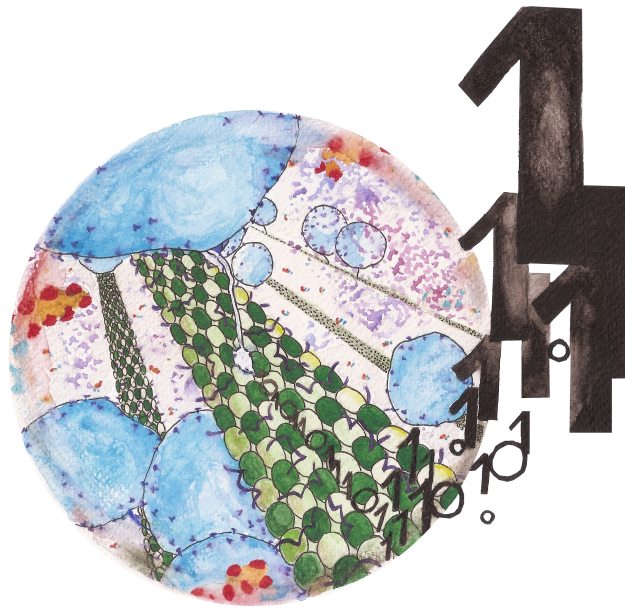


JAVIER SPARACINO

CUANTIFICADORES DE INFORMACIÓN EN FÍSICA

CUANTIFICADORES DE INFORMACIÓN EN FÍSICA

JAVIER SPARACINO



Aplicaciones al estudio de la dinámica del motor molecular KIF_{1A}
y al problema de distinguibilidad entre estados cuánticos

Presentada ante la Facultad de Matemática, Astronomía y Física
como parte de los requerimientos para la obtención del grado de
Doctor en Física de la Universidad Nacional de Córdoba

Mayo, 2013

Director: Pedro Walter Lamberti

Javier Sparacino: *Cuantificadores de Información en Física*, Aplicaciones al estudio de la dinámica del motor molecular KIF_{1A} y al problema de distinguibilidad entre estados cuánticos, © FaMAF-UNC, Mayo, 2013.

A mis padres: Eleonor y Mario.

RESUMEN

El núcleo de este trabajo lo constituye el estudio de varios aspectos de la dinámica del motor molecular KIF_{1A} a partir de conceptos de Mecánica Estadística y Teoría de la Información. La función biológica de este motor molecular es el transporte de cargas, en especial de vesículas sinápticas a lo largo de los axones de las neuronas. Defectos y disrupciones en el transporte intracelular llevado a cabo por motores moleculares ha demostrado ser la base molecular de varias enfermedades neurodegenerativas.

En una primera parte desarrollamos una simulación Monte Carlo para la dinámica de los motores moleculares KIF_{1A} basada en el modelo de Nishinari y colaboradores (Phys. Rev. Lett. 95: 118101, 2005) y construimos un método de análisis que nos permite estudiar el estado estacionario del modelo. El método consiste en el mapeo del estado del filamento en secuencias simbólicas y su posterior análisis a partir de un procedimiento de segmentación basado en la divergencia de Jensen-Shannon. De esta manera fue posible identificar de manera automática si el estado estacionario del sistema presenta una única fase o coexistencia de fases con una pared de dominio (interfase estable) separando una región de baja densidad de motores y una región de alta densidad en la cual el transporte se ve interrumpido.

En una segunda parte se realizaron mejoras al modelo para la dinámica de los motores KIF_{1A}, en particular se incorporó el efecto de la proteína asociada al microtúbulo tau. Los filamentos de las células eucariotas están abundantemente decorados de proteínas estáticas que cumplen diversas funciones en la organización del citoesqueleto. De todas ellas la más abundante es la proteína tau. Propusimos un modelo de ecuaciones maestras que incorpora los efectos de tau en la dinámica de KIF_{1A}, estudiamos algunos límites analíticos del modelo y desarrollamos una simulación Monte Carlo para estudiar su estado estacionario. Además, aplicamos el método de segmentación desarrollado con anterioridad para estudiar la formación de paredes de dominio en presencia de tau.

