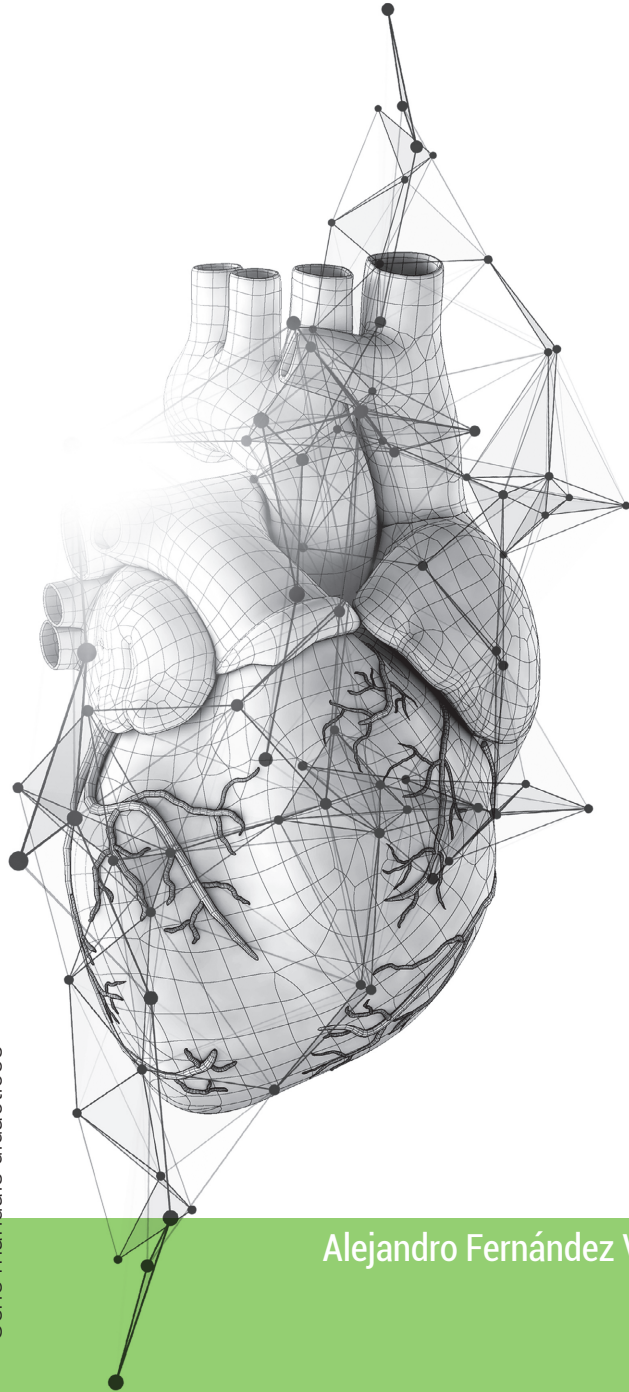


Modelado e Simulación de Sistemas Biomédicos



Manuais
Série manuais didáticos

Alejandro Fernández Villaverde



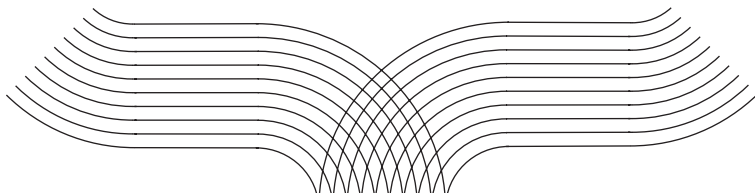
Alejandro Fernández Villaverde é enxeñeiro industrial superior (2004) e doutor europeo (2009) pola Universidade de Vigo. Tras un breve paso pola empresa privada encamiñou a súa carreira cara a investigación, especializándose en enxeñaría de control. Desde 2005 a 2009 foi investigador predoutoral FPI na Universidade de Vigo. Entre 2009-2015 e 2016-2020 foi investigador posdoutoral no CSIC, participando en varios proxectos europeos de bioloxía de sistemas, biotecnoloxía e biomedicina computacional. Entre ambos períodos desfrutou dunha axuda do plan I2C

da Xunta de Galicia. Ten realizado tamén estadías breves nas universidades de Groningen, MIT, Stanford, Minho e Oxford. Desde 2021 é investigador Ramón y Cajal na Universidade de Vigo, onde realiza labores docentes no departamento de Enxeñaría de Sistemas e Automática. É membro do Grupo de Control Non Linear, dentro do cal dirixe un subgrupo dedicado ao estudo dos biosistemas e á súa identificación, control e optimización. É tamén investigador vinculado ao Centro de Investigación e Tecnoloxía Matemática de Galicia (CITMAga). A súa investiga-

ción céntrase no desenvolvemento de metodoloxías para o modelado e análise de sistemas dinámicos non lineares, con énfase en aplicacións biolóxicas, biomédicas e biotecnolóxicas. Ten dirixido varios proxectos de I+D+i e publicado máis de 70 artigos nestas temáticas, particularmente sobre identificabilidade e observabilidade.

Servizo de Publicacións

Universidade de Vigo



Manuais

Serie de manuais didácticos

n.º 084

Edición

Universidade de Vigo
Servizo de Publicacións
Rúa de Leonardo da Vinci, s/n
36310 Vigo

Deseño da portada

Tania Sueiro Graña
Área de Imaxe
Vicerreitoría de Comunicacións e Relacións Institucionais

Maquetación

Tórculo Comunicación Gráfica, S. A.

Fotografía da portada

Adobe Stock

Impresión

Tórculo Comunicación Gráfica, S. A.

ISBN (Libro impreso)

978-84-1188-001-5

Depósito legal

VG 112-2024

© Servizo de Publicacións da Universidade de Vigo, 2024

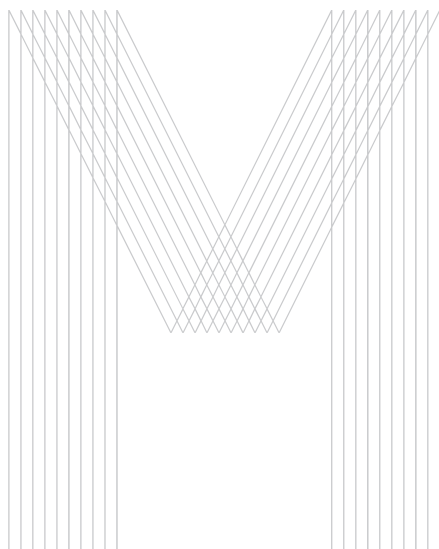
© Alejandro Fernández Villaverde

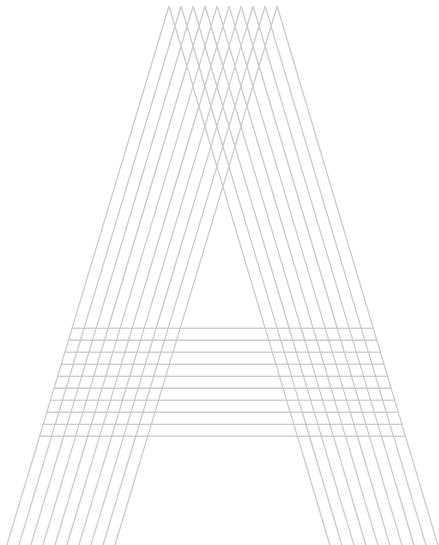
Sen o permiso escrito do Servizo de Publicacións da Universidade de Vigo, queda prohibida a reprodución ou a transmisión total e parcial deste libro a través de ningún procedemento electrónico ou mecánico, incluídos a fotocopia, a gravación magnética ou calquera almacenamento de información e sistema de recuperación.

Ao ser esta editorial membro da **une**, garántense a difusión e a comercialización das súas publicacións no ámbito nacional e internacional.

Servizo de Publicacións

Universidade de Vigo





Este volumen publícase co financiamento da



**XUNTA
DE GALICIA**

Modelado e Simulación de Sistemas Biomédicos

Autor

Alejandro Fernández Villaverde

Índice xeral

Limiar	1
1. Introducción	5
1.1. Motivación e historia do modelado matemático en biomedicina	5
1.2. Modelado dinámico: compoñentes e paradigmas	9
1.3. Tipos de modelos dinámicos	11
1.3.1. Grafos	11
1.3.2. Ecuacións diferenciais	14
1.3.3. Combinacións de modelos	16
1.4. Exemplo: modelando un biosistema con varios formalismos . .	17
1.4.1. Modelos do exemplo baseados en grafos	19
1.4.2. Modelos do exemplo baseados en ecuacións diferenciais	20
1.5. Panorama da materia	22
2. Modelos de biosistemas	23
2.1. Sobre os tipos de biosistemas de interese	23
2.2. Cinética das reaccións bioquímicas	25
2.3. Nivel celular	27
2.3.1. Metabolismo	29
2.3.2. Sinalización celular	31
2.3.3. Expresión xénica	34
2.4. Nivel orgánico	36
2.4.1. Electrofisioloxía	36
2.4.2. Regulación da glicosa	37
2.4.3. Farmacocinética e farmacodinámica	39

2.5. Nivel de poboacións	41
2.5.1. Epidemioloxía	41
2.5.2. Comunidades microbianas	43
3. Simulación	51
3.1. Integración de EDOs lineares	52
3.1.1. Transformada de Laplace.	52
3.1.2. Función de transferencia (FT)	58
3.2. Integración de EDOs nonlineares	61
3.2.1. Métodos de paso fixo	61
3.2.2. Métodos de paso variábel	65
3.2.3. Concluindo: como facer a integración?	67
3.3. Integración de ecuacións estocásticas	69
3.3.1. Algoritmo de Gillespie	70
3.4. Software de simulación	72
3.4.1. Entornos de programación de propósito xeral	72
3.4.2. Ferramentas de simulación especializadas	74
3.5. Estándares, formatos e repositorios	75
4. Calibración	79
4.0. PASO 0: obter as ecuacións do modelo	81
4.1. PASO 1: analizar observabilidade e identificabilidade estrutural	84
4.2. PASO 2: definir a función obxectivo	92
4.3. PASO 3: optimización dos parámetros	96
4.3.1. Métodos locais	96
4.3.2. Métodos globais.	99
4.3.3. Definición do problema de optimización.	100
4.4. PASO 4: análise da bondade do axuste	102
4.5. PASO 5: cuantificación da incerteza dos parámetros	103
4.6. PASO 6: cuantificación da incerteza nas predicións	107
4.7. Deseño de experimentos	108
4.8. Selección de modelos	109
4.9. Recursos software	110

5. Dinámica	113
5.1. Sistemas lineares e linearización	114
5.2. Equilibrio e estabilidade	116
5.2.1. Caracterización matemática da estabilidade	117
5.3. Bifurcacións	121
5.3.1. Diagramas de fase	121
5.3.2. Gráficas de bifurcación	122
5.4. Oscilacións	126
5.5. Robustez	127
5.5.1. Redundancia	129
5.5.2. Insensibilidade paramétrica.	129
5.5.3. Realimentación	131
5.5.4. Prealimentación	132
5.6. Redución de modelos	133

Limiar

ESTAS NOTAS foron creadas como material docente para a materia “Modelado e Simulación de Sistemas Biomédicos”, do Mestrado en Enxeñaría Biomédica (MEB) da Universidade de Vigo. O seu obxectivo é compendiar os coñecementos necesarios para construír modelos dinámicos de biosistemas en diferentes contextos, con énfase nos procesos e sistemas de interese na enxeñaría biomédica.

Aínda que empregamos a palabra “biosistema” nun senso amplo, non trataremos todos os tipos de “sistema” que se poden encontrar na enxeñaría biomédica. Así pois, este texto non aborda a simulación biomecánica nin o modelado de dispositivos médicos, temas que se estudan noutras materias do MEB. Pola contra, adoptamos o marco conceptual do que se coñece como bioloxía de sistemas, disciplina que naceu aproximadamente co século XXI e que xa conta por tanto cunha certa tradición, polo que estudaremos sistemas biolóxicos que van desde redes de reaccións bioquímicas que teñen lugar a nivel intracelular, até interaccións entre organismos – como por exemplo as estudadas en epidemioloxía.

En canto ao formalismo matemático, centrámonos fundamentalmente en sistemas descritos por ecuacións diferenciais ordinarias. Isto é así por dous motivos. En primeiro lugar, é un tipo de descrición adecuado para unha ampla variedade de procesos, e a maioría dos modelos dinámicos usados en enxeñaría biomédica adoptan este formalismo. En segundo lugar, as ecuacións en derivadas parciais vense con detalle noutras materias do MEB.

Para que a lectura do texto sexa o máis proveitosa posíbel, cómpre contar cun mínimo de familiaridade con conceptos básicos de bioloxía e matemáticas, pero non son precisos coñecementos avanzados previos. O texto pretende ser

autocontido, polo que non debería ser preciso ler bibliografía adicional, aínda que naturalmente se proporcionan as referencias pertinentes. Dito isto, se alguén quixer profundar na materia recoméndanse especialmente os seguintes libros:

- “Dynamic systems biology modeling and simulation”, de Joseph DiStefano III (Academic Press, 2015).
- “Systems biology: a textbook, 2nd Edition”, de Edda Klipp, Wolfram Liebermeister, Christoph Wierling & Axel Kowald (Wiley-Backwell, 2016).
- “Mathematical Modeling in Systems Biology: An Introduction”, de Brian Ingalls (MIT Press, 2013), dispoñíbel gratuitamente en:
<https://www.math.uwaterloo.ca/~bingalls/MMSB/>.
- “Biomolecular feedback systems”, de D. del Vecchio e R.M. Murray (Princeton University Press, 2014), dispoñíbel gratuitamente en:
http://www.cds.caltech.edu/~murray/BFSwiki/index.php/Main_Page
(principalmente para a Lección 5, “Comportamento dinámico”).

En caso de encontrar erratas no texto, agradécese que se envíen a afvillaverde@uvigo.gal. Calquera suxestión ou comentario será tamén benvido.

Finalmente, quero dar as grazas a Elba, Mari Carmen, e toda a miña familia polo seu apoio incondicional.

Vigo, 20 de outubro do 2023.

Dedicado a Julio Rodríguez Banga, pioneiro da bioloxía de sistemas con quen
tiven a sorte de traballar e aprender durante dez anos no CSIC.

Introdución

“All models are wrong, but some are useful.”

– George E. P. Box

Para introducir e motivar o estudo desta materia, comezaremos por definir o concepto de modelo, veremos que tipos de modelos hai, e como se aplican no contexto da enxeñaría biomédica.

1.1. Motivación e historia do modelado matemático en biomedicina

QUE É UN MODELO? Nun sentido amplo, calquera representación da realidade – por ambigua ou imprecisa que sexa – que facilite a súa comprensión é un “modelo”. Segundo esta definición todas as persoas usamos modelos continuamente, mesmo de forma inconsciente (ao facermos unha predición estamos a executar algún modelo mental).

Nós estamos interesados especificamente en *modelos biomédicos*. Nun informe de 1998, o National Research Council dos EUA daba a seguinte definición de modelo biomédico [41]: “un substituto do ser humano ou dun sistema biolóxico humano, que pode usarse para comprender o funcionamento normal e anormal do xenotipo ao fenotipo, e para proporcionar unha base para

a intervención terapéutica ou preventiva en doenzas humanas”. Nese sentido (o modelo como substituto total ou parcial do ser humano) os modelos poden ser de tres tipos:

- Modelos *in vivo*: animais que presentan semellanzas co ser humano.
- Modelos *in vitro*: cultivos de laboratorio, tanto celulares como de outro tipo.
- Modelos *in silico*: modelos matemáticos implementados computacionalmente.

O devandito informe centrábase no primeiro tipo. En contraste, nós usaremos a palabra “modelo” para referirnos ao terceiro tipo: a *modelos matemáticos, computacionais ou in silico*; é dicir, modelos representados por ecuacións, que implementaremos e simularemos nunha computadora. Nesta materia traballaremos con modelos dinámicos. Ditos modelos están formados por ecuacións diferenciais que describen a evolución temporal dun sistema. Xa que logo, de aquí en adiante ao falarmos de “modelos” estaremos a referirnos especificamente a *modelos matemáticos dinámicos*, salvo que se especifique o contrario.

Para que serven os modelos matemáticos? Os modelos dinámicos son unha ferramenta fundamental en moitas ramas da ciencia e da enxeñaría. Entre as súas utilidades podemos salientar as seguintes [20]:

1. Explicar o mecanismo polo que acontece un fenómeno.
2. Describir os aspectos máis importantes da súa dinámica.
3. Suxerir analoxías.
4. Dar lugar a novas preguntas.
5. Fomentar o razoamento cuantitativo e baseado en evidencia.
6. Guiar o deseño de experimentos e a recollida de datos.
7. Permitir reproducir resultados ou invalidar hipóteses.

8. Proporcionar rangos de resultados posíbeis.
9. Identificar e cuantificar as incertezas.
10. Facer predicións cuantitativas.

A derradeira é a que se adoita mencionar como razón principal do uso de modelos matemáticos en bioloxía: a necesidade de obter predicións cuantitativas sobre sistemas dinámicos complexos, cuxo comportamento non é predicíbel a través do razoamento intuitivo. Esta capacidade posibilita, por exemplo:

- identificar o mecanismo biolóxico máis plausíbel de entre varias alternativas para explicar un comportamento;
- predicir o resultado dun tratamento;
- optimizar o deseño de dito tratamento (selección de medicamento, dose, periodicidade...);
- monitorizar o estado dun paciente;
- deseñar estratexias de control.

Na liña do aforismo de Box que abre este capítulo, poderíase argumentar que a utilidade dun modelo non provén da exactitude con que represente a realidade, senón en como pode ser usado para xerar, comprobar, e refinar hipóteses [19, 51]. Esta visión enfatiza o valor do *proceso* de modelado, máis que o seu *resultado* (o modelo en si). Valórase que este proceso é unha forma de pensar sobre o sistema (o cal é dinámico, xeralmente non linear, e con frecuencia complexo), e que, aínda que a abstracción inherente a un modelo implica necesariamente unha simplificación, tamén leva a definir de forma explícita as suposicións introducidas, o que permite interpretar as predicións en función delas.

Perspectiva histórica do modelado de sistemas biolóxicos. En comparación cos problemas tratados noutras ramas da enxeñaría, que son o resultado do deseño humano, os sistemas biolóxicos evoluíron de forma non dirixida, polo que adoitan ter unha elevada complexidade e están mal caracterizados.

Isto ten dificultado a aplicación directa de métodos de modelado desenvolvidos noutras enxeñarías a procesos biolóxicos.

Na década de 1940 varios autores fixeron contribucións desde perspectivas moi diferentes (Fig. 1.1). Por exemplo, Erwin Schrödinger propuxo o estudo dos procesos celulares desde o punto de vista das leis físicas. O biólogo Ludwig von Bertalanffy visualizou unha teoría xeral dos sistemas (sociais, tecnolóxicos, biolóxicos). Pola súa parte, o matemático Norbert Wiener cuñou o termo “cibernética” para referirse aos procesos de control e comunicación tanto en organismos vivos como en máquinas, liña que foi continuada por Ashby e outros.

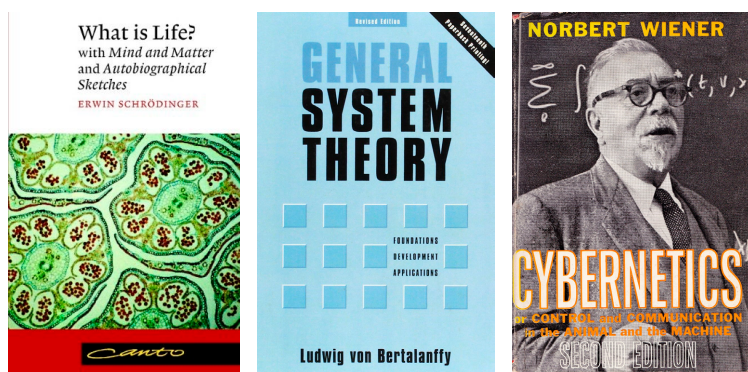


Figura 1.1: Nos anos 40 do século XX colle pulo o modelado matemático de biosistemas, grazas a autores como Schrödinger, Bertalanffy ou Wiener.

Ao comezo, estas propostas foran maiormente teóricas, pero axiña comezaron a aplicarse a problemas concretos. Un fito importante produciuse en 1952, cando Hodgkin e Huxley propuxeron un modelo dinámico para describir como se inician e transmiten os potenciais de acción nas neuronas [26]. Este traballo valeríalles o Premio Nobel de fisioloxía e medicina, e abriría o camiño a modelos computacionais como os de Guyton [25] ou Denis Noble [43]. Na década de 1960 temos exemplos como o modelado de sistemas oscilatorios de Goodwin [24], o estudo da autoorganización de Kauffman [32] ou o

desenvolvemento das ideas de Bertalanffy a cargo de Mesarovic.

Nos anos 70 existe xa unha comunidade de enxeñaría biomédica que usa o modelado dinámico como ferramenta fundamental, en áreas como fisioloxía ou farmacocinética. Os retos metodolóxicos que presentan os sistemas biolóxicos deron lugar a novos desenvolvementos en identificación de sistemas, con conceptos como a identificabilidade estrutural [5]. Nesta época destacan investigadores como Claudio Cobelli ou Joseph DiStefano III. A finais de década, esta área atopábase xa relativamente establecida; un estudo das aplicacións destas metodoloxías a sistemas biolóxicos recollía exemplos de modelos cardiovasculares, respiratorios, neuronais, renais, metabólicos e esqueléticos, entre outros [4].

O modelado biolóxico no século XXI: bioloxía de sistemas. Arredor do ano 2000 os avances experimentais en bioloxía molecular dan lugar a incrementos moi rápidos da cantidade de datos “ómicos” dispoñíbeis. Isto propicia a aparición dun novo paradigma, a chamada *bioloxía de sistemas* [33], que se caracteriza por un enfoque holístico (é dicir, centrado nas interconexións entre diferentes elementos, ou subsistemas, e en caracterizar as propiedades emerxentes que xorden das súas relacións) en contraposición a enfoques reduccionistas que se centran na caracterización de compoñentes por separado. Para realizar predicións cuantitativas sobre sistemas complexos, a bioloxía de sistemas baséase no modelado computacional. O termo *biomedicina de sistemas* úsase para referirse a este enfoque no contexto biomédico.

1.2. Modelado dinámico: compoñentes e paradigmas

A Figura 1.2 ilustra a visión do modelado dinámico típica de teoría de sistemas. O *estado* dun sistema vén dado en todo momento polo vector $x(t) = x_1(t), \dots, x_n(t)$, cuxos elementos son as variábeis de estado, ou simplemente estados. Como regra xeral, os estados non deben ser redundantes (por exemplo, se $x_i = 1 - x_j$, só unha das dúas, x_i ou x_j , debe ser escollida como variable de estado). O seu número (n) determina a dimensión do sistema.

A evolución temporal do estado é determinada pola resposta dinámica do sistema a entradas externas (tanto coñecidas, $u(t)$, como descoñecidas,

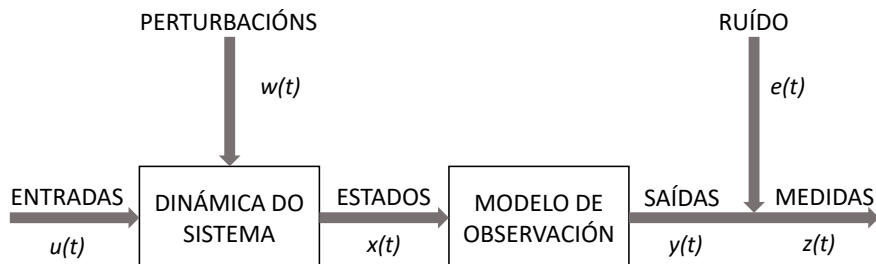


Figura 1.2: Unha posíbel clasificación dos compoñentes dun modelo dinámico. Todas as variábeis – representadas en cursiva – son en xeral vectoriais

$w(t)$), a partir da condición inicial ($x(t_0)$). As entradas coñecidas $u(t)$ poden ser controis, se o sistema é controlado; nese caso, polo xeral son calculadas nun lazo de realimentación. Un sistema sen entradas dise que é autónomo. As entradas descoñecidas $w(t)$ son perturbacións.

Polo xeral non somos capaces de medir ou acceder a todos os estados do sistema, senón só un subconxunto deles (ou unha función dun subconxunto), que se chaman saídas ($y(t)$). O mapa dos estados ás saídas é o modelo de observación. Ademais, en xeral existirá ruído experimental, $e(t)$, polo que as medidas obtidas, $z(t)$, diferirán das saídas teóricas.

As cantidades mencionadas son en principio variábeis no tempo. Un modelo pode ter tamén cantidades de valor fixo, chamadas *parámetros* (θ). Os parámetros coñecidos chámanse *constantes* (k).

É frecuente diferenciar entre dous paradigmas de modelado: “top-down” e “bottom-up”. O modelado “top-down” adopta un enfoque holístico. Parte dunha concepción global do sistema, enfocándose nas súas características xerais, e a partir delas vai engadindo detalles. O modelado “bottom-up” é considerado reduccionista. Comeza por caracterizar os subsistemas e despois interconectalos para obter o sistema xeral. A distinción entre ambas concepcións é até certo punto unha simplificación filosófica, xa que na práctica pódense adoptar enfoques intermedios.

1.3. Tipos de modelos dinámicos

Podemos clasificar os modelos en función de se son cualitativos ou cuantitativos, estatísticos ou mecanísticos, continuos ou discretos, deterministas ou estocásticos. A continuación descríbense varias clases de modelos, clasificados en dúas grandes familias: modelos baseados en grafos e modelos en ecuacións diferenciais.

1.3.1. Grafos

Muitos sistemas de interese en biomedicina poden modelarse naturalmente como grafos. Unha rede ou grafo consiste nun conxunto de nodos e un conxunto de ligazóns entre eles. Nas redes celulares os nodos son entidades moleculares como xenes, proteínas ou metabolitos. Os enlaces ou ligazóns son as interaccións ou relacións entre nodos; p.ex., as reaccións químicas nas que están presentes as moléculas, ou unha regulación que implica varias reaccións.

Os grafos son formalismos moi adecuados para construír modelos de interaccións a partir de datos (“data-driven approach”). Para cuantificar a asociación entre variables pódense usar medidas de correlación, teoría da información, ou inferencia bayesiana (ver Fig. 1.3). O cálculo de ditas medidas a partir de datos experimentais (por exemplo, da concentración de cada entidade correspondente a un nodo) permite determinar as relacións existentes entre nodos. Este procedemento coñécese como *inferencia de redes*.

Podemos clasificar os grafos en función do nivel de detalle co que especifican as interaccións entre os seus nodos:

Grafos non orientados. Son o tipo máis simple. Limitáanse a representar se existe unha relación directa entre dúas variábeis (nese caso, únense cunha ligazón) ou non. Este tipo de redes son estáticas, non posúen dinámica. Un exemplo amósase na parte superior dereita da Fig. 1.3.

Grafos orientados. Inclúen ademais información sobre a dirección da ligazón, as cales se representan como frechas. A dirección pode representar causalidade, fluxo, etc. Por tanto, teñen un certo carácter dinámico, aínda que máis

(1) Obter datos experimentais

(2) Cuantificar dependencias entre pares de variábeis

(3) Inferir interaccións

t	V	W	X	Y	Z
1	0.47	0.11	0.67	0.95	0.27
2	0.36	0.12	0.66	0.86	0.26
3	0.30	0.13	0.60	0.80	0.20
4	0.42	0.15	0.62	0.72	0.22
5	0.41	0.17	0.61	0.71	0.21
6	0.37	0.20	0.67	0.67	0.27
7	0.35	0.23	0.65	0.65	0.25
8	0.28	0.21	0.58	0.58	0.38
9	0.24	0.20	0.54	0.54	0.40
10	0.22	0.20	0.52	0.52	0.42
11	0.15	0.20	0.55	0.45	0.43

Correlación

$$r(X, Y) = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^n (X_i - \bar{X})^2} \sqrt{\sum_{i=1}^n (Y_i - \bar{Y})^2}}$$

Teoría da información

$$I(X, Y) = H(X) - H(X|Y)$$

Inferencia Bayesiana

$$p(X|Y) = \frac{p(X|Y)p(X)}{p(Y)}$$

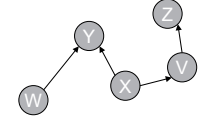
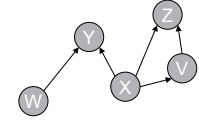
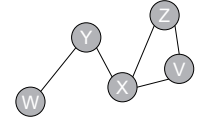
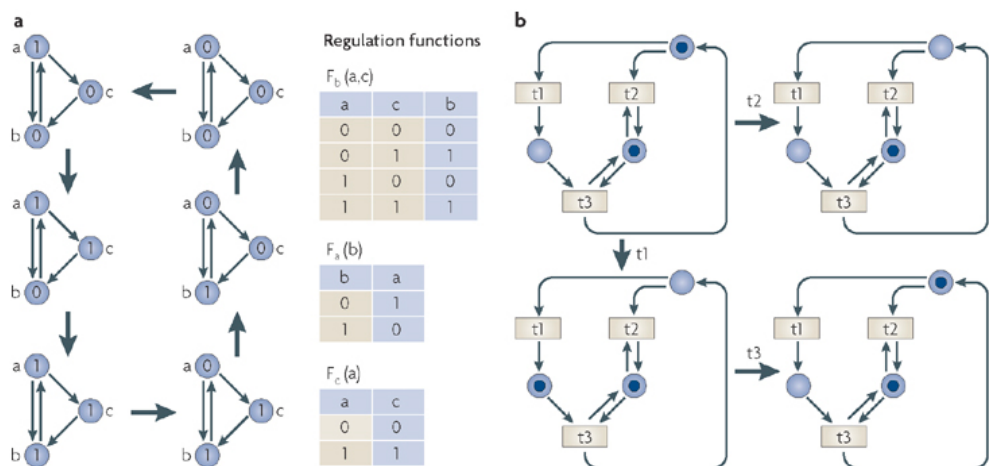


Figura 1.3: Inferencia de rede a partir de datos experimentais [52].

ben implícito. Un exemplo amósase nos grafos medio e inferior da dereita da Fig. 1.3.

Redes lógicas ou booleanas. Poden considerarse un tipo particular de grafos orientados. Nelas, cada nodo pode ter un valor binario (0 = falso, 1 = verdadeiro), que indica se a actividade está por enriba ou por baixo dun limiar. Bioloxicamente, un compoñente está activo cando o seu nivel de actividade é suficiente alto para influír nos niveis de actividade dun compoñente obxectivo. As interaccións entre nodos son relacións lógicas (AND, OR, etc). As redes booleanas son sistemas dinámicos. Foron introducidas por Kauffman en 1969 [32] para representar redes de regulación xenética. Como exemplo, o panel A da Fig. 1.4 mostra unha rede booleana formada por tres nodos, e a súa evolución ao longo do tempo. Cada paso temporal está representado por unha frecha grosa. O estado global da rede en cada instante temporal está formado polos estados dos seus tres compoñentes. O sistema do exemplo evolúe de forma cíclica entre seis estados globais, de acordo coas regras definidas nas táboas da dereita.



Nature Reviews | Molecular Cell Biology

Figura 1.4: (A) Rede booleana. (B) Rede de Petri. Fonte: [30]; Wikimedia Commons (<https://commons.wikimedia.org/wiki/File:Nrm2503-f1.jpg>), licenza CC-BY-SA-4.0.

Redes de Petri. Formalismo moi similar ás redes booleanas (ver Fig. 1.4.B). Ademais de nodos (“lugares”) e ligazóns orientadas (“arcos”) conteñen “marcas” para especificar o estado do sistema (puntos azul escuro na figura), e “transicións” ($t_1, t_2, \dots$ na figura) para determinar a súa evolución dun estado a outro. Unha transición ten lugar (“dispárase”) se os estados necesarios que apuntan a ela contan con suficientes marcas, e a condición que representan se cumpre. Nese caso, o marcado dos lugares que apuntan á transición decrece, e o dos lugares aos que apunta increméntase. As redes de Petri son moi usadas en enxeñaría automática para o modelado de sistemas concorrentes. O seu uso en bioloxía non é tan común, aínda que existen exemplos sobre todo para a descrición de redes de sinalización celular.

Grafos de ligazón (“bond graphs”). Son representacións de sistemas dinámicos nos que os arcos representan intercambio de enerxía. Foron inventados por Henry Paynter en 1961 e son utilizados en física e enxeñaría, sobre todo

para modelar sistemas eléctricos e mecánicos. Raramente se usan para representar biosistemas, con algunhas excepcións de sistemas de reaccións bioquímicas nas que se lle concede moita importancia a aspectos termodinámicos.

1.3.2. Ecuacións diferenciais

Ecuacións diferenciais ordinarias. As ecuacións diferenciais ordinarias (EDOs) son a forma máis habitual de modelar sistemas dinámicos. Un modelo en EDOs vén dado por:

$$\mathcal{M} = \begin{cases} \dot{x}(t) = f(x(t), \theta, u(t), w(t)) & (1.1) \\ y(t) = h(x(t), \theta, u(t), w(t)) & (1.2) \end{cases}$$

onde f e h funcións analíticas (en xeral non-lineares) dos estados $x(t) \in \mathbb{R}^{n_x}$, das entradas coñecidas $u(t) \in \mathbb{R}^{n_u}$, dos parámetros (constantes descoñecidas) $\theta \in \mathbb{R}^{n_\theta}$, e das entradas descoñecidas / perturbacións / parámetros variantes no tempo $w(t) \in \mathbb{R}^{n_w}$. A saída $y(t) \in \mathbb{R}^{n_y}$ representa as funcións medíbeis das variábeis do modelo. As expresións (1.1–1.2) son moi xerais e poden representar un amplo abano de estruturas.

— Os **sistemas lineares** son un caso particular de modelos en EDOs. Se – como é habitual – non consideramos entradas descoñecidas, as ecuacións dun sistema linear pódense escribir como:

$$\mathcal{M} = \begin{cases} \dot{x}(t) = A(\theta) \cdot x(t) + B(\theta) \cdot u(t) & (1.3) \\ y(t) = C(\theta) \cdot x(t) + D(\theta) \cdot u(t) & (1.4) \end{cases}$$

Onde A , B , C , D son matrices que poden conter parámetros descoñecidos (θ). Nótese que estas ecuacións son lineares respecto aos estados, pero serían non-lineares se consideramos as relacións entre parámetros e estados. A notación (1.3) coñécese como representación en variables de estado, ou no espazo de estados.

— Os **modelos compartimentais** son unha clase particular de modelos lineares. Ditos modelos baséanse na idea de compartimentalizar variábeis en entidades homoxéneas. Por exemplo, cada compartimento pode representar un órgano, dentro do cal suponse que a concentración dunha sustancia está homoxeneamente distribuída. Moitos sistemas de interese en biomedicina poden ser representados por modelos compartimentais, como se verá no próximo capítulo.

Exemplo de sistema linear: modelo farmacocinético. O modelo compartimental farmacocinético mostrado na figura 1.5 ten as seguintes ecuacións:

$$\mathcal{M} = \begin{cases} \dot{x}_1 = -(k_{12} + k_{10}) \cdot x_1 + k_{21} \cdot x_2 + k_{1u} \cdot u, & (1.5) \\ \dot{x}_2 = k_{12} \cdot x_1 - k_{21} \cdot x_2, & (1.6) \\ y = x_2; & (1.7) \end{cases}$$

as cales poden poñerse da forma (1.3)–(1.4), facendo:

$$A = \begin{bmatrix} -(k_{12} + k_{10}) & k_{21} \\ k_{12} & -k_{21} \end{bmatrix}, B = \begin{bmatrix} k_{1u} \\ 0 \end{bmatrix}, C = [0 \quad 1], D = 0 \quad (1.8)$$

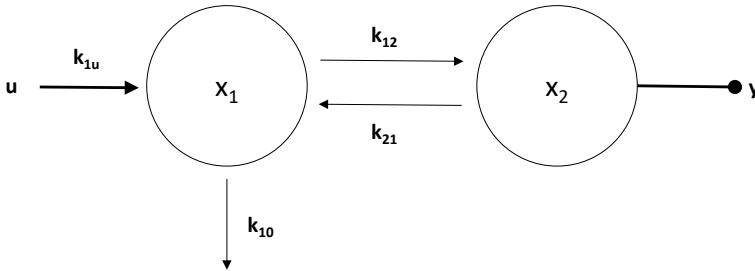


Figura 1.5: Modelo farmacocinético de dous compartimentos (= estados), unha entrada e unha saída.

Modelos de efectos mixtos. Nun modelo en EDOs, os parámetros θ considéranse fixos, é dicir, teñen un único valor real (aínda que sexa descoñecido). Os modelos de efectos mixtos consideran que cada parámetro ten un compoñente fixo e outro aleatorio. Así, poden representar a variabilidade existente nos valores do parámetro nunha poboación. Por exemplo, en lugar de considerar que un parámetro ten un valor real $\theta = 2$, considérase que cada membro da poboación ten un valor diferente, cuxa media é 2 e cuxa desviación típica é por exemplo 0.5. Este formalismo de particular utilidade en áreas como a farmacocinética, nas que estes modelos son moi comúns. Os modelos de efectos mixtos son máis difíciles de calibrar.

