceso de transcripción ocurre sin el apoyo o impedimento de otras moléculas se dice que es constitutiva. Por otro lado, existen proteínas capaces de unirse específicamente a secuencias cortas de ADN localizadas en los promotores y de interactuar con el complejo de la transcripción para inducir (un **activador**) o inhibir (un **represor**) la actividad de la ARNp (Nakashima et al., 2009). A estas moléculas se les conoce como Factores de Transcripción (TF).

La Traducción se produce cuando un **ribosoma** se ensambla al ARNm generado en el proceso anterior y “lee” la información contenida en él para producir una proteína. Los ribosomas son complejos formados por proteínas y ARN que reconocen secuencias concretas del ARNm, conocidas como *ribosome binding site* (RBS), a las cuales se adhieren. La facilidad con la que los ribosomas se enlazan o se separan del RBS determina la tasa de producción de la proteína (Boada, 2018). En ocasiones no genera una proteína completa, sino que codifica para un **polipéptido** o una cadena de aminoácidos. Un polipéptido puede formar una proteína funcional o puede requerir unirse o ensamblarse con otros para conformar una proteína funcional (OpenStax College, Concepts of Biology, 2013). Se suelen denominar como proteínas los polipéptidos cuya masa molecular es superior a 10000. Las proteínas constituyen la maquinaria que lleva a cabo las tareas más esenciales de las células.

2.3. Circuitos genéticos sintéticos

Como se ha comentado anteriormente, uno de los objetivos o aplicaciones principales de la Biología Sintética es la creación, control y programación de procesos biológicos y del comportamiento celular de los mismos. Las redes genéticas sintéticas se modelan como sistemas de reacciones químicas que suceden en el interior de las células, donde las especies químicas interactúan entre sí y adquieren un comportamiento tipo circuito (Hsiao, Swaminathan y Murray 2018). Por ello, se conocen como **circuitos genéticos sintéticos**. Estos sistemas están constituidos por Unidades Transcripcionales (TU) que son los elementos mínimos necesarios para poder producir proteína y que, a su vez, están compuestas por partes biológicas estándar o *bioparts* (promotores, sitio de unión ribosómico (RBS), región codificante (CDS) o gen y terminadores) que llevan a cabo un amplio rango de funciones específicas si estas partes se combinan adecuadamente (Wu y Wang, 2015). La representación gráfica de estas Unidades Transcripcionales y de los *bioparts* que las forman queda definido por el *Synthetic Biology Open Language* (SBOL) tal y como se puede observar en la Figura 2.2. Este lenguaje ha sido desarrollado como un estándar para apoyar el intercambio de información de diseños biológicos (Beal, Cox et al., 2016).

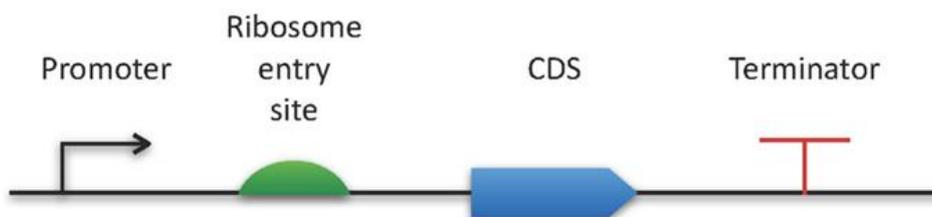


Figura 2.2 Expresión de una Unidad Transcripcional mediante el SBOL

	promoter		primer binding site
	cds		restriction site
	ribosome entry site		blunt restriction site
	terminator		5' sticky restriction site
	operator		3' sticky restriction site
	insulator		5' overhang
	ribonuclease site		3' overhang
	rna stability element		assembly scar
	protease site		signature
	protein stability element		user defined
	origin of replication		

Figura 2.3 Simbología y otros elementos del SBOL

Cada una de las cuatro partes que componen una Unidad Transcripcional tienen características específicas.

1. Promotor

Como se describe en el apartado del Dogma Central de la Biología Molecular un promotor es una secuencia concreta de ADN que normalmente se encuentra muy próxima al gen. En esta secuencia es donde se inicia el proceso de transcripción descrito anteriormente. Si este proceso sucede a un ritmo constante sin verse afectada por otros factores externos se dice que el promotor es **constitutivo**. Como ya se avanzó también en apartados anteriores, existen factores de transcripción que consiguen alterar este proceso. Los promotores que pueden alterar su ritmo de transcripción se conocen como **inducibles**. Hay dos tipos:

-**Activadores**. Son proteínas que inducen a las ARNp a adherirse a ellos para iniciar la transcripción.

-**Represores**. En contraposición con los anteriores son proteínas que inhiben el proceso de transcripción dificultando la adhesión de las ARNp.

2. Sitio de unión ribosómico (RBS)

Es una secuencia específica del ARNm donde los ribosomas pueden unirse para iniciar así la traducción. La producción de proteína en el proceso de traducción viene determinada por la afinidad para unirse entre un ribosoma y esta secuencia de unión ribosómica. Esta afinidad caracterizará el ritmo de la traducción (M. Eriksen et al., 2016).

3. Región codificante (CDS)

Es una región de ADN o ARN cuya secuencia determina la secuencia específica de aminoácidos (péptidos) en una proteína.

4. Terminador

El grupo BioROM lo define en su guía de aprendizaje como “la secuencia de ADN que permite a la ARNp separarse del ADN y posibilita que la ARN recién generada se libere y se complete la transcripción. Además, solo permite la finalización de la transcripción del gen que ha sido previamente «recorrido» por la ARNp.”

3. Métodos y materiales

3.1. Teoría de control en la Biología Sintética

Una vez definidas las partes y procesos más importantes que aparecen en la síntesis de proteínas, mediante la Teoría de Control, se trata de regular estos procesos de manera que se consigan objetivos específicos. Por lo general, se utilizan las proteínas para este tipo de estudios ya que son estructuras muy estables tanto en aspectos termodinámicos como cinéticos. Este último aspecto, que se detallará más adelante a través de la Ley de Acción de Masas, será el aspecto clave para definir las ecuaciones que rigen los sistemas o circuitos genéticos y que serán de vital importancia conocer para poder controlarlos.

Por tanto, la Teoría de Control se utiliza como una poderosa herramienta para aclarar los principios subyacentes que llevan a cabo las redes genéticas y predecir su dinámica y robustez ante variaciones o perturbaciones que puedan alterar el sistema (Menolascina et al., 2014).

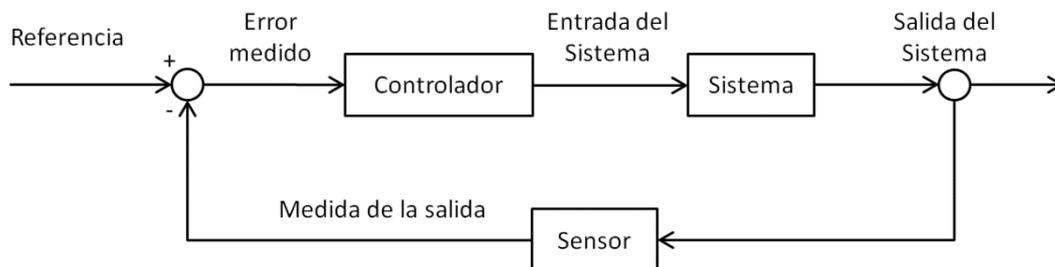


Figura 3.1 Sistema de control de lazo cerrado con retroalimentación

En la Figura 3.1 se tiene un sistema de control de lazo cerrado con retroalimentación. Este sistema permite que el proceso se autorregule a través de la **retroalimentación**. En el esquema la acción del controlador depende de la salida del sistema. El valor de la salida, medida a través de un sensor, retroalimenta el circuito y se compara con el valor de referencia o de entrada. El valor de la comparación entre el término de salida y de entrada se conoce como **error**. Este error es la variable que entra o recibe el controlador y a partir de ella genera una acción sobre el sistema que permitirá que este último alcance los valores que se hayan tomado como objetivo. Los sistemas retroalimentados son los más utilizados cuando el objetivo de control necesita relacionar las variables de salida con las de entrada o se quiere generar una dependencia entre ambas.

Según las acciones de control que se quieran o se deban adoptar sobre el sistema se pueden utilizar tres tipos de controladores (Aström y Murray, 2010).

-Proporcional (P)

Como su propio nombre indica, la acción de control $U(s)$ es proporcional al error $e(s)$. Esta proporcionalidad viene definida por la constante K_c . Según sea el valor de esta constante la dinámica será más rápida o lenta, es decir, la señal alcanzará el valor de referencia en mayor o menor tiempo. Esto puede generar problemas de estabilidad ya que si la dinámica es muy rápida alcanzará el valor deseado muy rápidamente pero no lo mantendrá en el tiempo. Además, en caso de que la acción de control sea necesaria para mantener ciertos valores dentro del sistema el error tiene que ser siempre distinto de cero.

$$U(s) = K_c (Y(s) - R(s)) = K_c \cdot e(s) \quad (3.1)$$

-Integral (I)

En este tipo de controladores se consigue un error nulo cuando el sistema se encuentra en estado estacionario, es decir, que sus variables no cambian a lo largo del tiempo. Con un controlador integral y, en definitiva, al incluir la integral del error, lo que se pretende es tener en cuenta el valor del error en el instante inmediatamente anterior para calcular la acción de control $U(s)$. La constante K_i determinará, igual que hace K_p en el controlador proporcional, la velocidad con la que la salida $Y(s)$ sigue a la referencia $R(s)$.

$$U(s) = K_i \int_0^t e(s) ds \quad (3.2)$$

-Derivativo (D)

La acción derivativa trata de predecir el valor del error de una manera aproximada. Esto lo consigue mediante una interpolación lineal. La acción derivativa tampoco garantiza un error nulo en estado estacionario.

$$e(t + Td) = e(t) + T_d \cdot \frac{e(t)}{dt} \quad (3.3)$$

Por tanto, una acción de control $U(s)$ regulada por un control derivativo quedaría de la siguiente forma:

$$U(s) = Kd \frac{e(s)}{ds} \quad (3.4)$$

Generalmente se utilizan combinaciones de los controladores anteriores para poder cumplir los requisitos del sistema. Algunas combinaciones podrían ser un controlador con acción proporcional-integral (PI), proporcional-derivativo (PD) o, el más utilizado de todos, un controlador con acción proporcional-integral-derivativa (PID). La mayoría de los procesos de control industriales (más del 90%) se llevan a cabo con este tipo de controladores (Aström y Murray, 2010).

Uno de los retos de la Biología Sintética es tratar de replicar las funcionalidades de estos controladores a nivel molecular y con especies biológicas, por ejemplo, para seguir una señal de entrada a partir la concentración resultante de otra especie biológica en un circuito (Hsiao et al., 2014). Además, hay que cerciorarse de que está implementación da una respuesta robusta ante las variaciones que pueda sufrir el sistema (Potvin-Trottier et al., 2016).

3.2. Ley de acción de masas

Para poder regular y controlar las reacciones químicas y biológicas que se llevan a cabo en el interior de las células es necesario conocer las dinámicas de los sistemas y elementos que las forman. La **Ley de Acción de Masas** (LAM) describe estos procesos que implican relaciones de equilibrio tanto de masa como de energía (Chellaboina et al., 2009). Una de las propiedades más útiles de este sistema es que no arroja soluciones negativas lo que, desde el punto de vista analítico, es muy importante ya que evita trabajar con resultados no realistas, por ejemplo, no pueden existir concentraciones de proteínas negativas. Con

estos modelos se pretende definir las reacciones bioquímicas y, en particular, cómo varían las especies cuando participan de estas reacciones a través de Ecuaciones Diferenciales Ordinarias (EDO). A continuación, se verá cómo se formulan estas ecuaciones a partir de un ejemplo:



Los elementos del lado izquierdo se conocen como reactivos y los del lado derecho se llaman productos. Existe una variable k , conocida como ritmo de reacción, que determina la velocidad con la que esta se lleva a cabo. Tanto los reactivos como los productos son referidos como especies de la reacción. Este es el caso más simplificado en el que la reacción es irreversible, pero podrían darse casos, más realistas, en los que la reacción sea reversible y existan velocidades de reacción (k_1 y k_2) en ambos sentidos de la misma.