Paralelamente en un contexto de Teoría de la Información Cuántica se propone una nueva métrica para estados cuánticos. La propuesta está fuertemente inspirada en la divergencia de Jensen-Shannon cuántica y tiene como ingredientes básicos la entropía y las purificaciones. Se estudian algunas propiedades de la nueva propuesta y se calcula explícitamente para el caso de un sistema de estados mixtos de 1-qubit.

PALABRAS CLAVE: Motores Moleculares, Teoría de la Información, Divergencia de Jensen–Shannon, Simulación Monte Carlo, Proteínas Motoras, Paredes de Dominio, Proteínas Asociadas al Microtúbulo, Tau, Purificación, Entropía, Estados Cuánticos.

PACS: 87.16.Nn, 05.40.-a, 89.70.Cf, 45.70.Vn, 87.16.Wd, 03.67.-a, 03.67.Mn.

ABSTRACT

The core of this work is the study of several aspects of the dynamics of the molecular motor KIF_{1A} using quantities from Statistical Mechanics and Information Theory. The main biological function of this molecular motor is cargo transport, specifically the transport of synaptic vesicle precursors in axons. Disruptions or defects of transport events have been determined to be the molecular basis for many neurodegenerative diseases.

In a first part, we developed a Monte Carlo simulation for the dynamics of the molecular motors KIF_{1A} based in the model by Nishinari and coworkers (Phys. Rev. Lett. 95: 118101, 2005) and we built a method that allow us to analyze the stationary state of the model. The method consists in the mapping of the state of the filament in symbolic sequences and their analysis by means of a segmentation procedure based on Jensen-Shannon divergence. In this way, it was possible to automatically identify if the stationary state of the system presents a unique phase or a phase coexistence with a domain wall (stable interphase) that separates a region with low density of motors from a region with high density of motors in which the intracellular transport is interrupted.

In a second part of this work, we improved the model for the dynamics of KIF_{1A} motors, particularly we incorporated the effect of the microtubule associated protein tau. The filaments of the eukaryotic cells are abundantly decorated by static proteins that have several functions in the organization of the cytoskeleton. Tau is the more abundant microtubule associated protein in axons. We proposed a model of master equations that incorporates the effects of tau in the dynamics of KIF_{1A}, we studied some analytical limits of the model and we developed a Monte Carlo simulation that allows us to study the stationary state of the model. Moreover, we applied the segmentation method that we have previously developed for studying the formation of domain walls in the presence of tau molecules.

In parallel in a quantum theoretical information context we proposed a new metric for quantum states. The proposal is strongly inspired by quantum Jensen-Shannon divergence and its basic ingredients are entropy and purifications. We studied some properties of the proposed metric and we explicitly evaluated it for 1-qubit mixed states.

KEYWORDS: Molecular Motors, Information Theory, Jensen-Shannon Divergence, Monte Carlo Simulation, Motor Proteins, Domain Walls, Microtubule Associated Proteins, Tau, Purification, Entropy, Quantum States.

PACS: 87.16.Nn Motor proteins
05.40.-a Fluctuation phenomena, random processes, noise, and Brownian motion.
89.70.Cf Entropy and other measures of information
45.70.Vn Granular models of complex systems; traffic flow
87.16.Wd Intracellular trafficking
03.67.-a Quantum information
03.67.Mn Entanglement measures, witnesses, and other characterizations

AGRADECIMIENTOS

El éxito es una excepción no es un continuo. Los seres humanos de vez en cuando triunfan pero habitualmente desarrollan, combaten, se esfuerzan y ganan de vez en cuando, muy de vez en cuando. No deberíamos difundir lo circunstancialmente exitoso si el objetivo es transmitir valores. La producción se mide en función de las posibilidades no exclusivamente en función de los logros. Pero nosotros estamos acostumbrados sólo a valorar a aquel que llega más arriba. He pensado mucho sobre lo que significa triunfar y lo que significa fracasar. Éxito y felicidad no funcionan como sinónimos.