Ecuacións en diferenzas. As EDOs describen a evolución temporal dun sistema en todo momento, é dicir, son sistemas en tempo continuo. Un tipo diferente de ecuacións, as ecuacións en diferenzas (EDs), representan un sistema en tempo discreto.

Ecuacións en derivadas parciais. As ecuacións en derivadas parciais (EDPs) conteñen unha ou máis variábeis dependentes, e polo menos dúas variábeis independentes. Por exemplo, as dúas variábeis independentes poden ser o tempo t e unha dimensión espacial x ; deste xeito, as EDPs permiten incorporar variación espacial a un modelo. As EDPs tamén son chamadas modelos distribuídos, ou modelos de parámetros distribuídos.

Ecuacións estocásticas. As reaccións químicas teñen lugar de forma determinista se hai números suficientemente elevados de moléculas e estas atópanse ben mesturadas, pero nalgúns casos estas hipóteses non se cumpren. Mentres que para modelar a heteroxeneidade espacial se poden usar EDPs, para describir relacións entre números pequenos de moléculas úsanse modelos estocásticos. Neles, as variábeis de estado son aleatorias. A súa simulación baséase no algoritmo de Gillespie [23]; para máis detalles ver o capítulo 3.

Multiaxentes. O paradigma de modelado baseado en axentes (“agent-based modeling”, ABM) permite usar regras, ecuacións alxébricas ou EDOs para definir como os axentes responden ao seu contorno. Permite incorporar heteroxeneidade, o que pode ser útil para modelar sistemas como cultivos celulares.

1.3.3. Combinacións de modelos

Modelos híbridos. Ás veces resulta conveniente combinar varios tipos de modelos, obtendo un modelo híbrido. Por exemplo, pódese combinar partes continuas en EDOs con eventos discretos, ou ben partes deterministas con estocásticas. Para un panorama recente dos moitos tipos de combinacións posibles véxase [35]. Como exemplo de aplicación, en [48] usouse unha combinación de modelado baseado en axentes (ABM) e en ecuacións diferenciais (EDO) para modelar a progresión da diabetes tipo 1. Consideráronse varios

axentes correspondentes a diferentes tipos de entidades (antíxenos, células β , linfocitos T, etc), e establecéronse regras para modelas as súas interaccións. Adicionalmente, usáronse ecuacións diferenciais para describir a dinámica interna de algúns axentes.

Conxuntos de modelos ou “ensemble modelling”. Con frecuencia, a información dispoñíbel sobre un sistema non é suficiente para inferir con detalle a súa estrutura, dinámica e/ou parámetros de forma inequívoca. Nestes casos existen varios modelos consistentes cos datos. Nestas circunstancias pódese usar un enfoque de modelado por conxuntos (“ensemble modelling”). En lugar de considerar un só modelo, considéranse todos os plausíbeis. Polo xeral, todos os modelos do ensemble son do mesmo tipo (por exemplo, un ensemble de modelos en EDOs, ou de redes booleanas). Unha posíbel utilidade do modelado por conxuntos é cuantificar a incerteza nas predicións feitas polos modelos: se as predicións de todos os modelos no ensemble presentan un alto grao de consenso, pódese ter confianza nelas; pola contra, se difiren considerablemente, presentan un alto grao de incerteza.

1.4. Exemplo: modelando un biosistema con varios formalismos

Este exemplo ilustra como un mesmo sistema pode ser representado por varios modelos. Úsase como caso de estudo un modelo de regulación xénica. Este tipo de biosistemas describírase con máis detalle na sección 2.3.3. A figura 1.6 mostra un circuíto de regulación xénica simple [33]. Está formado por catro xenes, representados en rectángulos a–d. Cada xene é “transcrito” no seu correspondente ARN mensaxeiro (mRNA), representado en óvalos a–d, o cal é “traducido” nas proteínas correspondentes (hexágonos A–D). As proteínas A e B poden formar un heterodímero que activa a expresión do xene c. A figura mostra outros mecanismos de activación e inhibición.

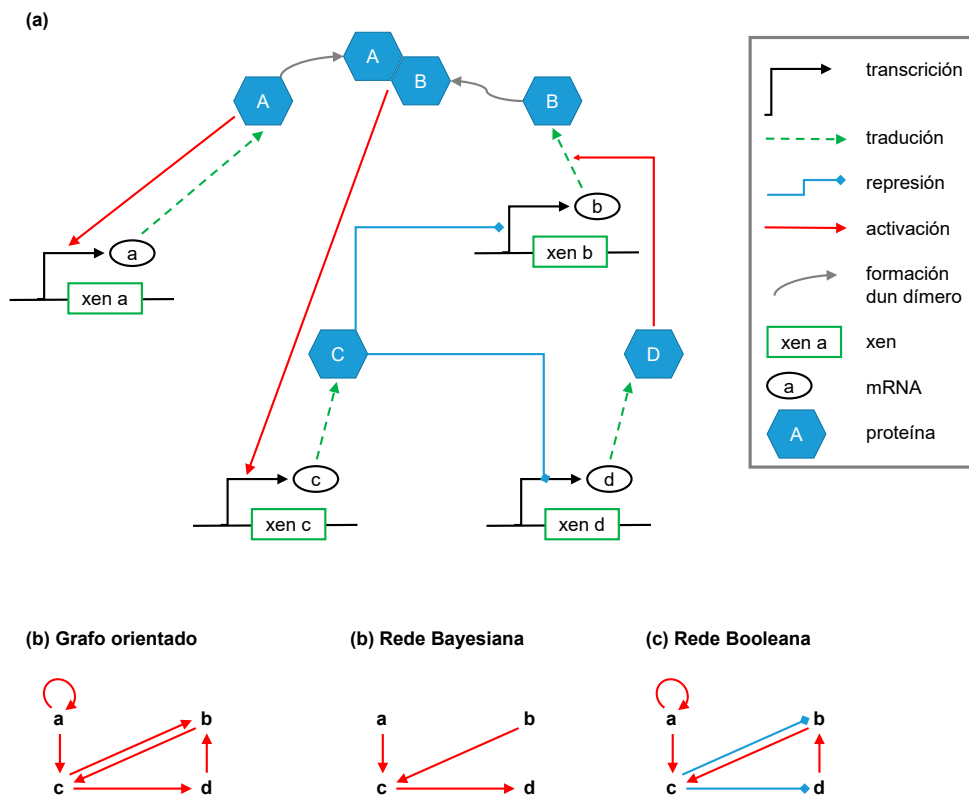


Figura 1.6: Parte superior (a): exemplo dunha pequena rede reguladora da expresión xénica. Parte inferior (b–d): varios modelos da rede tipo grafo.

1.4.1. Modelos do exemplo baseados en grafos

Hai varios xeitos de modelar o sistema da figura 1.6 como un grafo, segundo se describiu no apartado 1.3.1.

O subpanel 1.6.(b) mostra un *grafo orientado*. O xene a non está regulado por ningún outro, pero actívase a si mesmo. Xunto con b , activan c . Á súa vez, b é inhibido por c , polo que a interacción entre ambos é bidireccional. En canto a d , é inhibido por c e activa a b . Matematicamente, o grafo orientado da figura represéntase polo conxunto dos seus vértices (V) e o conxunto de ligazóns entre eles, co signo correspondente (L):

$$V = \{a, b, c, d\}$$

$$L = \{(a, a, +), (a, c, +), (b, c, +), (c, b, -), (c, d, -), (d, b, +)\}$$

Outro tipo de grafo é o mostrado en 1.6.(c). As *redes bayesianas* non conteñen bucles, e están totalmente definidas polas súas distribucións de probabilidades:

$$p(x_a)$$

$$p(x_b)$$

$$p(x_c | x_a, x_b)$$

$$p(x_d | x_c)$$

Outra opción é unha *rede booleana*. Este formalismo incorpora máis información que o grafo orientado, xa que permite especificar o sentido das interaccións (activación ou inhibición). As transicións definidas pola rede booleana do panel 1.6.(d) son as seguintes:

$$a(t+1) = a(t)$$

$$b(t+1) = (\text{NOT } c(t)) \text{ AND } d(t)$$

$$c(t+1) = a(t) \text{ AND } b(t)$$

$$d(t+1) = \text{NOT } c(t)$$

Representando o estado de cada xene como 1 (expresado) ou 0 (non expresado), temos un vector $[a, b, c, d]$ que pode ter $2^4 = 16$ estados. A Táboa 1.1 mostra todos os estados iniciais posíbeis e os estados sucesivos aos que se accede desde eles. As regras booleanas determinan as posíbeis traxectorias do sistema. Existe un número finito de traxectorias posíbeis. En concreto, a partir da táboa vese que esta rede ten dous tipos de comportamento estacionario,

dependendo do estado inicial do xene a : se $a(t) = 0$, a rede acaba no estado [0101], é dicir, cos xenes b e d expresados e a e c sen expresar. Se $a(t) = 1$, chégase a un comportamento cíclico: $1000 \rightarrow 1001 \rightarrow 1101 \rightarrow 1111 \rightarrow 1010 \rightarrow 1000$.

0000 $\rightarrow$ 0001	0100 $\rightarrow$ 0001	1000 $\rightarrow$ 1001	1100 $\rightarrow$ 1011
0001 $\rightarrow$ 0101	0101 $\rightarrow$ 0101	1001 $\rightarrow$ 1101	1101 $\rightarrow$ 1111
0010 $\rightarrow$ 0000	0110 $\rightarrow$ 0000	1010 $\rightarrow$ 1000	1110 $\rightarrow$ 1010
0011 $\rightarrow$ 0000	0111 $\rightarrow$ 0000	1011 $\rightarrow$ 1000	1111 $\rightarrow$ 1010

Cadro 1.1: Estados sucesivos da rede booleana da Fig. 1.6.(d).

Utilidade destes modelos. O grafo orientado e a rede bayesiana non caracterizan a dinámica da rede, pero poden proporcionar outro tipo de información útil. Por exemplo, cuantificar a súa conectividade, identificar lazos de realimentación, ou comparar a topoloxía das redes en diferentes organismos para inferir relacións evolutivas. Un modelo como a rede booleana proporciona máis información, por exemplo para indicar posíbeis dianas terapéuticas. Así, se quixésemos suprimir a expresión do xene c , poderíamos bloquealo (“gene knockout”); se dito bloqueo non fose práctico, unha alternativa sería bloquear a , xa que como vimos, se $a = 0$ chégase a un estado estacionario no que $c = 0$.

1.4.2. Modelos do exemplo baseados en ecuacións diferenciais

Construír un modelo en *ecuacións diferenciais ordinarias* do sistema da Fig. 1.6 implica definir as ecuacións (1.1). Definimos un vector de estado correspondente ao nivel de expresión de cada xene en cada momento: $x(t) = [a(t), b(t), c(t), d(t)]$; para simplificar a notación, omitimos a dependencia temporal: $x = [a, b, c, d]$. Para obter ecuacións diferenciais hai que asignar un tipo de cinética a cada interacción. Veremos as posíbeis cinéticas que se poden usar na sección 2.2. Para este exemplo escolleremos a cinética tipo Hill (2.8), que

dá lugar ás seguintes ecuacións (a súa dedución explícase embaixo):

$$\dot{a} = v_a - k_a \cdot a \quad (1.9)$$

$$\dot{b} = \frac{V_b \cdot (d/K_b)^{n_d}}{(1 + (d/K_b)^{n_d})(1 + (c/K_{Ic})^{n_c})} - k_b \cdot b \quad (1.10)$$

$$\dot{c} = \frac{V_c \cdot (a \cdot b/K_c)^{n_{ab}}}{1 + (a \cdot b/K_c)^{n_{ab}}} - k_c \cdot c \quad (1.11)$$

$$\dot{d} = \frac{V_d}{1 + (c/K_{Ic})^{n_c}} - k_d \cdot d. \quad (1.12)$$

Cada ecuación conta cun termo de degradación, polo que o mRNA decae proporcionalmente ($k_i \cdot i$); este termo introduce un parámetro constante k_i por cada estado. O xene a ten unha taxa de expresión constante, v_a , e os restantes xenes actívanse mediante unha cinética tipo Hill. O xene b é activado por d , o que escribimos como: $\frac{V_b \cdot (d/K_b)^{n_d}}{1 + (d/K_b)^{n_d}}$. Esta expresión introduce tres parámetros: V_b (velocidade máxima), K_b (taxa de disociación) e n_d (coeficiente de Hill). Como b está tamén inhibido por c , engádese o termo $(1 + (c/K_{Ic})^{n_c})$ no denominador. No caso de c , só ten termo de activación (polo complexo ab). No caso de d , só ten termo de inhibición (por c).

As EDOs (1.9)–(1.12) asumen que a regulación xénica é determinista e en tempo continuo, o que é unha simplificación. Debido ao baixo número de copias de cada molécula presente no proceso, é máis realista modelalo como un *proceso estocástico*. A integración de modelos estocásticos verase no capítulo 3.3.

Utilidade destes modelos. As ecuacións (1.9)–(1.12) permiten simular o comportamento dinámico da rede a partir dun estado inicial. Non é preciso asumir (como no caso das redes booleanas) que cada estado ten un valor igual a cero ou a un; a simulación pode tomar valores continuos. Para integrar as ecuacións é necesario coñecer os valores dos parámetros, o que en xeral se fai mediante calibración con datos experimentais. A esta tarefa está dedicado o capítulo 4, e en particular la sección 4.3. Por agora, simplemente suporemos que os parámetros teñen os seguintes valores: $v_a = 1$, $k_a = 1$, $V_b = 3$, $K_b = 0.1$, $K_{Ic} = 0.4$, $k_b = 2$, $V_c = 3$, $K_c = 0.1$, $k_c = 2$, $V_d = 3$, $k_d = 1$, $n_{ab} = n_c = n_d = 4$, obtendo as simulacións da Figura 1.7: Nótese

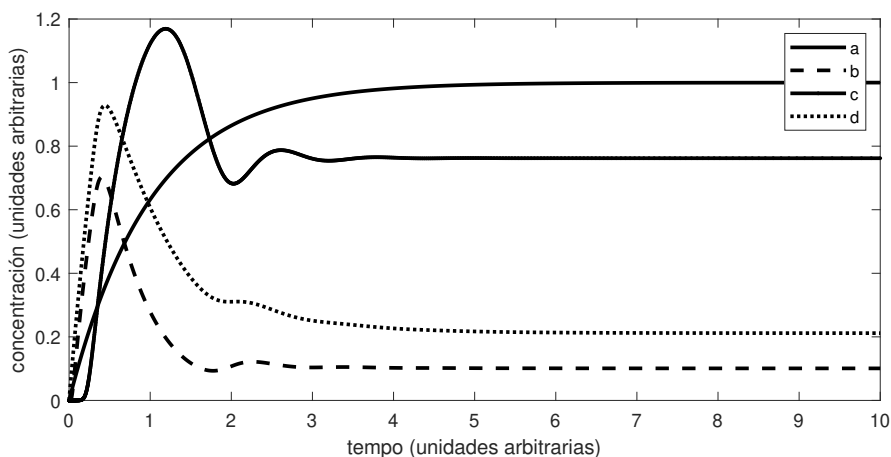


Figura 1.7: Simulación das EDOs (1.9)–(1.12)

que para simular a dinámica mediante o modelo booleano (Táboa 1.1) non foi preciso determinar ningún parámetro. Así pois, aínda que os modelos en EDOs proporcionan máis información que os booleanos, tamén precisan máis datos para a súa simulación.

1.5. Panorama da materia

Na **Lección 2** veremos distintos tipos de biosistemas de interese en medicina e, de forma análoga ao modelo de regulación xénica que acabamos de ver na sección 1.4, aprenderemos como modelalos.

Na **Lección 3** coñeceremos os métodos numéricos relevantes para a simulación dos modelos, é dicir, para integrar as súas ecuacións diferenciais.

Na **Lección 4** aprenderemos a construír e identificar un modelo, é dicir, a estimar os seus parámetros cun grao de certeza aceptábel e que así as súas simulacións teñan carácter predictivo.

Na **Lección 5** analizaremos o comportamento dinámico dos modelos, centrándonos en certos comportamentos de particular interese.

2

Modelos de biosistemas

“The best material model of a cat is another, or preferably the same, cat.”

– Norbert Wiener

“Don’t model bulldozers with quarks.”

– Leo Kadanoff

Nesta unidade veremos as características dos sistemas dinámicos máis estudados no ámbito biomédico, e cales son os enfoques máis apropiados para construír modelos deles.

2.1. Sobre os tipos de biosistemas de interese

Os procesos biolóxicos de interese en biomedicina son de moi diversos tipos, e teñen lugar nun rango moi amplo de escalas espazo-temporais (Fig. 2.1). Nesta lección veremos as características de algúns dos máis relevantes, clasificándoos en tres niveis:

- Nivel celular. Son os tipicamente estudados en bioloxía de sistemas.
- Nivel organismo. P.ex., regulación fisiolóxica ou farmacocinética.

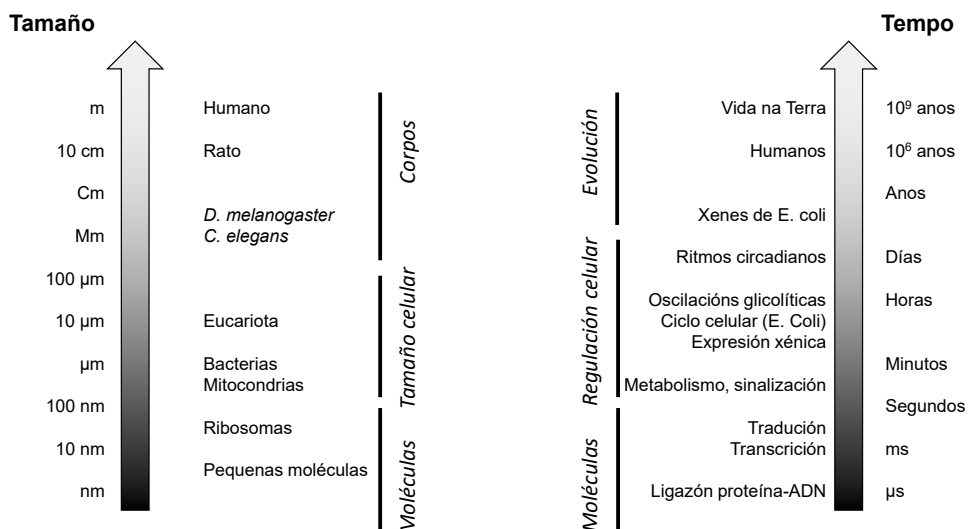


Figura 2.1: Distintas escalas espazo-temporais nas que teñen lugar os procesos biolóxicos. Elaborado a partir de [33].

- Nivel comunitario. Fenómenos que involucran a varios individuos dunha especie ou de varias.

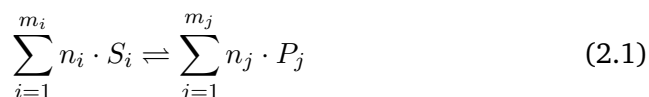
Veremos cales son os enfoques máis apropiados para modelalos. Por regra xeral, o “mellor” modelo será o máis simple de entre os capaces de responder as preguntas que se lle quere formular. Por exemplo, se queremos coñecer a distribución de fluxos dunha rede metabólica en estado estacionario, non precisamos detalles cinéticos: un modelo estequiométrico será suficiente. Pola contra, se queremos predicir o tempo no que se alcanza dito estado, necesitamos un modelo dinámico. É importante ter en conta que a dificultade e o custo computacional de identificar e simular un modelo aumenta rapidamente co seu tamaño. Por iso, un modelo non debe ser innecesariamente complexo.

Muitos dos biosistemas que imos ver inclúen reaccións bioquímicas, ou poden ser modelados de forma análoga ás redes bioquímicas. Por iso, antes de entrar nos detalles dos diferentes tipos de biosistemas, imos ver as cinéticas máis comunmente usadas para describir ditas reaccións.

2.2. Cinética das reaccións bioquímicas

Existen diferentes formas de modelar a cinética das reaccións bioquímicas.

Lei de acción de masas. Supoñamos unha reacción química na que m_i substratos, cuxas concentracións se denotan por S_i , reaccionan para dar m_j produtos de concentracións P_j . As molecularidades de cada substrato e cada produto (é dicir, o número polo que aparecen multiplicados na reacción) son n_i e n_j respectivamente. Dita reacción ten a seguinte expresión:



A velocidade á que ten lugar esta relación é proporcional á probabilidade de colisión dos reactivos, é dicir:

$$v = v_+ - v_- = k_+ \prod_{i=1}^{m_i} S_i^{n_i} - k_- \prod_{j=1}^{m_j} P_j^{n_j} \quad (2.2)$$

No equilibrio $v = 0$, polo que a constante de equilibrio K_{eq} é:

$$K_{eq} = \frac{k_+}{k_-} = \frac{\prod_{j=1}^{m_j} P_{j,eq}^{n_j}}{\prod_{i=1}^{m_i} S_{i,eq}^{n_i}} \quad (2.3)$$

Cando non se está en equilibrio, as concentracións das especies participantes na reacción varían segundo a seguinte dinámica:

$$\frac{d}{dt} S_i = -n_i \cdot v, \quad \frac{d}{dt} P_j = n_j \cdot v \quad (2.4)$$

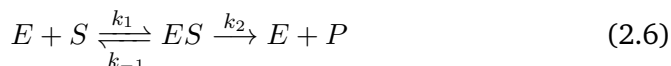
A evolución temporal das concentracións das especies calcúlase integrando as ecuacións dinámicas. Isto non é trivial porque n_i e n_j son constantes pero a velocidade de reacción v non o é, senón que é función das concentracións e outros parámetros.

S-systems. O chamado “S-systems approach”, ou “Biochemical Systems Theory”, é unha xeneralización da cinética de acción de masas desenvolvida por Savageau [46]. Para reflectir o feito de que na práctica a velocidade non sempre depende linearmente das concentracións, introduce potencias (“power law kinetics”), que poden ter valores fraccionais e mesmo negativos:

$$\frac{dX_i}{dt} = \alpha_i \prod_{j=1}^{n+m} X_j^{g_{ij}} - \beta_i \prod_{j=1}^{n+m} X_j^{h_{ij}} \quad (2.5)$$

As reaccións bioquímicas son catalizadas por enzimas (proteínas específicas), e a expresión de v depende do tipo de cinética enzimática. Hai dous tipos principais, Michaelis-Menten e Hill, que se describen a continuación:

Michaelis-Menten. Este tipo de cinética asume que hai un substrato e un produto, e que a reacción é irreversible:



Baixo determinados supostos, a velocidade de reacción ven dada pola expresión:

$$v = \frac{V_{\max} S}{S + K_m} \quad (2.7)$$

na que aparecen dous parámetros: a velocidade máxima, $V_{\max} = k_2 E_{\text{total}}$, e a constante de Michaelis, $K_m = (k_{-1} + k_2)/k_1$, que é a concentración de substrato á cal $v = V_{\max}/2$.

Cooperatividade: a ecuación de Hill. Moitas moléculas bioloxicamente importantes, incluíndo algúns enzimas, teñen múltiples sitios de unión. Nas moléculas *cooperativas*, a unión dun ligando nun sitio altera a afinidade doutros sitios de unión polos seus ligandos. Este fenómeno é coñecido tamén como *regulación alostérica*.

Un exemplo de molécula cooperativa é a hemoglobina, a molécula portadora de osíxeno nos glóbulos vermellos, que une catro moléculas de osíxeno, cada unha nun sitio de unión distinto. A hemoglobina recolle catro moléculas de O_2 nos pulmóns, nun ambiente rico en osíxeno, e entrégaa en tecidos

cunha concentración de osíxeno muito menor. Ao ser cooperativa, a unión dunha molécula de O_2 nun sitio aumenta a afinidade dos outros sitios por O_2 , facilitando a unión de máis O_2 . Pola contra, a medida que cada molécula de O_2 é liberada nun tecido, a afinidade dos sitios restantes por O_2 diminúe, facilitando a disociación de máis moléculas de O_2 . Así pois, a cooperatividade é fundamental para a función da hemoglobina.

Unha cinética cooperativa modélase mediante a ecuación de Hill:

$$v = \frac{V_{\max} S^n}{K^n + S^n} = \frac{V_{\max} (S/K)^n}{1 + (S/K)^n} \quad (2.8)$$

A respecto da cinética Michaelis-Menten (2.2), a de Hill introduce un parámetro adicional, n , que é chamado coeficiente de Hill e indica o grao de cooperatividade.

- Se $n = 1$ non existe cooperatividade, a ecuación de Hill redúcese á de Michaelis-Menten.
- Se $n > 1$ a cooperatividade é positiva: a unión dunha molécula de substrato aumenta a velocidade de unión doutros substratos.
- Se $n < 1$ a cooperatividade é negativa (a velocidade diminúe).

A gráfica 2.2 mostra curvas de velocidade fronte a substrato para varios tipos de cooperatividade. Se a cooperatividade é positiva as curvas son de tipo sigmoide. Valores típicos son $n = 2$, $n = 3$. A cooperatividade moi alta tamén se lle chama ultrasensibilidade. Canto maior é n , máis se aproxima o comportamento a un interruptor: para concentracións baixas de substrato a reacción (case) non ten lugar, mentres que a partir dun certo valor a reacción pasa a producirse a velocidade máxima (ou case). Un exemplo deste comportamento é a vía MAPK que veremos en 2.9.

2.3. Nivel celular

Dentro dunha célula teñen lugar distintos tipos de procesos químicos:

- Producción e degradación de substancias.

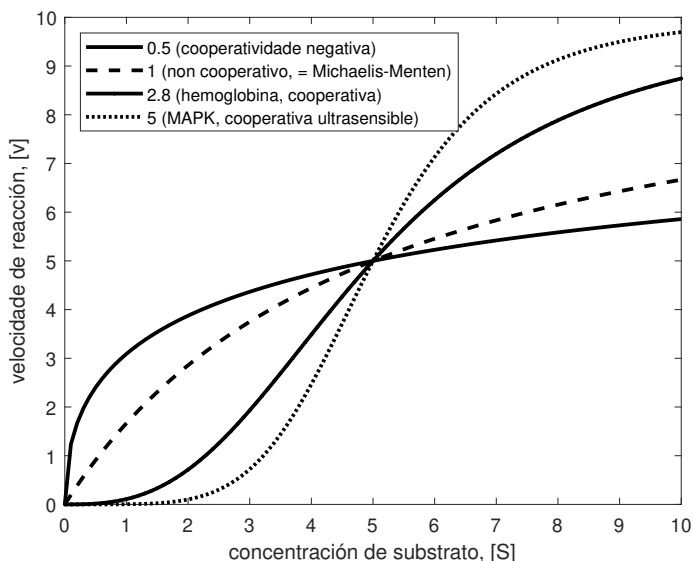


Figura 2.2: Curvas de velocidade de reacción fronte a concentración de substrato, para cinética de Hill con diferentes tipos de cooperatividade.

- Modificacións moleculares como fosforilación ou metilación.
- Activación e inhibición de reaccións.

Estes procesos son executados por redes de reaccións bioquímicas, clasificábeis segundo a súa función. Por exemplo, as redes metabólicas encárganse de asegurar a enerxía e os materiais de construción que as células precisan para sobrevivir e reproducirse. A sinalización celular encárgase de detectar sinais externas e procesar a súa información, permitindo a toma de decisións.

Nas últimas décadas téñense producido importantes avances no modelado dinámico de redes celulares individuais. Ditas redes non funcionan por separado, senón que están conectadas entre si. Por tanto, sería desexábel integrar os distintos modelos individuais para obter un modelo de célula completa (“whole cell model”), capaz de predicir o seu comportamento [50]. Non obstante, a pesar de que se teñen publicado algúns modelos deste tipo (sendo o primeiro o modelo de *E. coli* de [31]), as súas capacidades predictivas son moi limitadas,

polo que aínda estamos lonxe de conseguir este obxectivo.

2.3.1. Metabolismo

Descrición xeral. O metabolismo celular é o conxunto de procesos mediante os cales as células obteñen enerxía e constrúen os seus compoñentes. Consta de reaccións *catabólicas*, que producen enerxía a partir da degradación de compostos complexos, e reaccións *anabólicas*, que producen compoñentes necesarios para o funcionamento celular. Nestas reaccións adoitan participar enzimas, sen as cales as reaccións serían extremadamente lentas.

Os procesos metabólicos están altamente organizados en *rotas metabólicas específicas*. A fig. 2.3 mostra parte dunha destas rotas, a glicólise. Estas rotas ou *vías* metabólicas básicas teñen un alto grao de semellanza incluso entre especies moi diferentes. Por exemplo, o ciclo de Krebs é común a bacterias unicelulares como a *Escherichia coli* e a organismos pluricelulares.

O metabolismo é a rede celular máis próxima ao fenotipo. É o resultado da multitude de capas de regulación intracelulares. É actualmente o único sistema celular suficientemente caracterizado como para ser modelado e simulado a escala xenómica [1], mediante os chamados “genome-scale models” ou GEMs. Simúlanse en base á súa estequiometría, adoptando un enfoque baseado en restricións. Polo seu interese, este enfoque descríbese a seguir.

Modelado estequiométrico baseado en restricións: FBA. A análise do fluxo de equilibrio – coñecido polas súas siglas en inglés, FBA (“Flux Balance Analysis”) – é un método para calcular os fluxos metabólicos dun sistema biolóxico. O método asume que o sistema está en estado estacionario. Esta suposición permite prescindir de detalles sobre a súa dinámica, polo que non precisa coñecer a cinética enzimática do sistema nin as concentracións de metabolitos. A cambio, proporciona menos información que outros métodos máis detallados, xa que só calcula os fluxos en estado estacionario.

En FBA os cambios de concentracións dunha rede metabólica descríbense como o produto da matriz estequiométrica S e o vector dos fluxos v . A matriz S contén os coeficientes estequiométricos dos metabolitos participantes nas reaccións bioquímicas. O resultado do produto $S \times v$ é un vector de ceros, xa que o sistema está en estado estacionario. O cálculo do vector de fluxos

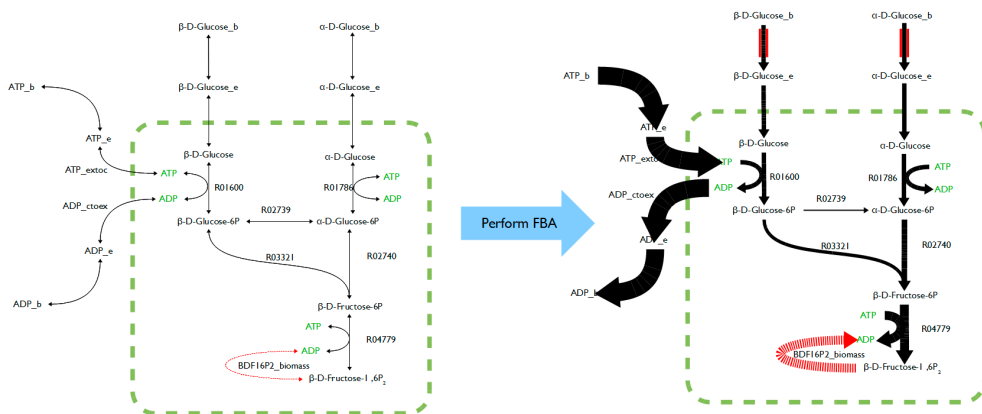


Figura 2.3: FBA. O grosor das liñas na figura da dereita representa os valores dos fluxos metabólicos calculados. Imaxe: Thomas Forth – Own work, CC BY-SA 3.0 <https://commons.wikimedia.org/w/index.php?curid=9709978>.

v é un problema indeterminado, ten infinitas solucións. Para resolvelo hai que introducir algún suposto adicional, tipicamente que os fluxos metabólicos maximizan a velocidade de crecemento do organismo. Isto implica definir a composición da biomasa e as reaccións que a forman. Unha vez definida, úsase programación lineal para calcular unha solución de fluxos correspondente ao estado estacionario. A Fig. 2.3 mostra un exemplo, no cal os valores dos fluxos obtidos se representan graficamente en función do grosor das liñas.

As simulacións realizadas usando FBA son computacionalmente eficientes, podéndose calcular os fluxos para modelos con millares de reaccións en poucos segundos en ordenadores persoais modernos. Un exemplo notable do éxito do FBA é a capacidade de predicir con precisión a taxa de crecemento da *E. coli* cando se cultiva en diferentes condicións.

Características principais dos sistemas metabólicos:

- Relevantes en: doenzas metabólicas, como diabetes ou síndrome metabólica; tamén noutras como os cancros.
- Escala temporal: minutos, horas.

- Formalismos adecuados: EDOs (para modelos de rutas curtas ou medianas), modelos estequiométricos baseados en restricións (para modelos grandes, até escala xenómica).
- Tamaños dos modelos: desde ~ 10 s compoñentes até escala xenómica con ~ 1000 s.
- Tipos de datos relevantes: transcriptómica, proteómica, metabolómica.
- Exemplos: glicólise, gliconeoxénese, ciclo de Krebs.

2.3.2. Sinalización celular

A sinalización celular é o intercambio de información entre unha célula e o seu entorno, incluíndo outras células. Esta comunicación realízase mediante receptores de membrana, que detectan a presenza de moléculas específicas (información entrante), e mediante a liberación de moléculas (información saínte). Dentro da célula a información é procesada mediante redes de reaccións bioquímicas chamadas *vías de sinalización* ou *transducción de sinal*.

O coñecemento actual das rutas de sinalización é incompleto. Polo xeral, nin se coñecen todos os compoñentes dunha ruta en particular, nin as interaccións entre eles (ou con compoñentes de outra ruta, “crosstalk”).

Cualitativamente, os procesos involucrados na sinalización celular son os mesmos que no metabolismo: produción e degradación de substancias, modificacións moleculares (fosforilación, metilación, etc), e activación e inhibición de reaccións. Porén, cuantitativamente son moi diferentes (para máis información, consultar a sección 3.4 de [33]).

Características principais dos sistemas de sinalización celular:

- Relevantes en: multitude de doenzas.
- Escala temporal: (segundos), minutos, (horas).
- Formalismos adecuados: grafos orientados, redes booleanas, EDOs.
- Tamaños dos modelos: de < 10 a ~ 100 s compoñentes.

- Tipos de datos relevantes: transcriptómica, proteómica, fosfoproteómica.
- Exemplos: JAK-STAT, MAPK, NF- κ B...

Exemplo 1: un modelo da cascada MAPK. A proteína quinase activada por mitóxeno (“mitogen-activated protein kinase”, MAPK) é unha rede de sinalización moi estudada. Unha proteína quinase é un enzima que modifica outras proteínas engadíndolles grupos fosfato (fosforilación). A fosforilación provoca un cambio funcional na proteína. As MAPK interveñen nas respostas celulares a un conxunto diverso de estímulos, como os mitóxenos, estrés osmótico, shock térmico e citocinas proinflamatorias.

A Fig. 2.4 mostra unha cascada de tres capas que se atopa na ruta de sinalización MAPK/ERK. Inclúe un lazo de realimentación negativo, así como realimentación positiva. Consta de tres especies, S_1 , S_2 , S_3 , e as súas correspondentes fosforilacións pS_1 , pS_2 , pS_3 . Como a cantidade total de cada especie (fosforilada, pS_i , e non fosforilada, S_i) é constante (S_{it}), o sistema pódese modelar con só 3 estados en lugar de 6, xa que $pS_i(t) = S_{it} - S_i(t)$. As súas ecuacións dinámicas son:

$$\begin{aligned} \dot{pS}_1 &= k_1 \cdot (S_{1t} - pS_1) \cdot \left(\frac{K_1^{n_1}}{K_1^{n_1} + pS_3^{n_1}} \right) - k_2 \cdot pS_1, \\ \dot{pS}_2 &= k_3 \cdot (S_{2t} - pS_2) \cdot pS_1 \cdot \left(1 + \frac{\alpha \cdot pS_3^{n_2}}{K_2^{n_2} + pS_3^{n_2}} \right) - k_4 \cdot pS_2, \\ \dot{pS}_3 &= k_5 \cdot (S_{3t} - pS_3) \cdot pS_2 - k_6 \cdot pS_3, \end{aligned} \quad (2.9)$$

onde se modelan dúas interaccións con cinéticas tipo Hill.

Este modelo da cascada de sinalización MAPK foi presentado en [42] e a súa identificabilidade estrutural analizouse en [54]. Como veremos na Lectión 5, ten un comportamento dinámico interesante que inclúe oscilacións e biestabilidade.