En este punto, se aplica la ley de acción de masas que, en términos técnicos, indica que “si se toma una reacción elemental, es decir, una reacción en la que los coeficientes estequiométricos de los reactivos sean uno, la velocidad de reacción es proporcional al producto de las concentraciones de los reactivos” (Chellaboina et al., 2009).



Donde X_i son las especies de la reacción y b es un entero positivo. Entonces, si se aplica la ley de acción de masas se obtiene que:

$$\dot{x}_1(t) = -kx_1(t)x_2(t) \quad (3.7)$$

$$\dot{x}_2(t) = bkx_1(t)x_2(t) \quad (3.8)$$

Donde $\dot{x}_i(t)$ representa las concentraciones de las especies X_i . Estos modelos han sido extraídos del estudio de Chellaboina et al., 2019 “*Modeling and analysis of mass-action kinetics*”.

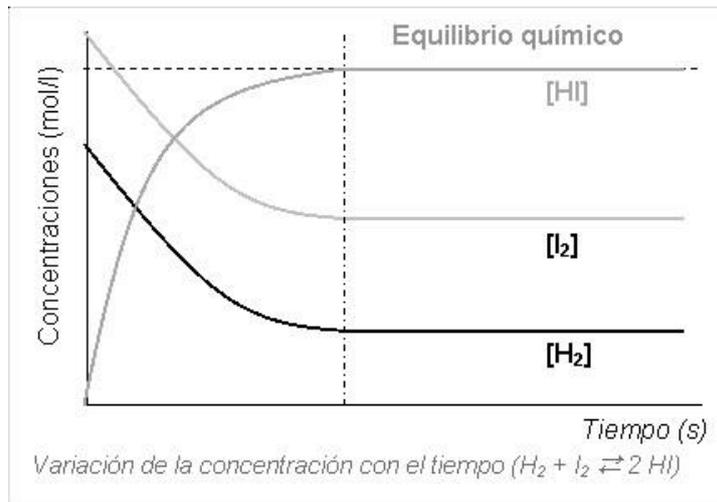


Figura 3.2 Representación de las dinámicas de las concentraciones de las sustancias hasta alcanzar el equilibrio químico

Todas las reacciones que se llevan a cabo dentro de la célula tienen una componente sometida al azar por naturaleza. En los modelos presentados anteriormente y en los que se presentarán más adelante no se tienen en cuenta las probabilidades que tienen esas reacciones de suceder, por tanto, se está usando un modelo **determinista**. En los modelos deterministas solo se tienen en cuenta la cantidad actual de especies, las velocidades de reacción y los coeficientes estequiométricos para determinar la cantidad de especies transformadas en la reacción (Boada 2018).

Actualmente existen dos grandes categorías para implementar y diseñar controladores que manejen los sistemas de circuitos genéticos (Del Vecchio et al., 2016).

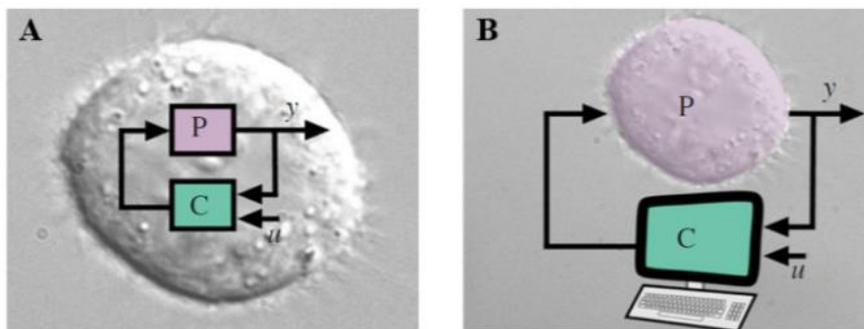


Figura 3.3 Estrategias de control para redes de circuitos genéticos

-Controlador *in vivo*. En esta primera aproximación, el control se lleva a cabo dentro de la propia célula a través de procesos biomoleculares. Los elementos que llevan a cabo

este control son los *bioparts* (explicados con detalle en apartados anteriores). Las posibles acciones de control que se pueden dar están limitadas a las interacciones que los *bioparts* tienen entre sí dentro de la célula, esto genera limitaciones y surgen problemas de robustez ya que las reacciones que se establecen están sujetas a variaciones estocásticas.

-Controlador *in silico*. Esta segunda categoría utiliza un ordenador, como se aprecia en la Figura 3.3 (B), para implementar la acción de control y el objeto sujeto a esta regulación es la célula en su conjunto. Esto amplía mucho las posibilidades de control a pesar de que el ámbito de aplicación no pueda ser una única célula.

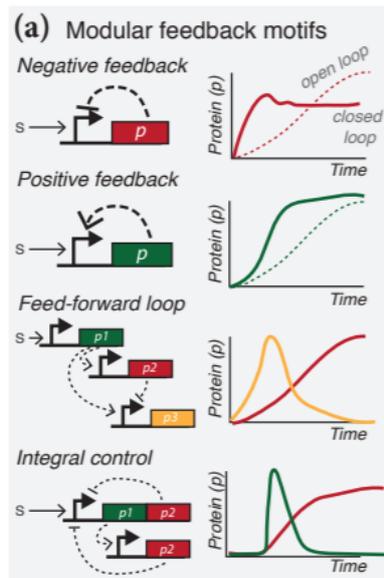


Figura 3.4 Implementación de control con retroalimentación de células vivas (Hsiao et al., 2018)

3.3. Circuito antitético

Un ejemplo de aplicación de los métodos de implementación de controladores y, en concreto, de una acción de control integral es el **Circuito Antitético** (Briat et al., 2016). Usa regiones específicas de las estructuras de la proteína definidas por una función particular y, por ello, en biología, se conoce como un **motivo antitético**. La propiedad principal de este circuito es que es capaz de implementar una respuesta robusta.

Este controlador necesita de un par de especies que presenten inactivación estequiométrica. Este par de especies son conocidos y pueden ser algunos tales como toxina/antitoxina, factor sigma/factor anti-sigma, ARNm/ARN anti sentido, entre otros. Los factores sigma y anti-sigma forman parte del grupo de **factores extra citoplasmáticos** (ECF).

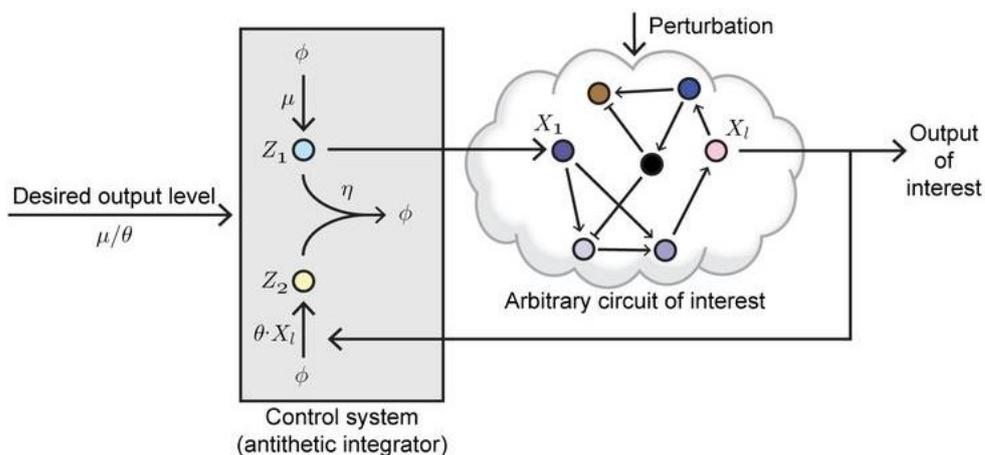


Figura 3.5 Descripción de las partes del circuito (Lillacci, Aoki et al., 2017)

Como se ha comentado este motivo antitético necesita, en este caso, de dos proteínas que se inactiven o anulen su actividad biológica cuando reaccionen entre ellas, por ejemplo, formando un dímero inerte (Lillacci, Aoki et al., 2017). Las especies utilizadas en este caso serán sigma (s) y anti- s que en la Figura 3.5 aparecen representadas como Z_1 y Z_2 , respectivamente.

Como se ha relatado en apartados anteriores, para realizar el control de este tipo de sistemas es necesario conocer la dinámica temporal de los mismos. Para ello, se utiliza la ya mencionada **Ley de Acción de Masas** para definir el modelo y demostrar cómo se obtiene el control integral de la señal de interés.

Partiendo de las dos especies ya presentadas (s y anti- s) se obtiene que:

$$\begin{aligned} \frac{Z_1}{dt} &= \mu - \eta Z_1 Z_2 \\ \frac{Z_2}{dt} &= \theta X_L - \eta Z_1 Z_2 \\ \frac{X_L}{dt} &= k Z_1 - d X_L \end{aligned} \quad (3.9)$$

Las ecuaciones que representan la dinámica temporal de las especies implicadas no son lineales. Dado que ambas especies tratan de aniquilarse entre sí cuando se juntan esto se puede representar mediante una resta, es decir, cuanto más cantidad o concentración haya de una especie menos habrá de la otra y viceversa.

Por ello:

$$\frac{Z_1}{dt} - \frac{Z_2}{dt} = \mu - \theta X_L \quad (3.10)$$

Esto hace que la ecuación resultante sí que sea lineal y que si se realiza la integral a ambos lados se obtenga:

$$(Z_1 - Z_2)(t) = \int_0^t (\mu - \theta X_L(\tau)) d\tau \quad (3.11)$$

Se demuestra así que la acción de control que realiza este motivo antitético coincide con la respuesta que da un controlador integral. En un modelo realista las variables que entran en juego son mucho más numerosas y complican mucho más las ecuaciones, pero a nivel práctico, el objetivo que se obtendría sería similar.

3.4. Optimización multiobjetivo

Hasta ahora se ha llegado a describir con modelos matemáticos de forma bastante precisa, aunque realizando ciertas simplificaciones, algunos de los procesos que se llevan a cabo en el interior de la célula. Estos procesos han sido explicados mediante dos factores principales (Boada, Picó y Vignoni, 2017): i) la naturaleza de los componentes y las interacciones que se llevan a cabo en la realización de ciertas funciones y ii) cómo estos componentes evolucionan a lo largo del tiempo al realizar estas funciones a través de las dinámicas de sus especies.

En la mayoría de los casos es muy difícil determinar o conocer los valores concretos que adoptan los parámetros involucrados en todos los procesos. Por ello, muchos de ellos se obtienen a partir de unos objetivos fijados, es decir, en base al cumplimiento de unos requisitos u otros, ciertas variables adoptarán unos valores u otros. Un método muy utilizado que cubre estas necesidades es la **Optimización Multiobjetivo**. Este modelo es muy útil cuando se tienen una serie de objetivos o requisitos que cumplir para alcanzar las metas propuestas. En los sistemas biológicos el ajuste mediante optimización multiobjetivo (MOOT) permite obtener los valores de los parámetros descritos en los modelos cinéticos o dinámicos de los circuitos sintéticos. Con el MOOT se logran trazar las pautas para seleccionar estos parámetros y, a su vez, se consiguen los valores de estos que permiten a los sistemas biológicos comportarse como se desee (Boada, Picó y Vignoni, 2017).