— Marcelo “el Loco” Bielsa [6]

La tarea de emprender un doctorado no hubiera sido posible sin la colaboración (en sentido amplio) de numerosas personas e instituciones con las que estoy sumamente agradecido.

A CONICET por permitirme continuar con mi formación, a través de las Becas de Postgrado Tipo I y II.

A SeCyT-UNC, la Red de MacroUniversidades Públicas de Latinoamérica y el Caribe, y la Red Latinoamericana de Ciencias Biológicas, que a través de su financiación me permitieron asistir a congresos internacionales y hacer estadías de investigación.

A Walter, por ser una persona accesible, con la que se puede dialogar y discutir. Por todo lo que aprendí en este tiempo, de física y de muchas más cosas. Un artículo, un libro o una noticia fueron disparadores de apasionantes conversaciones en numerosas ocasiones.

A Cristina Turner por su enorme generosidad, su calidad de persona.

Al personal de la facultad, en especial a María José (de Despacho de Alumnos), a Diego (de Personal), a Pancho (de Electrónica), a Eze (de Biblioteca), a Luis (de la Guardia) y a Don Julio (de la Playa de Estacionamiento). Todos ellos se han portado de excelente manera conmigo estos años.

A los profesores con los que tuve la suerte de compartir cátedra y me permitieron aprender algunas cosas sobre el hermoso oficio de la docencia, en especial a Cristina (¡de nuevo!), Damián Knopoff, Oscar Reula, Lorenzo Iparraguirre y Gustavo Sibona.

A la *muchachada* de la 324. A los actuales y también a aquellos con los que compartí oficina y se fueron recibiendo. Trabajar en un ambiente de compañerismo y buena onda fue fundamental para venir a trabajar con alegría y también para hacer más llevaderos los bajones.

A Raúl y Andrés, mis matemáticos de cabecera, por compartir gran parte del trayecto, por los almuerzos, las conversaciones, el apoyo mutuo.

A Alvi, por estar siempre, por las discusiones, el interés (o disimulo) cada vez que le fui con alguna fascinación o angustia científica. Y en especial por su visión artística que se encuentra decorando la tapa de esta tesis.

A Mario Agustín, por bajarme a la realidad muchas veces que me hacía falta, por los consejos y las conversaciones *astronómicas*.

A Elena, por todos los días. Y también por leer con paciencia parte del manuscrito y hacerme numerosos e interesantes comentarios.

A mi familia, por su apoyo y amor incondicional.

Va, para todos ellos, mi humilde agradecimiento por haber acompañado los *desarrollos, combates* y *esfuerzos* durante estos años de mi formación doctoral.

Un prefacio debería llamarse: pararrayos.

— *Georg C. Lichtenberg*¹

Ojalá, como era el deseo de Lichtenberg, este texto se sostuviera sólo sin necesidad de una protección, pero no puedo resistir la tentación de decir algunas cosas.

Esta tesis fue escrita con la idea de que pudiera ser leída por el público más amplio posible. Está dividida esencialmente en tres partes: la Parte [i](#), introductoria, la Parte [ii](#), de aplicaciones biológicas y la Parte [iii](#), de aplicaciones cuánticas.

Cada parte tiene, en cierto sentido, vida propia. La parte introductoria constituye en sí misma una introducción a los conceptos básicos de Teoría de la Información (principalmente el [Capítulo 2](#)). La parte de aplicaciones biológicas es independiente de la parte de aplicaciones cuánticas y se puede leer una sin leer la otra. Ambas requieren la parte introductoria, las aplicaciones cuánticas requieren sólo el [Capítulo 2](#), mientras que las aplicaciones biológicas requieren la parte introductoria completa.