Exemplo 2: uso dun modelo para evitar efectos adversos en tratamento anticanceroso. O IGF1R (“Insulin-like Growth Factor type 1 Receptor”, ou receptor do factor de crecemento similar á insulina tipo I) está ligado ao crecemento e propagación do cancro. Inhibindo a actividade de IGF1R retárdase

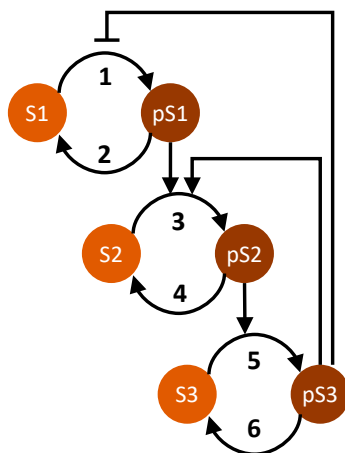


Figura 2.4: Esquema dunha cascada de sinalización MAPK con 3 capas.

o crecemento do cancro. Desafortunadamente, as moléculas que se unen ao IGF1R tamén se unen ao receptor de insulina (InsR), que está moi relacionado co IGF1R, perturbando así o metabolismo dun xeito similar á diabetes. Para conseguir alterar a actividade do IGF1R sen afectar á do InsR hai que determinar as diferenzas entre os mecanismos de sinalización Ins e IGF1. No artigo [21] construíuse un modelo computacional para responder esta pregunta. Dito modelo conta con 38 estados e 66 parámetros. O modelo suxeriu que a proteína ribosómica S6 quinase (RPS6K) xoga un papel crítico na regulación dos niveis de activación MAPK e Akt en resposta á estimulación con Ins e IGF1. Para validar experimentalmente esta predición perturbouse a quinase RPS6K, obténdose que a estimulación con Ins provocou unha maior activación de Akt que a estimulación con IGF1. Este descubrimento proporciona oportunidades para mellorar as terapias contra o cancro, ao posibilitar inhibir a actividade de IGF1R sen interromper a sinalización InsR.

2.3.3. Expresión xénica

A expresión xénica é o proceso polo que a información codificada nos xenes se utiliza para sintetizar produtos funcionais (frecuentemente proteínas). A información xénica almacénase no ADN. O chamado “dogma central” [13] da bioloxía molecular sostén que o ADN fai o ARN (*transcrición*) e o ARN fai as proteínas (*tradución*), e non hai fluxo de información xenética en sentido inverso. Esta visión simplificada ilústrase na Fig. 2.5.

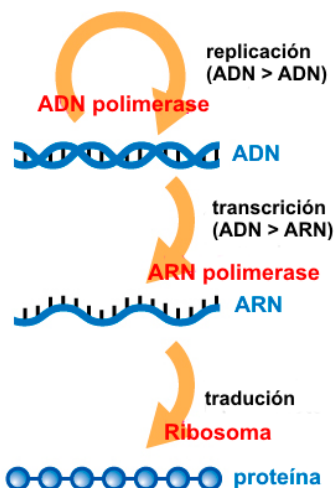


Figura 2.5: O “dogma central da bioloxía molecular”. Fonte: Wikimedia Commons, licenza CC BY-SA 3.0 (https://upload.wikimedia.org/wikipedia/commons/6/6b/Central_Dogma_of_Molecular_Biochemistry_with_Enzymes_gl.png).

En realidade, o proceso é máis complexo e consta de máis fases. En células eucariotas considérase que a expresión xénica é controlada en seis pasos. O proceso comeza cando certas proteínas, chamadas factores de transcrición, se unen a secuencias de ADN para iniciar o proceso de transcrición. Este mecanismo chámase *control transcricional*, e é o máis importante dos que interveñen na expresión xénica.

Mentres que a transcrición de ADN a ARN ten lugar no núcleo, a tradución de ARN a proteína ten lugar en ribosomas localizados no citoplasma. Por

tanto, é necesario que o ARN sexa transportado do núcleo aos ribosomas, a través de poros da membrana nuclear (un paso que, como os outros, tamén é regulado). Este ARN é denominado ARN mensaxeiro ou ARNm. Un paso clave é a *regulación post-transcricional*, pola que pequenas moléculas de ARN chamadas micro-ARNs se unen a secuencias de ARNm, actuando como represores da tradución.

Como os xenes deben expresarse en función do estado da célula e as condicións ambientais, hai intercambios de información con rutas de sinalización (para compilar información do ambiente–véxase sección 2.3.2) e con vías metabólicas (para obter información sobre o estado da célula–véxase sección 2.3.1). É dicir, a expresión xénica é regulada por *redes* nas que interveñen moléculas de ADN, ARN, proteínas, e outras. Un *operón* é unha unidade funcional do ADN que contén un grupo de xenes que están baixo o control dun só sinal regulador ou promotor.

Características principais dos modelos de expresión xénica:

- Relevantes en: ciclo celular, mantemento da vitalidade das células, resposta a estímulos.
- Escala temporal: minutos.
- Formalismos adecuados: grafos (orientados), redes booleanas, EDOs.
- Tamaños dos modelos: de < 10 a ~ 100 s de compoñentes.
- Tipos de datos relevantes: transcriptómica.
- Exemplos: un exemplo clásico é o operón Lac, o cal regula – en *E. coli* e outras bacterias – a produción dos enzimas necesarios para aproveitar lactosa como fonte de enerxía en lugar de glicosa, en contextos nos que esta última escasea.

Exemplo 3: un sistema, varios modelos. Na Sección 1.4 vimos como construír varios modelos diferentes dun sistema de expresión xénica.

2.4. Nivel orgánico

2.4.1. Electrofisioloxía

A electrofisioloxía cardíaca é unha das áreas da biomedicina onde máis se emprega o modelado matemático e computacional. Os modelos electrofisiolóxicos describen como as correntes iónicas producen o potencial de acción e dan lugar a propiedades emerxentes en tecidos enfermos e sans. As canles iónicas regulan o fluxo de ións a través das membranas celulares. A súa condutancia pode ser descrita polos modelos en EDOs, de Hodgkin-Huxley (HH) e de Modelos de procesos de decisión de Markov (MM). Os MM son da forma

$$\begin{aligned}\dot{x} &= M(p, u) \cdot x, \\ y &= \sum_{i=1}^{n_1} x_i\end{aligned}\tag{2.10}$$

onde u é o potencial de membrana, x é o vector de n estados condutores normalizados (abertos) ou non condutores (fechados), a observación y é a suma de todos os estados condutores de n_1 , e M é unha matriz $n \times n$, normalmente dispersa. A matriz M conserva a masa, é dicir, $\sum_{i=1}^n x_i(t)$ é constante. En experimentos, a condutancia $y(t)$ cambia co paso do tempo a medida que o potencial de membrana forzado, $u(t)$, é variado polos experimentadores, alterando así as velocidades de reacción. A influencia do potencial da membrana nas velocidades de reacción pode modelarse como:

$$M_{ij}(u(t)) = \exp(a_{ij} + b_{ij} \cdot u(t)).\tag{2.11}$$

Os MM son unha xeneralización dos modelos HH que son máis versátiles grazas ao uso de máis parámetros. Non obstante, esta maior flexibilidade tamén os fai máis propensos a problemas de identificabilidade.

Características principais dos modelos electrofisiolóxicos (MM ou HH):

- Relevantes en: estudo de doenzas cardíacas, p.ex. arritmias.
- Escala temporal: milisegundos, segundos.
- Formalismos adecuados: EDOs (son fortemente non-lineares).

- Tamaños dos modelos: de < 10 a ~ 10 s de compoñentes.
- Tipos de datos relevantes: eléctricos (intensidades e voltaxes).
- Exemplos: modelos de células ventriculares cardíacas.

Exemplos:

- O modelo clásico de Hodgkin-Huxley (ex. 8.6.4 do libro de Ingalls, [28]).
- Un exemplo recente dun modelo electrofisiolóxico complexo: <https://doi.org/10.1371/journal.pcbi.1009137>.
- Modelo electrofisiolóxico para ondas peristálticas: <https://doi.org/10.1371/journal.pcbi.1009644>.

2.4.2. Regulación da glicosa

A insulina, a hormona que regula o nivel de glicosa en plasma, é segregada nas células β do páncreas. En pacientes de diabetes tipo 1, a destrución auto-inmune das células β tradúcese en elevadas concentracións de glicosa, o que ten efectos prexudiciais que poden ser graves a longo prazo. Entre os esforzos dedicados nas últimas décadas a tratar a diabetes destacan os enfocados no desenvolvemento dun páncreas artificial (AP). Un AP é un sistema de control de glicosa en circuío fechado, que dispensa insulina automaticamente. Para levar a cabo o control, é crucial ser capaz de estimar e predicir correctamente a concentración de insulina [7]. Un compoñente básico dos APs son os sistemas de seguimento continuo de glucosa (CGM). En 2016, a FDA (a administración estadounidense para os alimentos e medicamentos) aprobou o primeiro sistema de AP.

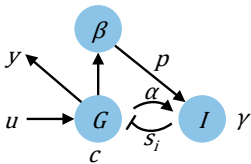
Características principais dos modelos de glicosa-insulina:

- Relevantes en: tratamento da diabetes, desenvolvemento de sistemas de páncreas artificial.

- Escala temporal: minutos, horas.
- Formalismos adecuados: típicamente modelos en EDOs, compartimentais (lineares).
- Tamaños dos modelos: de < 10 a ~ 10 s de compoñentes.
- Tipos de datos relevantes: concentracións en plasma.
- Exemplos: tres modelos clásicos, en nivel de complexidade ascendente, son os de [6], [27], e [14].

Exemplo: modelo β IG. O diagrama (A) da figura 2.6 representa a regulación do nivel de glicosa en plasma mediante insulina [29].

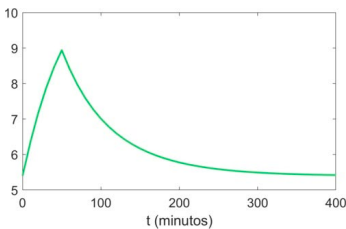
(A) Sistema dinámico



(B) Modelo mecanicista en EDOs

$$\begin{aligned} \dot{G} &= u + (c + s_i I)G \\ \dot{\beta} &= \beta \left(\frac{1.4583 \cdot 10^{-5}}{1 + \left(\frac{8.4}{G}\right)^{1.7}} - \frac{1.7361 \cdot 10^{-5}}{1 + \left(\frac{G}{4.8}\right)^{8.5}} \right) \\ \dot{I} &= p\beta \frac{G^2}{\alpha^2 + G^2} - \gamma I \\ y &= G; \\ G(0) &\text{ coñecido; } \beta(0) \text{ e } I(0) \text{ descoñecidos} \end{aligned}$$

(C) Saída medida: G (glicosa)



(D) Predición: I (insulina)

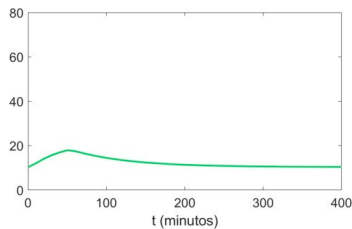


Figura 2.6: Circuito de regulación da glicosa en sangue: modelo “ β IG” [37].

A entrada (u) é a inxestión de glicosa. A concentración de glicosa (G) afecta ás células β , as cales segregan insulina (I) a velocidade p , e a insulina fai

baixar a concentración de glicosa a velocidade s_i . Este sistema dinámico pode describirse cun modelo compartimental con EDOs non-lineares, mostradas no panel (B). A gráfica (C) mostra a simulación da curva de glicosa, e a (D) a de insulina. Nas Leccións 4 (Sección 4.1) e 5 (Seccións 5.5.3 e 5.6) revisaremos este modelo.

2.4.3. Farmacocinética e farmacodinámica

A farmacocinética (*pharmacokinetics*, PK) é a parte da farmacoloxía que estuda os procesos aos que é sometido unha sustancia desde que é inxerida até a súa eliminación. Estes procesos coñécense como LADME (Liberación, Absorción, Distribución, Metabolismo e Excreción). Ás veces os dous primeiros agrúpanse nun só. A farmacocinética cuantifica a variación ao longo do tempo da concentración da sustancia (un medicamento, un aditivo, etc) en diferentes compartimentos do corpo humano. A complexidade dos modelos varía en función do número de compartimentos considerados, que soen ser poucos (xeralmente non máis de dous), do nivel de detalle con que se modelan os procesos, e do tipo de cinética considerada.

Pola súa banda, a farmacodinámica (*pharmacodynamics*, PD) estuda o efecto das drogas no corpo humano (ou, en xeral, nun organismo). É dicir, mentres que a PK estuda o efecto do corpo nunha sustancia, a PD estuda o efecto da sustancia no corpo. A maioría das drogas causan efectos nas células debido a que son detectadas por receptores situados na superficie celular. Así, ao igual que os PK, os modelos PD tamén describen reaccións bioquímicas: neste caso, as reaccións entre ligando (droga) e receptor.

É posíbel combinar ambos tipos de modelos, o que dá lugar a modelos PKPD.

Características principais dos modelos PK / PD:

- Relevantes en: deseño de fármacos e tratamentos.
- Escala temporal: minutos, horas.
- Formalismos adecuados: modelos en EDOs (para individuo), tipicamente compartimentais; e modelos non-lineares de efectos mixtos (NON-MEM, para poboacións).

- Tamaños dos modelos: pequenos (< 10 estados e parámetros).
- Tipos de datos relevantes: medidas de concentracións en plasma, etc.
- Exemplos: modelos PK de 1 ou 2 compartimentos, como o seguinte:

Exemplo. Consideremos o seguinte modelo de dous compartimentos: (1) o compartimento central x_1 , formado polo plasma e outros tecidos onde o medicamento se distribúe de forma case instantánea, e (2) o compartimento periférico x_2 , no que a distribución do medicamento é máis lenta:

$$\begin{aligned}
 \dot{x}_1 &= -(k_{12} + k_{10})x_1 + k_{21} \cdot x_2, \\
 \dot{x}_2 &= k_{12} \cdot x_1 - k_{21} \cdot x_2, \\
 y &= x_1, \\
 x_1(0) &= \text{dose}, \\
 x_2(0) &= 0
 \end{aligned}
 \tag{2.12}$$

Asúmese que se mide a concentración do medicamento no plasma.

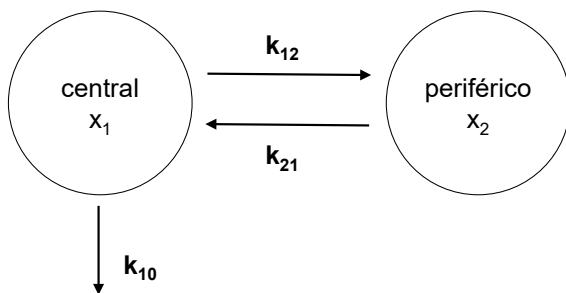


Figura 2.7: O modelo PK de 2 compartimentos das ecuación (2.12). Os procesos de distribución-metabolismo-eliminación módelanse con cinéticas de primeira orde k_* . A liberación-absorción considérase instantánea ao recibir a dose. Poderíase modelar de forma explícita con certa cinética, o cal engadiría outra frecha entrante ao compartimento 1.

2.5. Nivel de poboacións

2.5.1. Epidemioloxía

O modelado matemático de epidemias coñeceu o seu momento de gloria en 2020, mais xa contaba cunha longa tradición. Os modelos máis usados en epidemioloxía son os compartimentais. Estes modelos clasifican a poboación en compartimentos homoxéneos, cada un dos cales se corresponde cunha fase da doenza. A variábel de estado asociada cuantifica o número de individuos que están nesa fase. A súa dinámica modélase cunha ecuación diferencial (normalmente unha EDO). As ecuacións conteñen parámetros que describen as velocidades ás que os individuos pasan dunha fase a outra.

Hai dúas grandes familias de modelos compartimentais epidemiolóxicos, segundo inclúan ou non un compartimento para individuos expostos: SIR e SEIR.

Modelos SIR. No modelo epidemiolóxico básico, o SIR, a poboación divídese en tres clases:

- *Susceptibles (S)*: individuos que non teñen inmunidade e poden infectarse se están expostos.
- *Infectados e infecciosos (I)*: individuos que contraeron a enfermidade e poden transmitir.
- *Recuperados (R)*: individuos inmunes á enfermidade e que non afectan a súa transmisión.

As persoas susceptibles inféctanse cunha incidencia de:

$$-\beta SI$$

onde $\beta = p \cdot c$ é a taxa de transmisión, c é a taxa de contacto e p a probabilidade de que un contacto cun individuo susceptible produza unha transmisión. Os individuos que se recuperan abandonan a clase infecciosa a velocidade γ , onde $1/\gamma$ é o período infeccioso medio. O tamaño total da poboación é coñecido

(N). O conxunto de ecuacións diferenciais que describen o modelo SIR básico vén dado por:

$$\begin{aligned}\dot{S}(t) &= -\frac{\beta S(t)I(t)}{N} \\ \dot{I}(t) &= \frac{\beta S(t)I(t)}{N} - \gamma I(t) \\ \dot{R}(t) &= \gamma I(t)\end{aligned}$$

Modelos SEIR. Outra clase de modelos epidemiolóxicos, os SEIR, inclúen un compartimento adicional para ter en conta a existencia dun período de latencia despois da transmisión:

- *Expostos (E)*: individuos vulnerábeis a contraer a doenza, que entraron en contacto con ela mais aínda non son infecciosos.

En comparación cos modelos SIR, o compartimento adicional E permite unha descrición máis precisa das enfermidades nas que o período de incubación e o período latente non coinciden, é dicir, o tempo no que un infectado se fai infeccioso. É por iso que os modelos SEIR son en principio máis axeitados para epidemias cun longo período de incubación como COVID-19.

Os individuos susceptibles pasan á clase exposta cunha taxa $\beta I(t)$, onde β é o parámetro da taxa de transmisión. Os individuos expostos infécense a razón de κ , onde $1/\kappa$ é o período latente medio. Os individuos infectados recupéranse a un ritmo γ , onde $1/\gamma$ é o período infeccioso medio. O conxunto de EDOs que describen o modelo SEIR é:

$$\begin{aligned}\dot{S}(t) &= -\frac{\beta S(t)I(t)}{N} \\ \dot{E}(t) &= \frac{\beta S(t)I(t)}{N} - \kappa E(t) \\ \dot{I}(t) &= \kappa E(t) - \gamma I(t) \\ \dot{R}(t) &= \gamma I(t)\end{aligned}$$

Tanto os modelos SIR como os SEIR son simplificacións que non consideran algúns aspectos importantes. Para facelos máis realistas teñen sido propostas moitas extensións. Por exemplo, a consideración de compartimentos adicionais para modelar, entre outros: individuos asintomáticos, hospitalizados, ou en corentena; o efecto de nacementos e mortes na poboación (en lugar de considerala constante); ou os efectos de medidas como a obrigatoriedade da distancia social ou o confinamento.

Características principais dos modelos epidemiolóxicos:

- Relevantes en: predición da evolución dunha epidemia, e para avaliar o efecto de posíbeis medidas paliativas.
- Escala temporal: días.
- Formalismos adecuados: típicamente modelos en EDOs, compartimentais.
- Tamaños dos modelos: pequenos (< 10 estados e parámetros)
- Tipos de datos relevantes: número de individuos hospitalizados, mortes, infecciosos detectados (hai que ter en conta que en moitos casos só se detecta unha porcentaxe).
- Exemplos: modelos de COVID-19 (p.ex. os do review [38]).

2.5.2. Comunidades microbianas

Os microorganismos dominan case todos os nichos ecolóxicos do noso planeta. Recentemente comezamos a obter información detallada sobre a composición das comunidades microbianas, grazas aos avances experimentais en bioloxía molecular (secuenciación de ADN). Nestas comunidades danse diferentes tipos de interaccións, como a competencia, o mutualismo, o comensalismo ou o amensalismo (Fig. 2.8).

Relevancia en biomedicina. As comunidades microbianas xogan un papel importante na *microbiota intestinal*, cuxa estabilidade é crucial para a saúde humana. Neste contexto, o modelado matemático xa posibilitou descubrimentos contraintuitivos, como que a estabilidade se beneficia polo predominio de interaccións competitivas en lugar de cooperativas. Existe un potencial inexplorado na aplicación de modelos dinámicos en problemas como o deseño de intervencións óptimas (composición e administración de subministracións nutricionais) ao longo do tempo, para restablecer o funcionamento normal da microbiota minimizando os efectos nocivos.

Unha aplicación emerxente é o deseño de cócteis de fagos. Os virus bacteriófagos, ou simplemente fagos, son un tipo particular de virus que infecta bacterias. Son moi abundantes: o seu número é superior ao de calquera outro microorganismo existente na terra. Pola súa capacidade de infectar e matar bacterias con alta especificidade, os fagos son unha alternativa ao uso de antibióticos.

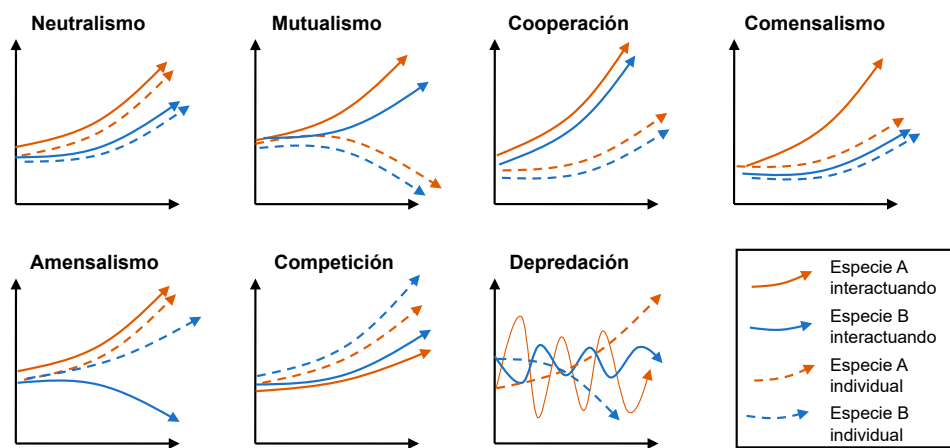


Figura 2.8: Algúns tipos de interaccións presentes nas comunidades microbianas [45]. As liñas a trazos mostran a evolución temporal de cada microorganismo en solitario; as liñas continuas a evolución en presenza de outros microorganismos.

Modelos de dinámica de poboacións. As dinámicas das comunidades de microorganismos poden describirse con modelos de poboacións en EDOs. Podemos clasificalos en función de se consideran as interaccións directas entre diferentes especies, ou se modelan ditas interaccións como competicións polos mesmos recursos.

- Modelos de especie–especie. Un exemplo clásico de modelos deste tipo son os de Lotka-Volterra; inicialmente introducidos en ecoloxía como modelos de presa-predador, representan dinámicas competitivas que

poden aparecer en aplicacións biomédicas. O *modelo básico de Lotka-Volterra* representa a interacción entre 2 especies, unha presa (x_1) e un depredador (x_2):

$$\begin{aligned}\dot{x}_1 &= a \cdot x_1 - b \cdot x_1 \cdot x_2, \\ \dot{x}_2 &= c \cdot x_1 \cdot x_2 - d \cdot x_2\end{aligned}\tag{2.13}$$

Un comportamento típico móstrase na Fig. 2.9.

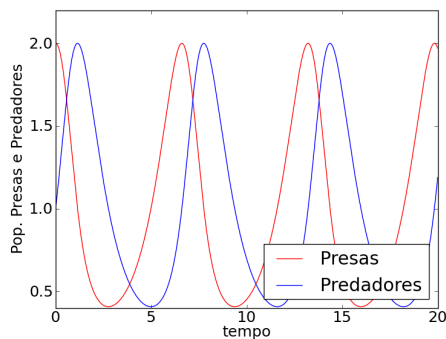


Figura 2.9: Comportamento dun modelo de Lotka-Volterra básico (2.13), con $a = b = c = d = 1$ (por Rsoaresp - Obra do propio, CC BY-SA 3.0, <https://commons.wikimedia.org/w/index.php?curid=18088284>).

Se consideramos n especies ($1, \dots, i, \dots, n$) que poden interactuar dúas a dúas (é dicir, sen considerar interaccións nas que interveñan 3 ou máis especies) obtemos o *modelo de Lotka-Volterra xeneralizado* (que ás veces se denota por “gL” polas súas siglas en inglés):

$$\dot{x}_i = x_i \left(a_i + \sum_{j=1}^n b_{ij} \cdot x_j \right)\tag{2.14}$$

onde os a_i son os parámetros de crecemento, a_i/b_{ii} é a capacidade de carga – é dicir, a máxima poboación da especie i que o medio ambiente pode soportar – e os parámetros $b_{ij}, i \neq j$ cuantifican as interaccións entre unha especie e todas as demais. É posíbel xeneralizar aínda máis

este modelo, considerando *interacciones non-lineares*:

$$\dot{x}_i = f_i(x_i) + \sum_{j=1}^n h_{ij}(x_i, x_j) \quad (2.15)$$

- Modelos de competición por recursos. Outros modelos describen a interacción entre especies en base á competición por recursos, non pola predación. Existen argumentos a favor de que este tipo de modelos reproduce de forma máis fiel a realidade [9]. Un modelo *SMI* (“species-metabolite interaction”) xeral sería:

$$\begin{aligned} \dot{x}_i &= x_i \sum_{j=1}^m f_j^i(x_i, y_j), \\ \dot{y}_j &= \sum_{i=1}^n \sum_{l=1}^m h_{il}^j(x_i, y_l) \end{aligned} \quad (2.16)$$

onde x_i é a masa de cada un dos n microorganismos, e y_j a concentración de cada un dos m metabolitos (ou, en xeral, a abundancia de cada recurso). As funcións de interacción $f()$ e $h()$ poden ser de tipo *cuadrático*:

$$\begin{aligned} \dot{x}_i &= x_i \cdot \left(\sum_{j=1}^m \psi_{ij} \cdot y_j - d_i \right), \\ \dot{y}_j &= y_j \cdot \left(- \sum_{i=1}^n k_{ij} \cdot x_i - d_j^* \right) + g_j + \sum_{i=1}^n \sum_{l=1}^m \phi_{il}^j \cdot x_i \cdot y_l, \end{aligned} \quad (2.17)$$

ou describir un crecemento tipo *Monod*:

$$\begin{aligned} \dot{x}_i &= x_i \cdot \left(\sum_{j=1}^m \frac{V_{ij} \cdot y_j}{K_{ij} + y_j \cdot x_i} - d_i \right), \\ \dot{y}_j &= y_j \cdot \left(- \sum_{i=1}^n \frac{V_{ij}^* \cdot x_i}{K_{ij}^* + y_j \cdot x_i} - d_j^* \right) + g_j + \sum_{i=1}^n \sum_{l=1}^m \frac{\phi_{il}^j \cdot x_i \cdot y_l}{K_{il}^* + x_i \cdot y_l}, \end{aligned} \quad (2.18)$$

En ambos casos hai termos de dilución, d , e de fluxo de entrada constante, g , así como outras constantes cinéticas.

Características principais dos modelos de comunidades microbianas:

- Relevantes en: análise da microbiota intestinal, deseño de suplementos nutricionais, tratamentos antibacterianos baseados en fagos.

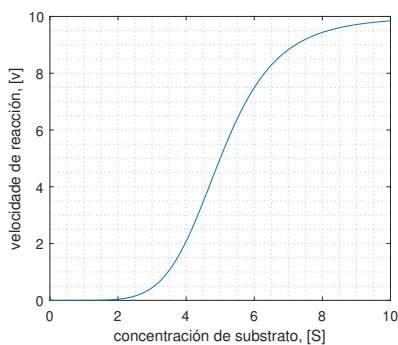
- Escala temporal: depende do sistema: poden ser horas/días para comunidades microbianas na microbiota, ou meses para interaccións entre outras especies.
- Formalismos adecuados: tipicamente modelos en EDOs.
- Tamaños dos modelos: variábel, tipicamente uns poucos estados (< 10).
- Tipos de datos relevantes: abundancia (relativa ou absoluta) das diferentes especies. O tipo de medida depende da aplicación (concentracións, número de individuos, etc).
- Exemplos: véxase por exemplo o [9].

Cuestións e problemas do Tema 2

C2.1. Supoñamos que unha reacción se produce con cinética Michaelis-Menten. Cal é a velocidade de reacción cando a concentración do substrato é a metade da constante de Michaelis?

C2.2. Nunha reacción que se produce con cinética Michaelis-Menten, cal é a velocidade de reacción cando a concentración do substrato tende a infinito?

C2.3. A partir da seguinte gráfica, determinar se un enzima ten cooperatividade, e que valor aproximado pode ter o seu coeficiente de Hill:



C2.4. Definicións de modelo farmacocinético e modelo farmacodinámico.

C2.5. Que formalismo de modelado se usa normalmente en epidemioloxía? Dentro del, cales son as dúas grandes familias de modelos epidemiolóxicos?

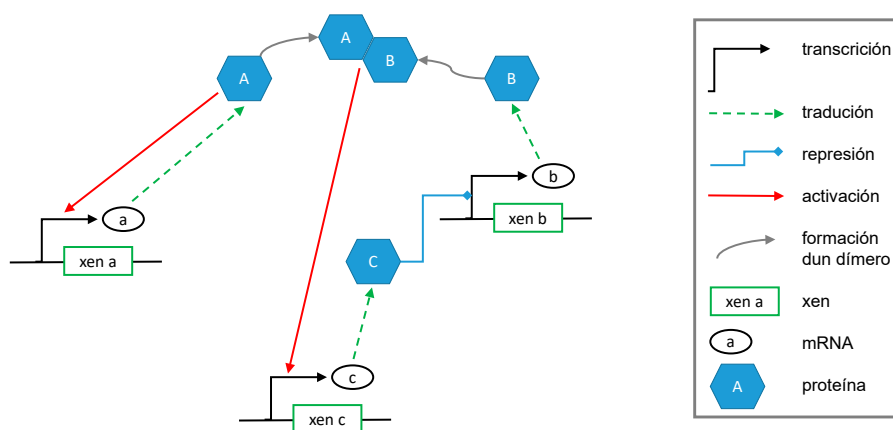
C2.6. Ecuacións dun modelo SIR. Describir os seus estados e parámetros.

C2.7. Ecuacións dun modelo SEIR. Describir os seus estados e parámetros.

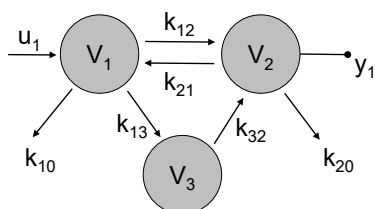
C2.8. Tipos de interaccións posibles en comunidades microbianas.

C2.9. Escribir as ecuacións de Lotka-Volterra. Que tipo de sistemas biolóxicos poden modelarse con elas?

C2.10. Escribir un sistema de EDOs adecuado para modelar o sistema de expresión xénica representado no seguinte diagrama. Describir as activacións e represións con cinéticas tipo Hill, excepto a auto-activación do xen 'a' que se suporá cunha taxa constante v_a . Incluír un termo de degradación para cada mRNA.



C2.11. Escribir un sistema de EDOs adecuado para o modelo compartimental representado no seguinte diagrama:



3

Simulación

“Everything should be made as simple as possible, but no simpler.”
– Albert Einstein

No tema anterior aprendemos a representar un sistema de interese biomédico mediante ecuacións diferenciais. Para simular o seu comportamento é preciso integrar ditas ecuacións. Nesta lección aprenderemos como levar a cabo esta integración de forma precisa e computacionalmente eficiente.

Comenzaremos por ver a integración de sistemas de EDOs lineares, que se pode facer de forma analítica, para o que resulta útil a transformada de Laplace. Despois introduciremos a base teórica dos métodos de integración numérica “clásicos”. Mencionaremos tamén o algoritmo de Gillespie para integrar ecuacións estocásticas. A continuación veremos entornos de programación nos que se poden implementar os métodos anteriores; para este fin pódense empregar linguaxes de programación de propósito máis xeral, ou ben ferramentas software especializadas en modelos biolóxicos. Por último veremos outros aspectos relacionados coas implementacións computacionais e a reproducibilidade: formatos, bases de datos, repositorios, etc.

Neste capítulo suporemos que todos os parámetros das ecuacións son coñecidos. En realidade, o máis habitual é ter que estimar os valores dalgúns

deles previamente á simulación. A estimación de parámetros verase no capítulo 4.

3.1. Integración de EDOs lineares

Lembremos que os sistemas lineares se poden representar en variábeis de estado como:

$$\mathcal{M} = \begin{cases} \dot{x} = A \cdot x + B \cdot u \\ y = C \cdot x + D \cdot u \end{cases} \quad (3.1)$$

$$(3.2)$$

3.1.1. Transformada de Laplace.

Para resolver algúns problemas en enxeñaría recórrese á substitución de funcións dunha variábel real (xeralmente o tempo ou a distancia) por funcións dunha variábel complexa que depende da frecuencia. (Un exemplo é o uso da transformada de Fourier para resolver problemas de enxeñaría eléctrica, como o cálculo da corrente nun punto dunha rede.) Na análise de sistemas dinámicos lineares en tempo continuo úsase a *transformada de Laplace*, que se define como:

$$\mathcal{L}[f(t)] = F(s) = \int_{0+}^{\infty} e^{s \cdot t} \cdot f(t) \cdot dt, \quad (3.3)$$

onde $s = \sigma + j\omega$; σ e ω son números reais e $j = \sqrt{-1}$.

A función orixinal $f(t)$ pode obterse de $F(s)$ mediante a *transformada inversa de Laplace*:

$$f(t) = \frac{1}{2\pi j} \int_{\sigma-j\infty}^{\sigma+j\infty} F(s) \cdot e^{s \cdot t} \cdot ds. \quad (3.4)$$

Transformada de Laplace de funcións típicas. Nas táboas 3.1 e 3.2 proporciónanse as transformadas de Laplace de funcións típicas.

Propiedades da transformada de Laplace. A táboa 3.3 presenta as principais propiedades da transformada de Laplace.