Desde el punto de vista matemático, estos modelos de MOOT se puede definir como:

$$\min_{\theta} J(\theta) = [J_1(\theta), \dots, J_m(\theta)]$$

Sujeto a:

$$\begin{aligned} g(\theta) &\leq 0 \\ h(\theta) &= 0 \\ \underline{\theta}_i \leq \theta_i &\leq \overline{\theta}_i, i = [1, \dots, n] \end{aligned} \quad (3.12)$$

Donde θ es el vector de las variables de decisión, es decir, las variables con las que el ajuste multiobjetivo ‘jugará’ para obtener los mejores valores para $J(\theta)$, que es el vector objetivo. Como se puede apreciar es un problema de minimización, pero se pueden resolver problemas de maximización si se define como $\max J_i(\theta) = -\min(-J_i(\theta))$. Por lo general, no se obtiene una única solución para los objetivos, por lo que se obtiene un **conjunto de Pareto**. Cada conjunto de Pareto ofrece un vector objetivo en el **Frente de Pareto**. Todo este conjunto de soluciones (Frente de Pareto) se dice que forman el óptimo de Pareto ya que son consideradas como igual de buenas y no existe una dominancia entre ellas (G. Reynoso 2014). Un objetivo se dice que es óptimo de Pareto si no existe otro vector objetivo mejor que él.

3.5. Materiales

Para poner en práctica todas las metodologías expuestas anteriormente y conseguir diseñar y ajustar con éxito el circuito seguidor serán necesarios una serie de elementos que se definirán a continuación.

Hardware

Para realizar tanto el diseño como las simulaciones de los diferentes modelos se ha utilizado un ordenador HP convertible de 8GB de RAM con procesador i5 Intel® Core™. El dispositivo viene equipado con una serie de Software que es el que permite implementar los modelos y realizar tanto la simulación como la optimización.

Software

Para poder realizar el estudio de los diferentes modelos y posibilidades se han necesitado herramientas informáticas para facilitar la resolución tanto de los sistemas de ecuaciones diferenciales como de los algoritmos de optimización. El programa utilizado para todo ello ha sido el **Matlab®** que utiliza el lenguaje de programación **M** para plasmar en él toda la información y operaciones necesarias.

4. Estudios previos y justificación

Los avances y descubrimientos que se han producido durante los últimos años en los campos presentados anteriormente, y en la Biología Sintética en general, han posibilitado su aplicación en diferentes estrategias de control sobre circuitos biomoleculares. Se han utilizado controladores basados en una acción proporcional para reducir la variabilidad en la expresión de proteínas (Becksei y Serrano, 2000). En otros casos, como ya se ha mencionado en puntos anteriores y como se verá en los siguientes, se ha empleado el motivo antitético para implementar acciones de control proporcional-integrales con realimentación (Briat et al., 2018). Por el contrario, no se ha hecho mucho hincapié en desarrollar circuitos sintéticos que estimen la derivada de una señal dada. Sin embargo, la estimación de la derivada de una señal está íntimamente relacionada y es un concepto clave para el seguimiento perfecto de esta. En todos ellos, la estrategia de control es *in vivo*.

Una primera aproximación para conseguir un seguimiento perfecto de una señal, y que también es *in vivo*, fue realizada por Picó, Vignoni et al. (2019) y servirá como punto de partida para realizar un modelo más realista y preciso que pretenda obtener un resultado similar.

El circuito consta (como se observa en la Figura 4.1) de un motivo Brink (Cuba Samaniego, Franco et al., 2017) realimentado por un motivo antitético. La y es la señal de entrada del circuito y a la cual hay que seguir y \hat{y} es la señal de salida del circuito (la señal que sigue a la entrada), que se utiliza para realimentar y simular la comparativa que se utiliza en teoría de control y que se lleva a cabo en el circuito antitético del motivo Brink ($e = y - \hat{y}$). Esta última comparativa se lleva a cabo realmente a través de las especies x_1 y x_2 en el proceso de secuestro dentro del motivo antitético. Dado que son dos especies pertenecientes al grupo de factores extra citoplasmáticos, tratan de interactuar entre si para formar el complejo x_1x_2 y, por tanto, cuanto más cantidad haya de una menos habrá de la otra y viceversa. El motivo Brink consiste en un motivo antitético seguido de un interruptor ultrasensible en el que la especie x_3 va variando entre su forma activa e inactiva debido a la acción de las especies x_1 y x_2 (Picó, Vignoni et al., 2019). Por último, la variable r , es una señal de entrada externa que se encarga de regular

el valor de la especie x_4 a través de x_5 en el segundo circuito antitético de la misma forma que lo hacen y e \hat{y} con x_1 y x_2 en la primera parte del sistema.

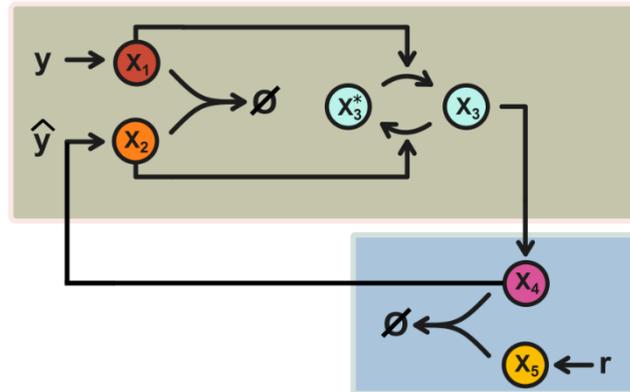


Figura 4.1 Representación del circuito biomolecular de seguimiento de señal

Ya en el modelo y la formulación utilizados en el estudio de Picó, Vignoni et al. (2019), que se realizó sin tener en cuenta ciertas interacciones entre especies y procesos que en el supuesto real sí que se llevan a cabo en este tipo de circuitos, se observa que no es trivial alcanzar el objetivo de seguir a una señal mediante el control de señales biológicas ni obtener los valores adecuados de los parámetros que lo logran y, también, cómo estos resultados dependen fuertemente de los valores que adoptan estos últimos (Figura 4.2). Por ello, nace la necesidad de ir ampliando la complejidad de los modelos utilizados para que se correspondan cada vez más a lo que sucede en la realidad, y de realizar un estudio exhaustivo sobre los modelos que vayan surgiendo para obtener los parámetros necesarios de las especies y de los elementos que las forman para que se cumplan los objetivos deseados, que en este caso es la implementación del **seguidor biomolecular**.

En la Figura 4.2 se aprecia como en el modelo simplificado con pequeños cambios el modelo ya no cumple su función y justifica que no es sencilla la obtención de valores para los parámetros que cumplan con ello.

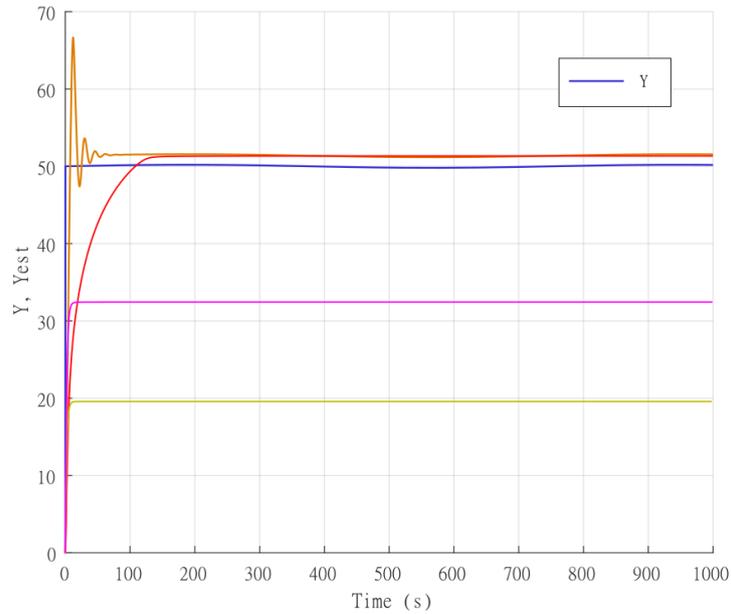


Figura 4.2 Barrido de simulaciones del modelo de Picó, Vignoni et al., 2019

5. Diseño y resultados

5.1. Diseño del modelo

Para la obtención de un nuevo modelo para este circuito se ha utilizado la metodología **determinista** y se ha desestimado la componente probabilística, estocástica, que contienen todas las reacciones bioquímicas. Sigue siendo una aproximación, pero es de mucha utilidad para seguir mejorando el estudio de estos circuitos.

Partiendo de la representación del circuito de la Figura 4.1, el modelo de nuestro seguidor biomolecular expresado en SBOL se puede describir como en la Figura 5.1