El contenido de esta tesis puede, por supuesto, ser leído de manera corrida pero también puede prescindirse de ciertas partes o capítulos, de acuerdo a los intereses personales.

Un lector con conocimientos de Teoría de la Información puede muy bien saltar el [Capítulo 2](#) y pasar directamente al [Capítulo 3](#) donde se presenta el método de segmentación.

Un físico que no ha tenido contacto con problemas de interés biológico puede sacar provecho del [Apéndice C](#), antes de adentrarse en la lectura de los [Capítulos 4 y 5](#), en los que se presentan las aplicaciones biológicas desarrolladas, pudiendo acceder, de este modo, a ciertos tecnicismos inevitables de una manera más apropiada. En cambio, un biólogo o un físico con conocimientos básicos de biología puede ir directamente a los capítulos sin pasar por el apéndice mencionado.

Un físico con interés en los problemas cuánticos puede saltar del [Capítulo 2](#) a la Parte [iii](#) de aplicaciones cuánticas directamente.

Ha sido la intención que esta tesis sea lo más autocontenida posible, para ello se incluyeron apéndices con demostraciones, con profundización de las discusiones y también con contenido adicional (como por ejemplo los códigos fuente de los programas). Esto definitivamente no se logró en la parte de aplicaciones cuánticas. Para comprender (y no sólo para extender) los contenidos de dicho capítulo, resulta

¹ Extraído de la Referencia [\[38\]](#).

necesario referirse en varios puntos importantes a la bibliografía que se cita.

Mucho del trabajo de estos años tuvo que ver con la formación no sólo con la creación (Walter dixit) y ha sido mi intención intentar transmitir el resultado de mi propia y variada digestión de conceptos. Muchos de los contenidos básicos de esta tesis pueden encontrarse en las referencias que se citan pero he tratado de agregar un toque personal, juntando ideas que fui adquiriendo de diferentes fuentes (bibliografía, profesores, congresos) y en la mayor parte simplemente agregando pequeños puntos que a mí me costó entender y en los que traté de agregar una interpretación o una cuenta completa.

Ojalá algunas de las intenciones tengan asidero en la realidad de estas páginas.

ÍNDICE GENERAL

I	PARTE INTRODUCTORIA	1
1	INTRODUCCIÓN	3
1.1	La versatilidad de la Teoría de la Información	3
1.2	Organización de la tesis	4
2	NOCIONES DE TEORÍA DE LA INFORMACIÓN	7
2.1	Entropía de Shannon	8
2.1.1	Deducción formal de la entropía	11
2.1.2	Algunos ejemplos sencillos	12
2.2	Entropía conjunta y entropía condicional	15
2.3	Información mutua	17
2.4	Divergencia de Kullback-Leibler	17
2.4.1	Propiedades de la divergencia de Kullback-Leibler	18
2.4.2	La información mutua como caso particular de la divergencia de Kullback-Leibler	19
2.5	Divergencia de Jensen-Shannon	20
2.5.1	Propiedades de la divergencia de Jensen-Shannon	20
2.5.2	Divergencia de Jensen-Shannon generalizada	21
2.5.3	Interpretaciones de la divergencia de Jensen-Shannon	22
2.6	Referencias comentadas	32
3	LA DJS COMO HERRAMIENTA PARA SEGMENTAR SECUENCIAS NO-ESTACIONARIAS	33
3.1	Repaso y consideraciones previas	33
3.2	¿Qué es significativamente mayor que cero?	34
3.3	Método de segmentación	36
3.3.1	Significancia máxima	37
3.4	Y con todo eso, ¿cómo se hace para segmentar secuencias no estacionarias?	38
3.5	Ejemplo de aplicación	39
II	APLICACIONES BIOLÓGICAS	47
4	DETECCIÓN DE PAREDES DE DOMINIO EN LA DINÁMICA DE KIF1A	49
4.1	Generalidades acerca del motor molecular KIF1A	50
4.1.1	Motivación: ¿qué? y ¿para qué?	50
4.1.2	¿Qué son los motores moleculares KIF1A?	53
4.1.3	Mecanismo de motilidad	54
4.2	Modelo para la dinámica de KIF1A	57
4.2.1	Ecuaciones maestras	58
4.2.2	Simulación	59
4.3	Localización de paredes de dominio	64
4.3.1	Procedimiento	65