$F(s) = \mathcal{L}\{f(t)\}$	$f(t) = \mathcal{L}^{-1}\{F(s)\}$
$\frac{1}{s}$	1
$\frac{1}{s^2}$	t
$\frac{1}{s^n}, \quad (n = 1, 2, 3, \dots)$	$\frac{t^{n-1}}{(n-1)!}$
$\frac{1}{\sqrt{s}},$	$\frac{1}{\sqrt{\pi t}}$
$\frac{1}{s^{\frac{3}{2}}},$	$2\sqrt{\frac{t}{\pi}}$
$\frac{1}{s^k}, \quad (k > 0)$	$\frac{t^{k-1}}{\Gamma(k)}$
$\frac{1}{s-a}$	e^{at}
$\frac{1}{(s-a)^2}$	te^{at}
$\frac{1}{(s-a)^n}, \quad (n = 1, 2, 3, \dots)$	$\frac{1}{(n-1)!} t^{n-1} e^{at}$
$\frac{1}{(s-a)^k}, \quad (k > 0)$	$\frac{1}{\Gamma(k)} t^{k-1} e^{at}$
$\frac{1}{(s-a)(s-b)}, \quad (a \neq b)$	$\frac{1}{a-b} (e^{at} - e^{bt})$
$\frac{s}{(s-a)(s-b)}, \quad (a \neq b)$	$\frac{1}{a-b} (ae^{at} - be^{bt})$
$\frac{1}{s^2 + w^2}$	$\frac{1}{w} \text{sen}(wt)$
$\frac{s}{s^2 + w^2}$	$\cos(wt)$
$\frac{1}{s^2 - a^2}$	$\frac{1}{a} \text{senh}(at)$
$\frac{s}{s^2 - a^2}$	$\cosh(at)$
$\frac{1}{(s-a)^2 + w^2}$	$\frac{1}{w} e^{at} \text{sen}(wt)$
$\frac{s-a}{(s-a)^2 + w^2}$	$e^{at} \cos(wt)$
$\frac{1}{s(s^2 + w^2)}$	$\frac{1}{w^2} (1 - \cos(wt))$
$\frac{1}{s^2(s^2 + w^2)}$	$\frac{1}{w^3} (wt - \text{sen}(wt))$
$\frac{1}{(s^2 + w^2)^2}$	$\frac{1}{2w^3} (\text{sen}(wt) - wt \cos(wt))$
$\frac{s}{(s^2 + w^2)^2}$	$\frac{t}{2w} \text{sen}(wt)$

Cadro 3.1: Táboa de transformadas de Laplace - parte 1

$F(s) = \mathcal{L}\{f(t)\}$	$f(t) = \mathcal{L}^{-1}\{F(s)\}$
$\frac{s^2}{(s^2 + w^2)^2}$	$\frac{1}{2w}(\text{sen}(wt) + wt \cos(wt))$
$\frac{s}{(s^2 + a^2)(s^2 + b^2)}, \quad (a^2 \neq b^2)$	$\frac{1}{b^2 - a^2}(\cos(at) - \cos(bt))$
$\frac{1}{(s^4 + 4a^4)}$	$\frac{1}{4a^3}(\text{sen}(at) \cosh(at) - \cos(at) \sinh(at))$
$\frac{s}{(s^4 + 4a^4)}$	$\frac{1}{2a^2} \text{sen}(at) \sinh(at)$
$\frac{1}{(s^4 - a^4)}$	$\frac{1}{2a^3}(\sinh(at) - \text{sen}(at))$
$\frac{s}{(s^4 - a^4)}$	$\frac{1}{2a^2}(\cosh(at) - \cos(at))$
$\sqrt{s-a} - \sqrt{s-b}$	$\frac{1}{2\sqrt{\pi t^3}}(e^{bt} - e^{at})$
$\frac{1}{\sqrt{s+a}\sqrt{s+b}}$	$e^{\frac{-(a+b)t}{2}} I_0\left(\frac{a-b}{2}t\right)$
$\frac{1}{\sqrt{s^2 + a^2}}$	$J_0(at)$
$\frac{s}{(s-a)^{\frac{3}{2}}}$	$\frac{1}{\sqrt{\pi t}} e^{at}(1 + 2at)$
$\frac{1}{(s^2 - a^2)^k}, \quad (k > 0)$	$\frac{\sqrt{\pi}}{\Gamma(k)} \left(\frac{t}{2a}\right)^{k-\frac{1}{2}} I_{k-\frac{1}{2}}(at)$
$\frac{1}{s} e^{-\frac{k}{s}}, \quad (k > 0)$	$J_0(2\sqrt{kt})$
$\frac{1}{\sqrt{s}} e^{-\frac{k}{s}}$	$\frac{1}{\sqrt{\pi t}} \cos(2\sqrt{kt})$
$\frac{1}{s^{\frac{3}{2}}} e^{-\frac{k}{s}}$	$\frac{1}{\sqrt{\pi t}} \sinh(2\sqrt{kt})$
$e^{-k\sqrt{s}}, \quad (k > 0)$	$\frac{k}{2\sqrt{\pi t^3}} e^{-\frac{k^2}{4t}}$
$\frac{1}{s} \ln(s)$	$-\ln(t) - \gamma, \quad (\gamma \approx 0,5772)$
$\ln\left(\frac{s-a}{s-b}\right)$	$\frac{1}{t}(e^{bt} - e^{at})$
$\ln\left(\frac{s^2 + w^2}{s^2}\right)$	$\frac{2}{t}(1 - \cos(wt))$
$\ln\left(\frac{s^2 - a^2}{s^2}\right)$	$\frac{2}{t}(1 - \cosh(at))$
$\tan^{-1}\left(\frac{w}{s}\right)$	$\frac{1}{t} \text{sen}(wt)$
$\frac{1}{s} \cot^{-1}(s)$	$\text{Si}(t)$

Cadro 3.2: Táboa de transformadas de Laplace - parte 2

Linearidade	$\mathcal{L}\{\alpha f(t) + \beta g(t)\} = \alpha \mathcal{L}\{f(t)\} + \beta \mathcal{L}\{g(t)\}$
Transformada da derivada	$\mathcal{L}\{f'(t)\} = s\mathcal{L}\{f(t)\} - f(0)$ $\mathcal{L}\{f''(t)\} = s^2\mathcal{L}\{f(t)\} - sf(0) - f'(0)$
Deslocamento no eixo s	$\mathcal{L}\{e^{at}f(t)\} = F(s-a)$
Deslocamento no eixo t	$\mathcal{L}\{u(t-a)f(t-a)\} = e^{-as}F(s)$ $\mathcal{L}\{u(t-a)\} = \frac{e^{-as}}{s}$
Transformada da integral	$\mathcal{L}\left\{\int_0^t f(\tau)d\tau\right\} = \frac{F(s)}{s}$
Transformada da Delta de Dirac	$\mathcal{L}\{\delta(t-a)\} = e^{-as}$
Teorema da Convolución	$\mathcal{L}\{(f * g)(t)\} = F(s)G(s),$ onde $(f * g)(t) = \int_0^t f(\tau)g(t-\tau)d\tau$
Transformada de funcións periódicas	$\mathcal{L}\{f(t)\} = \frac{1}{1-e^{-sT}} \int_0^T e^{-s\tau} f(\tau)d\tau$
Derivada da transformada	$\mathcal{L}\{tf(t)\} = -\frac{dF(s)}{ds}$
Integral da transformada	$\mathcal{L}\left\{\frac{f(t)}{t}\right\} = \int_s^\infty F(\hat{s})d\hat{s}$

Cadro 3.3: Propiedades da transformada de Laplace

Utilidade da transformada de Laplace. A transformada de Laplace pódese usar para resolver ecuacións diferenciais lineares con coeficientes constantes. Para iso, séguense estes pasos:

1. Facer a transformada de Laplace das funcións temporais presentes nas ecuacións de $\dot{x}(t)$ (consultar táboas) $\rightarrow$ obtemos ecuacións en función de s .
2. Despexar as incógnitas, e descompoñer as expresións que son función de s en fraccións simples.
3. Facer a transformada inversa das funcións resultantes (consultar táboas) $\rightarrow$ obtemos expresións dos estados, $x(t)$ (en lugar das súas derivadas como antes), en función de t .

Con este procedemento podemos integrar as EDOs dun modelo linear, obtendo así a expresión analítica de $x(t)$.

Exemplo 1: solución das ecuacións diferenciais dun modelo. Na Figura 4.3 do Tema 4 móstrase o modelo dun proceso de tradución de mRNA a unha proteína. O modelo consta de dous estados e as súas ecuacións son:

$$\mathcal{M} = \begin{cases} m\dot{RNA}(t) = -\gamma \cdot mRNA(t); & mRNA(0) = mRNA_0, \\ \dot{GFP}(t) = -\delta \cdot GFP(t) + k \cdot mRNA(t); & GFP(0) = 0. \end{cases} \quad (3.5)$$

A saída do modelo é $y(t) = s_1 \cdot GFP(t)$, onde modificamos a notación orixinal, substituíndo o parámetro “ s ” por “ s_1 ” para evitar confusión coa variábel frecuencial “ s ”. Para calcular a evolución temporal da saída, $y(t)$, temos que resolver as ecuacións diferenciais, obtendo a expresión de $GFP(t)$. Para iso, aplicamos a transformada de Laplace do xeito indicado no punto anterior:

1. Facemos as transformadas de Laplace das ecuacións diferenciais:

$$s \cdot mRNA(s) - mRNA_0 = -\gamma \cdot mRNA(s)$$

$$s \cdot GFP(s) = -\delta \cdot GFP(s) + k \cdot mRNA(s)$$

2. Despexamos $GFP(s)$ da segunda ecuación:

$$GFP(s) = \frac{k \cdot mRNA(s)}{s + \delta}$$

E substituímos nela expresión de $mRNA(s)$ que obtemos da primeira ecuación, é dicir, $mRNA(s) = \frac{mRNA_0}{s + \gamma}$, o que nos dá:

$$GFP(s) = k \cdot mRNA_0 \cdot \frac{1}{(s + \delta)(s + \gamma)}$$

3. Facemos a transformada inversa de Laplace, consultando as fórmulas nas táboas, e obtemos:

$$GFP(t) = k \cdot mRNA_0 \cdot \frac{e^{-\delta \cdot t} - e^{-\gamma \cdot t}}{\gamma - \delta}$$

Por tanto:

$$y(t) = s_1 \cdot GFP(t) = s_1 \cdot k \cdot mRNA_0 \cdot \frac{e^{-\delta \cdot t} - e^{-\gamma \cdot t}}{\gamma - \delta}$$

Exemplo 2: solución con descomposición en fracciones simples. Dada a seguinte ecuación:

$$\frac{d^2y}{dt^2} + 2\frac{dy}{dt} + y = \frac{du}{dt} - 0.5 \cdot u; \quad y(0) = 0; \quad \frac{dy}{dt}(0) = 0; \quad u(t) = e^{-2t} \text{ para } t \geq 0.$$

Tomando transformadas de Laplace, temos:

$$s^2Y(s) + 2sY(s) + Y(s) = sU(s) - 0.5U(s) \Rightarrow Y(s) = \frac{s - 0.5}{s^2 + 2s + 1}U(s),$$

Por outra banda, a transformada de Laplace da entrada é:

$$U(s) = \mathcal{L}[u(t)] = \mathcal{L}[e^{-2t}] = \frac{1}{s + 2},$$

e por tanto:

$$Y(s) = \frac{s - 0.5}{s^2 + 2s + 1} \frac{1}{s + 2}.$$

Para facer a transformada inversa de Laplace desta función, hai que descompoñela en fraccións simples:

$$\begin{aligned} \frac{s - 0.5}{s^2 + 2s + 1} \frac{1}{s + 2} &= \frac{s - 0.5}{(s + 1)^2} \frac{1}{s + 2} = \frac{a}{s + 2} + \frac{b}{s + 1} + \frac{c}{(s + 1)^2} \\ \frac{s - 0.5}{(s + 1)^2} \frac{1}{s + 2} &= \frac{a(s + 1)^2}{(s + 1)^2(s + 2)} + \frac{b(s + 1)(s + 2)}{(s + 1)^2(s + 2)} + \frac{c(s + 2)}{(s + 1)^2(s + 2)} \end{aligned}$$

Para $s = -1 \Rightarrow c = -1.5$

Para $s = -2 \Rightarrow a = -2.5$

Para $s = 0 \Rightarrow -0.5 = a + 2b + 2c = -5.5 + 2b \Rightarrow b = 2.5.$

$$y(t) = \mathcal{L}^{-1} \left[\frac{-2.5}{s + 2} + \frac{2.5}{s + 1} + \frac{-1.5}{(s + 1)^2} \right] = -2.5e^{-2t} + 2.5e^{-t} - 1.5te^{-t}$$

3.1.2. Función de transferencia (FT)

A función de transferencia (FT) dun sistema linear é o cociente entre as transformadas de Laplace da súa saída e da súa entrada, supoñendo condicións iniciais nulas. As FT son unha ferramenta fundamental da enxeñaría de control; úsanse para a análise de sistemas dinámicos e para o deseño de controladores.

Se lle chamamos $H(s)$ á FT dun sistema linear, a súa saída pódese escribir como $Y(s) = H(s) \cdot U(s)$ (+ os termos correspondentes ás condicións iniciais, se son non nulas). Como a transformada de Laplace dun impulso unitario é $U(s) = 1$, $H(s)$ representa a transformada de Laplace da resposta do sistema a unha entrada impulso unitario. Ademais, a FT pódese usar para obter a resposta a outras entradas, multiplicando $H(s)$ pola transformada de Laplace de ditas entradas $U(s)$.

Poñamos a FT en función do seu numerador e denominador: $H(s) = \frac{N(s)}{D(s)}$.

- O denominador $D(s)$ dunha FT é o *polinomio característico*. A ecuación característica é $D(s) = 0$.

- Ás raíces do denominador (solucións da ecuación característica, $D(s) = 0$) chámaselles *polos*.
- Ás raíces do numerador (solucións de $N(s) = 0$) chámaselles *ceros* (xa que anulan a FT).

Exemplo de función de transferencia: modelo farmacocinético. A partir das ecuacións do modelo compartimental farmacocinético da Fig. 1.5, obter a súa función de transferencia de $u(t)$ a $y(t)$.

A saída do modelo é $y(t) = x_2(t)$, por tanto temos que calcular a FT de $U(s)$ a $X_2(s)$.

A transformada de Laplace da ecuación (1.5) é:

$$\begin{aligned} s \cdot X_1(s) &= -(k_{12} + k_{10}) \cdot X_1(s) + k_{21} \cdot X_2(s) + k_{1u} \cdot U(s) \\ \Rightarrow (s + k_{12} + k_{10}) \cdot X_1(s) &= k_{21} \cdot X_2(s) + k_{1u} \cdot U(s) \end{aligned}$$

A transformada de Laplace da ecuación (1.6) é:

$$s \cdot X_2(s) = k_{12} \cdot X_1(s) - k_{21} \cdot X_2(s) \Rightarrow (s + k_{21}) \cdot X_2(s) = k_{12} \cdot X_1(s)$$

Na 2ª ecuación podemos despexar $X_1(s) = \frac{s+k_{21}}{k_{12}} X_2(s)$, e substituíndo esta expresión na 1ª ecuación obtemos:

$$\frac{(s + k_{12} + k_{10}) \cdot (s + k_{21})}{k_{12}} X_2(s) = k_{21} \cdot X_2(s) + k_{1u} \cdot U(s)$$

Reordenando:

$$\frac{s^2 + (k_{21} + k_{12} + k_{10}) \cdot s + \cancel{k_{21} \cdot k_{12}} + k_{10} \cdot k_{21} - \cancel{k_{21} \cdot k_{12}}}{k_{12}} X_2(s) = k_{1u} \cdot U(s)$$

Despexando $X_2(s)$ nesta última ecuación, chegamos a:

$$X_2(s) = \frac{k_{12} \cdot k_{1u}}{s^2 + (k_{21} + k_{12} + k_{10}) \cdot s + k_{10} \cdot k_{21}} \cdot U(s) \quad (3.7)$$

Cando $y(t) = x_2(t)$, a FT de $u(t)$ a $y(t)$ – isto é, o cociente $Y(s)/U(s) = H(s)$ – é a expresión resaltada en vermello.

Paso de VE a FT e viceversa. Un sistema en variábeis de estado (VE), dado por

$$\mathcal{M} = \begin{cases} \dot{x} = A \cdot x + B \cdot u \\ y = C \cdot x + D \cdot u, \end{cases} \quad (3.8)$$

$$(3.9)$$

ten unha única FT dada por

$$H(s) = C \cdot (s \cdot I - A)^{-1} \cdot B + D.$$

A correspondencia inversa non é única, é dicir, para unha FT hai infinitas representacións en VE posibles. Loxicamente, o significado físico das variábeis de estado variará dunha representación a outra. Porén, a relación entre a entrada e a saída do sistema (a súa resposta dinámica) é a mesma, de aí que a FT sexa única.

Paso de VE a FT do modelo farmacocinético. O modelo da Fig. 1.5 está descrito polas ecuacións (1.5)–(1.6), que se poden poñer en VE como (1.8). Calculamos a súa FT mediante a fórmula:

$$H(s) = C \cdot (s \cdot I - A)^{-1} \cdot B + D$$

Substituíndo as matrices A , B , C e D dadas na ecuación (1.8):

$$\begin{aligned} H(s) &= [0 \quad 1] \cdot \left(s \cdot \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} -(k_{12} + k_{10}) & k_{21} \\ k_{12} & -k_{21} \end{bmatrix} \right)^{-1} \cdot \begin{bmatrix} k_{1u} \\ 0 \end{bmatrix} + 0 = \\ &= [0 \quad 1] \cdot \left(\begin{bmatrix} s + (k_{12} + k_{10}) & -k_{21} \\ -k_{12} & s + k_{21} \end{bmatrix} \right)^{-1} \cdot \begin{bmatrix} k_{1u} \\ 0 \end{bmatrix} = \\ &= \frac{k_{12} \cdot k_{1u}}{s^2 + (k_{21} + k_{12} + k_{10}) \cdot s + k_{10} \cdot k_{21}}. \end{aligned}$$

Comprobamos que a FT obtida desta forma coincide coa que calculáramos a partir das transformadas de Laplace na ecuación (3.7).

3.2. Integración de EDOs nonlineares

Para o caso xeral de modelos nonlineares non é posíbel obter a solución analítica das ecuacións, tendo que recorrer á integración numérica. Lembremos que un sistema nonlinear descrito por EDOs ten a forma:

$$\mathcal{M} = \begin{cases} \dot{x} = f(x, \theta, u) \\ y = h(x, \theta, u) \end{cases} \quad (3.10)$$

$$(3.11)$$

Neste capítulo non modelamos de forma explícita a existencia de entradas descoñecidas, polo que, a diferenza de (1.1), o modelo non inclúe w .

Queremos integrar $\dot{x}(t) = f(x(t), t)$ para obter os valores de $x(t)$ nun intervalo temporal $t_0 \leq t \leq t_f$, partindo dos valores das variábeis de estado no instante inicial, $x(t_0)$. Este é un problema de valor inicial. Resólvese iterativamente, calculando en cada instante t os valores no seguinte instante temporal, isto é, $x(t + \Delta t)$. Podemos distinguir dúas grandes clases de métodos, dependendo de se usan un paso Δt fixo ou variábel ao longo da integración.

3.2.1. Métodos de paso fixo

Método de series de Taylor. A forma clásica de realizar a integración numérica baséase na expansión da expresión de $x(t)$ en series de Taylor. Para iso, o valor de $x(t)$ nun instante temporal distanciado Δt obtense como:

$$x(t + \Delta t) = \sum_{k=0}^{\infty} \frac{(\Delta t)^k}{k!} \left. \frac{d^k x}{dt^k} \right|_t \quad (3.12)$$

$$= x(t) + \left. \frac{dx}{dt} \right|_t \Delta t + \frac{1}{2!} \left. \frac{d^2 x}{dt^2} \right|_t \Delta t^2 + \frac{1}{3!} \left. \frac{d^3 x}{dt^3} \right|_t \Delta t^3 \dots \quad (3.13)$$

$$= x(t) + \left. \frac{dx}{dt} \right|_t \Delta t + \frac{1}{2!} \left. \frac{d^2 x}{dt^2} \right|_t \Delta t^2 + O(\Delta t^3) \quad (3.14)$$

onde $O(\Delta t^3)$ indica que os termos restantes son no peor dos casos de orde Δt^3 . Este termo, $O(\Delta t^3)$, representa o *erro de truncado*. Para valores pequenos de Δt , os termos de orde superior serán desprezábais. (Nótese que x depende de t , mais non sempre o explicitamos para simplificar a notación.)

O valor de $\frac{dx}{dt}$ é coñecido, xa que por definición $\frac{dx}{dt} = f(x, t)$. Por tanto, podemos calcular o primeiro paso na solución de $\dot{x}(t) = f(x(t), t)$ como

$$x(t + \Delta t) = x(t) + f(x(t), t)\Delta t + \frac{\dot{f}(x(t), t)}{2!}\Delta t^2 + O(\Delta t^3) \quad (3.15)$$

A solución da ecuación diferencial orixinal pode calcularse de forma recursiva. É dicir, dado un instante temporal t_m , a fórmula anterior pode aplicarse para obter o valor en t_{m+1} , e así sucesivamente. Escribindo, para simplificar, $x(t_m) \equiv x^m$, $x(t_{m+1}) \equiv x^{m+1}$, e $f(x(t_m), t_m) \equiv f^m$, a ecuación anterior queda:

$$x^{m+1} = x^m + f^m \Delta t + \frac{\dot{f}^m}{2!}\Delta t^2 + O(\Delta t^3) \quad (3.16)$$

Para poder calcular (3.16) hai que calcular a derivada de $\dot{f}$ a tempo t_m . Pola regra da cadea,

$$\dot{f}^m = \left[\frac{\partial f}{\partial x} \frac{dx}{dt} + \frac{\partial f}{\partial t} \frac{dt}{dt} \right] \Big|_{t_m} = \frac{\partial f^m}{\partial x} f^m + \frac{\partial f^m}{\partial t} \quad (3.17)$$

Por tanto, a seguinte ecuación proporciona unha *solución algorítmica recursiva* para a integración de $\dot{x}$:

$$x^{m+1} = x^m + \Delta t \left[f^m + \frac{\Delta t}{2} \left(\frac{\partial f^m}{\partial x} f^m + \frac{\partial f^m}{\partial t} \right) \right] + O(\Delta t^3) \quad (3.18)$$

A avaliación de (3.18) require calcular derivadas parciais de f . Poden obterse de forma analítica, mediante cálculo simbólico (dispoñíbel en programas como Matlab, Mathematica, Maple, etc), ou por aproximacións numéricas, como se describe a continuación.

Cálculo numérico de derivadas. Aproximación numérica polo método de diferenzas finitas *cara atrás* (“backwards-difference”, BD):

$$\dot{f}^m = \frac{f(t_m) - f(t_{m-1})}{t_m - t_{m-1}} + E_{BD} \approx \frac{f(t_m) - f(t_{m-1})}{t_m - t_{m-1}} \quad (3.19)$$

Este método só require coñecer un valor anterior. Outra fórmula, a da *diferenza central* (“central-difference”, CD), require de dous valores anteriores:

$$\dot{f}^m = \frac{f(t_{m+1}) - f(t_{m-1}))}{t_{m+1} - t_{m-1}} + E_{DC} \approx \frac{f(t_{m+1}) - f(t_{m-1}))}{t_{m+1} - t_{m-1}} \quad (3.20)$$

Ambas fórmulas incorren en erros de truncado, E_{BD} e E_{CD} .

Coa expresión de BD, podemos calcular:

$$\frac{\partial f^m}{\partial x} \approx \frac{f(t_m) - f(t_{m-1}))}{x_m - x_{m-1}}, \quad (3.21)$$

$$\frac{\partial f^m}{\partial t} \approx \frac{f(t_m) - f(t_{m-1}))}{t_m - t_{m-1}} \quad (3.22)$$

Co que a **solución de Taylor** queda:

$$x^{m+1} = x^m + \Delta t \left[f^m + \frac{\Delta t}{2} \left(\frac{f(t_m) - f(t_{m-1}))}{x_m - x_{m-1}} f^m + \frac{f(t_m) - f(t_{m-1}))}{t_m - t_{m-1}} \right) \right] + O(\Delta t^3) + E_{BD} \quad (3.23)$$

A fórmula (3.23) pódese usar para resolver iterativamente o problema de valor inicial, pero ten un inconveniente. Para o cálculo do valor no instante t_{m+1} precisa coñecer os *dous* valores anteriores de x , en t_m e t_{m-1} . Por tanto, para calcular o primeiro punto $x(1)$ precisaríamos $x(0)$ e $x(-1)$, e este último non está definido. Se tentásemos evitar o problema calculando directamente $x(2)$, precisaríamos coñecer $x(1)$ e $x(0)$, mais $x(1)$ non estaría dispoñíbel xa que non foi calculado previamente. Dise que (3.23) non é auto-iniciábel.

A seguir descríbense dous métodos auto-iniciábeis para obter solucións aproximadas de sistemas de EDOs: o de Euler e os de Runge-Kutta.

Método de Euler. É o algoritmo auto-iniciábel máis simple. Aproximación a ecuación (3.23) como:

$$x^{m+1} \approx x^m + f^m \Delta t \quad (3.24)$$

Como só precisa do valor das variábeis no instante anterior ao que se calcula, é auto-iniciábel. Por tanto, pode usarse para integrar unha EDO a partir das condicións iniciais. Pero pode dar lugar a erros de truncado grandes, xa que aproxima unha curva pola súa pendente. Para diminuír eses erros podería pensarse en empregar un Δt moi pequeno, mais se se escolle pequeno demais

aumentan os erros de redondeo. Por iso é preferíbel utilizar outros métodos, como os de Runge-Kutta.

Erros numéricos e estabilidade.

Na resolución numérica das EDOs cométense erros de dous tipos:

- *Erro de truncado*, E_T : son erros deste tipo o do truncado da serie de Taylor (3.14), e o da aproximación da derivada (3.19 ou 3.20). Na ecuación (3.23) vese como ambos erros se suman.
- *Erro de redondeo*, E_R : son causados pola necesidade de representar os números nun computador cunha lonxitude de palabra finita (por exemplo, ao representar $1/3 = 0.3333 \dots$ cun número finito de díxitos).

O *erro relativo total* a tempo t é:

$$E(t) = \frac{\text{erro computacional total}}{\text{valor real}} = \frac{E_T + E_R}{x^*(t)} \quad (3.25)$$

A integración dun sistema de EDOs é *numericamente estábel* se $E(t)$ permanece acoutado a medida que progresa a simulación, isto é, para $t \rightarrow \infty$. Aínda que un sistema teña un comportamento dinámico estábel, a súa integración pode ser numericamente inestábel. Os problemas de inestabilidade numérica poden aparecer especialmente en sistemas “stiff”, descritos na sección 3.2.2.

Métodos de Runge-Kutta (R-K). Os métodos R-K son métodos auto-iniciábeis máis sofisticados que o de Euler, e dan o mesmo resultado que o método de Taylor da mesma orde. Fan unha expansión en series de Taylor de f en dúas variábeis, x e t , en lugar de só en t . Isto permite evitar avaliar derivadas de f , xa que as derivadas de f que aparecen na expansión son substituídas por valores intermedios de x entre (x^m, t_m) e (x^{m+1}, t_{m+1}) .

O método de R-K máis popular é o de cuarta orde. Fai a seguinte aproxi-

mación:

$$x^{m+1} \approx x^m + \frac{\Delta t}{6}(k_1 + 2k_2 + 2k_3 + k_4) \quad (3.26)$$

$$k_1 = f(x^m, t_m) \quad (3.27)$$

$$k_2 = f\left(x^m + \frac{\Delta t}{2}k_1, t_m + \frac{\Delta t}{2}\right) \quad (3.28)$$

$$k_3 = f\left(x^m + \frac{\Delta t}{2}k_2, t_m + \frac{\Delta t}{2}\right) \quad (3.29)$$

$$k_4 = f(x^m + \Delta t k_3, t_m + \Delta t) \quad (3.30)$$

Para cada $m = 1, 2, \dots$, o algoritmo computa primeiro as k_i , e despois calcula (3.26).

3.2.2. Métodos de paso variábel

Os métodos de Taylor, Euler e Runge-Kutta usan un paso de integración fixo Δt , que é preciso especificar. Un Δt menor permite en principio maior precisión (erros de truncado menores), pero require de máis pasos e implica maior custo computacional. Por outra banda, un Δt moi pequeno pode ser contraproducente para a precisión: se a dinámica é moi lenta, a variación do estado Δx nun Δt moi pequeno será tamén moi pequena, o que pode aumentar os erros de redondeo. Por tanto a lonxitude de paso ideal depende do problema, e é difícil de saber a priori.

A alternativa é usar métodos de paso variábel. Ditos métodos estiman en cada paso o erro de truncado absoluto e relativo, e permiten especificar tolerancias absolutas e relativas, que fan de límites superiores de ditos erros. Así, a lonxitude do paso Δt vaise adaptando en cada iteración; se a estimación do erro absoluto ou do relativo supera a tolerancia correspondente, repítese o paso cunha lonxitude inferior.

Métodos para ecuacións ríxidas (“stiff”). Un sistema de EDOs non lineares é *dinamicamente estábel* en $f(x, t)$ se os autovalores $\lambda_i(x, t)$ da súa matriz Jacobiana ($\frac{\partial f}{\partial x}$) teñen partes reais negativas. O cociente entre o maior e menor autovalor é o “stiffness ratio”. Se este ratio é moi grande (p.ex., > 1000), o

sistema de ecuacións é *ríxido* (“stiff”). Fisicamente, isto correspóndese con sistemas que teñen algúns modos ou partes da súa dinámica moi rápidos e outros moi lentos. Este é o caso de algúns modelos multi-escala, que inclúen procesos que teñen lugar ao longo de diferentes escalas temporais.

A integración deste tipo de sistemas é problemática, xa que cada modo requiriría de pasos de integración diferentes. Nestas circunstancias, un método de paso variábel estándar tende a baixar o tamaño do paso até un valor esaxeradamente pequeno, mesmo en rexións onde a curva de solución é suave. Con estes tamaños de pasos, os métodos estándar poden precisar millóns de avaliacións mesmo para percorrer un curto intervalo de tempo, polo que resultan inaceptabelmente lentos. Ademais, a resolución dun sistema de EDOs ríxido pode ser numericamente inestábel mesmo se o sistema é dinamicamente estábel.

Existen métodos deseñados expresamente para EDOs ríxidas, chamados integradores ríxidos (“stiff solvers” en inglés). Ditos métodos fan cálculos adicionais en cada por paso, pero a cambio son capaces de dar pasos moito máis grandes e mellorar a estabilidade numérica en comparación cos métodos estándar (non ríxido)s. Os entornos de simulación modernos adoitan incluír integradores adecuados para sistemas stiff. Matlab conta con varios, sendo `ode15s` o que mellor funciona na maioría das aplicacións.

Exemplo: sistema ríxido. Considérese un sistema coa seguinte solución analítica:

$$x(t) = \underbrace{e^{-1000t}}_{x_1(t)} - \underbrace{e^{-0.1t}}_{x_2(t)} \quad (3.31)$$

A figura 3.1 mostra a evolución temporal de $x_1(t)$ e $x_2(t)$. O primeiro termo varía moi rapidamente ao comezo da simulación (requirindo un paso de integración moi pequeno), mentres que o segundo varía moi lentamente (requirindo un paso moito maior). Neste caso xa coñecemos a solución $x(t)$ (é un exemplo trivial para ilustrar o problema), pero en xeral, se temos un sistema de comportamento similar do cal coñecemos a súa dinámica $\dot{x}(t)$, deberemos integralo cun método apropiado para sistemas “stiff”.

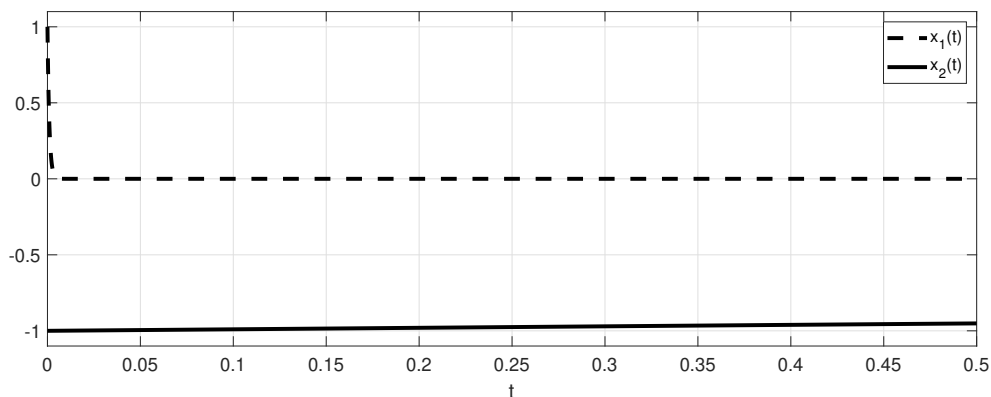


Figura 3.1: Simulación do sistema da ecuación (3.31).

3.2.3. Concluindo: como facer a integración?

Idealmente, débese escoller un integrador que proporcione a solución correcta no mínimo tempo posíbel. Como non sabemos cal é a solución correcta, escolleremos polo menos dous métodos diferentes; se as solucións son iguais, é probábel que sexan correctas.

Hai que ter en conta que o tempo de cálculo da integración pode ter máis ou menos importancia, dependendo da finalidade. Por exemplo, se simplemente queremos simular o sistema unha vez, ou unhas poucas veces, a diferenza entre que tarde unha centésima ou unha décima de segundo non é importante. Pola contra, se queremos facer unha estimación de parámetros – o que implica un procedemento de optimización que require de milleiros de simulacións – o tempo de cada simulación pasa a ser relevante.

Unha boa práctica é a seguinte:

1. Escoller un integrador de paso variábel, e adecuado para sistemas ríxidos → por exemplo, en Matlab, comezar por `ode15s`.
2. Axustar as tolerancias absoluta e relativa a valores moi estritos → por exemplo, 10^{-10} ou 10^{-12} ; en Matlab, isto faise con: `odeset('RelTol',1e-10,'AbsTol',1e-12)`.

3. Simular o modelo, gardando o resultado da simulación e o tempo de cálculo requirido.
4. Probar con outro integrador, usando tamén tolerancias moi estritas.
5. Comparar os resultados dos pasos (3) e (4); se coinciden, podemos ter a certeza razoábel de que a solución obtida é correcta. En tal caso pasaremos ao paso seguinte (6). En caso contrario, probamos con outro(s) integrador(es) e tolerancias máis severas, até chegar a un certo consenso.
6. Relaxar as tolerancias para baixar o tempo de cálculo. Se a solución comeza a variar apreciablemente, é sinal de que se relaxaron en exceso. Como regra xeral, non se deben baixar de 10^{-4} , pero moitas veces son precisos valores máis estritos (é dicir, 10^{-6} , 10^{-8} ...).

Exemplo: integración dun sistema con varios solvers. A seguinte ecuación describe, dunha forma simplificada, a auto-regulación do factor de transcripción dunha proteína x , sen ter en conta o retardo (adaptado de [16], pág. 101):

$$\dot{x} = b + k_1 \cdot \frac{x^2}{(x^2 + k_2)} - k_{deg} \cdot x; \quad (3.32)$$

A figura 3.2 mostra o resultado da integración desta ecuación con diferentes métodos: o método de Runge–Kutta de cuarta orde (3.26), con tres pasos diferentes de integración ($\Delta t = [0.05; 0.5; 3]$); e o método de paso variábel ode15s, con dúas tolerancias diferentes (tolerancia absoluta = tolerancia relativa = $[10^{-2}; 10^{-6}]$).

Obsérvase que as integracións coinciden para os métodos R–K con pasos de integración baixos (0.05 e 0.50) e o método de paso variábel con tolerancias precisas (10^{-6}). As curvas correspondentes superpóñense, e aproximan ben a solución verdadeira. Pola contra, se integramos o modelo con tolerancias demasiado relaxadas (10^{-2}) ou, no caso de de paso fixo, cun paso demasiado grande ($\Delta t = 3$), as curvas obtidas afástanse da realidade.

En canto ao custo computacional, ode15s é máis lento que R–K por ser aplicábel a sistemas stiff. Dentro de cada método, maior precisión implica maior tempo de cálculo. Os valores aproximados en segundos son:

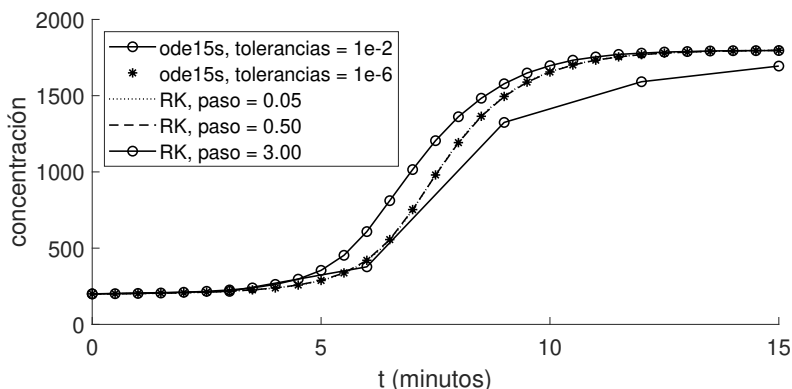


Figura 3.2: Integración dun sistema dinámico con métodos de paso fixo (Runge-Kutta, 'RK', con diferentes pasos), e de paso variábel (ode15s, con diferentes tolerancias).

ode15s, tolerancias = $10^{-6} \rightarrow 0.02$ s.

ode15s, tolerancias = $10^{-2} \rightarrow 0.01$ s.

R-K, paso = 0.05 $\rightarrow 0.006$ s.

R-K, paso = 0.50 $\rightarrow 0.002$ s.

R-K, paso = 3.00 $\rightarrow 0.0002$ s.

3.3. Integración de ecuaciones estocásticas

Alguns procesos requiren de ecuacións estocásticas para seren modelados con precisión. Isto acontece cando as entidades representadas polas variábeis de estado son pouco numerosas. É o caso, por exemplo, de reaccións químicas nas que participan poucas moléculas (como nalguns procesos intracelulares), ou de sistemas ecolóxicos nos que interveñen poucos individuos de cada especie. Por exemplo, as flutuacións nas concentracións de especies químicas escalan cun factor de $1/\sqrt{(N)}$; así, para $N = 1000$ ditas flutuacións son de $\approx 3\%$ ($1/\sqrt{1000} (\times 100\%)$), mentres que para $N = 10$ teríamos $1/\sqrt{10} (\times 100\%) \approx 32\%$. No primeiro caso é razoábel usar unha aproximación determinista (describindo o proceso cunha EDO), pero no segundo sería

desexábel usar un modelo estocástico.

Un proceso dinámico estocástico defínese por tres elementos:

- O seu *estado*, que vén dado polo conxunto das súas variábeis de estado $(X_1, X_2, \dots, X_n)$ nun determinado momento.
- As posíbeis *transicións* entre estados, que veñen dado pola topoloxía e a dinámica, é dicir polas reaccións $R_1, R_2, \dots, R_M$.
- As *propensións*, $a_1, a_2, \dots, a_M$, ou probabilidades de cada transición por unidade de tempo.

Nunha reacción química as propensións indican a probabilidade de colisión de dous reactantes. Considérase que as reaccións son *unidireccionais*; unha ecuación reversíbel represéntase mediante dúas reaccións, unha en cada sentido. Tamén se considera que son *elementais*, isto é, só poden intervir dúas especies, xa que a colisión simultánea de tres ou máis especies ten unha probabilidade desprezábel.

3.3.1. Algoritmo de Gillespie

O clásico algoritmo de Gillespie [23] proporciona a resposta (aleatoria) a dúas preguntas: *que* reacción terá lugar, e *cando*. Para determinalo, en cada paso cóllense dous números aleatorios:

- Un número aleatorio r_1 , extraído dunha distribución uniforme en $[0, 1]$. Con el determínase o *tempo* τ que pasará até a seguinte reacción: $\tau = \frac{\ln(1/r_1)}{\sum_{j=1}^M a_j}$. (Isto é equivalente a extraer τ dunha distribución exponencial de probabilidades, pero máis simple.) As a_j son as propensións de cada unha das M reaccións, as cales se determinan a partir de principios de acción de masas. O algoritmo de Gillespie é un método de paso variábel, no que o paso é o tempo τ .
- Outro número aleatorio r_2 , extraído ao igual que r_1 dunha distribución uniforme en $[0, 1]$. Con el determínase a reacción R_j que se producirá no tempo $t + \tau$. A reacción depende das propensións a_i : escóllese a reacción R_j tal que $\sum_{i=1}^{j-1} a_i < r_2 \sum_{i=1}^M a_i < \sum_{i=1}^j a_i$

Despois de ter lugar a reacción, o número de moléculas x_j é actualizado e execútase o seguinte paso, e así até chegar ao tempo final. Un diagrama do algoritmo móstrase na Figura 3.3. Como o resultado do algoritmo é aleatorio, tipicamente se executa moitas veces, obtendo así un conxunto de traxectorias das cales se adoita calcular estatísticas como a media, varianza, etc. O algoritmo de Gillespie pode resultar moi custoso computacionalmente, especialmente para modelos ríxidos. Existen outros algoritmos posteriores máis eficientes.