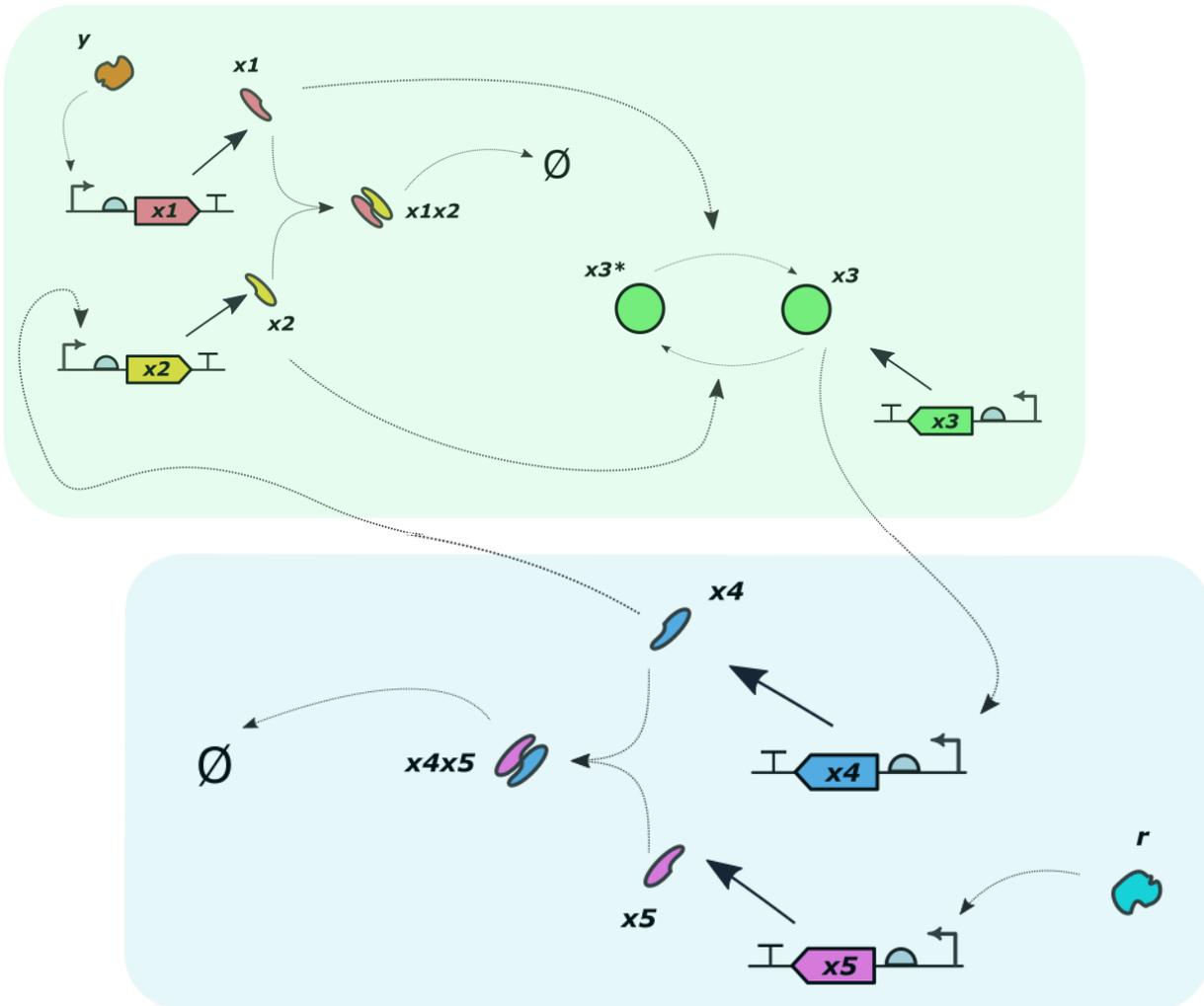
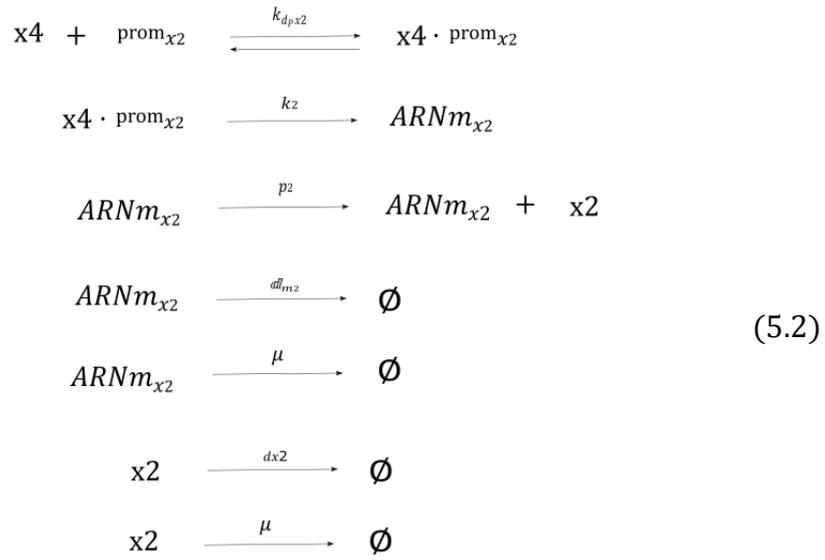
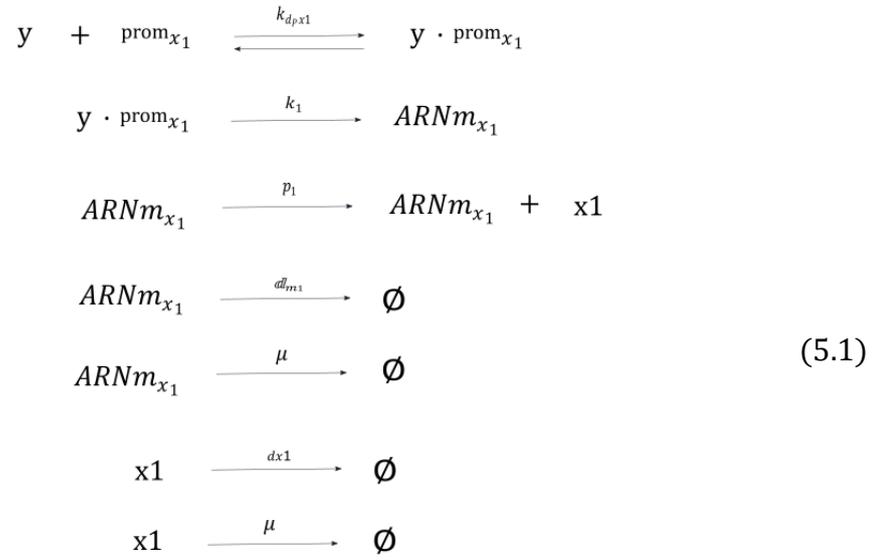
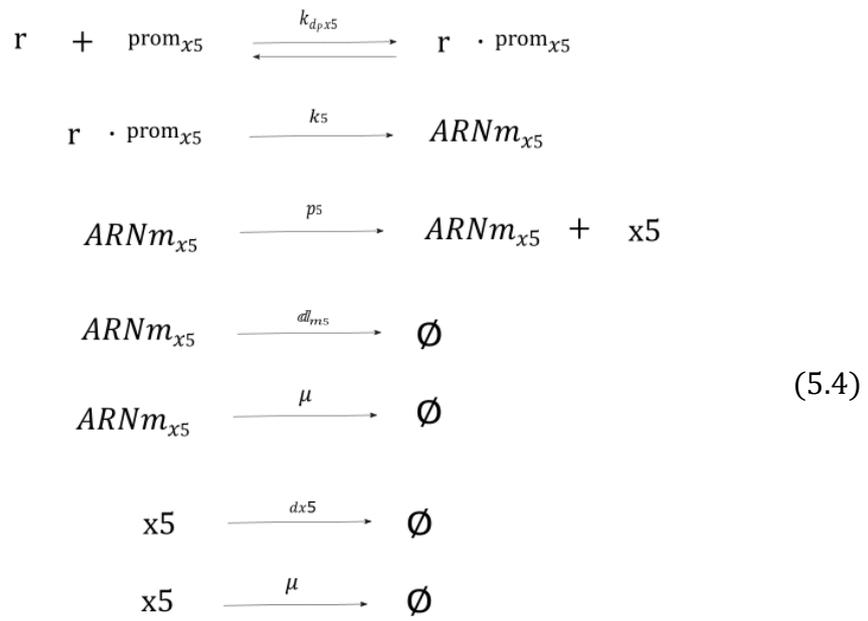
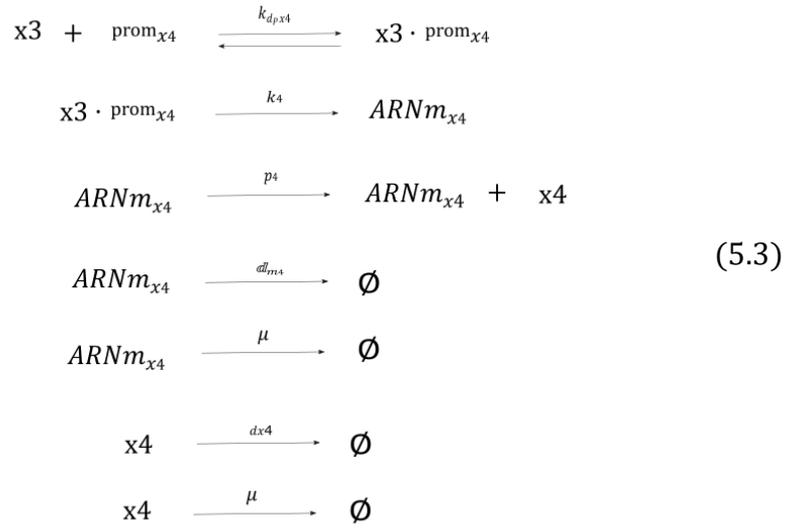
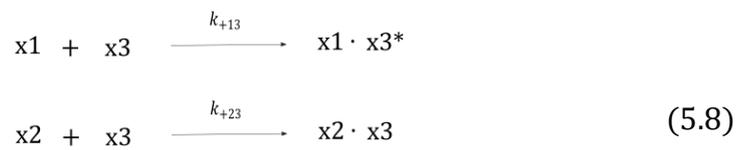
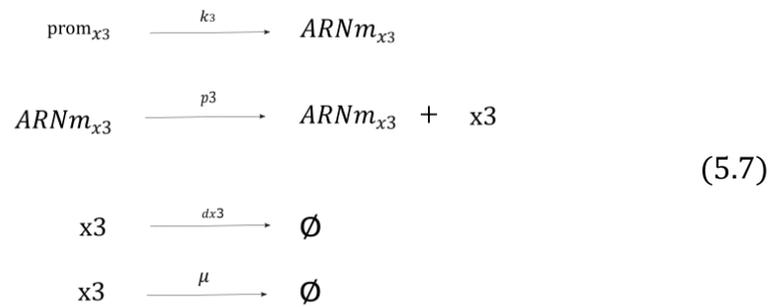
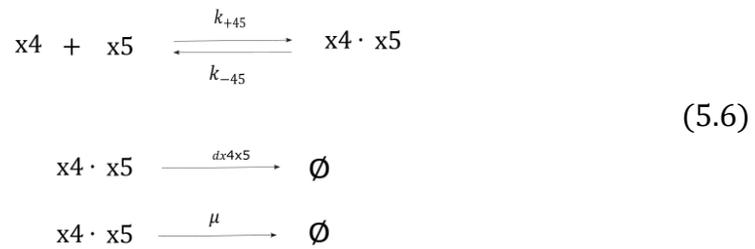
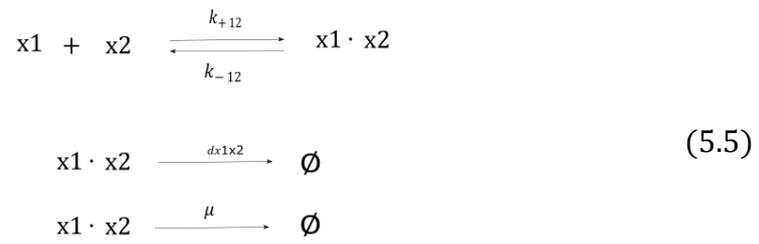


Figura 5.1 Descripción del circuito con lenguaje SBOL.

Haciendo uso de la topología del modelo se puede observar como las siguientes reacciones se llevan a cabo dentro del sistema. En ellas, se describen los procesos de transcripción, traducción, degradación, dilución, etc. de las especies involucradas en nuestro circuito.







Estas reacciones se obtienen de los procesos de transcripción y traducción que se llevan a cabo en las especies de nuestro circuito. También se han tenido en cuenta los factores de degradación y crecimiento celular de cada una de las proteínas y complejos generados.

-Supuestos particulares

Para generar los modelos y las ecuaciones de nuestro sistema se han tenido en cuenta ciertos supuestos adicionales a valorar:

1. Solamente se tiene en cuenta la dinámica de las proteínas ya que se considera que la transcripción se produce muy rápidamente en comparación con la traducción por lo que se asume que se encuentra en un estado cuasi-estacionario. La **aproximación cuasi-estacionaria (QSSA)** permite reducir el volumen de los modelos considerando que la evolución dinámica de ciertos procesos dentro del sistema sucede a velocidades muy elevadas en comparación con otros y que, por tanto, la evolución en el tiempo de las especies biológicas involucradas es nula.
2. El proceso de traducción se ha modelado como una reacción irreversible a pesar de que, realmente, el proceso de unión de los ribosomas al RBS es un proceso reversible.
3. Para todas las especies se consideran la degradación y la dilución debido al crecimiento celular. Esto se tiene en cuenta mediante una ratio de degradación d_{x_j} y otra ratio para el crecimiento específico de la célula μ .
4. Para cada promotor, cualquier proceso de transcripción que incluya factores de transcripción (TF) se han modelado utilizando una función de tipo Hill con la modificación realizada en Trabelsi et al., 2018. En esta aproximación se tiene en cuenta que el número de TF y sitios de unión se escalan con el número de copias del plásmido, C_N (Boada et al., 2020).

Si se tienen en cuenta todas las suposiciones anteriores y se utilizan las conocidas como **funciones de Hill** para expresar los promotores de las especies del circuito, tras aplicar la **Ley de Acción de Masas** a todas las proteínas del sistema se obtiene el sistema de ecuaciones diferencias del modelo reducido, que representa las concentraciones de las proteínas que son de interés a lo largo del tiempo.