4.3.2	Resultados y comentarios acerca del procedimiento	66	
4.4	Conclusiones del capítulo	73	
5	EFEECTO DE LA PROTEÍNA TAU EN LA DINÁMICA DE KIF1A		79
5.1	Generalidades acerca de tau	79	
5.1.1	Motivación: ¿por qué es importante incluir a tau en el modelo?	79	
5.1.2	Interacción kinesina-tau	80	
5.2	Modelo para la dinámica de KIF1A en presencia de tau	82	
5.2.1	Ecuaciones maestras	83	
5.2.2	Simulación	85	
5.2.3	Límite de alta densidad	88	
5.2.4	Reducción al caso sin tau	89	
5.3	Tiempos de ligadura y longitudes de corrida	90	
5.4	Localización de paredes de dominio en presencia de tau	92	
5.5	Conclusiones del capítulo	94	
III APLICACIONES CUÁNTICAS 103			
6	UNA NUEVA MÉTRICA PARA TEORÍA DE LA INFORMACIÓN CUÁNTICA	105	
6.1	Distancias en mecánica cuántica	105	
6.2	Entropía y Purificación	108	
6.3	Entropía de von Neumann como una métrica para estados puros	110	
6.4	Extensión de la definición de la métrica para estados mixtos	112	
6.5	Evaluación de la métrica para estados mixtos de 1-qubit		114
6.6	Conclusiones del capítulo	116	
IV EPÍLOGO 119			
7	A MODO DE ÉNFASIS	121	
V APÉNDICES 123			
A	EXTRA, EXTRA, . . . INFORMACIÓN DE ÚLTIMO MOMENTO		125
A.1	Entropía conjunta	125	
A.2	La información no hace daño	125	
A.2.1	Desigualdad de Jensen	126	
A.2.2	Aplicaciones de la desigualdad de Jensen	127	
B	FUNCIONES ESPECIALES Y CONSIDERACIONES ESTADÍSTICAS	131	
B.1	Funciones especiales	131	
B.1.1	Función gamma	131	
B.1.2	Función error	131	
B.2	¿Son dos distribuciones diferentes?		132
B.2.1	Chi-cuadrado	132	

B.3	Significancia	133
B.3.1	Significancia máxima	135
B.3.2	Valor crítico de la DJS	137
C	BREVE INTRODUCCIÓN A LA CÉLULA	139
C.1	Células	139
C.1.1	Procariotas y eucariotas	140
C.1.2	La vida requiere energía libre	141
C.2	Nucleótidos	142
C.2.1	Hidrólisis de ATP	142
C.3	Proteínas	143
C.3.1	Plegamiento de proteínas	144
C.3.2	Interacciones de las proteínas	145
C.3.3	Proteínas motoras	145
D	SOBRE KIF1A	147
D.1	Parámetros del modelo para la dinámica de KIF _{1A}	147
D.1.1	Valores de los parámetros del modelo	149
D.1.2	Relación con los procesos físicos del mecanismo ratchet	150
D.2	Núcleo de la simulación	151
D.2.1	Sitio del bulk	151
D.2.2	Sitios en los extremos	153
D.3	Código fuente para el modelo de KIF _{1A}	155
D.3.1	Código fuente constantes.f90	156
D.3.2	Código fuente opciones.f90	157
D.3.3	Código fuente extremos.f90	159
D.3.4	Código fuente main.f90	162
D.4	Formas alternativas para construir las secuencias	165
E	SOBRE LA INTERACCIÓN KIF1A-TAU	171
E.1	Código fuente para el modelo de KIF _{1A} en presencia de tau	171
E.1.1	Código fuente constantes.f90	172
E.1.2	Código fuente concentracion.f90	172
E.1.3	Código fuente opciones.f90	173
E.1.4	Código fuente extremos.f90	181
E.1.5	Código fuente main.f90	190
	BIBLIOGRAFÍA	199