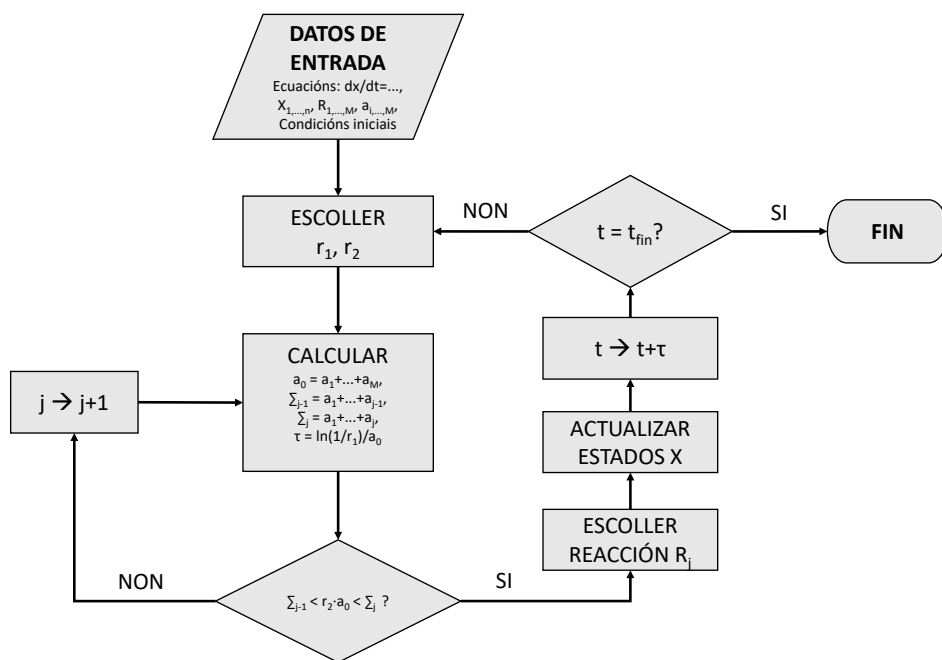
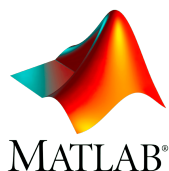


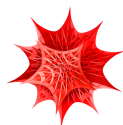
Figura 3.3: Algoritmo de Gillespie.

3.4. Software de simulación

3.4.1. Entornos de programación de propósito xeral



MATLAB (<https://es.mathworks.com/products/matlab.html>) é a linguaxe máis comunmente usada para simular, calibrar e analizar modelos biomédicos. Proporciona un amplo abano de toolboxes especializadas. O seu principal inconveniente é que é de pago; existe unha versión aberta, Octave, pero que non posúe tantas funcionalidades.



Mathematica

(<https://www.wolfram.com/mathematica/>), ao igual que Matlab, é un software científico comercial de propósito xeral. A súa principal vantaxe respecto a Matlab é no campo do cálculo simbólico. O seu uso non está tan estendido na comunidade de modelado biolóxico.



Python (<https://www.python.org/>) é unha linguaxe de programación de propósito xeral, cuxa popularidade entre a comunidade do modelado biolóxico vai en aumento. A diferenza de Matlab, é de código aberto. Conta con moitas librarías que proporcionan funcionalidades especializadas. Un exemplo é SciPy, con integradores de EDOs, métodos de optimización, etc.

```

clear;

% Opciones de integración:
tfin      = 15;
tspan     = [0:0.5:tfin];
my_opts1  = odeset('RelTol',1e-2,'AbsTol',1e-2);
my_opts2  = odeset('RelTol',1e-6,'AbsTol',1e-6);
xzero     = 200;

% Integrar o sistema de diferentes formas (medindo o tempo con tic-toc):
tic, [t1b,y1b] = ode15s(@dxtdt,tspan,xzero,my_opts2); toc
tic, [t1a,y1a] = ode15s(@dxtdt,tspan,xzero,my_opts1); toc
tic, [t1c,y1c] = runge_kutta_4(@dxtdt,tfin,xzero,0.05); toc
tic, [t1d,y1d] = runge_kutta_4(@dxtdt,tfin,xzero,0.5); toc
tic, [t1e,y1e] = runge_kutta_4(@dxtdt,tfin,xzero,3); toc

% Crear figura:
figure()
hold on
my_line_width = 1.0;
plot(t1a,y1a,'-o','Color','b','LineWidth',my_line_width);
plot(t1b,y1b,'*','Color','b','LineWidth',my_line_width);
plot(t1c,y1c,':','Color',[0.5 0.5 0.5],'LineWidth',my_line_width);
plot(t1d,y1d,'--','Color',[0.5 0.5 0.5],'LineWidth',my_line_width);
plot(t1e,y1e,'-o','Color',[0.5 0.5 0.5],'LineWidth',my_line_width);
xlabel('t (minutos)','FontSize',14)
ylabel('concentración','FontSize',14)
legend('ode15s, tolerancias = 1e-2','ode15s, tolerancias = 1e-6',...
       'RK, paso = 0.05','RK, paso = 0.50','RK, paso = 3.00','Location','NW')
set(gca,'FontSize',14);
xlabel('t (minutos)')
set(gca,'FontSize',14);
xlim([0 tfin])

% Función que define a dinámica do sistema (EDO):

function dxdt = dxtdt(t,x)
    b = 20; k1 = 2000; k2 = 400000; kdeg = 1;
    dxdt = b + k1*x^2/(x^2+k2) -kdeg*x;
end

```

Figura 3.4: Exemplo de programación en Matlab: integración dun sistema con varios *solvers*. Código que integra un sistema dinámico e crea a Figura 3.2.



Julia (<https://julialang.org/>) é unha linguaxe de programación científica de código aberto, orientada ao modelado dinámico. É máis recente que Matlab e Python (foi creado en 2009), e tamén máis inmaturo. Aínda que conta con moitos defensores entre a comunidade de modelado biolóxico, aínda non proporciona tantas utilidades como as alternativas anteriores.

3.4.2. Ferramentas de simulación especializadas

Matlab, Mathematica, Python, Julia – e outros – son entornos moi potentes, pero que esixen unha aprendizaxe por parte do usuario que pode ser lenta. Por outra banda, existen ferramentas máis especializadas, que posúen menos funcionalidades pero a cambio permiten traballar de forma máis intuitiva, sobre todo para persoas con limitada experiencia computacional. Unha das mellores é COPASI:



COPASI (<http://copasi.org/>) é un programa que permite a simulación, análise, calibración, visualización de sistemas bioquímicos. Proporciona integradores continuos e estocásticos. Conta cunha longa traxectoria (desde 2006) e é gratuíto. Un inconveniente é que as ecuacións dos modelos non poden ser introducidas directamente, senón que o modelo debe ser definido a partir de reaccións químicas; as ecuacións correspondentes son deducidas polo programa.

Na actualidade existen dúcias de ferramentas de simulación con moi diferentes características. Algunhas delas lístanse e descríbense no libro de Ed-da Klipp [33] (na sección 2.4.2 e no capítulo 17). Unha panorámica actualizada en 2021 encóntrase en [40]. Un recurso desenvolvido recentemente é <https://biosimulators.org/>, que serve como directorio de ferramentas de simulación e proporciona interfaces para algunhas delas.

3.5. Estándares, formatos e repositorios

“The nice thing about standards is that you have so many to choose from.”

– Andrew S. Tanenbaum

Un problema que presenta o uso de ferramentas especializadas é que frecuentemente é preciso usar varias delas, ao non existir unha soa que leve a cabo todas as tarefas necesarias. Xorde entón o problema da incompatibilidade de formatos: a forma en que se especifica o modelo ou as simulacións, a forma en que se gardan os resultados, etc. A solución (en teoría) sería o establecemento de estándares para ditas especificacións, o que simplificaría (e idealmente automatizaría) o intercambio de información entre ferramentas. No ámbito da bioloxía de sistemas existen moitos esforzos neste sentido, pero infelizmente aínda se está lonxe de alcanzar a estandarización existente en disciplinas máis maduras como a enxeñaría eléctrica e electrónica. Nesta sección menciónanse algúns dos avances máis significativos.

COMBINE. É a principal iniciativa (<https://co.mbine.org/>) para coordinar o desenvolvemento de formatos para modelos computacionais en bioloxía, fomentando a súa interoperabilidade. Para iso, COMBINE define una serie de especificacións que deben cumprir ditos formatos para seren considerados “estándares”. Os que COMBINE considera como tais están listados en <https://co.mbine.org/standards>. Entre eles destacamos os seguintes:

- **SBML**, Systems Biology Markup Language (<http://sbml.org/>), é un formato de ficheiros tipo XML para especificar *modelos*. É seguramente o formato máis usado na comunidade, aínda que o seu uso dista de ser sistemático. Existen toolboxes que facilitan o seu uso xunto con outras ferramentas. Outro formato similar é **CellML** (<https://www.cellml.org/>).
- **SED-ML**, Simulation Experiment Description Markup Language (<https://sed-ml.org/>), é un formato que define como especificar *simulacións*.
- **SBGN**, Systems Biology Graphical Notation (<https://sbgn.github.io/>), describe como representar *diagramas* de procesos biolóxicos.

Cómpre mencionar tamén **PEtab** [47], actualmente candidato a estándar de COMBINE. É un formato para especificar problemas de estimación de parámetros (tarefa que se estudará no próximo capítulo), baseado en SBML e en ficheiros de datos separados por tabulacións (.TSV).

Existen numerosas ferramentas que procuran facilitar o intercambio de información baseándose nestes formatos. Un exemplo recente é a **Systems Biology Simulation Core Library** (<https://draeger-lab.github.io/SBSCL/>), unha librería de Java para a simulación numérica de modelos que soporta os estándares SBML e SED-ML [44].

Cuestións e problemas do Tema 3

C3.1. Definición e utilidade da transformada de Laplace.

C3.2. Mediante o uso da transformada de Laplace, obter a solución das ecuacións (1.5)–(1.6), que describen un modelo de dous compartimentos, para unha entrada $u(t)$ impulso unitario e $k_{12} = 2$, $k_{10} = 1$, $k_{21} = 3$, $k_{1u} = 4$.

C3.3. Obter a función de transferencia do modelo dado polas ecuacións (1.5)–(1.6), cando $y(t) = x_1(t)$.

C3.4. Describir a solución algorítmica recursiva para a integración de $\dot{x}$ baseada en series de Taylor. Este método é auto-iniciábel?

C3.5. Que significa que un método de integración de EDOs é auto-iniciábel? Dicar dous métodos auto-iniciábeis de paso fixo.

C3.6 Que inconveniente presentan os métodos de integración de paso fixo?

C3.7. Que tipos de erros se cometen na resolución numérica de EDOs, e cales son as súas causas?

C3.8. Método de Euler para integración de EDOs. Dedución a partir do desenvolvemento en series de Taylor.

C3.9. Sistemas de ecuacións “ríxidos” (stiff): como se caracterizan matematicamente, e con que tipos de fenómenos físicos se corresponden?

C3.10. A solución obtida ao integrar numericamente un modelo dinámico está influída polo integrador escollido e as opcións usadas. Como a solución correcta non é coñecida, que procedemento podemos usar para tentar obter unha boa solución nun tempo razoábel?

- C3.11.** Que elementos hai que definir para caracterizar un proceso estocástico?
- C3.12.** Que algoritmo se usa para integrar ecuacións estocásticas? Describi-lo.

4

Calibración

“With four parameters I can fit an elephant, and with five I can make him wiggle his trunk.”

– John von Neumann

No tema 2 aprendemos a obter as ecuacións diferenciais dun modelo dinámico, e no tema 3 vimos como integralas para simular o comportamento do sistema. Pero aínda non tivemos en conta que polo xeral as ecuacións conteñen parámetros descoñecidos, cuxo valor é preciso estimar. Esta tarefa é máis difícil do que parece a primeira vista. Nesta unidade aprenderemos como se fai.

A construción de modelos matemáticos de sistemas dinámicos a partir de datos chámase **identificación de sistemas**. É un proceso consistente nunha serie de subrutinas, que pode requirir de varias iteracións até completarse satisfactoriamente. O estudo da identificación de sistemas é propio da enxeñaría de sistemas e control [36] e conta cunha longa tradición de aplicación a sistemas biolóxicos [4]. O groso dos esforzos destínanse a determinar os valores dos parámetros descoñecidos do modelo [58]. A Figura 4.1 mostra un ciclo típico de modelado, segundo o protocolo proposto en [55].

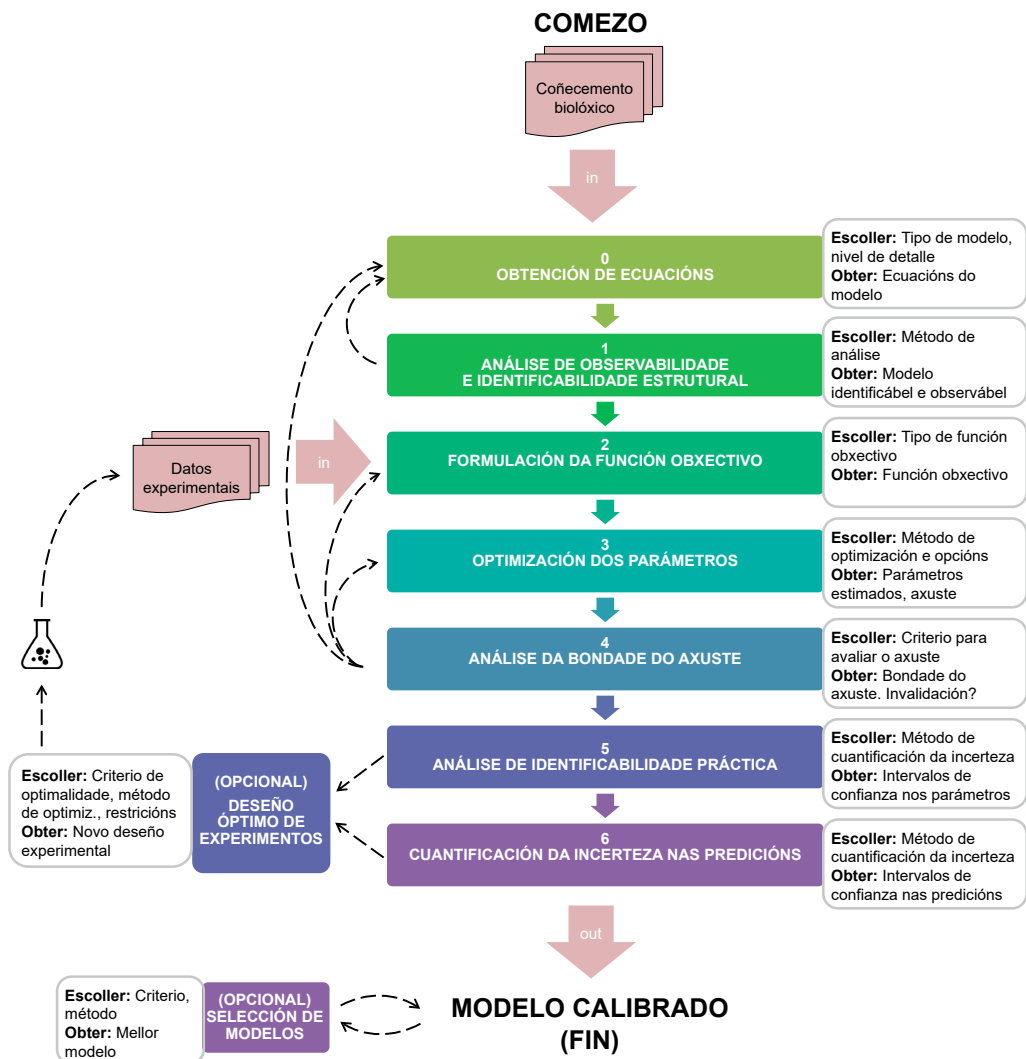


Figura 4.1: O ciclo de modelado, ou identificación de sistemas [55].

4.0. PASO 0: obter as ecuacións do modelo

PASO 0.1. O punto de partida é *formular a pregunta que debe responder* o modelo. Como vimos no Capítulo 2, isto determina o nivel de detalle necesario. En base a isto debemos **escoller o tipo de modelo**, de entre os descritos no Capítulo 1. A clase de modelos máis comunmente usada será o de ecuacións diferenciais ordinarias.

PASO 0.2. Unha vez decidido isto, podemos **determinar as súas ecuacións**. Para iso comezamos por *buscar o coñecemento* biolóxico (ou bioquímico, etc) dispoñíbel. No ámbito da bioloxía molecular isto implica determinar as reaccións bioquímicas relevantes para o proceso, e as especies que interveñen nelas. Para este fin, ademais da *literatura científica* pódense consultar *bases de datos* como a KEGG (Kyoto Encyclopedia of Genes and Genomes, <https://www.genome.jp/kegg/>), STRING (<https://string-db.org/>), ou REACTOME (<https://reactome.org/>). O seguinte paso é asignar *cinética* ás reaccións, como se explicou na sección 2.2.

PASO 0.3. Deste forma obtense un modelo matemático dinámico, que xeralmente terá algúns parámetros descoñecidos. Pode **arrecadar información sobre os parámetros** en bases de datos como Sabio-RK (<http://sabio.h-its.org/>) ou BRENDA (<https://www.brenda-enzymes.info/statistics.php>), mais en xeral será preciso estimalos a partir de datos experimentais – noutras palabras, haberá que *calibrar* o modelo (véxase a sección 4.3).

Nótese que ás veces o modelo que nos interesa pode estar dispoñíbel nun repositorio como BioModels (<https://www.ebi.ac.uk/biomodels/>) ou JWS Online (<https://fair-dom.org/jws-online/>), aínda que non sempre se corresponderán exactamente coas nosas necesidades, e ademais en xeral terán que ser (re)calibrados.

Exemplo do paso 0: a ruta de sinalización Akt. Supoñamos que queremos esclarecer o comportamento temporal da ruta de sinalización Akt, involucrada no crecemento celular en mamíferos. A Akt é coñecida tamén como proteína quinase B (PKB). A súa activación depende do factor de crecemento epidérmi-

co (“epidermal growth factor”, EGF, e o seu receptor EGFR). O resultado final é a fosforilación da proteína ribosómica S6, que regula o tamaño celular. Como nos interesa o comportamento temporal deste sistema, e queremos obter resultados cuantitativos, precisamos un modelo dinámico. Para obtelo, procuramos información na literatura científica. Buscando “agf akt pathway model” nalgún buscador especializado (p.ex. Google Scholar) atopamos, entre outros, o artigo [22]. No artigo non veñen as ecuacións, pero proporciónase o modelo en COPASI e SBML como información suplementaria. As EDOs son as seguintes:

$$\begin{aligned}
 \dot{EGFR} &= 68190 \cdot p_{13} - EGF \cdot EGFR \cdot p_1 + EGF_EGFR \cdot p_2 - EGFR \cdot p_{13}, \\
 \dot{EGF_EGFR} &= EGF \cdot EGFR \cdot p_1 - EGF_EGFR \cdot p_2 - EGF_EGFR \cdot p_3 \\
 \dot{pEGFR} &= EGF_EGFR \cdot p_3 - pEGFR \cdot p_4 + pEGFR_Akt \cdot (p_5 + p_6) - Akt \cdot pEGFR \cdot p_7, \\
 \dot{pEGFR_Akt} &= Akt \cdot pEGFR \cdot p_7 - pEGFR_Akt \cdot (p_6 - p_5), \\
 \dot{Akt} &= pAkt \cdot p_8 + pEGFR_Akt \cdot p_5 - Akt \cdot pEGFR \cdot p_7, \\
 \dot{pAkt} &= pAkt_S6 \cdot p_9 - pAkt \cdot p_8 + pAkt_S6 \cdot p_{10} + pEGFR_Akt \cdot p_6 - S6 \cdot pAkt \cdot p_{11}, \\
 \dot{S6} &= pAkt_S6 \cdot p_9 + pS6 \cdot p_{12} - S6 \cdot pAkt \cdot p_{11}, \\
 \dot{pS6} &= pAkt_S6 \cdot p_{10} - pS6 \cdot p_{12}, \\
 \dot{pAkt_S6} &= S6 \cdot pAkt \cdot p_{11} - pAkt_S6 \cdot p_{10} - pAkt_S6 \cdot p_9,
 \end{aligned} \tag{4.1}$$

EGF é a entrada externa (estímulo), e hai 9 estados dinámicos e 13 parámetros cinéticos, ademais dunha constante coñecida. Só dúas reaccións se consideran enzimáticas, para modelalas inclúense dous estados adicionais. A Fig. 4.2 representa as interaccións entre as variábeis do modelo.

Na definición das saídas do sistema aparecen 3 parámetros adicionais:

$$\begin{aligned}
 y_1 &= p_{14} \cdot (pEGFR + pEGFR_Akt), \\
 y_2 &= p_{15} \cdot (pAkt + pAkt_S6), \\
 y_3 &= p_{16} \cdot pS6
 \end{aligned} \tag{4.2}$$

Implementamos as ecuacións (4.1)–(4.2) nun script de Matlab para proceder ao paso seguinte (análise da observabilidade e identificabilidade estrutural), que se describe a continuación.

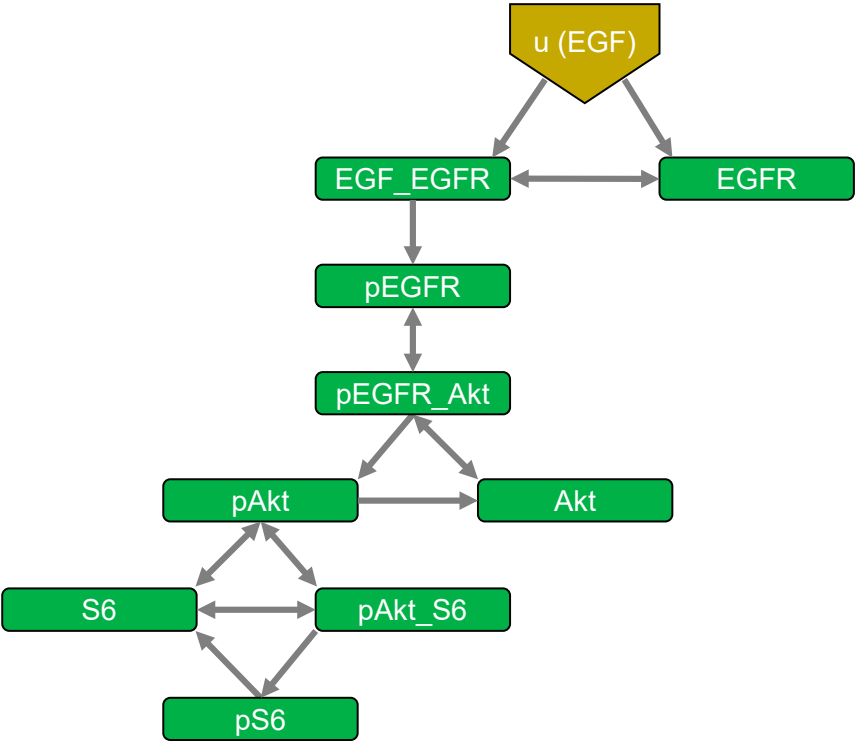


Figura 4.2: Diagrama da ruta de sinalización Akt.

4.1. PASO 1: analizar observabilidade e identificabilidade estrutural

Conceptos. A identificabilidade é unha propiedade dun modelo que di se é posíbel determinar os valores dos parámetros descoñecidos a partir do coñecemento das súas entradas e saídas, así como das súas ecuacións.

- A identificabilidade *estrutural* responde a pregunta asumindo medidas da saída libres de erros e con infinita resolución. Baixo estas circunstancias, se hai non-identificabilidade é debido a simetrías ou redundancias na estrutura do modelo, que son consecuencia dunha sobreparametrización: o modelo contén máis parámetros dos que se poden estimar medindo a saída.
- Outro tipo de falta de identificabilidade é a *numérica* ou *práctica*, cuxa causa é a mala calidade ou cantidade dos datos experimentais; verémola na sección 4.5.

Unha propiedade análoga á identificabilidade estrutural é a *observabilidade*, que nos di se é posíbel determinar os estados non medidos a partir das entradas, saídas e ecuacións do modelo. A falta de observabilidade soe estar ligada á falta de identificabilidade: a excesiva flexibilidade do modelo dificulta a súa capacidade de predicir.

Relevancia. Se un modelo non é estruturalmente identificábel...

- non será posíbel determinar o valor verdadeiro dos seus parámetros. Isto é perigoso, porque en ocasións estes parámetros teñen un significado mecanístico, e un valor erróneo pode levarnos a conclusións erróneas.
- Ademais, é frecuente que a falta de identificabilidade provoque falta de observabilidade, o que quere dicir que as simulacións do modelo sobre os estados non observábeis serán incorrectas.

En conclusión, é fundamental analizar estas propiedades para evitar extraer conclusións que nos poden levar a tomar decisións equivocadas. A Fig. 4.3 ilustra cun exemplo moi simple varias causas posíbeis de falta de observabilidade e identificabilidade estrutural, e mostra os seus efectos adversos.

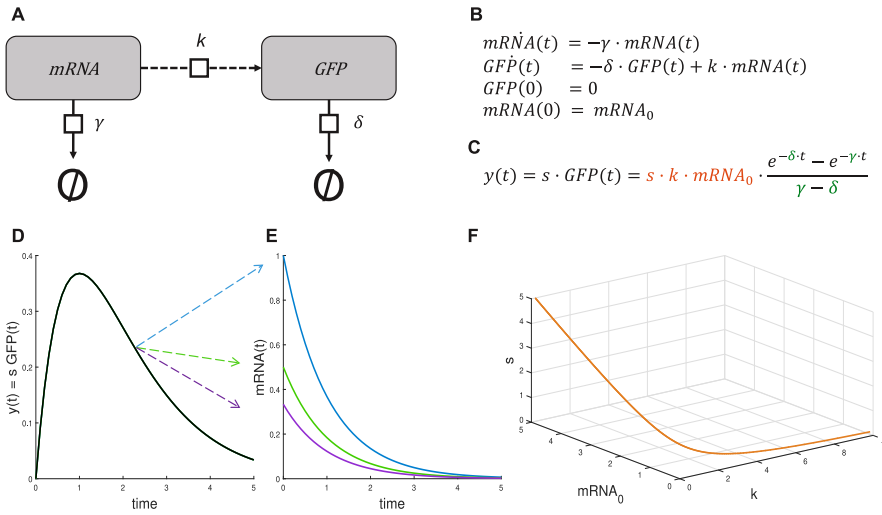


Figura 4.3: Análise de observabilidade e identificabilidade estrutural. **(A)** Diagrama dun proceso de tradución de mRNA a unha proteína. **(B)** Modelo en EDOs [2] con dous estados, mRNA e GFP. **(C)** A saída do modelo é a intensidade da fluorescencia, que é proporcional ao nivel de GFP. O modelo ten 5 parámetros descoñecidos: a concentración inicial $mRNA_0$, tres parámetros cinéticos (γ, δ, k) e un de escalado (s). Dada a súa sinxeleza, é posíbel calcular analiticamente $y(t)$, que é función do produto de tres parámetros ($s \cdot k \cdot mRNA_0$) mostrados en laranxa, e contén outro termo que inclúe δ e γ en verde. Esta última expresión é simétrica a respecto de δ e γ : pódense intercambiar os seus valores sen alterar o resultado. Por tanto, estes dous parámetros non son estruturalmente identificábeis globalmente, senón só localmente, con dúas solucións. O produto ($s \cdot k \cdot mRNA_0$) ten o mesmo valor para infinitas combinacións de valores dos tres parámetros, polo que son estruturalmente non-identificábeis. **(D)** Falta de identificabilidade estrutural: a saída do modelo é idéntica para un número infinito de vectores de parámetros. **(E)** A falta de identificabilidade provoca falta de observabilidade: móstranse tres vectores diferentes de parámetros que, aínda que producen a mesma saída ($GFP(t)$), dan lugar a diferentes simulacións de $mRNA(t)$; por tanto, ese estado non pode ser determinado. **(F)** Correlación entre os 3 parámetros non identificábeis: a liña indica combinacións dos parámetros para os cales a saída é idéntica.

PASO 1.1. Análise: métodos. A análise destas propiedades realízase simbolicamente. Para analizar as propiedades dun punto de vista *local* pódese adoptar o enfoque resumido na caixa “Observabilidade e identificabilidade estrutural *local*”. Para analízalas desde o punto de vista *global*, hai que usar álgebra diferencial (que non veremos). En xeral, ambos métodos resultan excesivamente complexos para facer os cálculos a man, polo que son precisas ferramentas computacionais. Entre o software de código aberto capaz de analizar ambas propiedades temos,

- en Matlab, GenSSI (<https://github.com/genssi-developer/GenSSI>) ou STRIKE-GOLDD (<https://github.com/afvillaverde/strike-goldd>);
- en Maple, SIAN (<https://github.com/pogudingleb/SIAN>, tamén dispoñíbel sen necesidade de instalación en <https://maple.cloud/app/5710317752942592/SIAN>) ou ObservabilityTest (<https://github.com/sedoglavic/ObservabilityTest>).
- en Julia, SIAN (<https://github.com/pogudingleb/SIAN>) e StructuralIdentifiability (<https://github.com/SciML/StructuralIdentifiability.jl>).

Observabilidade de modelos lineares

Modelos lineares. Sexa un modelo no espazo de estados, representado na nosa notación habitual como:

$$\mathcal{M}_L = \begin{cases} \dot{x}(t) = A \cdot x(t) + B \cdot u(t), \\ y(t) = C \cdot x(t), \end{cases}$$

A matriz de observabilidade é $\mathcal{O} = \begin{pmatrix} C \\ C \cdot A \\ C \cdot A^2 \\ \vdots \\ C \cdot A^{(n-1)} \end{pmatrix}$, e o modelo é obser-

vábel se $\text{rango}(\mathcal{O}) = n$.

Observabilidade e identificabilidade estrutural local de modelos non-lineares

Consideremos un modelo non-linear, con parámetros descoñecidos θ e (posibelmente) entradas descoñecidas $w(t)$:

$$\mathcal{M}_{NL} = \begin{cases} \dot{x}(t) = f(x(t), \theta, u(t), w(t)) \\ y(t) = g(x(t), \theta, u(t), w(t)) \end{cases}$$

Incluimos θ e $w(t)$, así como as súas derivadas até orde l , nun vector de estados aumentado, $\tilde{x}(t) = \begin{pmatrix} x(t)^T & \theta^T & w(t)^T & \dots & w(t)^{(l)^T} \end{pmatrix}^T$. Así, a dinámica aumentada de orde l é:

$$\dot{\tilde{x}}(t) = \tilde{f}(\tilde{x}(t), u(t)) = \begin{pmatrix} f(x(t), \theta, u(t), w(t)) \\ 0_{n_\theta \times 1} \\ \dot{w}(t) \\ \vdots \\ w^{(l+1)}(t) \end{pmatrix},$$

A matriz de observabilidade constrúese con derivadas de Lie:

- Derivada de Lie de orde 0: é a saída do sistema, $L_{\tilde{f}}^0 g = g$.

- Derivada de Lie de orde 1:

$$L_{\tilde{f}} g(\tilde{x}, u) = \frac{\partial g}{\partial \tilde{x}}(\tilde{x}, u) \tilde{f}(\tilde{x}, u) + \frac{\partial g}{\partial u}(\tilde{x}, u) \dot{u}.$$

- Derivada de Lie de orde superior (i):

$$L_{\tilde{f}}^i g(\tilde{x}, u) = \frac{\partial L_{\tilde{f}}^{i-1} g}{\partial \tilde{x}}(\tilde{x}, u) \tilde{f}(\tilde{x}, u) + \sum_{j=0}^{i-1} \frac{\partial L_{\tilde{f}}^{i-1} g}{\partial u^{(j)}}(\tilde{x}, u) u^{(j+1)}.$$

A matriz de observabilidade e identificabilidade é:

$$\mathcal{O}_I(\tilde{x}, u) = \frac{\partial}{\partial \tilde{x}} \begin{pmatrix} L_{\tilde{f}}^0 g(\tilde{x}, u) \\ L_{\tilde{f}} g(\tilde{x}, u) \\ \vdots \\ L_{\tilde{f}}^{n_{\tilde{x}}-1} g(\tilde{x}, u) \end{pmatrix}.$$

O modelo é observábel e estruturalmente identificábel (localmente) se $\text{rango}(\mathcal{O}_I(\tilde{x}_0, u)) = n_{\tilde{x}} = n_x + n_\theta + n_w$. Se $\text{rango}(\mathcal{O}_I) < n_{\tilde{x}}$, pode haber variábeis identificábeis / observábeis. Para determinar a observabilidade da variábel j -ésima, quitamos a columna j -ésima de $\mathcal{O}_I$ e recalcamos o rango. Se o rango diminúe, a variábel é identificábel / observábel; se o rango permanece constante, non.

Adicionalmente, como software fechado existe DAISY (<https://daisy.dei.unipd.it/>) en REDUCE, ou EAR (<http://www.fcc.chalmers.se/software/other-software/identifiabilityanalysis/>) en Mathematica.

Como alternativa ao cálculo computacional, é posíbel aplicar un método simple que se pode facer a man [10], o cal non presenta garantía total de exactitude pero dá o resultado correcto en moitos casos. Consiste en comprobar se a saída do modelo posúe simetrías respecto a transformacións de escalado. En caso afirmativo, os parámetros e variábeis involucrados en ditas simetrías non son identificábeis / observábeis. O problema é que pode haber modelos que non posúan simetrías de escala, pero si de outro tipo; neste caso o método clasificaría erroneamente o modelo como identificábel e observábel.

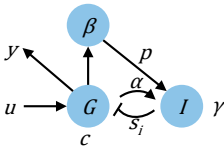
PASO 1.2. Accións post-análise. Se hai parámetros estruturalmente non identificábeis ou variábeis de estado non observábeis, é aconsellábel adoptar unha das seguintes medidas:

- a. *Reformular o modelo* fusionando os parámetros non identificábeis e/ou os estados non observábeis en combinacións que si o sexan. Isto non sempre é factíbel, e cando o é, non sempre resulta útil. Por exemplo, o modelo da Figura 4.3 pódese reformular usando a función AutoRepar do software STRIKE-GOLDD (<https://github.com/afvillaverde/strike-goldd/tree/master/STRIKE-GOLDD>), obtendo unha reparametrización estruturalmente identificable cuxas novas variábeis son $M = k \cdot s \cdot mRNA$, $P = s \cdot GFP$, $M_0 = k \cdot s \cdot mRNA_0$. As novas ecuacións son $\dot{M} = -\gamma \cdot M$, $\dot{P} = -\delta \cdot P + M$, $y = P$. Nótese que, aínda que o modelo resultante é estruturalmente identificábel e observábel, as súas variábeis xa non teñen o seu significado orixinal. Isto é moi frecuente, aínda que en ocasións é posíbel obter manter certas variábeis de interese inalteradas. En calquera caso, o usuario do modelo debe decidir se a transformación obtida é aceptábel para os fins do modelo.
- b. *Fixar os parámetros non identificábeis* a valores tomados da literatura ou de bases de datos de coñecemento biolóxico. De tomar esta decisión, hai que telo en conta ao analizar posteriormente a incerteza nas predicións do modelo (xa que ao fixar o valor dun parámetro que en principio é descoñecido estamos a diminuír artificialmente a incerteza do modelo).

- c. *Realizar experimentos adicionais*, se é posíbel, con novas condicións experimentais e/ou medidas adicionais. Na maior parte dos casos, desta forma pódese mellorar a identificabilidade e observabilidade, pero ten un custo adicional e non sempre é posíbel ou práctico.

Un exemplo do paso 1: modelo β IG. O modelo da figura 4.4 foi introducido na Sección 2.4.2, e representa a regulación do nivel de glicosa en plasma mediante insulina [29]. Lembremos que a entrada $u(t)$ é a inxestión de glicosa, $G(t)$ a concentración de glicosa, $I(t)$ a de insulina, e $\beta(t)$ a masa de células β . Este sistema, mostrado graficamente no panel (A), pode describirse cun modelo compartimental con EDOs non-lineares, mostradas en (B), que conteñen 5 parámetros descoñecidos ($p, s_i, c, \alpha, \gamma$).

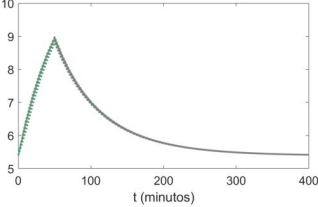
(A) Sistema dinámico



(B) Modelo mecanicista en EDOs

$$\begin{aligned}\dot{G} &= u - (c + s_i I)G \\ \dot{\beta} &= \beta \left(\frac{1.4583 \cdot 10^{-5}}{1 + \left(\frac{8.4}{G}\right)^{1.7}} - \frac{1.7361 \cdot 10^{-5}}{1 + \left(\frac{G}{4.8}\right)^{8.5}} \right) \\ \dot{I} &= p\beta \frac{G^2}{\alpha^2 + G^2} - \gamma I \\ y &= G; \\ G(0) &\text{ coñecido; } \beta(0) \text{ e } I(0) \text{ descoñecidos}\end{aligned}$$

(C) Saída medida: G (glicosa)



(D) Predición: I (insulina)

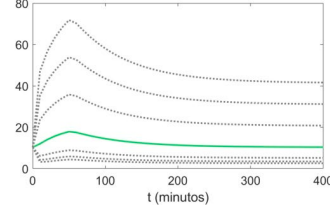


Figura 4.4: Circuito de regulación da glicosa en sangue: modelo “ β IG” [37].