Se obtiene entonces el siguiente modelo reducido:

$$\begin{aligned} \dot{x}_1 &= \frac{p_1 C_{N_1} k_1}{d_{m_1} + \mu} \left(\frac{[y]^2}{k_{d_{p_{x1}}} \cdot (k_{d_{x1}} \cdot C_{N_1})^2 + [y]^2} \right) - k_{+13}[x_1]([c] - [x_3]) - k_{+12}[x_1][x_2] + k_{-12}[x_1 x_2] - (d_{x_1} + \mu)[x_1] \\ \dot{x}_2 &= \frac{p_2 C_{N_2} k_2}{d_{m_2} + \mu} \left(\frac{[x_4]^2}{k_{d_{p_{x2}}} \cdot (k_{d_{x2}} \cdot C_{N_2})^2 + [x_4]^2} \right) - k_{+23}[x_2][x_3] - k_{+12}[x_1][x_2] + k_{-12}[x_1 x_2] - (d_{x_2} + \mu)[x_2] \\ \frac{d[x_1 x_2]}{dt} &= k_{+12}[x_1][x_2] - k_{-12}[x_1 x_2] - (d_{x_1 x_2} + \mu)[x_1 x_2] \\ \dot{x}_3 &= k_{+13}[x_1]([c] - [x_3]) - k_{+23}[x_2][x_3] - (d_{x_3} + \mu)[x_3] \\ \dot{x}_4 &= \frac{p_4 C_{N_4} k_4}{d_{m_4} + \mu} \left(\frac{[x_3]^2}{k_{d_{p_{x4}}} \cdot (k_{d_{x4}} \cdot C_{N_4})^2 + [x_3]^2} \right) - k_{+45}[x_4][x_5] + k_{-45}[x_4 x_5] - (d_{x_4} + \mu)[x_4] \\ \dot{x}_5 &= \frac{p_5 C_{N_5} k_5}{d_{m_5} + \mu} \left(\frac{[r]^2}{k_{d_{p_{x5}}} \cdot (k_{d_{x5}} \cdot C_{N_5})^2 + [r]^2} \right) - k_{+45}[x_4][x_5] + k_{-45}[x_4 x_5] - (d_{x_5} + \mu)[x_5] \\ \frac{d[x_4 x_5]}{dt} &= k_{+45}[x_4][x_5] - k_{-45}[x_4 x_5] - (d_{x_4 x_5} + \mu)[x_4 x_5] \end{aligned} \tag{5.9}$$

Variable	Descripción
y	Señal de entrada
x_1	Especie 1
x_2	Especie 2
x_3	Especie 3
c	Suma de componentes activa e inactiva de x_3 ($c = x_3 + x_3^*$)
x_4	Especie 4
x_5	Especie 5
$x_1 x_2$	Complejo $x_1 x_2$
$x_4 x_5$	Complejo $x_4 x_5$

Tabla 5.1 Especies del modelo reducido

Diseño y ajuste de un circuito seguidor biomolecular

Parámetros	Descripción	Unidades
Generales		
μ	Tasa de crecimiento	min^{-1}
x_1		
C_{N_1}	Número de copias del plásmido	copias
k_1	Ratio de transcripción	min^{-1}
p_1	Ratio de traducción	min^{-1}
d_{m_1}	Ratio de degradación de $ARNm_{x_1}$	min^{-1}
k_{dpx1}	Constante de disociación a promotor	moléculas
k_{dx1}	Constante de disociación de x_1	moléculas
d_{x1}	Ratio de degradación de x_1	min^{-1}
x_2		
C_{N_2}	Número de copias del plásmido	copias
k_2	Ratio de transcripción	min^{-1}
p_2	Ratio de traducción	min^{-1}
d_{m_2}	Ratio de degradación de $ARNm_{x_2}$	min^{-1}
k_{dpx2}	Constante de disociación a promotor	moléculas
k_{dx2}	Constante de disociación de x_2	moléculas
d_{x2}	Ratio de degradación de x_2	min^{-1}
Complejo x_1x_2		
k_{+12}	Ratio de unión de x_1 con x_2	$min^{-1} \text{ moléculas}^{-1}$
k_{-12}	Ratio de disociación de x_1x_2	min^{-1}
d_{x1x2}	Ratio de degradación de x_1x_2	min^{-1}
x_3		
k_{+13}	Ratio de unión de x_1 con x_3^*	$min^{-1} \text{ moléculas}^{-1}$
k_{+23}	Ratio de unión de x_1 con x_3	$min^{-1} \text{ moléculas}^{-1}$
x_4		
C_{N_4}	Número de copias del plásmido	copias
k_4	Ratio de transcripción	min^{-1}
p_4	Ratio de traducción	min^{-1}
d_{m_4}	Ratio de degradación de $ARNm_{x_4}$	min^{-1}
k_{dpx4}	Constante de disociación a promotor	moléculas
k_{dx4}	Constante de disociación de x_4	moléculas
d_{x4}	Ratio de degradación de x_4	min^{-1}
x_5		
C_{N_5}	Número de copias del plásmido	copias
k_5	Ratio de transcripción	min^{-1}
p_5	Ratio de traducción	min^{-1}
d_{m_5}	Ratio de degradación de $ARNm_{x_5}$	min^{-1}
k_{dpx5}	Constante de disociación a promotor	moléculas
k_{dx5}	Constante de disociación de x_5	moléculas
d_{x5}	Ratio de degradación de x_5	min^{-1}
Complejo x_4x_5		
k_{+45}	Ratio de unión de x_4 con x_5	$min^{-1} \text{ moléculas}^{-1}$
k_{-45}	Ratio de disociación de x_4x_5	min^{-1}
d_{x4x5}	Ratio de degradación de x_4x_5	min^{-1}

Tabla 5.2 Parámetros del modelo reducido

Una vez obtenido el modelo reducido de ecuaciones que representan la dinámica de las especies utilizadas el siguiente paso es realizar la implementación en MATLAB® y calcular las primeras simulaciones para resolver el sistema diferencial de ecuaciones del modelo.

Para las simulaciones se han elaborado dos archivos y una función (ver Anexo A.1). El archivo **param.m** contiene los valores de los parámetros del modelo, el archivo **ode_newmodelJaimeAnton.m** contiene las ecuaciones diferenciales del modelo implementadas y, por último, la función **main_Jaimeprogram.m** “llama” a los otros dos programas y ejecuta el *solver* ODE23 para obtener los resultados del sistema de ecuaciones del modelo reducido partiendo de unas condiciones iniciales (Tabla 5.1). Se ha escogido el ODE23 ya que es un buen *solver* para sistemas rígidos y con poca tolerancia al error.

Variable	Descripción	Valor	Unidades
tfin	Tiempo de simulación	1000	segundos
Step	Paso temporal de la simulación	0.0032	segundos
AbsTol	Tolerancia absoluta del <i>solver</i>	1×10^{-3}	adimensional
RelTol	Tolerancia relativa del <i>solver</i>	1×10^{-6}	adimensional
Estado inicial	Valor inicial de las especies	0	moléculas

Tabla 5.3 Parámetros iniciales de las simulaciones

Para realizar una primera simulación es necesario dotar de valores numéricos a los parámetros del modelo (Tabla 5.2). Para ello, se realizará un primer barrido con valores dentro de los rangos típicos que abarcan estos parámetros. Estos rangos y valores se han obtenido de trabajos previos como R. Monfort (2019) o Y. Boada (2018), entre otros.

Como se puede observar en la Figura 5.2 ninguna de las combinaciones de parámetros consigue el objetivo. Esto confirma que no es trivial encontrar una combinación de valores que hagan que la especie x_4 , que es la que estamos tratando de controlar, siga a la señal de entrada y .

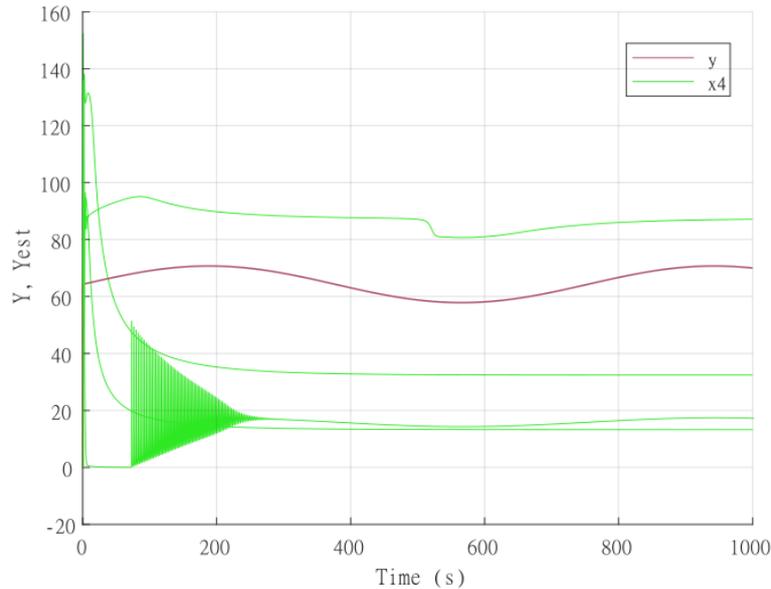


Figura 5.2 Simulaciones con un barrido de parámetros. La y es la señal de entrada y $x4$ la variable que trata de seguirla y la cual estamos tratando de controlar y modelar

Para obtener los resultados deseados se va a utilizar la **Optimización Multiobjetivo**. Como base, se van a utilizar los algoritmos desarrollados por G. Reynoso Meza (2014), pero los vamos a adaptar para que se ajusten a nuestro modelo reducido y cumplan los objetivos que se necesitan. El optimizador consta de tres programas (ver Anexo A.2). El programa **spMODEx.m** implementa el algoritmo que permite la optimización, en el **MOEAparam.m** se determinan el número de objetivos, las variables de decisión, los rangos que estas últimas pueden tener y los valores que se consideran aceptables o no para los objetivos. Por último, la **CostFunction.m** incluye parte del modelo reducido y arroja como resultados los valores del conjunto y el frente de Pareto.

Los objetivos que se han implementado son:

- Integral del error en valor absoluto (IAE)
- Integral del error absoluto ponderado en el tiempo (ITAE). Se penalizan con mayor dureza los errores en la estimación conforme el tiempo de simulación avanza.
- Función de cruce por cero. Trata de reducir el número de veces que la señal estimada cruza la señal de entrada a seguir, es decir, es una función que evita una oscilación excesiva de la señal estimada.

Como variables de decisión se tomaron los siguientes parámetros:

Parámetros	Descripción
p_1	Ratio de traducción
p_4	Ratio de traducción
p_5	Ratio de traducción
d_{m_4}	Ratio de degradación de $ARNm_{x_4}$
d_{m_5}	Ratio de degradación de $ARNm_{x_5}$
k_{+12}	Ratio de unión de x_1 con x_2

Tabla 5.4 Variables de decisión

5.2. Resultados y discusión

Tras varias simulaciones y ajustes tanto en los rangos de los parámetros como en los valores aceptables de los objetivos se obtuvo la siguiente solución de parámetros (ver Tabla 5.5). Esta solución incluye tanto los valores de los parámetros que en las optimizaciones eran fijos como los valores finales de las variables de decisión obtenidas vía optimización.