ÍNDICE DE FIGURAS

Figura 1	Entropía para una variable aleatoria binaria.	13
Figura 2	Preguntas binarias para determinar X .	14
Figura 3	Ejemplo del método de segmentación. Paso 1.	40
Figura 4	Ejemplo del método de segmentación. Paso 2a.	41
Figura 5	Ejemplo del método de segmentación. Paso 2b.	41
Figura 6	Ejemplo del método de segmentación. Paso 3a.	42
Figura 7	Ejemplo del método de segmentación. Paso 3b.	43
Figura 8	Ejemplo del método de segmentación. Paso 4a.	43
Figura 9	Ejemplo del método de segmentación. Paso 4b.	44
Figura 10	Ejemplo del método de segmentación. Paso 5a.	45
Figura 11	Ejemplo del método de segmentación. Paso 5b.	45
Figura 12	Transporte intracelular por motores moleculares en una neurona.	51
Figura 13	Acumulación de kinesina en el extremo del microtúbulo.	52
Figura 14	Esquema de un protofilamento del microtúbulo con un motor KIF1A ligado.	53
Figura 15	Acoplamiento mecano-químico del motor KIF1A.	56
Figura 16	Esquema del modelo: estado 0.	60
Figura 17	Esquema del modelo: estado 1.	61
Figura 18	Esquema del modelo: estado 2.	61
Figura 19	Esquema del modelo mostrando las posibilidades de actualización y las tasas de transición asociadas.	62
Figura 20	Localización de una pared de dominio. Valores de la DJS promedio para cada posición del cursor y gráfico espacial.	67
Figura 21	Localización de una pared de dominio. Análisis de las subsecuencias.	68
Figura 22	Detalles del procedimiento de localización de <i>shocks</i> para diferentes concentraciones de ATP y motores KIF1A.	75
Figura 23	Posición del <i>shock</i> para diferentes concentraciones de ATP .	76
Figura 24	Posición del <i>shock</i> para diferentes concentraciones de motores KIF1A.	76
Figura 25	Posición del <i>shock</i> para diferentes concentraciones de motores KIF1A con el gráfico espacial correspondiente a cada uno de los casos.	77
Figura 26	Esquema del modelo con τ para un sitio del bulk.	85

Figura 27	Esquema del modelo con tau válido para el extremo (-). 87
Figura 28	Esquema del modelo con tau válido para el extremo (+). 87
Figura 29	Perfiles de densidad en el estado estacionario. Comparación de las simulaciones y la aproximación analítica. 96
Figura 30	Tiempo de ligadura promedio en función de la tasa de hidrólisis para diferentes concentraciones de tau. 97
Figura 31	Longitud de corrida promedio como función de la tasa de hidrólisis para diferentes concentraciones de tau. 97
Figura 32	Histogramas para la longitud de corrida de KIF1A para diferentes concentraciones de tau. 98
Figura 33	Análisis de la localización de una pared de dominio con $C_{\text{tau}} = 0$. 99
Figura 34	Densidades de motores para $C_{\text{tau}} = 0$. 99
Figura 35	Análisis de la localización de una pared de dominio en presencia de tau con $C_{\text{tau}} = 0,005$. 100
Figura 36	Densidades de motores para $C_{\text{tau}} = 0,005$. 100
Figura 37	Análisis de la localización de una pared de dominio en presencia de tau con $C_{\text{tau}} = 0,01$. 101
Figura 38	Densidades de motores para $C_{\text{tau}} = 0,01$. 101
Figura 39	Análisis de la localización de una pared de dominio en presencia de tau con $C_{\text{tau}} = 0,1$. 102
Figura 40	Densidades de motores para $C_{\text{tau}} = 0,1$. 102
Figura 41	Distancia con la métrica propuesta entre un estado de 1-qubit y el estado correspondiente que emerge de un canal de depolarización. 116
Figura 42	Valor crítico para la DJS en función de la longitud de secuencia para un alfabeto binario y un umbral $s_0 = 0,999$. 138
Figura 43	Características de una célula eucariota. 141
Figura 44	La molécula de ATP como portadora de energía en la célula. 143
Figura 45	Localización de una pared de dominio. Secuencias construidas en la forma adoptada. 167
Figura 46	Localización de una pared de dominio. Secuencias construidas siguiendo la alternativa 1. 168