Este modelo presenta falta de identificabilidade e observabilidade: medindo a concentración de glicosa ao longo do tempo ($y(t) = G(t)$), dous parámetros (p, s_i) son estruturalmente non identificables. Hai un número infinito de pares de valores (p, s_i) que xeran o mesmo curso temporal de glicosa. A gráfica (C) mostra que as simulacións do modelo de G para varios destes pares son idénti-

cas, polo que non é posible inferir os valores correctos ($s_i = 0,0005, p = 0,03$) das medicións da glicosa. Ademais, esta falta de identificabilidade está ligada a unha falta de observabilidade das variábeis de estado non medidas (I, β), polo que as predicións dos seus cursos temporais son erróneas. A gráfica **(D)** mostra as curvas de insulina simuladas cos pares (p, s_i) do panel C, que difiren considerabelmente da curva verdadeira (mostrada en verde) incluso para valores razoables de (p, s_i) . A causa desta ambigüidade é estrutural.

Posíbeis solucións:

- a. Reformular o modelo: aplicando o método AutoRepar [37], incluído na toolbox de Matlab STRIKE-GOLDD (<https://github.com/afvillaverde/strike-goldd>), obtense a seguinte transformación das variábeis de estado:

$$\begin{aligned}\beta^*(t) &= \beta(t)p, \\ \tilde{\beta}(t) &= \beta(t)^* s_i = \beta(t) s_i p, \\ \tilde{I}(t) &= I(t) s_i\end{aligned}$$

que dá lugar ao seguinte modelo reparametrizado:

$$\begin{aligned}\dot{G}(t) &= u(t) + (c + \tilde{I}(t))G(t), \\ \dot{\tilde{\beta}}(t) &= \tilde{\beta}(t)(\lambda^+ - \lambda^-), \\ \dot{\tilde{I}}(t) &= \tilde{\beta}(t)\rho_G - \gamma\tilde{I}(t), \\ y(t) &= G(t)\end{aligned}$$

Desta forma é necesario transformar todos os estados non observábeis ($I(t), \beta(t)$). Así, aínda que se obtén un modelo totalmente identificábel e observábel, non podemos empregalo para inferir ditas variábeis, o que limita a súa utilidade.

Non obstante, hai un xeito de acadar este obxectivo: aproveitando a flexibilidade adicional que dá supoñer que se descoñece a entrada externa. Se supoñemos que $u(t)$ é descoñecida, existe unha reparametrización que

volve observábel a insulina, $I(t)$, sen transformala:

$$\begin{aligned}\dot{G}(t) &= u^*(t) + (c + I(t))G(t), \\ \dot{\tilde{\beta}}(t) &= \tilde{\beta}(t)(\lambda^+ - \lambda^-), \\ \dot{I}(t) &= \tilde{\beta}(t)\rho_G - \gamma I(t), \\ y(t) &= G(t)\end{aligned}$$

onde:

$$\begin{aligned}u^*(t) &= u(t) + G(t)I(t)(s_i + 1), \\ \tilde{\beta}(t) &= \beta(t)p\end{aligned}$$

- b. Fixar o valor de algún(s) parámetros: neste caso, require fixar (p, s_i) . Mais tampouco é unha solución moi útil, xa que a menos que se poda coñecer o seu valor verdadeiro, dará lugar a erros na predición de $(I(t), \beta(t))$.
- c. Realizar experimentos adicionais: se podemos medir $(I(t), \beta(t))$, todos os parámetros son estruturalmente identificábeis. Nótese que isto implica ter que medir todos os estados.

Neste caso, se o que se quere é inferir o curso de $I(t)$, a mellor solución é a reparametrización asumindo $u(t)$ descoñecido. Alternativamente, tamén se obtén unha solución práctica combinando a 2ª e 3ª opcións: fixando o valor de p , e tomando medidas de $G(t)$ e $\beta(t)$ (o que pode ser práctico xa que $\beta(t)$ varía moi lentamente, polo que chega con tomar medidas “offline” ocasionalmente), obtense un sistema observábel e identificábel.

Outro exemplo do paso 1: a ruta de sinalización Akt. A observabilidade e identificabilidade estrutural deste modelo foron analizadas en detalle en [55], usando a toolbox STRIKE-GOLDD [54]. Na seguinte ligazón hai un live script de Matlab que leva a cabo ditas análises, e unha versión en pdf: https://github.com/afvillaverde/model_calibration_protocol/blob/main/step1_structural_identifiability/. Temos as seguintes condicións experimentais:

- A entrada EGF non pode ser manipulada en tempo continuo, así que se considera constante durante un experimento.
- Pódense realizar varios experimentos, cada un deles con diferentes niveis de EGF.
- As condicións iniciais de seis estados son sempre cero.

Con estas restricións, o resultado das análises é que hai 4 parámetros non identificábeis (p_7, p_8, p_9, p_{16}) e un estado non observábel ($pS6$). Esta falta de identificabilidade e observabilidade é debida ás condicións iniciais iguais a cero: se fosen distintas, o modelo sería totalmente observábel e identificábel. O tempo de computación necesario para levar a cabo as análises é de varios minutos nun PC estándar.

Posíbeis solucións:

- a. Reformular o modelo (non resulta práctico para este caso).
- b. Fixar o valor de 2 dos 4 parámetros non identificábeis, asignándolles valores plausíbeis sacados da literatura ou de bases de datos.
- c. Realizar experimentos adicionais: por exemplo, realizando dous experimentos con condicións iniciais distintas de cero, o modelo é estruturalmente identificábel e observábel.

4.2. PASO 2: definir a función obxectivo

Os valores dos parámetros descoñecidos estímense mediante un proceso de *optimización* computacional que se describirá no PASO 3, consistente en minimizar unha *función obxectivo* ou *función de custo*.

Compoñentes das funcións obxectivo. Unha función de custo pode constar de un ou máis dos seguintes termos:

1. *Termo de probabilidade*, $p(\mathcal{D}|\theta)$: cuantifica a bondade do axuste entre a saída do modelo e os datos experimentais. Mide a probabilidade de medir os datos $\mathcal{D}$ para uns valores dos parámetros θ . Este termo encapsula as características das medidas do proceso.

2. *Distribución a priori*, $p(\theta)$: encapsula o coñecemento previo (ou mellor dito, as crenzas preexistentes) sobre os valores dos parámetros.
3. *Outras consideracións* que se queiran introducir.

Tipos de funcións obxectivo. Todas as funcións de custo deben incluír un termo de probabilidade. A depender do que inclúan adicionalmente, podemos ter os seguintes tipos de funcións:

1. **Funcións frecuentistas:** só usan a función de probabilidade. As máis comúns son:

- *Función de verosimilitude logarítmica (log-likelihood).* Asumindo medidas independentes con ruído normalmente distribuído, a función de verosimilitude logarítmica (“negative log-likelihood”) é

$$J(\theta) = \frac{1}{2} \sum_{j=1}^{n_y} \sum_{k=1}^{n_t} \left[\log(2\pi\sigma_j^2(t_k)) + \left(\frac{y_j^m(t_k) - y_j(t_k, \theta)}{\sigma_j(t_k)} \right)^2 \right], \quad (4.3)$$

onde $y_j^m(t_k)$ son os valores medidos e $y_j(t_k, \theta)$ é o resultado da simulación da j -ésima saída no instante t_k . A desviación típica do ruído de medida, $\sigma_j(t_k)$, pode ser coñecida (p.ex. calculándoa a partir de múltiples medidas replicadas) ou ser función dos parámetros. As transformacións logarítmicas úsanse por ser máis fáciles de avaliar numericamente e facer que o optimizador converxa máis rapidamente.

- *Función de mínimos cadrados ponderada (weighted least squares).* Se as desviacións típicas son coñecidas, os sumandos $\log(2\pi\sigma_j^2(t_k))$ son independentes do vector de parámetros, e poden ignorarse, dando lugar á función de mínimos cadrados ponderada:

$$J(\theta) = \sum_{j=1}^{n_y} \sum_{k=1}^{n_t} w_j(t_k) (y_j^m(t_k) - y_j(t_k, \theta))^2 \quad (4.4)$$

onde $w_j(t_k) = \sigma_j^{-2}(t_k)$. Se os pesos w_j teñen outro valor, esta función tamén pode usarse, mais non ten unha interpretación frecuentista rigorosa.

2. **Funcións Bayesianas:** consideran tanto a función de probabilidade como a distribución a priori, o que dá lugar ao log-posterior negativo non-normalizado, $J(\theta) = -\log p(\mathcal{D}|\theta) - \log p(\theta)$. A súa avaliación é máis custosa que a das funcións frecuentistas, xa que require *estimar* unha distribución de probabilidade, o que se adoita facer con métodos de mostraxe como Markov chain Monte Carlo methods (MCMC).

Ás veces úsanse funcións con *regularización*, para tentar reducir o número de parámetros distintos de cero. Outra posibilidade é a optimización *multi-obxectivo*, na que se quere minimizar simultaneamente varios obxectivos, procurando un balance entre eles.

Implementación. Unha vez escollido o tipo de función obxectivo, é preciso implementala computacionalmente. Isto implica:

1. Escribir un código que simule o modelo para as condicións experimentais. (Lembrar que os métodos de simulación foron descritos no Capítulo 3.) Se se dispón de datos de varios experimentos en condicións diferentes, hai que simular o modelo tantas veces como experimentos.
2. Aliñar as saídas do modelo cos datos experimentais dispoñíbeis para a optimización. Esta tarefa pode non ser trivial, especialmente se hai varios conxuntos de medidas obtidas en diferentes experimentos e o mapeo das saídas do modelo cos datos varía dun experimento a outro. Formatos como PTab (<https://github.com/PEtab-dev/PEtab>) axudan a sistematizar a estrutura dos ficheiros de datos experimentais, o que permite automatizar a construción da función obxectivo.

Exemplo do paso 2: función de custo en Matlab do modelo β IG. O seguinte código calcula en Matlab a función de custo “log-likelihood” (4.3) deste modelo:

```

% Función obxectivo:
function custo = fun_obx_log_likelihood_BIG(params)

    % Integrar o modelo:
    tsim = [0:1:100];
    options = odeset('RelTol',1e-6,'AbsTol',1e-6);
    cond_inic = [0.5, 0, params(3)];
    [t,xsim] = ode15s(@(t,x) edos_BIG(t,x,params),tsim,cond_inic,options);

    % Cargar datos experimentais:
    load('experim_data.mat','tempos','medidas','desv')

    % Aliñar simulacións e dados:
    indices_tempos = [0,5,20,50];
    indices_estados = 1;
    xsim = xsim(indices_tempos,indices_estados);

    % Calcular función de custo:
    custo = 0;
    for j=1:size(medidas,2)
        custo = custo+0.5*sum(sum(log(2*pi*desv.^2)+((medidas(:,j)-xsim(:,j))./desv).^2));
    end

end

% Función que define a dinámica do sistema (EDOs):

```

```

function dxdt = edos_BIG(t,x,params)

    % Ler parámetros:
    u      = params(1);
    C      = params(2);
    Si     = params(3);
    p      = params(4);
    alpha  = params(5);
    gamma  = params(6);
    muplus = params(7);
    muminus = params(8);

    % EDOs:
    dxdt = [u-(C+Si*x(2))*x(1);
            p*x(3)*(1/(1+(alpha/x(1))^2))-gamma*x(2);
            x(3)*(muplus*(1/(1+(8.4/x(1))^1.7))-muminus*(1/(1+(x(1)/4.8)^8.5))]];

end

```

4.3. PASO 3: optimización dos parámetros

Unha vez definida a función de custo, é preciso minimizala usando un método de optimización. Para resolver este problema úsanse algoritmos iterativos, que parten dun punto inicial e van refinando a solución. Hai moitos algoritmos deste tipo dispoñíbeis, que podemos clasificar en dúas grandes categorías: locais e globais. Os métodos de optimización locais encontran o óptimo local máis próximo ao punto de partida. Na práctica, as funcións obxectivo que se encontran na calibración de modelos biolóxicos son tipicamente non-lineares e non-convexas, polo que presentan múltiples óptimos locais. Para encontrar o óptimo global é preciso usar unha estratexia global. Porén, é común usar métodos locais en combinación con estratexias globais, para aproveitar os puntos fortes dos primeiros. Por iso, veremos en primeiro lugar cales son os métodos locais máis útiles.

4.3.1. Métodos locais

Podemos clasificar os métodos locais en función da información sobre a función obxectivo que usan. Moitos métodos explotan información sobre a curvatura da función obxectivo, tanto de 1ª orde (gradiente) como de 2ª (Hesiano). Dita información permite acelerar a converxencia, mais non sempre resulta produtiva xa que non adoita estar dispoñíbel e é preciso estimala, o que ten un custo computacional asociado e pode resultar pouco fiable dependendo do método empregado.

Métodos que avalían o gradiente. Os métodos baseados en gradientes explotan o coñecemento sobre o gradiente da función obxectivo para guiar a procura. Ademais de diferenciarse por como utilizan (o que dá lugar a diversas variantes de métodos), un aspecto importante é como obteñen esa información, xa que polo xeral o gradiente non se coñece de forma analítica, senón que o hai que aproximar numericamente. Isto pode facerse de tres formas diferentes:

1. *Diferenzas finitas*. É a forma máis simple de calcular o gradiente, e tamén a menos fiable. A sensibilidade da función J respecto ao parámetro i -ésimo é:

$$\frac{\partial J}{\partial \theta_i} = \frac{J(\theta + a \cdot e_i) - J(\theta - b \cdot e_i)}{a + b},$$

onde a, b son constantes positivas e e_i é o vector unitario do compoñente i -ésimo. O problema que presenta esta aproximación é elixir valores adecuados para a, b , o que depende dos parámetros.

2. *Forward sensitivities*. Con este método, primeiro obtéñense as expresións das sensibilidades do estado e da saída diferenciando as ecuacións do modelo, (1.1)–(1.2), respecto a cada parámetro θ_i . Despois, as expresións obtidas avalíanse na traxectoria do estado, $x(t)$. Os gradientes así obtidos son máis precisos que con diferenzas finitas, pero o seu cálculo resulta ineficiente para modelos grandes, xa que o número de sensibilidades se incrementa linearmente co número dos parámetros.
3. *Adjoint sensitivities*. Este método é menos intuitivo que os dous anteriores. Baséase en calcular a traxectoria dun estado adxunto (que é continuo a intervalos e cero a tempo final) cara atrás no tempo, mediante unha secuencia de ecuacións diferenciais cara atrás. Este estado adxunto úsase para calcular o gradiente. Desta forma, o tempo de cálculo do gradiente é maiormente independente do número de parámetros. Por ser máis preciso que as diferenzas finitas e máis eficiente que as “forward sensitivities”, este método soe ser a mellor opción para modelos de tamaño mediano e grande.

Métodos que avalían o Hessiano. Estes métodos non só aproveitan a información sobre a primeira derivada da función obxectivo (o gradiente, que nos dá a dirección de maior descenso) senón tamén a da segunda derivada, é dicir o Hessiano, que nos dá a curvatura e pode permitir unha converxencia máis rápida (ver Fig. 4.5).

Dous dos principais son:

1. *Método de Newton*. Baséase en aproximar a función $J(\theta)$ mediante series de Taylor de 2ª orde. Se na iteración k se ten o vector θ_k , no seguinte

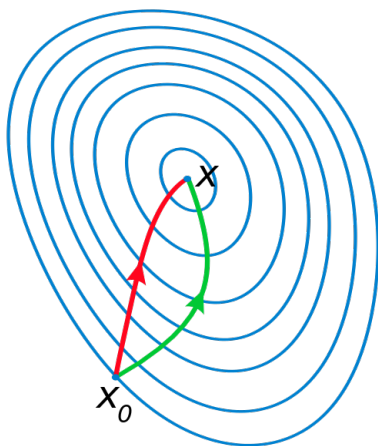


Figura 4.5: Gradiente vs Hessiano. Comparativa do descenso do gradiente (verde) e do método de Newton (vermello) para minimizar unha función. O método de Newton utiliza información de curvatura para tomar unha ruta máis directa. (Fonte: Oleg Alexandrov, Wikimedia Commons)

paso obtense:

$$\theta_{k+1} = \theta_k - [J''(\theta_k)]^{-1} J'(\theta_k)$$

É dicir, úsase información do gradiente, J' , e da inversa do Hessiano, $[J'']^{-1}$, da función obxectivo.

2. *Métodos de punto interior*. É unha familia de métodos para optimización con restricións. Os problemas de estimación de parámetros son problemas deste tipo, xa que polo xeral é posíbel especificar valores máximos e mínimos admisíbeis para os parámetros. Baséanse en percorrer a rexión de valores admisíbeis. Unha implementación moi usada é a da función de Matlab “fmincon”.

Métodos que avalían só o valor da función obxectivo. Esta familia de métodos inclúe métodos de interpolación, busca directa, procura de patróns, e heurísticas como “Hill climbing” ou o de Nelder-Mead.

4.3.2. Métodos globais.

Existen varios tipos de estratexias de optimización global. Unha clase son os métodos *deterministas*, que proporcionan garantías teóricas de que a solución encontrada é efectivamente a real, dentro dunha certa tolerancia. Sen embargo, na práctica estes métodos teñen un custo computacional moi elevado, polo que o seu uso non resulta viábel.

No seu lugar úsanse outras estratexias, que non garanten alcanzar o óptimo global, mais permiten obter solucións suficientemente boas en tempos razoábeis. As estratexias globais máis comúns na práctica son de dous tipos:

Multi-start: consiste en levar a cabo varias optimizacións locais, comezando desde diferentes puntos do espazo de parámetros. Débese escoller un número suficientemente elevado de puntos iniciais, e deben estar distribuídos no espazo. A idea ilústrase na Fig. 4.6.

Metaheurísticas: as heurísticas son técnicas deseñadas para encontrar unha solución aproximada cando os métodos clásicos non son capaces de encontrar unha solución exacta nun tempo razoábel. Sacrifican precisión (a solución obtida pode non ser exactamente a óptima) a cambio de velocidade de converxencia. Muitos destes métodos inspíranse na forma en que a natureza atopa solucións óptimas, por exemplo:

- Métodos evolutivos, como os algoritmos xenéticos.
- Algoritmos inspirados no comportamento animal, como enxames (“swarm optimization”), formigas (“ant colony optimization”), etc.
- “Simulated annealing”, ou recocido simulado.

É frecuente combinar unha heurística cun método de optimización local, o que a veces se denomina metaheurística. Tanto o multi-start como as metaheurísticas son métodos híbridos, pois combinan estratexias locais (para acelerar a converxencia no entorno dun óptimo) e globais (para procurar explorar todo o espazo de parámetros, evitando ficar atrapado nun óptimo local).

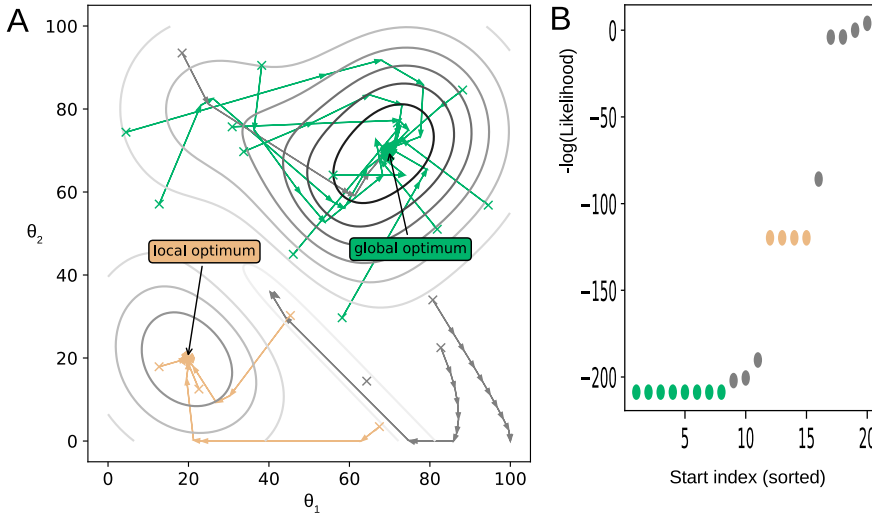


Figura 4.6: Optimización de parámetros. (A) A optimización multi-start consta de múltiples procuras locais, lanzadas desde diferentes vectores iniciais distribuídos no espazo de parámetros. En problemas con moitos óptimos locais poden precisarse moitas procuras para encontrar o óptimo global. Na figura, as traxectorias seguidas durante a optimización sinálanse con frechas, e os puntos iniciais con “x”. As curvas de nivel indican valores da función de custo, con liñas máis escuras canto mellor (é dicir, menor) sexa o valor. (B) Un “gráfico de cascada”, onde a existencia de (múltiples) chairas denota converxencia, xa que indica que a mesma solución é encontrada de forma repetida desde diferentes puntos iniciais. A súa ausencia pode solucionarse lanzando procuras adicionais, seleccionando outros puntos iniciais, ou usando métodos de optimización diferentes.

4.3.3. Definición do problema de optimización.

Ademais de escoller un método de optimización é preciso indicar diversas opcións, como:

- rangos de valores admisíbeis para os parámetros,
- escolla de puntos iniciais / forma de samplear aleatoriamente o espazo

de parámetros,

- algún(s) criterio(s) de terminación do algoritmo:
 - por tempo máximo de computación,
 - por número máximo de avaliacións da función obxectivo,
 - tolerancias numéricas para a terminación: se se realiza unha procura local, esta tolerancia especifica cando se considera que xa non se está a mellorar a solución.

Un exemplo do paso 3: modelo β IG. O seguinte código define o en Matlab o problema de optimización dos parámetros deste modelo, e lanza unha optimización coa toolbox MEIGO:

```
clear all

% Engadir directorios ao path de Matlab:
addpath(genpath(pwd));
addpath(genpath('K:\example_folder\MEIGO_toolbox'));
addpath(genpath('K:\example_folder\model_data_files'));

% Vector inicial de parámetros (guess) e rango de valores admisíbel (escala logarítmica):
u0=0.03; C=0.001; Si=0.0005; p=0.03; alpha=7.85; gamma=0.3; mup=0.000015; mum=0.000017;
init_guess_p = [u0,C,Si,o,alpha,gamma,mup,mum];
problem.x_0 = log10(init_guess_p);
problem.x_L = problem.x_0-2; % Límite inferior dos valores dos parámetros
problem.x_U = problem.x_0+2; % Límite superior dos valores dos parámetros

% Función obxectivo a minimizar:
problem.f = 'fun_obx_log_likelihood_BIG';

% Algoritmo de optimización:
optim_algor = 'multistart'; % multi-start de procuras locais

% Opcións do optimizador:
opts.local.solver = 'fmincon'; % Método de optimización local escollido
opts.ndiverse = 500; % nº de vectores iniciais (aleatorios)
opts.maxtime = 2*3600; % Límite de tempo
opts.maxeval = 1e6; % Nº máx. de avaliacións da f. obx.

% Chamar ao algoritmo
Results = MEIGO(problem,opts,optim_algorithm);

% Gardar os resultados:
save('results_eSS_alpha_pinene')
```

4.4. PASO 4: análise da bondade do axuste

Unha vez concluída a optimización, sempre hai que comprobar a bondade do axuste aos datos mediante inspección visual. Adicionalmente, pódense usar métricas cuantitativas para este fin. Se o axuste non é bo, isto pode indicar que a optimización non tivo éxito ou que o modelo é inadecuado. Dependendo do caso, podemos modificar o algoritmo de optimización ou o modelo.

Métricas cuantitativas

- *RMSE e NRMSE.* Se se coñece o nivel de ruído, a raíz cadrada do erro cadrado medio (RMSE nas súas siglas en inglés, “root mean square error”), proporciona unha métrica da bondade do axuste:

$$\text{RMSE} = \sqrt{\frac{\sum_{j=1}^{n_y} \sum_{k=1}^{n_t} (y_j^m(t_k) - y_j(t_k, \theta))^2}{n_y \cdot n_t}}$$

O erro cadrado medio normalizado de raíz (NRMSE) obtense (polo xeral, aínda que hai outras normalizacións posíbeis) dividindo o RMSE polo rango das medidas, e adoita ser máis útil xa que permite unha comparación directa entre diferentes observábeis.

- *chi-cadrado (χ^2).* Para avaliar se o valor da función obxectivo alcanzado é plausíbel, pódese comparar co valor esperado. Para certos problemas (de regresión linear, con ruído de medida normalmente distribuído e desviación estándar coñecida), a suma de residuos cadrados segue unha distribución chi-cadrado con $n_{\mathcal{D}} - n_{\theta}$ graos de liberdade, onde $n_{\mathcal{D}} = n_y \cdot n_t$ é o número de datos e n_{θ} é o número de parámetros estimados. Isto tamén se usa a miúdo como aproximación para modelos en EDOs.

Overfitting. Pode acontecer que o valor obtido da función obxectivo sexa moito menor do esperado. Este é un sinal de sobreaxuste ou *overfitting*, o que significa que non só se axusta o sinal dos datos de medición, senón tamén o ruído. Isto é unha posíbel causa de predicións incorrectas, e debe evitarse.

Validación. Un xeito de controlar o overfitting é realizar unha “validación cruzada” (*cross-validation*). Para iso, é preciso reservar no PASO 3 un subconxunto dos datos experimentais, que non usamos na optimización. Despois, utilizaremos o modelo calibrado para realizar predicións acerca deste último subconxunto de datos. Se o modelo consegue un bo axuste no primeiro subconxunto, pero non no segundo (é dicir, non consegue xeneralizar as predicións para condicións experimentais coas que non foi adestrado), significa que existe *overfitting*. Hai moitas formas de repartir os datos orixinais entre subconxuntos para estimación e para validación. Por exemplo, é común reservar o 30 %–40 % dos datos para a fase de validación, pero a elección depende en gran parte do problema.

Exemplo do paso 4: a ruta de sinalización Akt. No artigo <https://doi.org/10.1093/bib/bbab387> (sección “Examples”, subsección “Akt pathway model”) móstranse os detalles deste paso.

1. Por inspección visual, advertimos que o resultado da optimización non reproduce ben os datos.
2. Para conseguir un mellor axuste, primeiro probamos a repetir as estimacións co mesmo método de optimización, L-BFGS-B, pero diferentes settings. Non o conseguimos.
3. Probamos cun método diferente, Fides, que consegue mellorar a calidade das solucións obtidas a cambio dun maior custo computacional. Encontramos así unha solución de calidade similar á reportada na publicación orixinal do modelo.

4.5. PASO 5: cuantificación da incerteza dos parámetros

O análise de identificabilidade práctica consiste en cuantificar a incerteza existente nas estimacións dos parámetros. O obxectivo é obter intervalos de confianza para os valores dos parámetros. Hai varios tipos de métodos que se poden usar para este fin.

PASO 5.1. Analizar a identificabilidade práctica con algún dos métodos descritos nos recadros seguintes. Se as incertezas son grandes, proceder ao PASO 5.2. Se son aceptábeis, proceder ao PASO 6.

Análise de identificabilidade práctica coa FIM

Para n_t medidas, a Matriz de información de Fisher (FIM) calcúlase como:

$$\sum_{j=1}^{n_y} \sum_{i=1}^{n_t} \left(\frac{\partial y_j(t_i)}{\partial \theta} \right) W^{(i)} \left(\frac{\partial y_j(t_i)}{\partial \theta} \right)^T \quad (4.5)$$

onde $\frac{\partial y(t_i)}{\partial \theta}$ son as sensibilidade das saídas do modelo respecto dos parámetros, e $W^{(i)}$ é unha matriz diagonal con $W_{jj}^{(i)} = 1/\sigma_j^2(t_i)$, onde $\sigma_j(t_i)$ é a desviación típica.

Segundo o teorema de Cramér-Rao [12], se $\hat{\theta}$ é unha estimación non nesgada de θ (é dicir, $E(\hat{\theta}) = \bar{\theta}$), a inversa da FIM proporciona un límite inferior para a matriz de covarianza:

$$\text{Cov}(\hat{\theta}) \geq \text{FIM}^{-1}(\hat{\theta})$$

Á súa vez, a matriz de covarianza proporciona información sobre a variabilidade dos parámetros – individualmente e en pares – para diferentes realización dos datos experimentais:

$$\text{Cov} = E \left[\left(\hat{\theta} - \bar{\theta} \right) \left(\hat{\theta} - \bar{\theta} \right)^T \right] \begin{bmatrix} \sigma^2(\hat{\theta}_1) & \dots & \text{cov}(\hat{\theta}_1 \hat{\theta}_{n_\theta}) \\ \vdots & \ddots & \vdots \\ \text{cov}(\hat{\theta}_{n_\theta} \hat{\theta}_1) & \dots & \sigma^2(\hat{\theta}_{n_\theta}) \end{bmatrix} \quad (4.6)$$

Os intervalos de confianza calculados coa FIM son sempre simétricos e poden ser excesivamente optimistas en presenza de nonlinearidades, xa que se basean nunha linearización do modelo. Existen alternativas máis exactas, aínda que con maior custo computacional:

Análise de identificabilidade práctica con métodos de sampleado

- **Frecuentistas.** Estes métodos baséanse en xerar datos artificiais, con diferente ruído, e usalos para resolver o problema de estimación de parámetros moitas veces. Obtense así unha nube de solucións, da que se extrae información estatística como intervalos de confianza. Hai diversas variantes desta idea. Na súa definición clásica, chamada *bootstrap* [18], obtéñense diferentes conxuntos de datos ($S_1, S_2, \dots$) mediante a mostraxe aleatoria (sampleado) con reempazo do conxunto orixinal, S . Alternativamente pódese resolver o mesmo problema de estimación varias veces, cos mesmos datos pero comezando desde diferentes puntos iniciais (de forma análoga á optimización “multi-start”) [3]. Noutra técnica de resampleado, chamada *jackknife*, cada mostra quítase do conxunto de datos unha vez [17].
- **Bayesianos.** Os enfoques bayesianos consideran os parámetros como variábeis estocásticas cunha certa distribución previa (*prior*), e calculan a distribución de probabilidade *posterior* tendo en conta os datos experimentais. Isto faise mediante sampleado por Markov chain Monte Carlo (MCMC), o que ten un alto custo computacional para modelos grandes [34].

Perfil de probabilidades (PL). A idea deste método é a seguinte: cada parámetro θ_i fíxase a un valor lixeiramente diferente do óptimo, $\theta_i = \theta_i^{\text{opt}} + \Delta_k$, e re-estímanse os restantes parámetros. Facendo o mesmo para varios valores de Δ_k ($k = 1, 2, \dots$), obtemos unha serie de valores (perfil) da función de custo. Se o parámetro é identificábel, os demais parámetros non serán capaces de compensar a súa desviación do óptimo, e a optimización resultará nun valor superior da función obxectivo, polo que o perfil terá un mínimo no óptimo. Pola contra, se non é identificábel, a optimización dará o mesmo valor, obtendo un perfil plano. Se hai falta de identificabilidade práctica, o perfil terá un mínimo con pouca curvatura, polo que o intervalo de confianza será grande. A figura 4.7 ilustra as casuísticas.

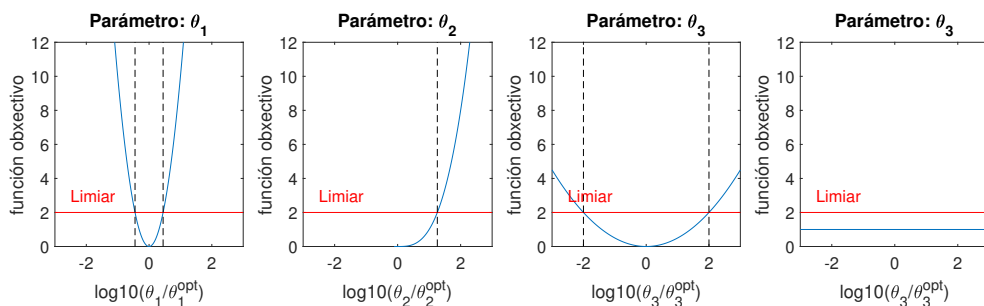


Figura 4.7: Perfil de probabilidade. A figura mostra catro casuísticas diferentes. En todas elas, o perfil de probabilidade é a liña azul, e o limiar do valor de confianza a liña vermella. O intervalo de confianza do parámetro é o limitado polas liñas verticais negras discontinuas. De esquerda a dereita: o rango de valores probábeis do parámetro θ_1 é finito e relativamente estreito, polo que é practicamente identificábel. Pola contra, θ_2 e θ_3 son practicamente non identificábeis. O perfil de θ_4 é plano, polo que é estruturalmente non identificábel. Cómpre sinalar que estas gráficas foron xeradas artificialmente para ilustrar a metodoloxía. Na práctica, non sempre é posíbel obter resultados tan claros.

PASO 5.2. En caso de incertezas elevadas:

1. Se é posíbel realizar novos experimentos $\rightarrow$ engadir máis datos experimentais. Idealmente, os experimentos adicionais deberían ser deseñados de forma óptima, para maximizar a información que proporcionan. O deseño óptimo de experimentos descríbese no 4.7.
2. Se non resulta práctico realizar novos experimentos $\rightarrow$ considerar se é posíbel reparametrizar o modelo, reducindo o número dos seus parámetros mantendo a interpretación biolóxica.
3. Se non se pode facer (1) nin (2) $\rightarrow$ usar coñecemento previo sobre os valores dos parámetros, ou sobre o rango de valores, en repositorios públicos.

A continuación, volver ao PASO 3.

No caso en que non sexa posíbel reducir as incertezas nos valores dos seus parámetros, é posíbel que o modelo aínda teña utilidade. Hai que comprobar se é capaz de realizar predicións con precisión aceptábel. Isto analízase no PASO 6.

Exemplo do paso 5: a ruta de sinalización Akt

PASO 5.1: Realizamos a análise de identificabilidade práctica co método MCMC. Na Fig. 6.D do artigo <https://doi.org/10.1093/bib/bbab387> vemos que os intervalos de confianza de algúns parámetros son moi amplos, é dicir, que son practicamente non identificábeis.

PASO 5.2: neste caso non é posíbel reparametrizar o modelo. Tampouco podemos realizar experimentos adicionais, polo que pasamos ao PASO 6.

4.6. PASO 6: cuantificación da incerteza nas predicións

Neste paso calcúlanse intervalos de confianza nas predicións temporais producidas polo modelo. A idea ilústrase na Figura 4.8.

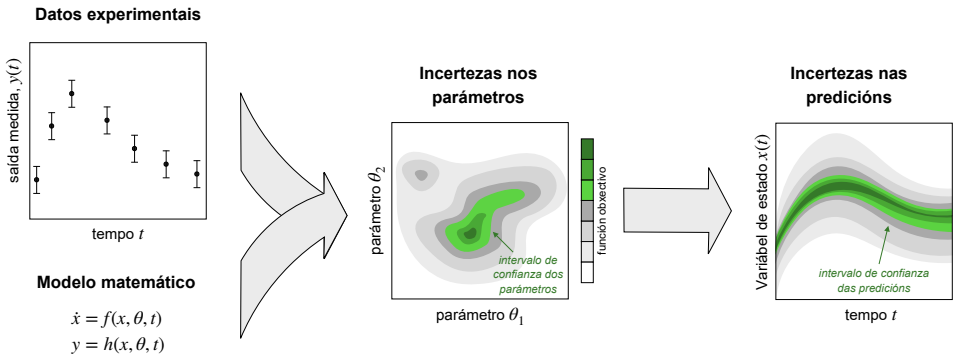


Figura 4.8: Incerteza das predicións. As limitacións nos datos experimentais levan a estimacións incertas dos parámetros, que á súa vez se trasladan ás predicións.

Para isto poden usarse variantes dos métodos usados no PASO 5, modificados para calcular intervalos de confianza dos estados en lugar de dos parámetros [56]. O custo computacional é superior para os estados que para os parámetros, polo que métodos como o PL poden non ser aplicábeis. Unha alternativa é o modelado por conxuntos (ensemble modelling).

Exemplo do paso 6: a ruta de sinalización Akt. Os intervalos de confianza obtidos a partir da mostraxe de MCMC indican que as incertezas nas traxectorias observables son razoablemente baixas. Non obstante, as traxectorias dos estados teñen maiores incertezas, o que fai que este modelo non sexa adecuado para facer predicións que involucren estes estados.

4.7. Deseño de experimentos

Os deseños experimentais alternativos poden aumentar o contido de información dos datos, mellorando así as estimacións dos parámetros. Se se busca o mellor experimento posible mediante optimización, o procedemento chámase deseño experimental óptimo (OED nas súas siglas en inglés). O OED fórmulase como un problema de optimización dinámica, no que as variábeis que se poden modificar son as condicións experimentais (perturbacións permitidas, observábeis medidos, número de experimentos, duración dos experimentos, número e localización dos tempos de mostraxe...), e o obxectivo a maximizar é algunha medida do contido de información dos datos. As restricións son a dinámica do sistema e as limitacións experimentais.