Parámetros	Valores	Unidades	Parámetros	Valores	Unidades
μ	8.5×10^{-3}	min^{-1}	C_{N_4}	10	copias
C_{N_1}	10	copias	k_4	0.4498	min^{-1}
k_1	9.4657	min^{-1}	p_4	500	min^{-1}
p_1	9.63998	min^{-1}	d_{m_4}	0.01	min^{-1}
d_{m_1}	0.231	min^{-1}	k_{d_px4}	50	moléculas
k_{d_px1}	50	moléculas	k_{dx4}	5	moléculas
k_{dx1}	5	moléculas	d_{x4}	3×10^{-4}	min^{-1}
d_{x1}	3×10^{-4}	min^{-1}	C_{N_5}	10	copias
C_{N_2}	10	copias	k_5	14.2672	min^{-1}
k_2	11.0286	min^{-1}	p_5	234.6479	min^{-1}
p_2	47.1758	min^{-1}	d_{m_5}	0.57432	min^{-1}
d_{m_2}	1.3999	min^{-1}	k_{d_px5}	50	moléculas
k_{d_px2}	50	moléculas	k_{dx5}	5	moléculas
k_{dx2}	5	moléculas	d_{x5}	3×10^{-4}	min^{-1}
d_{x2}	3×10^{-4}	min^{-1}	k_{+45}	0.0001	$min^{-1} \text{ moléculas}^{-1}$
k_{+12}	69.5121	$min^{-1} \text{ moléculas}^{-1}$	k_{-45}	0.01	min^{-1}
k_{-12}	0.01	min^{-1}	d_{x4x5}	3×10^{-4}	min^{-1}
d_{x1x2}	3×10^{-4}	min^{-1}	c	300	moléculas
k_{+13}	0.01	$min^{-1} \text{ moléculas}^{-1}$	r	500	moléculas
k_{+23}	0.01	$min^{-1} \text{ moléculas}^{-1}$			

Tabla 5.5 Valores de los parámetros del modelo reducido

Si volvemos a simular ahora el modelo reducido en MATLAB® con los valores de los parámetros se obtiene lo siguiente:

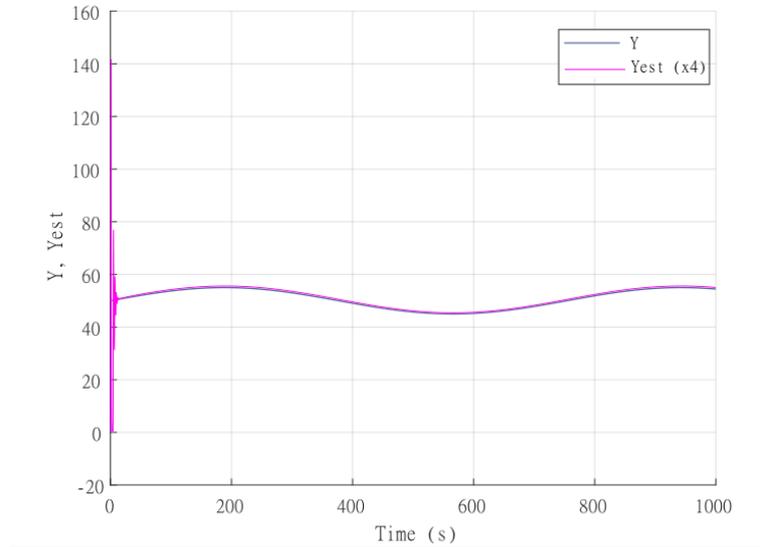


Figura 5.3 Simulación final del modelo

Como se puede apreciar en la Figura 5.3 ahora sí se consigue el objetivo y nuestra señal estimada (x_4) consigue seguir a la señal de entrada dada, en este caso una señal de tipo sinusoidal con un valor medio de 50 moléculas, en un tiempo aceptable, inferior a los 15 segundos. A continuación, se representará la evolución de cada una de las especies:

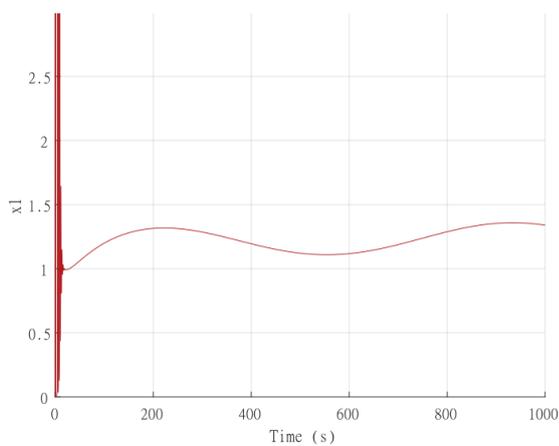


Figura 5.4 Evolución de la concentración de la especie x_1

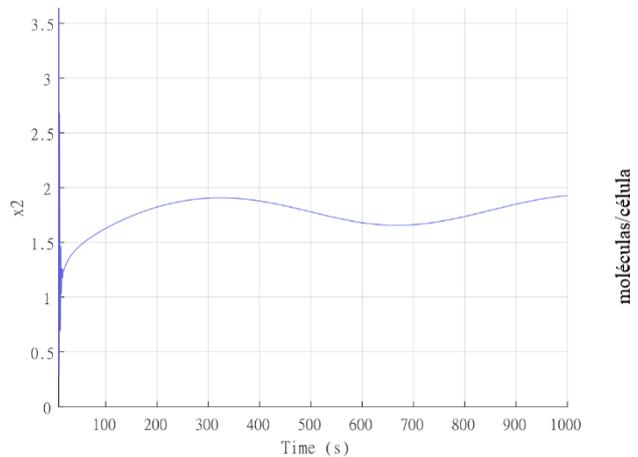


Figura 5.5 Evolución de la concentración de la especie x_2

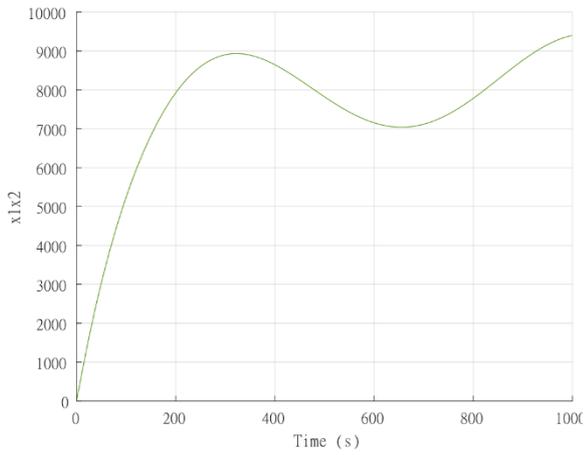


Figura 5.6 Evolución de la concentración de la especie x_{1x2}

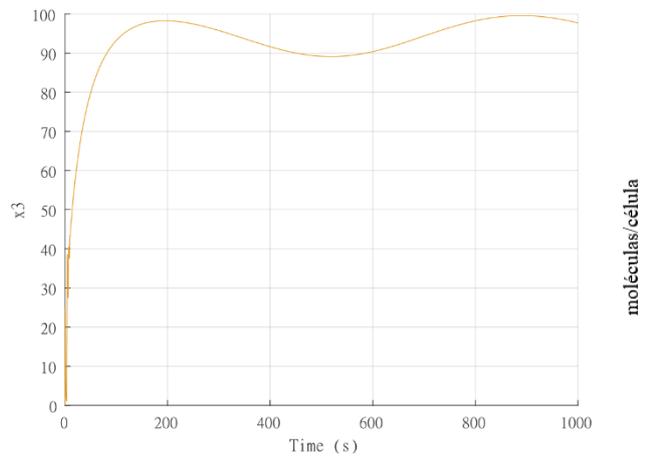


Figura 5.7 Evolución de la concentración de la especie x_3

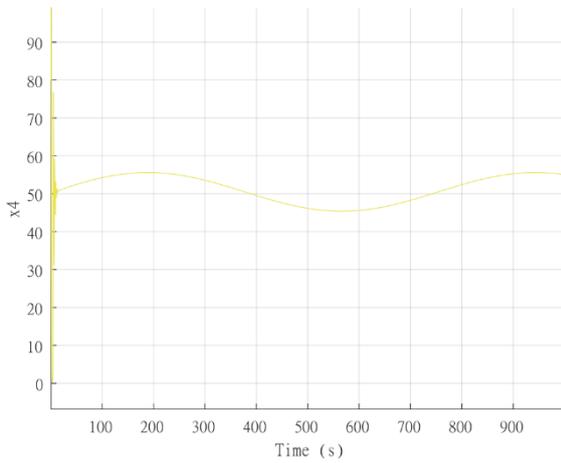


Figura 5.8 Evolución de la concentración de la especie x_4

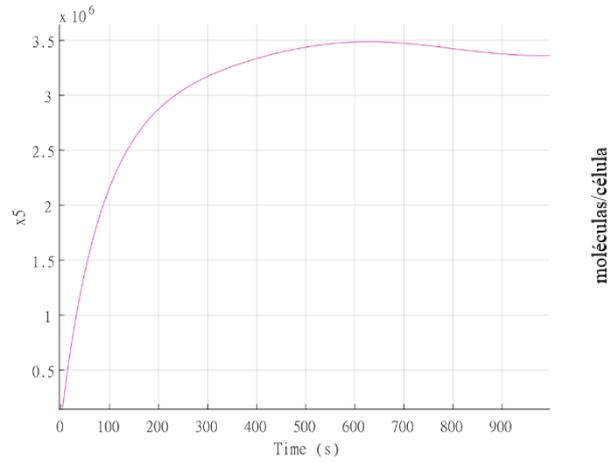


Figura 5.9 Evolución de la concentración de la especie x_5

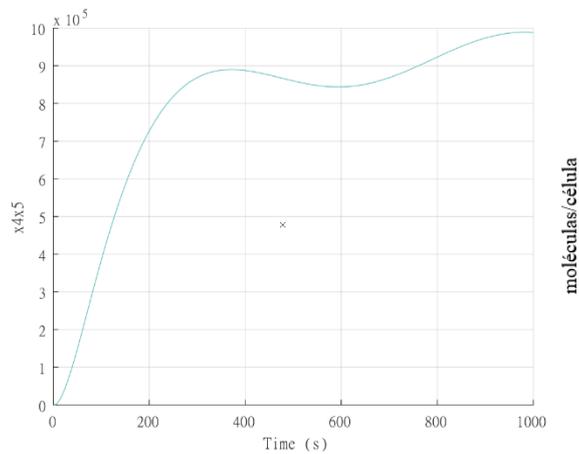


Figura 5.10 Evolución de la concentración de la especie x_{4x5}

También se puede observar (Figura 5.11) como el modelo sirve para señales de distinta magnitud, es decir, que es adecuado para cualquier amplitud que se le dé a la señal de entrada.