Figura 47 Localización de una pared de dominio. Secuencias construidas siguiendo la alternativa 2. 169

ÍNDICE DE TABLAS

Tabla 1	Longitudes y distribuciones de las subsecuencias del ejemplo. 39
Tabla 2	Subsecuencias reconstruidas. 46
Tabla 3	Parámetros del modelo para la Figura 20. 66
Tabla 4	Parámetros del modelo para la Figura 22. 69
Tabla 5	Desviación estándar de la posición del <i>shock</i> para algunos casos de la Figura 22. 71
Tabla 6	Parámetros del modelo para la Figura 23. 72
Tabla 7	Parámetros del modelo para las Figuras 24 y 25. 72
Tabla 8	Parámetros del modelo para la Figura 29. 89
Tabla 9	Parámetros del modelo para las Figuras 30 y 31. 90
Tabla 10	Parámetros del modelo para la Figura 32. 93
Tabla 11	Parámetros para la significancia máxima. 136
Tabla 12	Parámetros del modelo para la Figura 45. 166

ACRÓNIMOS

TI	Teoría de la Información
DKL	divergencia de Kullback-Leibler
DJS	divergencia de Jensen-Shannon
MT	microtúbulo
ADN	ácido desoxirribonucleico
ARN	ácido ribonucleico
ATP	adenosín trifosfato
ADP	adenosín difosfato
ASEP	proceso de exclusión simple asimétrico (por sus siglas en inglés)

- MCS paso Monte Carlo (por sus siglas en inglés)
- MAP proteína asociada al microtúbulo (por sus siglas en inglés)
- TIC Teoría de la Información Cuántica
- DJSC divergencia de Jensen-Shannon cuántica

Parte I

PARTE INTRODUCTORIA

INTRODUCCIÓN

1.1 LA VERSATILIDAD DE LA TEORÍA DE LA INFORMACIÓN

Este trabajo de tesis puede enmarcarse dentro de las tradiciones de la Biofísica y la Física Computacional (Parte [ii](#)) y de la Mecánica Cuántica (Parte [iii](#)). El hilo que conecta estas disciplinas, en apariencia dispares, es la Teoría de la Información.

Desde los pilares de la Teoría de la Información y, en especial, desde las nociones de distancia que de ella surgen, con la divergencia de Jensen-Shannon liderando el pelotón, hemos atacado esencialmente dos problemas: el estudio de la dinámica del motor molecular KIF_{1A} y la definición de una nueva métrica para estados cuánticos.

Nuestra primera contribución al estudio de la proteína motora KIF_{1A} fue desarrollar un método de análisis para detectar los casos patológicos que surgen de efectos colectivos de estos motores en el estado estacionario. Cuando el estado estacionario presenta coexistencia de fases y el filamento se embotella, el transporte intracelular se ve interrumpido.

En estos casos patológicos se produce una pared de dominio estable que separa una región de alta densidad de motores de otra de baja densidad. Con el método de análisis desarrollado podemos localizar, de manera automática, la posición en el filamento en la que se produce la pared de dominio.

El interés biofísico radica no sólo en entender los efectos colectivos entre estos motores moleculares sino también en que los problemas en el transporte intracelular se encuentran asociados a enfermedades de tipo neurodegenerativo.