O OED pódese realizar con varios propósitos, como diminuír a incerteza nas estimacións dos parámetros, diminuír a incerteza nas predicións do modelo, ou discriminar entre varios modelos alternativos. A función obxectivo a optimizar depende deste obxectivo final. Co propósito de diminuír a incerteza dos parámetros é común empregar funcións obxectivo baseadas na FIM. As opcións típicas son os criterios D-óptimo e E-óptimo, que maximizan o determinante da FIM e o seu valor propio mínimo, respectivamente. O criterio D minimiza a media xeométrica dos erros dos parámetros, mentres que o criterio E minimiza o maior erro.

4.8. Selección de modelos

Ás veces a estrutura do modelo non está totalmente definida, existindo varias alternativas correspondentes a diferentes hipóteses biolóxicas.

Unha forma sinxela de comparar modelos é ver se a calidade dos seus axustes difire dun xeito estatisticamente significativo. Isto coñécese como proba de razón de verosimilitude. Un modelo con máis parámetros é máis flexíbel, polo que é máis fácil conseguir un mellor axuste. Non obstante, un modelo sobreparametrizado pode presentar sobreaxuste (overfitting), o que non é desexable. Para ter isto en conta, ao seleccionar un modelo hai que buscar un equilibrio entre a complexidade do modelo e a bondade de axuste. Medidas como o criterio de información de Akaike (AIC) [8] e o criterio de información bayesiano (BIC) [57] teñen en conta a bondade do axuste e unha penalización baseada no número de parámetros estimados, polo que poden ser usados para cuantificar esta compensación. Unha definición do AIC é:

$$\text{AIC} = 2 \cdot n_{\theta} - 2 \cdot \ln(J(\theta)) \quad (4.7)$$

onde n_{θ} é o número de parámetros a estimar, e $J(\theta)$ o valor óptimo da función de custo. Existen outras versións, por exemplo para casos con poucos datos. Pola súa parte, o BIC é:

$$\text{BIC} = \ln(n_D) \cdot n_{\theta} - 2 \cdot \ln(J(\theta)) \quad (4.8)$$

onde n_D é o número total de datos.

Se é factible realizar novos experimentos, pódense deseñar especificamente para a discriminación de modelos, aplicando un procedemento de deseño experimental (*OED*) que busque maximizar a diferenza entre os resultados dos modelos candidatos.

Cando non sexa posíbel discriminar entre os modelos candidatos, unha posibilidade é telos en conta todos. Isto pódese facer construíndo un conxunto de modelos (*ensemble modelling*) que conteña non só modelos con diferentes vectores de parámetros, senón tamén con diferentes ecuacións.

Se varias modelos con estruturas diferentes poden producir a mesma saída, dise que son estruturalmente *indistinguíbeis* (de xeito similar a como un parámetro se chama estruturalmente non identificábel se pode ter un número infinito de valores que conducen á mesma saída do modelo).

4.9. Recursos software

A Táboa 4.1 mostra algunhas ferramentas dispoñíbeis para levar a cabo diferentes tarefas relacionadas coa calibración de modelos.

Nome	Pasos	Website	Entorno
STRIKE-GOLDD	1	https://github.com/afvillaverde/strike-goldd	MATLAB
SIAN (web)	1	https://maple.cloud/app/5710317752942592/SIAN	web
SIAN	1	https://github.com/pogudingleb/SIAN	Maple
IdentifiabilityAnalysis	1	http://www.fcc.chalmers.se/software/other-software/identifiabilityanalysis/	Mathematica
DifferentialEquations.jl	Simulación	https://github.com/SciML/DifferentialEquations.jl	Julia
AMICI	Simulación	https://github.com/AMICI-dev/AMICI	Python
COPASI	Simulación, 3,5	http://copasi.org/	stand-alone
MEIGO	3	https://bitbucket.org/jrbanga_/meigo64/src/master/	Matlab
AMIGO2	3, 5, OED	https://sites.google.com/site/amigo2toolbox/home	Matlab
pyPESTO	3, 5, 6	https://github.com/ICB-DCM/pyPESTO	Python
Data2Dynamics	3, 5, 6, OED	http://www.data2dynamics.org	MATLAB

Cadro 4.1: Algúns recursos software dispoñíbeis para a calibración de modelos dinámicos.

No referente ao hardware necesario, cómpre ter en conta que moitos algoritmos usados na calibración son paralelizábeis. Por tanto, se se dispón dun clúster de computación ou unha computadora multi-core, a *paralelización* permite diminuír os tempos de cálculo.

Cuestións e problemas do Tema 4

C4.1. Definicións de identificabilidade estrutural e práctica.

C4.2. Determinar se o modelo dado polas seguintes ecuacións ten parámetros non identificables. En caso afirmativo, dicir cales.

(NOTA: usamos notación estándar, é dicir, x_* son variábeis de estado, y é a saída (medida), u é unha entrada coñecida, e os k_* son parámetros descoñecidos.)

$$\begin{aligned}\dot{x}_1 &= u - (k_1 + k_2)x_1, \\ \dot{x}_2 &= (k_1 + k_2)x_1 - (k_3 + k_5)x_2, \\ \dot{x}_3 &= k_2x_1 + k_3x_2 - k_4x_3, \\ y &= x_2.\end{aligned}$$

C4.3. As seguintes ecuacións representan un modelo de infección de gripe A, onde T son as células non infectadas, I_1 as células expostas (nas cales a infección é latente), I_2 as células infectadas de forma efectiva, e V é a carga viral [39]. As restantes variábeis que aparecen no modelo son parámetros descoñecidos. Supoñamos que o único estado medido é a carga viral. Cales variábeis de estado son non observábeis? E cales parámetros son non-identificábeis?

$$\dot{T} = -\beta TV, \tag{4.9}$$

$$\dot{I}_1 = \beta TV - kI_1, \tag{4.10}$$

$$\dot{I}_2 = kI_1 - \delta I_2, \tag{4.11}$$

$$\dot{V} = pI_2 - cV. \tag{4.12}$$

C4.4. Que podemos facer se un modelo presenta falta de observabilidade e/ou identificabilidade estrutural?

C4.5. Fórmula da función de verosimilitude logarítmica (log-likelihood). Diferenza coa de mínimos cadrados ponderada.

- C4.6.** Tipos de algoritmos de optimización usados para estimación de parámetros. Vantaxes e inconvenientes.
- C4.7.** Que métricas se poden usar para cuantificar a bondade de axuste cos datos?
- C4.8.** Fórmula da matriz de información de Fisher. Como se pode usar para cuantificar incertezas nos parámetros? E nos estados?
- C4.9.** Que métodos existen para cuantificar a incerteza nas estimacións dos parámetros, e que vantaxes e inconvenientes presentan?
- C4.10.** Que métodos existen para cuantificar a incerteza nas predicións, e que vantaxes e inconvenientes presentan?
- C4.11.** Como podemos formular un problema de deseño óptimo de experimentos?
- C4.12.** Criterios para a selección de modelos.

5

Dinámica

“This model will be a simplification and an idealization, and consequently a falsification. It is to be hoped that the features retained for discussion are those of greatest importance.”

– Alan Turing

Unha vez construído un modelo do noso sistema biolóxico, podemos usalo para simular e analizar o seu comportamento. Se o modelo foi desenvolvido correctamente, será capaz de reproducir os aspectos relevantes da dinámica do sistema.

NESTE tema caracterízanse algúns comportamentos de particular interese nos sistemas biomédicos.

Ao analizar o comportamento dos sistemas dinámicos é común diferenciar entre transitorio e estacionario. Cando un sistema dinámico é sometido a unha perturbación ou entrada exterior, prodúcese inicialmente unha resposta *transitoria*, que ao cabo dun tempo dá lugar a un estado *estacionario*. Este estado final pode ser idéntico ou diferente ao inicial, dependendo de como responda o sistema a perturbacións.

En ausencia de perturbacións, os sistemas biolóxicos encóntranse con frecuencia nun punto de *equilibrio*, asociado a un estado estacionario. As características dese estado conforman o fenotipo do sistema. Chámaselle *fenotipo* a

calquera característica detectábel dun organismo, determinada pola interacción entre o xenotipo e o medio. Ditas características poden ser estruturais ou relativas ao comportamento.

Comunmente asóciase a idea de fenotipo a unha característica estática, ligada á existencia dun único punto de equilibrio e dun estado estacionario. Pero un sistema tamén pode ter múltiples puntos de equilibrio; no caso máis sinxelo podemos ter biestabilidade, na que hai dous puntos de equilibrio. Dependendo das condicións iniciais e das entradas externas, un determinado sistema pode acabar preto dun ou doutro punto de equilibrio, dando lugar a dous fenotipos distintos. Un modelo dinámico con biestabilidade (ou multiestabilidade) representa unha forma de modelar a memoria nun sistema: a célula ou organismo lembra a súa historia en virtude do punto de equilibrio ao que converxeu. Tamén poden existir comportamentos (fenotipos) máis complexos, lonxe do réxime estacionario, nos que existan patróns temporais como ciclos límite (órbitas periódicas) ou respostas de entrada/saída afastadas do equilibrio.

Outra característica importante nun biosistema é a *robustez* do seu comportamento en condicións variábeis, que debe combinarse coa capacidade de responder a determinados estímulos externos.

Para estudar estes aspectos poderemos traballar con modelos nonlineares, pero tamén usaremos ás veces modelos lineares, que son máis fáciles de analizar, ou versións linearizadas de modelos nonlineares.

5.1. Sistemas lineares e linearización

Como se viu ao longo deste libro, os sistemas biomédicos son con carácter xeral *nonlineares*. Un sistema nonlinear descrito por EDOs ten a forma:

$$\mathcal{M} = \begin{cases} \dot{x} = f(x, \theta, u) \\ y = h(x, \theta, u) \end{cases} \quad (5.1)$$

$$(5.2)$$

Neste capítulo non modelamos de forma explícita a existencia de entradas descoñecidas, polo que, a diferenza de (1.1), o modelo non inclúe w .

Os sistemas *lineares* son un caso particular dos nonlineares. Pódense re-

presentar como:

$$\mathcal{M} = \begin{cases} \dot{x} = A \cdot x + B \cdot u \\ y = C \cdot x + D \cdot u \end{cases} \quad (5.3)$$

$$(5.4)$$

As ecuacións (5.3)–(5.4) usan a representación en variábeis de estado, xa vista no Tema 1 (1.3), e da que volveremos falar en seguintes subseccións.

Para a análise de sistemas lineares existen técnicas máis desenvolvidas que para os nonlineares. Por iso ás veces é útil aproximar un sistema nonlineal por un linear (linearización). Nalgúns casos, a linearización dun sistema nonlineal nas proximidades do equilibrio proporciona unha boa aproximación do seu comportamento. Neste apartado imos ver como linearizar un modelo nonlineal.

Supoñamos que un sistema posúe un estado estacionario $\bar{x}$, e que nel $u = 0$ por simplicidade. Podemos escribir $x(t) = \bar{x} + \hat{x}(t)$, onde $\hat{x}(t)$ é a desviación respecto do estado estacionario. A súa ecuación dinámica é:

$$\dot{x}(t) = f(\bar{x} + \hat{x}(t)) = \frac{d}{dt}(\bar{x} + \hat{x}(t)) = \frac{d}{dt}\hat{x}(t). \quad (5.5)$$

A variación temporal de cada estado é

$$\frac{d}{dt}\hat{x}_i = f_i(\bar{x}_1 + \hat{x}_1, \dots, \bar{x}_n + \hat{x}_n).$$

Omitindo a dependencia de t por simplicidade, a expansión en series de Taylor dá:

$$\frac{d}{dt}\hat{x}_i = f_i(\bar{x}_1, \dots, \bar{x}_n) + \sum_{j=1}^n \frac{\partial f_i}{\partial x_j} \hat{x}_j + \frac{1}{2} \sum_{j=1}^n \sum_{k=1}^n \frac{\partial^2 f_i}{\partial x_j \partial x_k} \hat{x}_j \hat{x}_k + \dots \quad (5.6)$$

Desprezando os termos de orde 2 e superior, e tendo en conta que no estado estacionario $f_i(\bar{x}_1, \dots, \bar{x}_n) = 0$, obtemos a seguinte aproximación linear:

$$\frac{d}{dt}\hat{x}_i = \sum_{j=1}^n \frac{\partial f_i}{\partial x_j} \hat{x}_j = \sum_{j=1}^n a_{ij} \hat{x}_j. \quad (5.7)$$

onde os coeficientes a_{ij} son as derivadas parciais $\frac{\partial f_i}{\partial x_j}$ avaliadas no estado estacionario, sendo por tanto constantes. Estes coeficientes forman o *Jacobiano*, que é a seguinte matriz:

$$J = \{a_{ij}\} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \cdots & \cdots & \ddots & \cdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix} \quad (5.8)$$

Por tanto, a ecuación (5.7) pódese escribir en forma matricial como:

$$\frac{d}{dt}\hat{x} = \dot{\hat{x}} = J \cdot \hat{x}. \quad (5.9)$$

Lembremos que, para sistemas lineares, $\dot{x} = A \cdot x + B \cdot u$. Comparando esta expresión con (5.9) vemos que o Jacobiano coincide coa matriz A , $J = A$ (lembremos que $u = 0$).

Para o caso máis xeral de sistemas nonlineares, o Jacobiano xoga un papel similar ao da matriz A .

5.2. Equilibrio e estabilidade

Se o estado estacionario dun sistema linearizado é estábel, o estado estacionario do sistema nonlinear tamén é estábel. En caso contrario, é inestábel. A análise do sistema linear é máis sinxela que a do nonlinear.

Equilibrio(s). Consideremos un sistema dinámico que, por simplicidade, non ten entradas (u) nin parámetros descoñecidos (θ), polo que a súa dinámica se pode representar por $\dot{x} = f(x)$. O sistema ten un *punto de equilibrio* en x_e se $f(x_e) = 0$. Se non existe ningún tipo de entrada externa (incluíndo perturbacións), unha vez que o sistema se encontra no punto de equilibrio nun instante t_e permanece nel para todo $t > t_e$. Un sistema pode ter 0, 1, ou máis puntos de equilibrio.

Estabilidade: definicións. A estabilidade dun punto de equilibrio determina se as solucións próximas a el permanecen próximas, se aproximan a el, ou se afastan. Para cada unha destas nocións existe a correspondente definición matemática, como veremos a continuación.

- *Estabilidade no sentido de Lyapunov.* Un punto de equilibrio x_e é estábel no sentido de Lyapunov se as solucións do sistema que parten da súa proximidade se manteñen próximas a el. Formalmente, x_e é estábel se:

$$\|x(t_0) - x_e\| < \delta \Rightarrow \|x(t) - x_e\| < \epsilon \quad \forall t > t_0$$

Nótese que esta definición non implica que as solucións se vaian aproximando ao punto de equilibrio, senón só que non se afastan indefinidamente del.

- *Estabilidade asíntótica.* É unha propiedade máis forte que a estabilidade de Lyapunov. Un punto de equilibrio x_e é asíntoticamente estábel se é estábel no sentido de Lyapunov, e ademais $x(t) \rightarrow x_e$ para $t \rightarrow \infty$, para $x(t_0)$ suficientemente próximos a x_e (estabilidade asíntótica *local*) ou ben para calquera $x(t_0)$ (estabilidade asíntótica *global*).
- *Inestabilidade.* Un punto de equilibrio x_e é inestábel se, dado un ϵ , non existe ningún $\delta > 0$ tal que, se $\|x(t_0) - x_e\| < \delta$, $\|x(t) - x_e\| < \epsilon \quad \forall t > t_0$.

Exemplo: tres sistemas con diferente estabilidade. A Figura 5.1 mostra a evolución temporal de tres sistemas que ilustran diferentes tipos de comportamento estábel.

5.2.1. Caracterización matemática da estabilidade

Caracterización das solucións de EDOs lineares. No capítulo 3 vimos como simular EDOs nonlineares. Como en xeral non é posíbel resolvelas analiticamente, a súa integración faise numericamente, existindo varios métodos para ese fin. En contraste, no caso de EDOs lineares si é posíbel achar unha solución analítica, para o cal é útil a transformada de Laplace, como xa vimos. Nos exemplos vistos até agora, a dependencia temporal das solucións viña dada por termos do tipo $e^{a \cdot t}$. De feito, este resultado é xeneralizábel. O

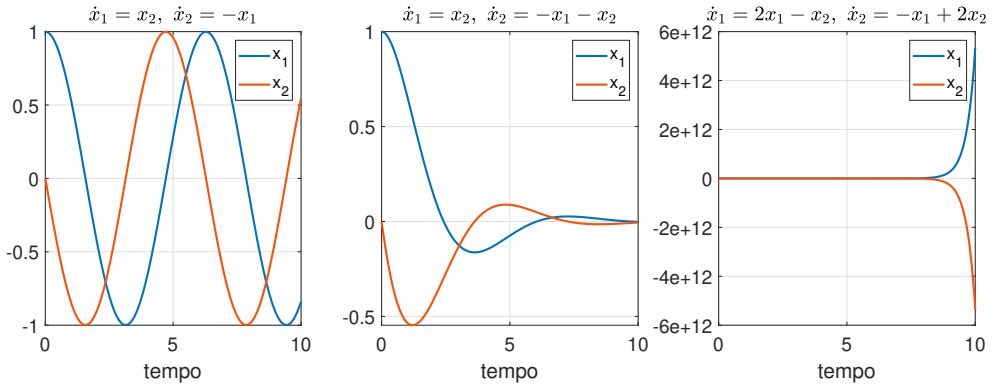


Figura 5.1: Estabilidade: tres exemplos. Os tres paneis mostran a evolución temporal dos seguintes sistemas desde o mesmo punto inicial ($x_1 = 1, x_2 = 0$):

- (1) Esquerda: $\dot{x}_1 = x_2, \dot{x}_2 = -x_1$
- (2) Centro: $\dot{x}_1 = x_2, \dot{x}_2 = -x_1 - x_2$
- (3) Dereita: $\dot{x}_1 = 2x_1 - x_2, \dot{x}_2 = -x_1 + 2x_2$

Os tres sistemas teñen un punto de equilibrio en $x_1 = 0, x_2 = 0$, pero a súa estabilidade é diferente: o primeiro sistema mantense no entorno do equilibrio (é estábel no sentido de Lyapunov); o segundo converge ao punto de equilibrio (é asintoticamente estábel); e o terceiro escapa cara o infinito (é inestábel).

feito de que as solucións sexan desta forma ten consecuencias para a estabilidade. Neste apartado imos ver cales son, e o papel que xogan os coeficientes tipo a que aparecen no expoñente. Consideremos en primeiro lugar sistemas homoxéneos (é dicir, nos que $u = 0$) dun só estado ($n = 1$). Para eles, temos simplemente

$$\dot{x}_1 = a_{11}x_1,$$

cuxa solución é

$$x_1(t) = x_1(0)e^{a_{11}t} \quad (5.10)$$

Analogamente, para modelos con $n > 1$ estados a solución é:

$$\mathbf{x}(t) = \sum_{i=1}^n c_i \mathbf{b}^{(i)} e^{\lambda_i t} \quad (5.11)$$

onde os λ_i son os autovalores da matriz A , e b_i os seus correspondentes vectores propios. (Nótese que en 5.11 escribimos $\mathbf{x}$ e $\mathbf{b}$ en negrita, ao contrario do que facemos habitualmente, para explicitar que son vectores.) Os coeficientes c_i calcúlanse a partir das condicións iniciais, resolvendo o seguinte sistema de ecuacións:

$$\mathbf{x}(0) = \sum_{j=1}^n c_j \mathbf{b}^{(j)}.$$

Até agora asumimos, por simplificar a exposición, que o modelo non tiña entradas u , polo que o sistema de EDOs era homoxéneo. En caso de que haxa entradas constantes, o sistema $\dot{x} = A \cdot x + B \cdot u$ pode facerse homoxéneo mediante un cambio de coordenadas arredor do equilibrio $\bar{x}$, é dicir, $\hat{x} = x - \bar{x}$:

$$\frac{d}{dt} \hat{x} = \frac{d}{dt} x - \frac{d}{dt} \bar{x} = A \cdot x + B \cdot u - A \cdot \bar{x} - B \cdot u = A \cdot (x - \bar{x}) = A \cdot \hat{x}$$

Así, resolvemos para $\hat{x}$ da forma xa vista, e despois obtemos x sumando $\bar{x}$ á solución ($\bar{x} = -B \cdot u/A$, xa que $\dot{\bar{x}} = 0 = A \cdot \bar{x} + B \cdot u$).

Análise da estabilidade asintótica local. Pódese determinar a estabilidade dun sistema a partir dos autovalores do sistema linearizado. Para ver isto comezamos polo caso de sistemas de un só estado, $n = 1$. A súa evolución temporal ven dada pola ecuación (5.11), na que o coeficiente a_{11} coincide co único autovalor do sistema ($\lambda_1 = a_{11}$). Claramente, se $\lambda_1 < 0$, o termo $e^{a_{11}t}$ diminúe no tempo, co que $x_1(t) \xrightarrow{t \rightarrow \infty} x_1(0)$, é dicir, o estado volve á súa condición inicial (o equilibrio): o estado é asintoticamente estábel. É dicir:

- Se $\lambda_1 < 0$, o estado estacionario é asintoticamente estábel.
- Se $\lambda_1 > 0$, o estado é inestábel (xa que crece indefinidamente).
- Se $\lambda_1 = 0$, poderíamos pensar que existe estabilidade no sentido de Lyapunov; porén, en realidade neste caso non podemos determinar a estabilidade a partir do sistema linearizado, xa que esta aparente “estabilidade neutra” pode verse afectada polos termos de orde superior que foran desprezados.

Para sistemas de varios estados ($n > 1$), se todos os autovalores teñen parte real estritamente negativa, o sistema é estábel. Se algún autovalor ten parte real estritamente positiva, é inestábel. Nótese que un só autovalor positivo implica inestabilidade, aínda que existan varios autovalores negativos. Se un sistema ten autovalores complexos, a súa solución $x(t)$ conterá compoñentes oscilatorias. A súa estabilidade virá dada pola parte real dos seus autovalores. Nótese que a estabilidade dun sistema linear depende só da matriz A , a cal é función dos parámetros pero non do punto de equilibrio. Polo tanto, no caso linear podemos falar de estabilidade do sistema en lugar de estabilidade dun punto de equilibrio específico.

Cálculo dos autovalores dun sistema linear(izado)

Sexa un sistema $\dot{x} = A \cdot x$ (que pode resultar da linearización dun sistema nonlinear $\dot{x} = f(x)$), cuxo Jacobiano é (5.8). Os seus autovalores calcúlanse en función do número de estados n :

1. Se $n = 1$, o sistema ten un só autovalor, $\lambda = A$.
2. Se $n = 2$, o sistema ten dous autovalores, que se obteñen como solucións da seguinte ecuación:

$$\lambda^2 - \text{Tr}(A) \cdot \lambda + \text{Det}(A) = 0,$$

é dicir:

$$\lambda_{1,2} = \frac{1}{2} \left(\text{Tr}(A) \pm \sqrt{\text{Tr}(A)^2 - 4 \cdot \text{Det}(A)} \right), \quad (5.12)$$

onde $\text{Tr}(A) = a_{11} + a_{22}$ é a traza do Jacobiano, e $\text{Det}(A) = a_{11}a_{22} - a_{12}a_{21}$ é o seu determinante.

3. Se $n \geq 3$, pódese usar o criterio de Routh-Hurwitz.

Na práctica, a estabilidade pódese calcular de forma sinxela en Matlab co comando `isstable()`.

Exemplo: análise da estabilidade dun biosistema. O seguinte modelo describe a expresión dun xene con realimentación negativa, onde x_1 é concentración de mRNA e x_2 concentración de proteína:

$$\frac{d}{dt}x_1 = \frac{1}{1+x_2} - x_1, \quad \frac{d}{dt}x_2 = x_1 - x_2 \quad (5.13)$$

O Jacobiano arredor dun estado estacionario xenérico, $x_e = (x_{1e}, x_{2e})$, é:

$$A = \left. \frac{\partial f}{\partial x} \right|_{x_e} = \begin{bmatrix} -1 & -\frac{1}{(1+x_{2e})^2} \\ 1 & -1 \end{bmatrix} \quad (5.14)$$

Como $\text{Tr}(A) = -2 < 0$, e $\text{Det}(A) = 1 + \frac{1}{(1+x_{2e})^2} > 0$, os dous autovalores teñen parte real negativa, independentemente do valor de x_e . Por tanto, o punto de equilibrio x_e é asintoticamente estábel.

5.3. Bifurcacións

A localización dos puntos de equilibrio dun modelo, así como a súa estabilidade, dependen non só da forma das EDOs, senón dos valores dos parámetros θ . A análise de bifurcación determina como varía o comportamento dinámico a medida que cambian os parámetros. Se o comportamento varía cualitativamente para un vector de parámetros específico $\theta = \theta^*$, o sistema ten un punto de bifurcación en $\theta = \theta^*$. Por exemplo, un punto de equilibrio x_e pode pasar de estábel a inestábel, ou viceversa; ou o número de puntos de equilibrio pode cambiar.

5.3.1. Diagramas de fase

Os *diagramas de fase* son unha forma de visualizar bifurcacións. Ditos diagramas mostran a evolución temporal de dous estados dun sistema, cada un dos cales corresponde cun eixo (X e Y).

Consideremos como exemplo o sistema dado polas seguintes ecuacións:

$$\frac{d}{dt}x_1 = x_2, \quad \frac{d}{dt}x_2 = -k \cdot x_1 - \theta x_2. \quad (5.15)$$

con $k, \theta > 0$. En notación matricial, este sistema escríbese así:

$$\dot{X} = \begin{bmatrix} 0 & 1 \\ -k & \theta \end{bmatrix} X. \quad (5.16)$$

Este sistema ten un punto de equilibrio na orixe ($x_{1e} = 0, x_{2e} = 0$). Aplicando a fórmula (5.12), os seus autovalores son:

$$\lambda_{1,2} = \frac{1}{2} \left(-\theta \pm \sqrt{\theta^2 - 4k} \right).$$

Os autovalores poden tomar valores positivos ou negativos en función dos valores do parámetro θ .

A parte esquerda da figura 5.2 mostra os diagramas de fase do sistema dado por (5.15), para $k = 5$ e catro valores diferentes de θ . A parte dereita da figura 5.2 representa cada estado en función do tempo.

Se θ é positivo e ten valores altos (fila superior), os autovalores do sistema son reais negativos, polo que o sistema é estábel e a resposta temporal non é oscilatoria, senón sobreamortecida. Para valores de θ positivos pero de inferior magnitude (fila 2), os autovalores pasan a ser complexos con parte real negativa; o sistema segue a ser estábel pero a resposta é oscilatoria subamortecida. Se $\theta = 0$ (fila 3), o sistema é estábel no sentido de Lyapunov, cun comportamento puramente oscilatorio; e se $\theta < 0$ (fila inferior), o sistema vira inestábel e as traxectorias afástanse do punto de equilibrio.

5.3.2. Gráficas de bifurcación

As *gráficas de bifurcación* mostran como varía un punto de equilibrio x_e en función dun determinado parámetro θ . Ao igual que os diagramas de fase, calcúlanse seguindo estes pasos:

1. Calcular o(s) punto(s) de equilibrio dun sistema, igualando as súas EDOs a cero ($\dot{x}_i = 0$). O número de equilibrios pode ser función de algún parámetro.
2. Calcular o Jacobiano en torno ao(s) punto(s) de equilibrio.
3. Deducir a estabilidade a partir dos autovalores.

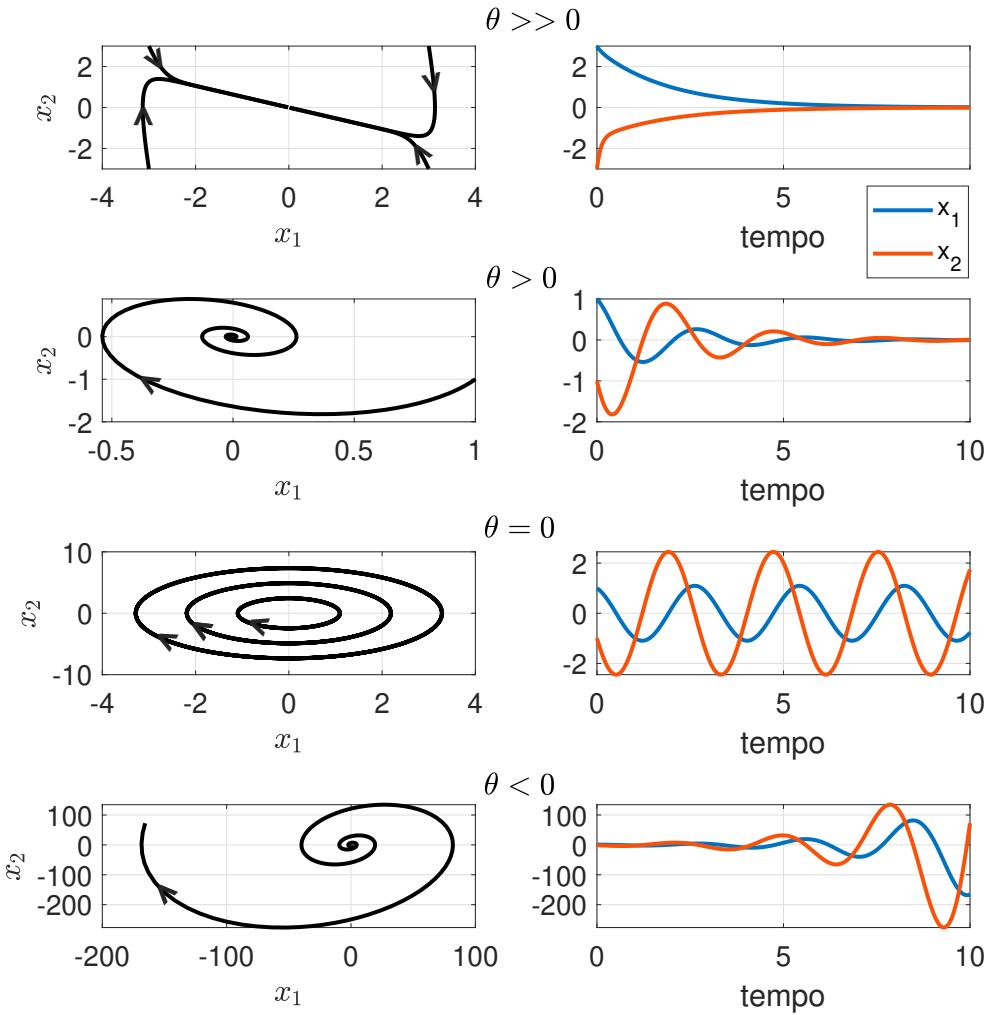


Figura 5.2: Diagramas de fase (columna da esquerda) e comportamento temporal (columna da esquerda) do sistema definido polas ecuacións (5.15) para varios valores do parámetro θ .

Unha vez obtida esta información, débúxase nunha gráfica a localización do(s) punto(s) de equilibrio (eixo vertical) en función dos valores do parámetro relevante (eixo horizontal), en trazo continuo ou discontinuo dependendo de se o equilibrio é estábel ou inestábel, respectivamente. Por exemplo, a gráfica de bifurcación do sistema dado por (5.15) e analizado na subsección seguinte é a representada no panel (A) da figura 5.3. Esta é a bifurcación máis sinxela: a localización do punto de equilibrio non varía, simplemente pasa de inestábel a estábel a partir dun certo valor do parámetro. En concreto, o seu equilibrio é sempre $x_{1e} = x_{2e} = 0$, e é inestábel para $\theta < 0$ e estábel para $\theta > 0$. Os restantes paneis da figura 5.3 (B–E) mostran outros catro tipos de bifurcacións máis complexas:

- *Bifurcación de sela-nó* (“saddle-point”). O punto de bifurcación separa un equilibrio estábel doutro inestábel. Este comportamento coñécese como *histérese*. A estabilidade para un valor do parámetro depende da dirección de cambio do seu valor. A figura 5.3 mostra un exemplo no que só hai un punto de bifurcación, e a transición é irreversíbel. Noutros casos, a curva ten forma de “S” e hai dous puntos de bifurcación que separan unha rexión central inestábel de dúas rexións estábeis. Neste último caso a transición é reversíbel e o sistema é *biestábel*; un exemplo típico sería un “switch” ou interruptor xenético.
- *Bifurcación transcritical*. Nesta bifurcación, dous puntos de equilibrio (un estábel e outro inestábel) aproxímanse a medida que varía θ , e na intersección intercambian a súa estabilidade.
- *Bifurcación tridente*. Ao chegar ao punto de bifurcación, un equilibrio estábel convértese en inestábel e aparecen dous novos equilibrios estábeis (biestabilidade).
- *Bifurcación de Hopf*. Un equilibrio estábel pasa a inestábel cando un par de autovalores complexos conjugados cruzan o eixo imaxinario, dando lugar a un ciclo límite (os ciclos límites veranse na Sec. 5.4).

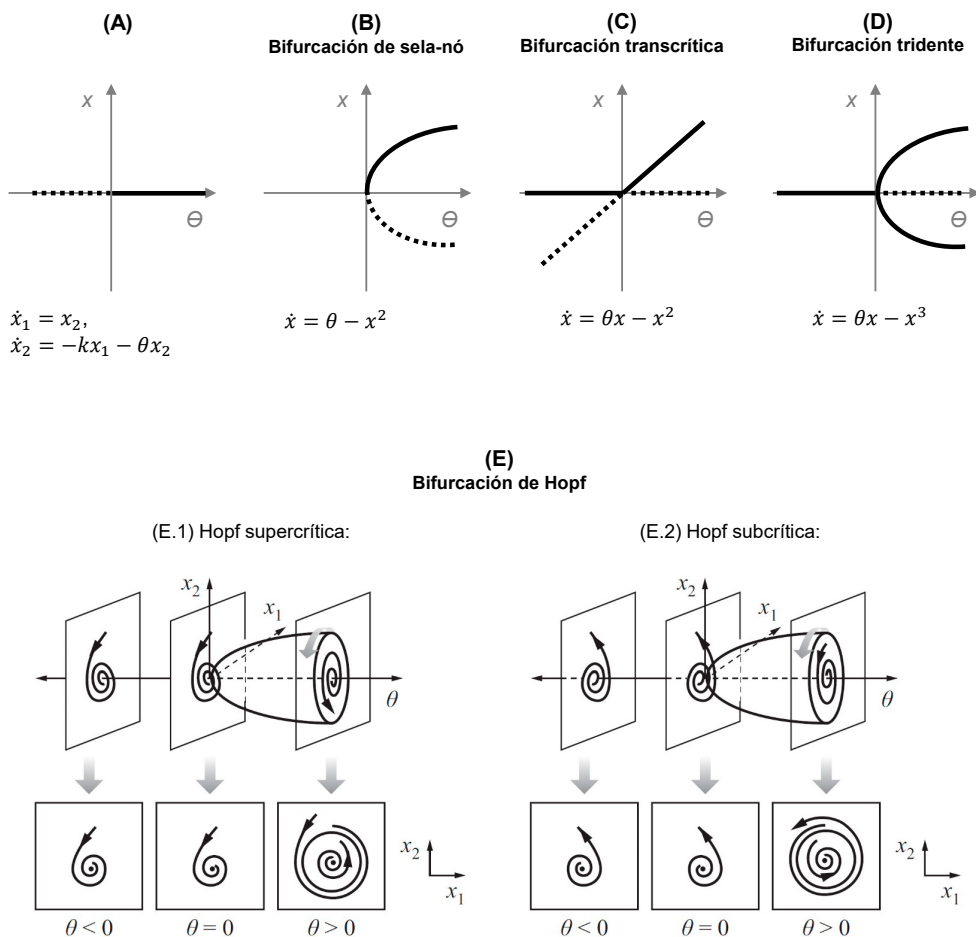


Figura 5.3: Tipos de bifurcacións. Os catro paneis superiores (A–D) mostran diferentes tipos de bifurcacións e as ecuacións correspondentes. As liñas continuas representan equilibrios estábeis, e as liñas a trazos inestábeis. Os paneis inferiores (E) mostran dous tipos de bifurcación de Hopf [15]: no caso supercrítico (E.1), a medida que aumenta θ aparece un ciclo límite estábel. No caso subcrítico (E.2) aparecen órbitas periódicas inestábeis.

5.4. Oscilacións

Moitos fenómenos biolóxicos presentan oscilacións. Un exemplo clásico son os ritmos circadianos, cun período igual á lonxitude do día, que afectan a moitos procesos. Nese caso, existen sinais externos periódicos (luz, temperatura) que contribúen a xerar dito ritmo (a maiores, existen mecanismos internos que tamén xogan un papel no seu mantemento). Outro tipo de oscilacións son *auto-sostidas*, é dicir, son xeradas internamente polo sistema. Un exemplo son os modelos de Lotka-Volterra, cuxo comportamento periódico se mostrou na Fig. 2.9.