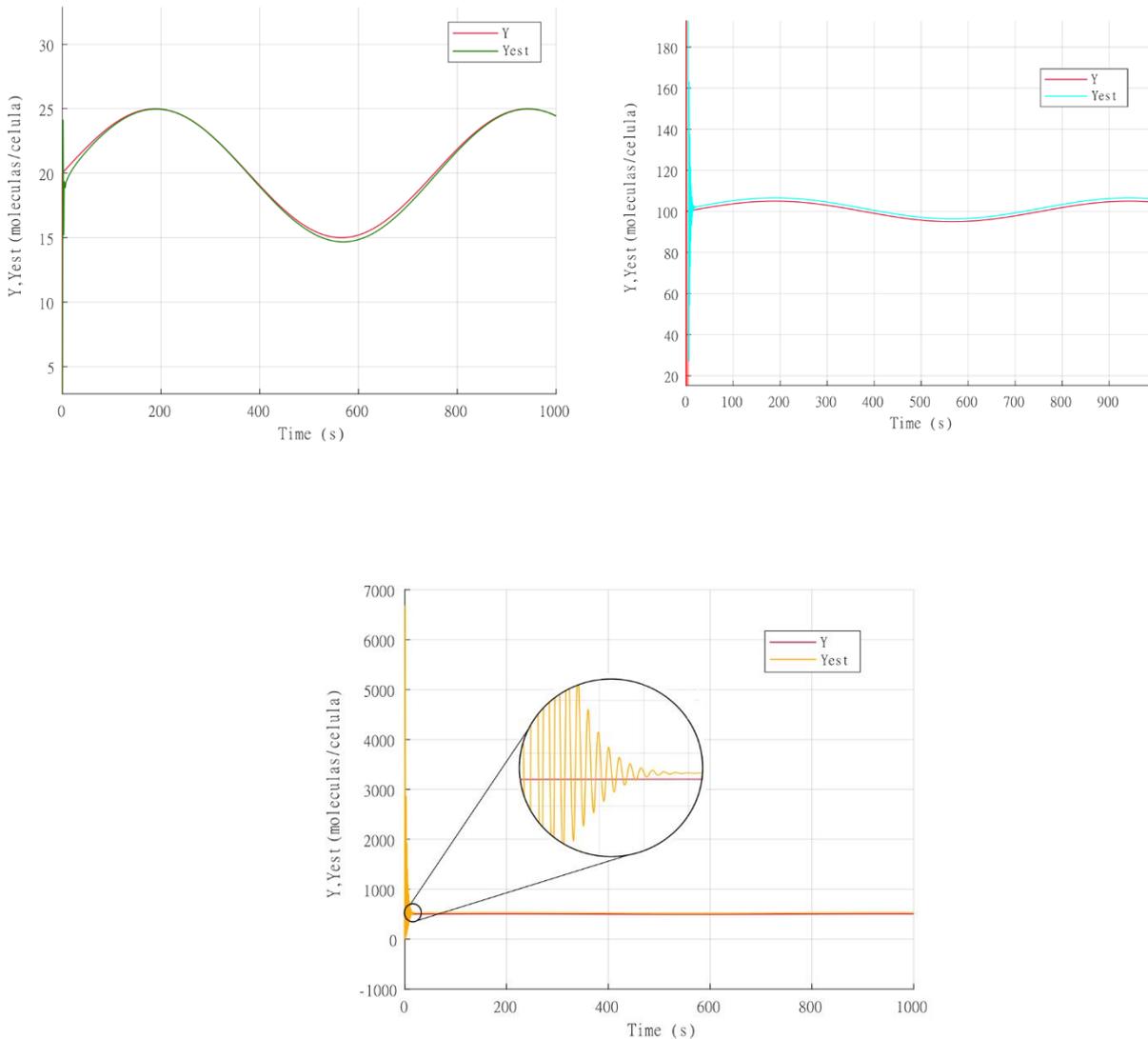
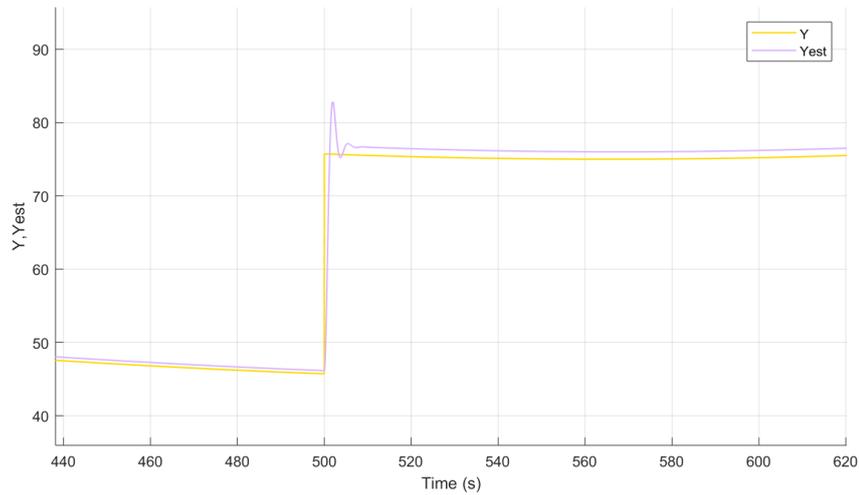


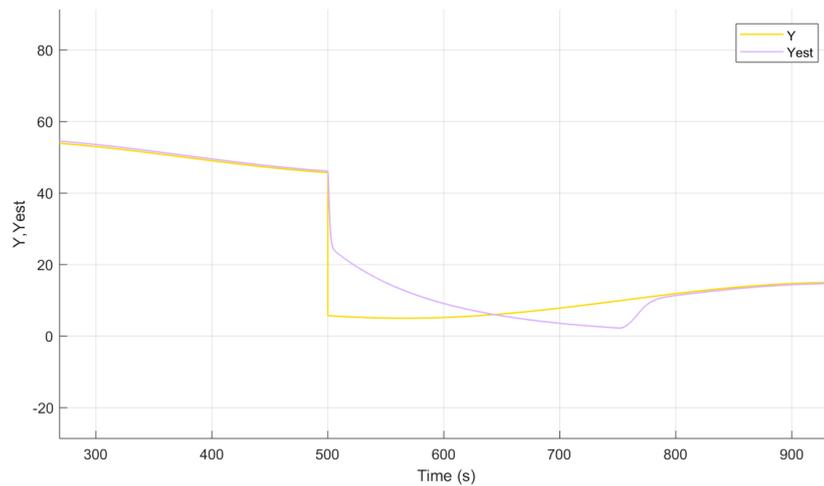
Figura 5.11 Respuesta frente a diferentes valores de la señal de entrada

En la Figura 5.11 se observa como a medida que se aumenta el valor medio de la señal de entrada la oscilación inicial es más pronunciada, pero logra seguir a la señal en un tiempo similar en todos los casos.

Otro resultado muy interesante es que es capaz de soportar perturbaciones o cambios bruscos en la señal de entrada y el tiempo de reacción para las mismas es muy reducido también (Figura 5.12).



(a)



(b)

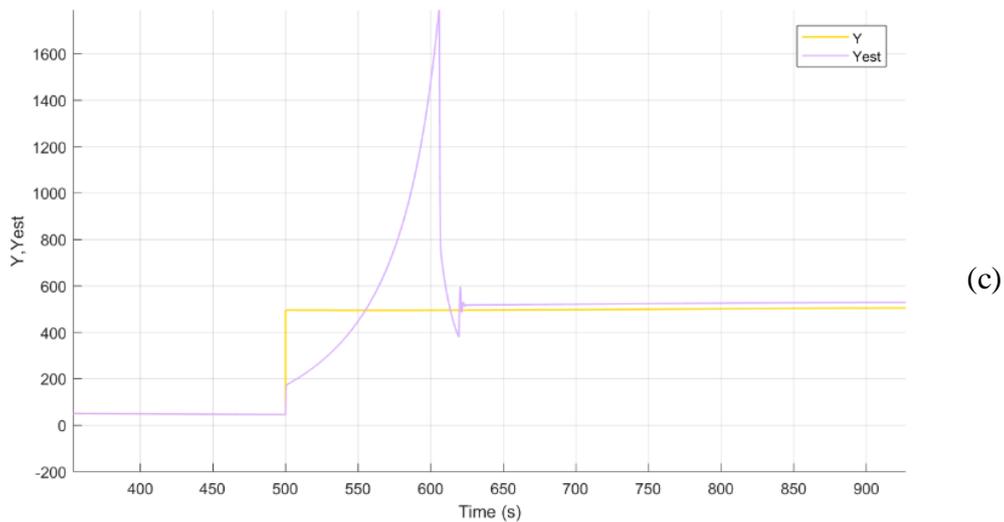


Figura 5.12 Diferentes respuestas frente a oscilaciones o perturbaciones

Se observa en la Figura 5.12 (a) cómo para pequeñas oscilaciones el modelo responde muy bien y se consigue seguir a la señal con un cierto error.

Se pueden establecer los límites de oscilación para los que este modelo es efectivo. Como se observa en la Figura 5.12 (b) cuando la variación supone una disminución del valor medio de la señal del 500% se aprecia como el modelo pierde velocidad y comienza a incrementarse de manera pronunciada su tiempo de respuesta. Para incrementos en el valor de la señal del 1000%, Figura 5.12 (c), se observa como el modelo tampoco consigue seguir a la señal de manera efectiva y se producen picos de gran magnitud. Por tanto, para los límites de oscilación que aceptaría el modelo se cuadrarán en ese rango.

Por todo ello, se puede decir que hemos conseguido un circuito que consigue el objetivo de seguir a diferentes señales de entrada y consigue adaptarse frente a oscilaciones que se puedan producir en la misma dentro de un rango definido. Además, se consigue todo esto en un tiempo muy reducido por lo que la respuesta temporal es buena.

Parte II

Presupuesto

Presupuesto

6.1. Introducción y objetivo

Para todo proyecto ingenieril resulta imprescindible conocer una valoración económica del mismo. Para ello, es necesario elaborar un presupuesto en el que se estudien todos los elementos necesarios para el desarrollo del Trabajo desde un punto de vista económico.

6.2. Presupuesto

A continuación, se expondrán las diferentes partidas que son necesarias para llevar a cabo el **Diseño y ajuste de un circuito seguidor biomolecular**. Como se puede observar, se han tenido en cuenta tanto los aspectos materiales como la mano de obra necesaria para llevarlo a cabo.

PRESUPUESTO DESCOMPUESTOS Y MEDICIONES

CÓDIGO	RESUMEN	CANTIDAD	PRECIO	IMPORTE
01	SEGUIDOR BIOMOLECULAR			
01.01	Ud Diseño y ajuste de un circuito seguidor biomolecular			
	Diseño, desarrollo y control de un circuito seguidor biomolecular mediante el uso de software especializado.			
	Descomposición			
I Sen	Hr Ingeniero Senior	100,00	12,50	1.250,00
I Jun	Hr Ingeniero Junior	300,00	6,00	1.800,00
HP360	Ud Ordenador portatil	1,00	854,00	854,00
WIN10	Ud Licencia Anual Microsoft Windows 10	0,25	150,00	37,50
MATR2017B	Ud Licencia Anual Matlab R2017b	0,25	250,00	62,50
INKS	Ud Licencia Anual Inkscape	0,25	0,00	0,00
%ind	% Costes Indirectos	40,04	5,00	200,20
Total cantidades alzadas		1,00		
		1,00	4.204,20	4.204,20
TOTAL 01				4.204,20
TOTAL				4.204,20

PRESUPUESTO Y MEDICIONES

CÓDIGO	RESUMEN	UDS	LONGITUD	ANCHURA	ALTURA	CANTIDAD	PRECIO	IMPORTE
01	SEGUIDOR BIOMOLECULAR							
01.01	Ud Diseño y ajuste de un circuito seguidor biomolecular Diseño, desarrollo y control de un circuito seguidor biomolecular mediante el uso de software especializado.							
						1,00	4.204,20	4.204,20
	TOTAL 01.....							4.204,20
	TOTAL.....							4.204,20

RESUMEN DE PRESUPUESTO

CAPÍTULO	RESUMEN	IMPORTE
01	SEGUIDOR BIOMOLECULAR	4.204,20
	PRESUPUESTO DE EJECUCIÓN MATERIAL	4.204,20
	13,00 % Gastos generales	546,55
	6,00 % Beneficio industrial	252,25
	Suma	798,80
	PRESUPUESTO DE EJECUCION POR CONTRATA SIN IVA	5.003,00
	21% IVA	1.050,63
	PRESUPUESTO DE EJECUCIÓN POR CONTRATA	6.053,63

Asciende el presupuesto a la expresada cantidad de SEIS MIL CINCUENTA Y TRES EUROS con SESENTA Y TRES CÉNTIMOS

Valencia, 3 de septiembre 2021.

Anexo A

A.1 Código de Matlab para la simulación del modelo

- **param.m.** Valores de los parámetros del modelo

```
function p=param()
%% Parameters Supertwisting:
p.a=5;
p.b=5;

%Tasa de crecimiento:
p.mu=8.5e-3; %min-1

%Tasas de degradación de la proteína:
p.dx1=3e-4;
p.dx2=3e-4;
p.dx3=3e-4;
p.dx4=3e-4; %min-1
p.dx5=3e-4;
p.dx1x2=3e-4;
p.dx4x5=3e-4;

%Número de copias de los plásmidos
p.Cn1=10;
p.Cn2=10;
p.Cn4=10; %Copias --> [1,100]
p.Cn5=10;

%Tasas de degradación del mRNA:
p.dm1=0.231;
p.dm2=1.3999; %min-1
p.dm4=0.01;
p.dm5=0.57432;

%Tasas de traducción:
p.p1=9.63998;
p.p2=47.1758;
p.p4=500; %[1,100]min-1
p.p5=234.6479;

%Tasas de transcripción:
p.k1=9.4657;
p.k2=11.0286;
p.k4=0.4498; %[0.5,10]min-1
p.k5=14.2672;

%Tasas de afinidad entre moléculas de x:
p.kdx1=5;
p.kdx2=5;
p.kdx4=5; %[1,500]molec
p.kdx5=5;

%Tasas de afinidad entre moléculas x y el promotor:
p.kdpx1=50;
p.kdpx2=50;
p.kdpx4=50; %[50,5500]molec
p.kdpx5=50;

% Sequestration in the antithetic motif x4x5
p.kplus45=0.0001;
```

```

% Sequestration in the antithetic motif x1x2
p.kplus12=69.5121;

% Sequestration in the brink comparator:
p.kplus13=0.01;
p.kplus23=0.01;

%Opposite of sequestration
p.kminus12=0.01;
p.kminus45=0.01;

% Parameters brink switch:
p.c=300; % Sum active+inactive states

% parameter r in the antithetic motif
p.r=500;

% Parameter input signal
p.d6=500;

end

```

- **ode_newmodelJaimeAnton.** Implementación de las ecuaciones del modelo reducido

```

function [out] = ode_newmodelJaimeAnton(t,x,p)
x1=x(1);
x2=x(2);
x12=x(3);
x3=x(4);
x4=x(5);
x5=x(6);
x45=x(7);
y=x(8);
x7=x(9);
x8=x(10);

%d_x1
out(1,1)=((p.p1*p.Cn1*p.k1/(p.dm1+p.mu))*(y^2/(p.kdpx1*(p.kdx1*p.Cn1)^2+y^2)))-p.kplus13*x1*(p.c-x3)-p.kplus12*x1*x2+p.kminus12*x12-(p.dx1+p.mu)*x1;

%d_x2
out(2,1)=(((p.p2*p.Cn2*p.k2)/(p.dm2+p.mu))*((x4^2)/(p.kdpx2*(p.kdx2*p.Cn2)^2+x4^2)))-p.kplus23*x2*x3-p.kplus12*x1*x2+p.kminus12*x12-(p.dx2+p.mu)*x2;

%d_x1x2
out(3,1)=p.kplus12*x1*x2-p.kminus12*x12-(p.dx1x2+p.mu)*x12;

%d_x3
out(4,1)=p.kplus13*x1*(p.c-x3)-p.kplus23*x2*x3-(p.dx3+p.mu)*x3;

%d_x4
out(5,1)=(((p.p4*p.Cn4*p.k4)/(p.dm4+p.mu))*((x3^2)/(p.kdpx4*(p.kdx4*p.Cn4)^2+x3^2)))-p.kplus45*x4*x5+p.kminus45*x45-(p.dx4+p.mu)*x4;

%d_x5
out(6,1)=(((p.p5*p.Cn5*p.k5)/(p.dm5+p.mu))*((p.r^2)/(p.kdpx5*(p.kdx5*p.Cn5)^2+p.r^2)))-p.kplus45*x4*x5+p.kminus45*x45-(p.dx5+p.mu)*x5;

%d_x4x5
out(7,1)=p.kplus45*x4*x5-p.kminus45*x45-(p.dx4x5+p.mu)*x45;

%y
out(8,1) = p.d6*(50+5*sin(t/120)+0*cos(t/30)-y);

```

```
if (t>500)
    out(8,1) = p.d6*(500+5*sin(t/120)+0*cos(t/30)-y);
end

% To compare with the ideal supertwisting derivator:
out(9,1) = p.a*sqrt(abs(y-x7))*tanh(1000*(y-x7))+x8;

out(10,1) = p.b*tanh(1000*(y-x8));
end
```

- **main_JaimeProgram.m.** Llamada a las dos funciones anteriores, inicialización de variables e implementación del *solver* para el modelo reducido.

```
clear all;

%% Simulation options
tfin = 1000; % Tiempo de simulacion
opts = odeset('AbsTol',1e-3,'RelTol',1e-6); % opciones para el simulador ode

%% Simulate Model

initial_states = 0.*abs(rand(1,10));
p = param(); %Default values of parameters

%If we start from random initial conditios, it still works:
[t_D,x_D] = ode23t(@(t,x)ode_newmodelJaimeAnton(t,x,p), [0
tfin],initial_states,opts);

%[t_D,x_D] = ode15s(@(t,x) ode_ReducedUtkinDerivator(t,x,p), [0
tfin],zeros(1,8),opts);
```

A.2 Código de Matlab para la implementación de la optimización multiobjetivo

- **spMODEx.m.** Implementación de los algoritmos de optimización (G. Reynoso, 2014)
- **MOEApam.m.** Definición del número de objetivos, variables de decisión, rangos de valores para las variables y valores admisibles de los objetivos. Definición de las partes más importantes:

```

%% Variables regarding the multi-objective problem

spMODExDat.NOBJ = 3; % Number of decision objectives.

spMODExDat.NRES = 0; % Number of constraints plus
                    % non-decision objectives. That is,
                    % variables that are not used to
                    % built the spherical grid, but used
                    % in the norm computation.

spMODExDat.NVAR = 6; % Number of decision variables

spMODExDat.FieldD = [1 100;...
                    0.01 2;... %Tasas de discociación de la proteina 50
                    0.5 300;...
                    0.01 2;...
                    0.0001 75;...
                    1 500];

%% Variables regarding pertinency (Global Physical Programming)
% The following values for each design objective (decision objective +
% non-decision objective) and constraint shall be defined:
%
% Physical Matrix Definition.
% HD Highly Desirable
% D Desirable
% T Tolerable
% U Untolerable
% HU Highly Untolerable
%
% HD - D - T - U - HD
spMODExDat.PhyMatrix{1} = [0.0 1.5 3 6 12 20;
                          0.0 7 14 30 50 70;
                          0.0 5 10 20 30 50];

```

- **CostFunction.m.** En esta función se implementan los objetivos a cumplir y se invoca al modelo a optimizar.

```

function [J, X]=CostFunction(X,Dat)

p = param2(); %Default values of parameters
%Simulate
tfin = 600; % Tiempo de simulacion
opts = odeset('AbsTol',1e-6,'RelTol',1e-4); % opciones para el simulador
ode

```

```

%% Simulate Model
initial_states = 0.*0.5*abs(rand(1,10));

for xpop =1:size(X ,1)
    % Parameters coming from Opt
    p.p1=X(xpop,1);
    p.dm4=X(xpop,2);
    p.p5=X(xpop,3);
    p.dm5=X(xpop,4);
    p.kplus12=X(xpop,5);
    p.p4=X(xpop,6);

    %If we start from random initial conditios, it still works:
    [t_D,x_D] = ode23t(@ (t,x)ode_newmodelJaimeAnton(t,x,p), [0
tfin],initial_states,opts);

    %Calcular Objectives
    AE = abs(x_D(:,8)-x_D(:,5));
    TAE = t_D.*AE;
    %SE = (x_D(:,8)-x_D(:,5)).^2;
    %TSE = t_D.*SE;

    CAE = cumsum(AE);
    MAE =1/(length(AE)).*sum(AE);

    CTAE = cumsum(TAE);
    MTAE =1/(length(TAE)).*sum(TAE);

    %CSE=cumsum(SE);
    %MSE=1/(length(SE)).*sum(SE);

    %Zero-crossing
    Zt=x_D(:,5)-x_D(:,8);
    Xt =(x_D(:,5)-x_D(:,8));
    CD=0;

    for i = 1:size(Zt,1)
        if (Zt(i) > 0)
            Xt(i) = 1;
        else Xt(i) = 0;
        end
    end

    for j = 1:(size(Zt,1)-1)
        D(j) = (Xt(j+1)-Xt(j))^2;
        CD = CD + D(j);
    end

    J(xpop,1) = MAE;    %J1
    J(xpop,2) = MTAE;  %J2
    J(xpop,3) = CD;    %J3

end
end

```


Bibliografía

- Aoki, Lillacci, Stephanie K et al. (2017). *A universal biomolecular integral feedback controller for robust perfect adaptation*. En: Nature, pág. 1
- Aström, Karl Johan y Richard M Murray (2010). *Feedback systems: an introduction for scientists and engineers*. Princeton university press
- Beal, Cox et al. (2016). *Synthetic biology open language visual (SBOL visual) version 2.0*. En: Journal of integrative bioinformatics
- Becskei, A. and Serrano, L. (2000). *Engineering stability in gene networks by autoregulation*. Nature, 405(6786), 590– 593.
- Boada Acosta, Yadira Fernanda (2018). *A systems engineering approach to model, tune and test synthetic gene circuits*. Tesis doctoral.
- Boada, Yadira, Alejandro Vignoni y Jesús Pico (2017). *Engineered control of genetic variability reveals interplay among quorum sensing, feedback regulation, and biochemical noise*. En: ACS synthetic biology 6.10, págs. 1901-1912
- Briat, C., Gupta, A., and Khammash, M. (2016). *Antithetic integral feedback ensures robust perfect adaptation in noisy biomolecular networks*. Cell Systems, 2(1), 15–26.

- Briat, C., Gupta, A., and Khammash, M. (2018). *Antithetic proportional-integral feedback for reduced variance and improved control performance of stochastic reaction networks*. Journal of The Royal Society Interface, 15.
- Cuba Samaniego, C. and Franco, E. (2017). *An ultrasensitive biomolecular network for robust feedback control*. IFAC PapersOnLine, 50(1), 10950–10956.
- Del Vecchio, Domitilla, Aaron J Dy y Yili Qian (2016). *Control theory meets synthetic biology*. En: Journal of The Royal Society Interface
- F. Crick. *Central dogma of molecular biology*. Nature, 227(5258):561, 1970.
- G. Reynoso-Meza, S. García-Nieto, J. Sanchis, and X. Blasco, (2014). *Controller tuning using multiobjective optimization algorithms: a global tuning framework*. IEEE Transactions on Control Systems Technology, 21(2):445–458.
- H. De Battista, J. Picó, E. Pico-Marco, A. Vignoni (2019). *Biomolecular signal tracker with fast time response*.
- Hsiao, V., de los Santos, E.L.C., Whitaker, W.R., Dueber, J.E., and Murray, R.M. (2014). *Design and implementation of a biomolecular concentration tracker*. ACS Synthetic Biology.
- Hsiao, Victoria, Anandh Swaminathan y Richard M Murray (2018). *Control theory for synthetic Biology: Recent advances in system characterization, control design, and controller implementation for synthetic biology*. En: IEEE Control Systems Magazine 38.3, págs. 30-60
- J. D. Watson, F. H. Crick, et al. *A structure for deoxyribose nucleic acid*. 1953.
- Kazuo Nakashima, Yusuke Ito, Kazuko Yamaguchi-Shinozaki (2009), *Transcriptional Regulatory Networks in Response to Abiotic Stresses in Arabidopsis and Grasses*, Plant Physiology, Volume 149.
- L. Potvin-Trottier, N. D. Lord, G. Vinnicombe, and J. Paulsson, (2016). *Synchronous longterm oscillations in a synthetic gene circuit*. Nature, 538(7626):514–517.
- Menolascina, Filippo y col. (2014). *In-vivo real-time control of protein expression from endogenous and synthetic gene networks*. En: PLoS computational biology 10.5, e1003625
- Monfort Llorens, R. (2019). *Diseño e implementación de un controlador proporcional-integral (PI) para el control de la producción de proteínas*.
- V. Chellaboina, S. Bhat, M. Haddad, and D. Bernstein. *Modeling and analysis of massaction kinetics*. Control Systems, IEEE, 29(4):60–78, Aug 2009. ISSN 1066-033X. doi: 10.1109/MCS.2009.932926.
- Wang Z, Wu M. (2015). *An integrated phylogenomic approach toward pinpointing the origin of mitochondria*.

Y. Boada, A. Vignoni, J. Picó, P. Carbonell (2020). *Extended metabolic biosensor design for dynamic pathway regulation of cell factories*