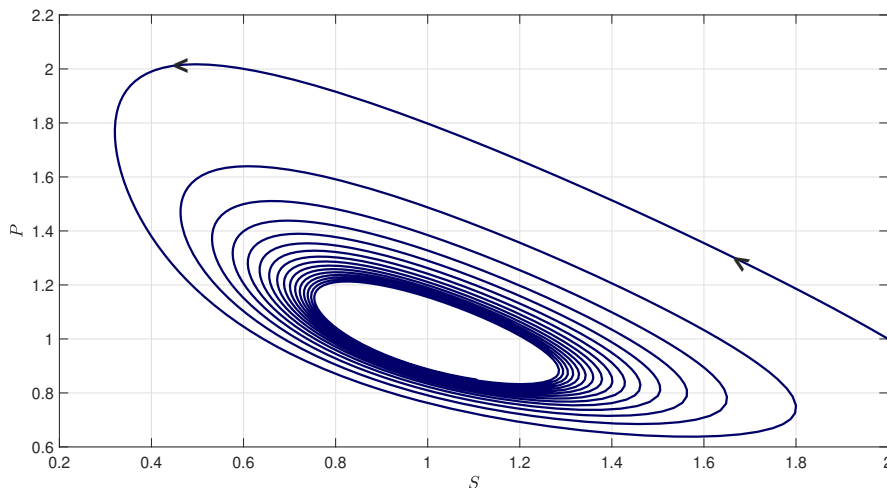
Os *ciclos límite* son un tipo de oscilacións auto-sostidas, caracterizadas por curvas fechadas no espazo de estados. Todas as traxectorias próximas a un ciclo límite son solucións periódicas que, a medida que $t \rightarrow \infty$, poden achegarse a el (ciclo límite estábel) ou afastarse (inestábel). Os ritmos circadianos ou os procesos de diferenciación celular en mamíferos presentan ciclos límite.

Os modelos dinámicos con comportamento oscilatorio son notabelmente difíciles de calibrar. Un motivo é a posibilidade de confusión de frecuencias: para poder reconstruír un sinal, o teorema de Shannon di que é necesario tomar medidas cunha frecuencia como mínimo igual ao duplo da frecuencia de oscilación. Como os biosistemas frecuentemente presentan limitacións importantes para a toma de medidas, é posíbel que isto non se cumpra. Ademais, o problema de optimización asociado á estimación de parámetros dun sistema oscilatorio soe ser numericamente complicado e é fácil obter como solución un mínimo local, en lugar do óptimo global.

Oscilacións glicolíticas: exemplo de ciclo límite e bifurcación de Hopf. (Sacado de [15], pp. 119–121 e 127.) Na sección 2.3.1 falouse da glicólise, unha rede de reaccións metabólicas coas que os organismos obteñen enerxía. Neste proceso téñense observado oscilacións, tanto amortecidas como sostidas. Un mecanismo involucrado neste proceso é o seguinte: un substrato S transfórmase nun produto P , o cal actúa como un enzima, catalizando a conversión de S a P . Este mecanismo chámase autocatálise e pode modelarse con cinética Hill como:

$$\dot{S} = v_0 - S \frac{\alpha(P/K)^2}{1 + (P/K)^2}, \quad \dot{P} = S \frac{\alpha(P/K)^2}{1 + (P/K)^2} - k_2 P. \quad (5.17)$$

Se $K \gg P$, o termo $F(P) = \frac{\alpha(P/K)^2}{1+(P/K)^2} \approx \frac{\alpha P^2}{K^2}$, e o sistema correspondente admite un ciclo límite estábel para valores apropiados dos parámetros, como se mostra na seguinte figura para $v_0 = 1, \alpha/K^2 = 1, k_2 = 1.00001$:



Vexamos se este sistema presenta unha bifurcación de Hopf. Para iso, calculamos os seus autovalores coa fórmula 5.12, obtendo:

$$\text{Tr}(J) = k_2 - k_1(v_0/k_2)^2, \text{Det}(J) = k_1 \cdot v_0^2/k_2$$

onde usamos $k_1 = \alpha/K^2$. Os autovalores son imaxinarios se $\text{Tr}(J) = 0$, é dicir, se $k_1 = k_2^3/v_0^2$. Por tanto, cando k_1 se aproxima a este valor, o sistema pasa por unha bifurcación de Hopf. Cando $k_1 \approx k_2^3/v_0^2$ aparece unha oscilación sinusoidal, cuxa frecuencia é $\omega = \sqrt{4\text{Det}(J)} = 2v_0\sqrt{k_1/k_2}$.

5.5. Robustez

Para sobrevivir, os organismos deben ser capaces de tolerar cambios externos (producidos no seu entorno) e internos (imperfeccións ou variabilidade nos seus compoñentes). O termo *robustez* alude á capacidade dun biosistema de seguir funcionando en condicións variábeis, sen alterar substancialmente o seu comportamento. Isto implica:

1. Preservar a estabilidade.
2. Que o comportamento dinámico sexa o mesmo cualitativamente.
3. Que o comportamento dinámico se desvíe pouco do nominal en termos cuantitativos: que o transitorio sexa parecido, que se chegue a un estado estacionario similar..

Pero, ademais de exhibir este comportamento robusto, os biosistemas deben ser capaces de detectar e responder adecuadamente a estímulos externos (é dicir, deben ser sensíbeis a certos sinais). Por tanto, existe un *balance entre sensibilidade e robustez*; un biosistema debe ser sensíbel a certas entradas e robusto fronte a outras. Pensemos na analoxía cun filtro paso baixo: está deseñado para eliminar altas frecuencias, que se consideran ruído, e deixar pasar sinais de baixa frecuencia, que se consideran relevantes. A analoxía con sistemas eléctricos é interesante porque, ao igual que neles, reducir o ruído a certas frecuencias pode implicar aumentalo para outras frecuencias: a sensibilidade e robustez perfectas non existen. A evolución selecciona organismos capaces de adaptarse a perturbacións frecuentes, aínda que non respondan tan ben a perturbacións extremas que teñen lugar esporadicamente.

Tipos de robustez biolóxica. A robustez implica insensibilidade fronte a cambios. Dependendo da natureza do que cambie, podemos distinguir entre diferentes tipos de robustez:

- *Homeostase*: capacidade de manter aproximadamente constante unha cantidade de interese en presenza de perturbacións externas.
- *Canalización*: capacidade de alcanzar o mesmo estado final, independentemente das condicións iniciais.
- *Insensibilidade paramétrica*: capacidade de manter o mesmo comportamento cualitativo para pequenas variacións dos parámetros internos.
- *Tolerancia a fallos*: capacidade de seguir funcionando se un ou varios compoñentes fallan.

Mecanismos para conseguir robustez. Existen varios tipos de mecanismos que tenden a facer robusto o comportamento dos biosistemas a todos os niveis, desde células a organismos. Nas seguintes subseccións veremos os principais.

5.5.1. Redundancia

Se un sistema posúe elementos duplicados, pode seguir funcionando se un deles falla. Este mecanismo é moi común nas redes celulares. Por exemplo, mediante a existencia de dous ou máis xenes que codifican a produción do mesmo enzima. Outro exemplo son as rutas metabólicas ou de sinalización redundantes. Por exemplo, moitas células prefiren producir enerxía en forma de ATP mediante fermentación da glicosa. Este proceso (glicólise) é rápido e eficiente. Porén, en ausencia de glicosa esta ruta metabólica non funciona, polo que moitas células pasan a usar outra ruta, o ciclo TCA, que produce enerxía mediante respiración, o que resulta menos eficiente.

5.5.2. Insensibilidade paramétrica.

Neste apartado vese como varían a saída e os estados dun sistema en función de θ . Para isto poderíamos tentar resolver as EDOs dun modelo analiticamente, obtendo así x_e para despois obter $y = h(x_e, \theta_0, u_0)$. Porén, con carácter xeral isto non é posíbel, excepto para casos particulares. Por iso, obteremos no seu lugar a resposta linearizada, isto é,

$$S_{x,\theta} = \left. \frac{dx_e}{d\theta} \right|_{\theta_0}, \quad S_{y,\theta} = \left. \frac{dy_e}{d\theta} \right|_{\theta_0}, \quad (5.18)$$

que son os cambios infinitesimais no estado e a saída no equilibrio debidos a un cambio no vector de parámetros.

Sexa $y_e(\theta_0, u_0)$ a saída dun modelo correspondente a un punto de equilibrio estábel, x_e , para uns vectores de parámetros θ_0 e u_0 . Por ser un punto de equilibrio, $f(x_e, \theta_0, u_0) = 0$. Supoñamos que $u = 0$ e escribamos, para simplificar a notación, $f(x, \theta)$ en lugar de $f(x, \theta, u)$.

Así, calculamos a *matriz de sensibilidade dos estados* como:

$$\frac{df}{d\theta} = \frac{\partial f}{\partial x} \frac{dx_e}{d\theta} + \frac{\partial f}{\partial \theta} \Rightarrow S_{x,\theta} = \frac{dx_e}{d\theta} = - \left(\frac{\partial f}{\partial x} \right)^{-1} \frac{\partial f}{\partial \theta} \Big|_{(x_e, \theta_0)} \quad (5.19)$$

Do mesmo xeito, podemos obter a sensibilidade fronte a cambios na entrada como:

$$\left. \frac{dx_e}{du} \right|_{(u_0, \theta_0)} = - \left(\frac{\partial f}{\partial x} \right)^{-1} \frac{\partial f}{\partial u} \Big|_{(x_e, \theta_0, u_0)} \quad (5.20)$$

Analogamente, obtemos a *matriz de sensibilidade das saídas* como:

$$S_{y, \theta} = \frac{dy_e}{d\theta} = \frac{\partial h}{\partial x} \frac{dx_e}{d\theta} + \frac{\partial h}{\partial \theta} = \left(\frac{\partial h}{\partial x} \left(\frac{\partial f}{\partial x} \right)^{-1} \frac{\partial f}{\partial \theta} - \frac{\partial h}{\partial \theta} \right) \Big|_{(x_e, \theta_0)} \quad (5.21)$$

Estas expresións poden ser avaliadas numericamente, cuantificando así o efecto de pequenos cambios dos parámetros no estado e na saída no equilibrio.

As versións normalizadas das matrices de sensibilidades dividen os parámetros polos seus valores nominais, e reescalan as saídas e os estados polos seus valores de equilibrio:

$$\bar{S}_{x, \theta} = (\text{diag}\{x_e\})^{-1} S_{x, \theta} \text{diag}\{\theta\}, \quad (5.22)$$

$$\bar{S}_{y, \theta} = (\text{diag}\{y_e\})^{-1} S_{y, \theta} \text{diag}\{\theta\}. \quad (5.23)$$

Implicacións para a calibración. Se a saída do modelo é pouco sensíbel a un parámetro, o parámetro será pouco identificábel na práctica. Poderíamos pensar que isto ten pouca importancia, xa que as predicións do modelo non se ven moi afectadas polo valor do parámetro. Porén, hai que ter coidado, xa que as sensibilidades definidas anteriormente son locais, é dicir, válidas no entorno dun punto de equilibrio determinado. Un modelo pode ser pouco sensíbel a un parámetro nun punto determinado do espazo de estados, e moi sensíbel noutro punto. Se calibramos o modelo con datos obtidos no entorno do primeiro punto, e despois simulamos o modelo no entorno do segundo punto, poderemos cometer erros importantes.

Metabolic control analysis (MCA). O MCA é un formalismo particular para estudar a resposta dunha rede metabólica en estado estacionario fronte a perturbacións nos valores de parámetros ou concentracións. Emprega para tal fin matrices similares ás presentadas nesta sección, e define unha serie de coeficientes (de elasticidade, de control de fluxo, de resposta). Para máis información pódese consultar a Sección 2.3.2 de [33].

5.5.3. Realimentación

A realimentación negativa (*feedback*) é unha forma moi potente de manter unha cantidade – tipicamente, a saída dun sistema – en torno a un valor desexado (consigna) fronte a perturbacións. A idea básica é aplicar ao sistema unha entrada que sexa función do erro (diferenza entre a saída e a consigna). A figura 5.4 (baseada na 1.2) representa esta configuración.

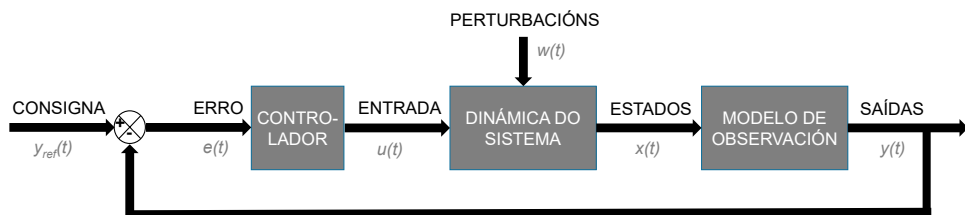


Figura 5.4: Realimentación negativa. O controlador aplica unha lei de control, como p.ex. a da ecuación (5.24), que calcula a entrada $u(t)$ ao sistema en función do erro $e(t)$, definido como a diferenza entre a consigna ou sinal de referencia $y_{ref}(t)$ e a saída real $y(t)$.

Na súa forma máis sinxela, a entrada pode ser unha función proporcional, isto é,

$$u = K \cdot e = K(y_{ref} - y), \quad (5.24)$$

onde e é o erro, e y_{ref} é a consigna ou saída de referencia. Tamén se poden empregar entradas que sexan funcións da derivada do erro (control derivativo) ou da súa integral (control integral). O control integral, en particular, ten a capacidade de eliminar o erro en estado estacionario en certo tipo de sistemas. A combinación dos tres tipos dá lugar a un controlador PID (proporcional-integral-derivativo), amplamente usado na industria. Moitos biosistemas posúen mecanismos de realimentación negativa, tanto a nivel celular como fisiolóxico.

5.5.4. Prealimentación

A **prealimentación** (*feedforward*) é un mecanismo moi frecuente en redes de transcripción e sinalización. Nelas, a entrada u actúa na saída y de dúas formas: unha directa e outra indirecta (a través dun estado intermedio, o que introduce un retardo). Se ambas interaccións teñen o mesmo signo (p.ex., se ambas teñen un efecto positivo na variábel final) dise que o lazo é *coherente*. Se teñen signo diferente, dise que é un lazo *incoherente*. Por exemplo, se a interacción directa é positiva – aumenta o nivel da variábel final – e a indirecta é negativa – tende a diminuír o nivel da variábel final, a través dun estado intermedio, o que introduce un retardo – temos un lazo de feedforward incoherente, como o representado na Fig. 5.5. Esta combinación de efectos positivos e negativos pode dar lugar, baixo condicións apropiadas, a unha adaptación.

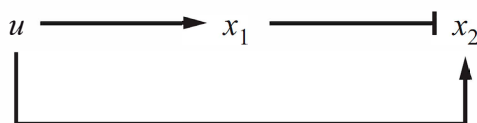


Figura 5.5: Lazo de prealimentación incoherente ou *incoherent feedforward loop*. A variábel u ten dous efectos contraditorios en x_2 : por unha banda, positivo (rama inferior, directa) e por outra, negativo (rama superior, indirecta mediante x_1).

De feito, algúns lazos de feedforward incoherentes son capaces de proporcionar unha propiedade de invarianza chamada **fold-change detection**, que é como segue: Supoñamos un sistema que se encontra en equilibrio cunha entrada $u = a$. Ao modificar a entrada a $u = a + b$, o sistema responde cunha saída $y_1(t)$. Supoñamos que ao mesmo sistema se lle aplica unha entrada $p \cdot a$, e se lle deixa tempo para adaptarse a ela. Unha vez en equilibrio, aplícaselle unha entrada $p \cdot (a + b)$, o que ocasiona unha saída $y_2(t)$. Se a nova saída é igual á antiga, é dicir $y_1(t) = y_2(t)$, o sistema ten a propiedade de “fold-change detection”. Noutras palabras, os sistemas que presentan este comportamento responden da mesma forma – despois dun período de adaptación – ante entradas cambiadas polo mesmo factor multiplicativo.

Outra propiedade similar é a chamada **compensación dinámica** (*dynamical compensation*, DC) [49]. A DC consiste caracterízase porque, fronte a cambios no valor dun parámetro θ , a saída do sistema é exactamente igual despois dun período de adaptación. (Nótese que, de non ser pola existencia dun período de adaptación, esta propiedade consistiría simplemente nunha sensibilidade de saída nula respecto a dito parámetro.)

En canto aos lazos de feedforward coherentes, poden actuar como **filtros**, contribuíndo desta forma a asegurar un comportamento robusto nun entorno ruidoso [11]. Por este motivo, este tipo de lazos abundan nas redes de sinalización. Outro mecanismo capaz de atenuar o ruído é a realimentación negativa.

Exemplo de “compensación dinámica”. Un dos modelos que temos visto neste curso, o βIG (véxanse as Figs. 2.6 e 4.4), foi presentado orixinalmente en [29] para ilustrar o fenómeno da compensación dinámica. Esta propiedade está relacionada coa falta de identificabilidade estrutural [53], e de feito, na Fig. 4.4 do presente texto usouse para ilustrar a falta de identificabilidade estrutural. Na Fig. 5.6 este modelo úsase como exemplo de compensación dinámica. O diagrama da esquerda mostra o esquema de interconexións das variábeis de estado do modelo (en círculos azuis) e dos seus parámetros. Obsérvase que a entrada u ten un efecto inmediato de incrementar a concentración de glucosa (frecha directa $u \rightarrow G$), pero tamén outro efecto, indirecto, mediante o que reduce dita concentración (camiño formado polas frechas vermellas, sendo a última inhibitoria). Así, este modelo conta cun tipo de prealimentación, que posibilita a adaptación descrita na figura.

5.6. Redución de modelos

O obxectivo da redución de modelos é obter un novo modelo que manteña as propiedades esenciais e o comportamento dinámico do modelo orixinal – ou polo menos, de certas variábeis de interese – pero con menos ecuacións e/ou parámetros. Existen varios métodos de redución de modelos.

Un tipo de técnicas baséanse na *separación de escalas temporais*, e pódense aplicar se un modelo conta con procesos que teñen lugar a escalas temporais

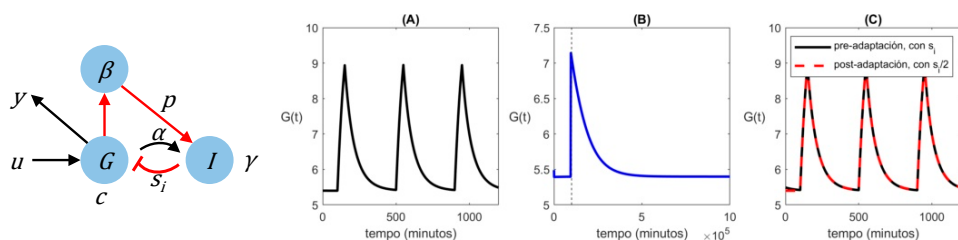


Figura 5.6: Compensación dinámica. As gráficas (A-B-C) ilustran a propiedade da compensación dinámica co modelo βIG , mostrado á esquerda. (A) Dinámica de glicosa anterior á adaptación, en resposta a inxestións periódicas. (B) En $t = 100000$, o valor de s_i cambia ($s_i \rightarrow s_i/2$), provocando unha lenta adaptación do valor basal de glicosa. (C) Compensación dinámica: despois da adaptación ao novo valor do parámetro, o comportamento fronte ás inxestións de glicosa é idéntico ao previo á adaptación.

moi diferentes. Se estamos interesados na dinámica máis lenta, os procesos de dinámica rápida poden ser aproximados polas súas solucións de equilibrio. Se estamos interesados na dinámica máis rápida, os estados de dinámica moi lenta poden considerarse constantes e iguais á súa condición inicial durante o período de tempo de interese.

Outro tipo de simplificación é a *omisión de elementos* que teñen pouca influencia nas predicións do modelo, como parámetros con baixa sensibilidade ou especies cuxa concentración é moi baixa e non afecta apreciabelmente as reaccións de interese.

A *linearización*, que temos visto para analizar a estabilidade, é tamén en si mesma un método de redución de modelos, xa que un modelo linear é menos complexo que un nonlinear.

Exemplo: redución do modelo βIG . Obsérvese na Fig. 5.7 a diferenza de valores dos tres estados do modelo (liñas de trazo negro continuo) nos eixos verticais: mentres que a glicosa e a insulina varían substancialmente en cuestión de poucas horas, a masa das células β varía apenas un 0.3 % no mesmo tempo (de 322 a 323). Por tanto, unha posíbel simplificación do modelo

sería considerala constante, o que reduciría os estados do modelo de 3 a 2. As simulacións correspondentes do modelo así reducido móstranse en trazos vermellos discontinuos, sendo para $G(t)$ e $I(t)$ indistinguíbeis na práctica das traxectorias do modelo orixinal. Por tanto, se estamos interesados nas predicións sobre $G(t)$ e $I(t)$, podemos usar esta simplificación.

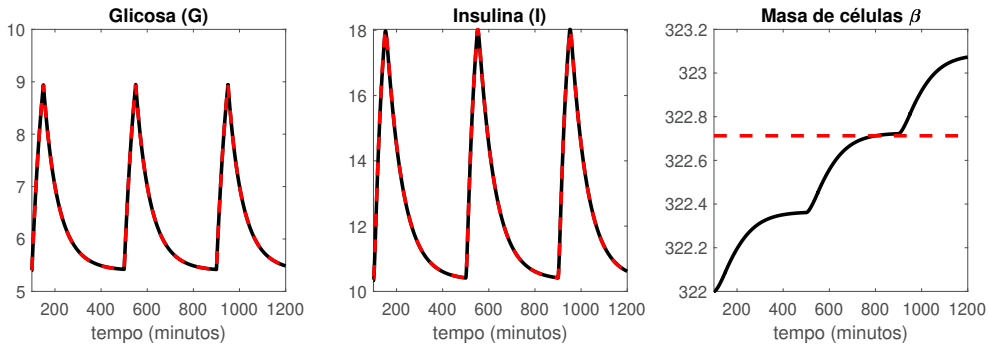


Figura 5.7: Modelo βIG . Traxectorias das variábeis de estado do modelo orixinal (trazo negro continuo) e do modelo reducido mediante a consideración da masa das células β como constante (trazo vermello discontinuo).

Cuestións e problemas do Tema 5

C5.1. O seguinte modelo representa un posíbel mecanismo para implementar o lazo de prealimentación incoherente da Fig. 5.5. A variábel intermedia, x_1 , promove a degradación da variábel final, x_2 .

$$\mathcal{M} = \begin{cases} \dot{x}_1 = \alpha u - \gamma x_1, & (5.25) \\ \dot{x}_2 = \beta u - \delta x_1 x_2, & (5.26) \\ y = x_2. & (5.27) \end{cases}$$

Pídese:

- Calcular o Jacobiano do sistema en torno a un punto de equilibrio para unha entrada constante $u \neq 0$.
- Dicar se o punto de equilibrio é estábel.

C5.2. O seguinte modelo representa outra forma de implementar un lazo de prealimentación incoherente. Pídese:

- Calcular o Jacobiano do sistema en torno a un punto de equilibrio para unha entrada constante $u \neq 0$.
- Dicar se o punto de equilibrio é estábel.

$$\mathcal{M} = \begin{cases} \dot{x}_1 = \alpha u - \gamma x_1, & (5.28) \\ \dot{x}_2 = \beta \frac{u}{x_1} - \delta x_2, & (5.29) \\ y = x_1; & (5.30) \end{cases}$$

C5.3. Definicións de estabilidade asintótica e estabilidade no sentido de Lyapunov.

C5.4. Como podemos determinar a estabilidade asintótica dun modelo de EDOs nonlineares?

C5.5. Sensibilidade: concepto e fórmulas das matrices de sensibilidade dos estados e das saídas.

C5.6. Tipos de bifurcaciones.

C5.7. Prealimentación: en que consiste, e que dous tipos hai?

C5.8. De que formas se pode reducir a dimensión dun modelo?

Bibliografía

- [1] C. Angione. Human systems biology and metabolic modelling: a review—from disease metabolism to precision medicine. *BioMed research international*, 2019:8304260, 2019.
- [2] B. Ballnus, S. Schaper, F. J. Theis, and J. Hasenauer. Bayesian parameter estimation for biochemical reaction networks using region-based adaptive parallel tempering. *Bioinformatics*, 34(13):i494–i501, 2018.
- [3] J. R. Banga and E. Balsa-Canto. Parameter estimation and optimal experimental design. *Essays in Biochemistry*, 45:195–210, 2008.
- [4] G. A. Bekey and J. E. Beneken. Identification of biological systems: a survey. *Automatica*, 14(1):41–47, 1978.
- [5] R. Bellman and K. J. Åström. On structural identifiability. *Mathematical biosciences*, 7(3-4):329–339, 1970.
- [6] R. Bergman, L. S. Phillips, C. Cobelli, et al. Physiologic evaluation of factors controlling glucose tolerance in man: measurement of insulin sensitivity and beta-cell glucose sensitivity from the response to intravenous glucose. *The Journal of clinical investigation*, 68(6):1456–1467, 1981.
- [7] J. Bondia, S. Romero-Vivo, B. Ricarte, and J. L. Diez. Insulin estimation and prediction: a review of the estimation and prediction of subcutaneous insulin pharmacokinetics in closed-loop glucose control. *IEEE Control Systems Magazine*, 38(1):47–66, 2018.

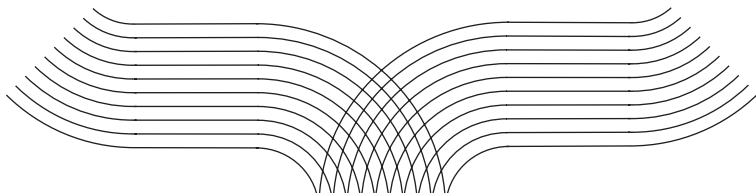
- [8] H. Bozdogan. Model selection and Akaike's information criterion (AIC): The general theory and its analytical extensions. *Psychometrika*, 52(3):345–370, 1987.
- [9] J. D. Brunner and N. Chia. Metabolite-mediated modelling of microbial community dynamics captures emergent behaviour more effectively than species–species modelling. *Journal of the Royal Society Interface*, 16(159):20190423, 2019.
- [10] M. Castro and R. J. de Boer. Testing structural identifiability by a simple scaling method. *PLOS Computational Biology*, 16(11):e1008248, 2020.
- [11] S. Chakravarty and A. Csikász-Nagy. Systematic analysis of noise reduction properties of coupled and isolated feed-forward loops. *PLOS Computational Biology*, 17(12):e1009622, 2021.
- [12] H. Cramér. *Mathematical Methods of Statistics (PMS-9)*, volume 9. Princeton university press, 2016.
- [13] F. Crick. Central dogma of molecular biology. *Nature*, 227(5258):561–563, 1970.
- [14] C. Dalla Man, R. A. Rizza, and C. Cobelli. Meal simulation model of the glucose-insulin system. *IEEE Transactions on biomedical engineering*, 54(10):1740–1749, 2007.
- [15] D. Del Vecchio and R. M. Murray. *Biomolecular feedback systems*. Princeton University Press, 2014.
- [16] J. DiStefano III. *Dynamic systems biology modeling and simulation*. Academic Press, 2015.
- [17] B. Efron and C. Stein. The jackknife estimate of variance. *The Annals of Statistics*, pages 586–596, 1981.
- [18] B. Efron and R. Tibshirani. Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. *Statistical science*, pages 54–75, 1986.

- [19] H. Enderling and O. Wolkenhauer. Are all models wrong? *Computational and systems oncology*, 1(1), 2020.
- [20] J. M. Epstein. Why model? *Journal of artificial societies and social simulation*, 11(4):12, 2008.
- [21] C. Erdem, A. V. Lee, D. L. Taylor, and T. R. Lezon. Inhibition of rps6k reveals context-dependent akt activity in luminal breast cancer cells. *PLOS Computational Biology*, 17(6):e1009125, 2021.
- [22] K. A. Fujita, Y. Toyoshima, S. Uda, Y.-i. Ozaki, H. Kubota, and S. Kuroda. Decoupling of receptor and downstream signals in the akt pathway by its low-pass filter characteristics. *Science Signaling*, 3(132):ra56–ra56, 2010.
- [23] D. T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *The journal of physical chemistry*, 81(25):2340–2361, 1977.
- [24] B. Goodwin. Oscillatory behavior in enzymatic control processes. *Advances in Enzyme Regulation*, 3:425–428, 1965.
- [25] A. C. Guyton, T. G. Coleman, and H. J. Granger. Circulation: overall regulation. *Annual review of physiology*, 34(1):13–44, 1972.
- [26] A. Hodgkin and A. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *Journal of Physiology*, 117:500–544, 1952.
- [27] R. Hovorka, V. Canonico, L. J. Chassin, U. Haueter, M. Massi-Benedetti, M. O. Federici, T. R. Pieber, H. C. Schaller, L. Schaupp, T. Vering, et al. Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes. *Physiological measurement*, 25(4):905, 2004.
- [28] B. P. Ingalls. *Mathematical modeling in systems biology: an introduction*. MIT press, 2013.
- [29] O. Karin, A. Swisa, B. Glaser, Y. Dor, and U. Alon. Dynamical compensation in physiological circuits. *Molecular systems biology*, 12(11):886, 2016.

- [30] G. Karlebach and R. Shamir. Modelling and analysis of gene regulatory networks. *Nature reviews Molecular cell biology*, 9(10):770–780, 2008.
- [31] J. R. Karr, J. C. Sanghvi, D.Ñ. Macklin, M. V. Gutschow, J. M. Jacobs, B. Bolival Jr, N. Assad-Garcia, J. I. Glass, and M. W. Covert. A whole-cell computational model predicts phenotype from genotype. *Cell*, 150(2):389–401, 2012.
- [32] S. A. Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of theoretical biology*, 22(3):437–467, 1969.
- [33] E. Klipp, W. Liebermeister, C. Wierling, and A. Kowald. *Systems biology: a textbook*. John Wiley & Sons, 2016.
- [34] J. Liepe, P. Kirk, S. Filippi, T. Toni, C. P. Barnes, and M. P. Stumpf. A framework for parameter estimation and model selection from experimental data in systems biology using approximate bayesian computation. *Nature protocols*, 9(2):439–456, 2014.
- [35] F. Liu, M. Heiner, and D. Gilbert. Hybrid modelling of biological systems: current progress and future prospects. *Briefings in Bioinformatics*, 23(3):bbac081, 2022.
- [36] L. Ljung. *System identification: Theory for the user*. Prentice Hall, New Jersey, 1999.
- [37] G. Massonis, J. R. Banga, and A. F. Villaverde. Autorepar: A method to obtain identifiable and observable reparameterizations of dynamic models with mechanistic insights. *International Journal of Robust and Nonlinear Control*, 2021.
- [38] G. Massonis, J. R. Banga, and A. F. Villaverde. Structural identifiability and observability of compartmental models of the covid-19 pandemic. *Annual reviews in control*, 51:440–459, 2021.
- [39] H. Miao, X. Xia, A. Perelson, and H. Wu. On identifiability of nonlinear ode models and applications in viral dynamics. *SIAM Review*, 53(1):3–39, 2011.

- [40] J. Musilova and K. Sedlar. Tools for time-course simulation in systems biology: a brief overview. *Briefings in Bioinformatics*, 22(5):bbaa392, 2021.
- [41] National Research Council. *Biomedical models and resources: current needs and future opportunities*. National Academies Press, 1998.
- [42] L. K. Nguyen, A. Degasperi, P. Cotter, and B.Ñ. Kholodenko. Dyvipac: an integrated analysis and visualisation framework to probe multi-dimensional biological networks. *Scientific reports*, 5(1):1–17, 2015.
- [43] D. Noble. Modeling the heart—from genes to cells to the whole organ. *Science*, 295(5560):1678–1682, 2002.
- [44] H. Panchiwala, S. Shah, H. Planatscher, M. Zakharchuk, M. König, and A. Dräger. The systems biology simulation core library. *Bioinformatics*, 38(3):864–865, 2022.
- [45] G. W. Roell, J. Zha, R. R. Carr, M. A. Koffas, S. S. Fong, and Y. J. Tang. Engineering microbial consortia by division of labor. *Microbial cell factories*, 18(1):1–11, 2019.
- [46] M. A. Savageau. Introduction to s-systems and the underlying power-law formalism. *Mathematical and Computer Modelling*, 11:546–551, 1988.
- [47] L. Schmiester, Y. Schälte, F. T. Bergmann, T. Camba, E. Dudkin, J. Egert, F. Fröhlich, L. Fuhrmann, A. L. Hauber, S. Kemmer, et al. Petab–interoperable specification of parameter estimation problems in systems biology. *PLOS computational biology*, 17(1):e1008646, 2021.
- [48] Z. Shi, Y. Li, and M. Jaber-Douraki. Hybrid computational modeling demonstrates the utility of simulating complex cellular networks in type 1 diabetes. *PLOS computational biology*, 17(9):e1009413, 2021.
- [49] E. D. Sontag. Dynamic compensation, parameter identifiability, and equivariances. *PLOS computational biology*, 13(4):e1005447, 2017.
- [50] M. Tomita. Whole-cell simulation: a grand challenge of the 21st century. *Trends in Biotechnology*, 19(6):205–210, 2001.

- [51] J. Vera, C. Lischer, M. Ćenov, S. Ćikolov, X. Lai, and M. Eberhardt. Mathematical modelling in biomedicine: A primer for the curious and the skeptic. *International journal of molecular sciences*, 22(2):547, 2021.
- [52] A. F. Villaverde and J. R. Banga. Reverse engineering and identification in systems biology: strategies, perspectives and challenges. *Journal of The Royal Society Interface*, 11(91):20130505, 2014.
- [53] A. F. Villaverde and J. R. Banga. Dynamical compensation and structural identifiability of biological models: Analysis, implications, and reconciliation. *PLOS computational biology*, 13(11):e1005878, 2017.
- [54] A. F. Villaverde, A. Barreiro, and A. Papachristodoulou. Structural identifiability of dynamic systems biology models. *PLOS Computational Biology*, 12(10):e1005153, 2016.
- [55] A. F. Villaverde, D. Pathirana, F. Fröhlich, J. Hasenauer, and J. R. Banga. A protocol for dynamic model calibration. *Briefings in bioinformatics*, 23(1):bbab387, 2022.
- [56] A. F. Villaverde, E. Raimúndez, J. Hasenauer, and J. R. Banga. Assessment of prediction uncertainty quantification methods in systems biology. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 20(3):1725–1736, 2022.
- [57] V. Vyshemirsky and M. A. Girolami. Bayesian ranking of biochemical system models. *Bioinformatics*, 24(6):833–839, 2008.
- [58] E. Walter and L. Pronzato. *Identification of Parametric Models from Experimental Data*. Springer, Masson, 1997.



Manuais

Serie de manuais didácticos

Últimas publicacións na colección

*Traducom comentada de textos didáticos e divulgadores.
Um modelo orientativo para tradutor em formaçom (2023)*
Carlos Garrido Rodríguez

Bioestatística para a Enxeñaría Biomédica (2023)
Juan Carlos Pardo Fernández

Fundamentos de Dereito de Defensa da Competencia (2022)
Julio Costas Comesaña

*Sistemas Fluidomecánicos no transporte: Prácticas de
simulacións numéricas (2022)*
María Concepción Paz Penín, Eduardo Suárez Porto,
Jesús Vence Fernández e Adrián Cabarcos Rey.

Álgebra linear: Historia, teoría e práctica (2021)
Ramón González Rodríguez



Modelado e Simulación de Sistemas Biomédicos

O uso do modelado matemático en bioloxía e medicina conta cunha longa tradición, cuxas orixes podemos rastrear como mínimo até a Idade Media. No século XX a utilización de métodos procedentes da enxeñaría tornouse práctica común en biomedicina. Polas súas características específicas, os procesos biomédicos presentan retos diferentes aos encontrados nas enxeñarías “clásicas”, o que motivou o desenvolvemento de novas técnicas e a adaptación das existentes. Creouse así un fluxo bidireccional de ideas, que fertilizou ambas disciplinas científicas e preparou o camiño para a aparición da moderna bioloxía de sistemas no século XXI. Na actualidade, o modelado e simulación de sistemas dinámicos é unha ferramenta fundamental na enxeñaría biomédica.

Este manual foi concibido orixinalmente como material docente para a materia “Modelado e Simulación de Sistemas Bio-

médicos”, impartida no primeiro curso do Mestrado en Enxeñaría Biomédica da Universidade de Vigo. Nel compéndianse os coñecementos necesarios para construír, analizar e aplicar modelos dinámicos de biosistemas, facendo énfase nos máis relevantes para a enxeñaría biomédica.

O libro estrutúrase en cinco capítulos, cada un dos cales inclúe múltiples exemplos e cuestións. Comeza cunha introdución e unha panorámica dos distintos tipos de modelos, centrándose nos baseados en ecuacións diferenciais ordinarias. A seguir, o segundo capítulo describe como modelar biosistemas de distintos tipos, desde redes de reaccións bioquímicas a nivel intracelular até sistemas de interaccións entre organismos. O terceiro capítulo mostra como simular ditos modelos, e o cuarto describe un protocolo para calibralos. Finalmente, o derradeiro capítulo ensina a analizar aspectos clave da súa dinámica.