El procedimiento tiene su raíz en la Teoría de la Información, particularmente en la adaptación de un método de segmentación de secuencias simbólicas no estacionarias, basado en la divergencia de Jensen-Shannon.

Como paso previo fue necesario desarrollar una simulación Monte Carlo que implementa el modelo para la dinámica del motor molecular KIF_{1A}.

Posteriormente extendimos el modelo para incluir el efecto de la interacción de los motores KIF_{1A} con una proteína estática (no-móvil) que también se liga al filamento, la proteína tau. Además de calcular casos límites analíticos para el modelo propuesto, se implementó la simulación para estudiar el estado estacionario. Se analiza no sólo la formación de paredes de dominio con el procedimiento desarrollado previamente, sino también otros observables de interés biofísico: el

tiempo promedio de ligadura del motor molecular y su longitud de corrida.

Por otro lado, la propuesta de una nueva métrica para estados cuánticos surgió de manera natural en el contexto de la Teoría de la Información y tiene su genealogía en la divergencia de Jensen-Shannon cuántica.

La versatilidad de la Teoría de la Información ha quedado demostrada con la gran cantidad de aplicaciones distintas que ha ido teniendo a lo largo de los años desde que fue introducida y formalizada en el año 1948. Si bien fue propuesta en el contexto de la teoría de comunicaciones encontró asidero en muchas áreas de la ciencia: en mecánica estadística, economía, ciencia de la computación, filosofía de la ciencia, etcétera.

De especial interés aquí es mencionar que se ha constituido en una herramienta fundamental en diversas áreas de la biofísica y la biología: en bioinformática, neurociencia, biología evolutiva, biología molecular y biología de sistemas, entre otras.

Entre las aplicaciones biofísicas concretas podemos mencionar la identificación de aquellas regiones de las secuencias que corresponden a sitios de ligaduras de proteínas en los ácidos nucleicos ([ADN](#) y [ARN](#)), la distinción de regiones que codifican de las que no codifican en el [ADN](#), el análisis de los trenes de disparos neuronales y de funciones superiores del sistema cognitivo.

Esta versatilidad propia de la Teoría de la Información nos permitió movernos con naturalidad en el ámbito de los problemas que estamos a punto de comenzar a discutir, a partir del uso de distintos cuantificadores de información.

1.2 ORGANIZACIÓN DE LA TESIS

En el [Capítulo 2](#) se hace una revisión de los conceptos básicos de Teoría de la Información, presentando aquellos que constituyen la base para el resto de la tesis, entre ellos la noción de entropía como información, la información mutua y medidas entrópicas de distinguibilidad.

En el [Capítulo 3](#) se introduce un método de segmentación de secuencias no estacionarias basado en la divergencia de Jensen-Shannon, con un ejemplo de aplicación para ilustrar su carácter recursivo.

En la [Parte ii](#) se presentan las aplicaciones biológicas desarrolladas. En el [Capítulo 4](#) se discute un procedimiento de localización de paredes de dominio para la dinámica de los motores moleculares KIF_{1A}. Este procedimiento está basado en el método de segmentación presentado en el [Capítulo 3](#). Se presentan las características básicas del motor molecular KIF_{1A}, el modelo matemático para su dinámica y se describe la simulación Monte Carlo desarrollada y las características del procedimiento por el cuál podemos identificar paredes de

dominio en el estado estacionario de este modelo. En el Capítulo 5 se presentan mejoras realizadas al modelo para la dinámica de KIF_{1A}, en particular se incluye la interacción con las proteínas tau que se encuentran en las sendas de los motores moleculares afectando la forma en que se ligan y en que se desplazan por las mismas. Se presentan las características básicas de estas proteínas y la forma en que interactúan con los motores moleculares KIF_{1A} y se describe el modelo propuesto para caracterizar esta interacción. Se estudian casos límites y se compara con el modelo anterior. Se analizan diversos observables de interés biológico para este nuevo modelo y se estudia la formación de *shocks* en presencia de esta proteína.

En la Parte iii se detallan las aplicaciones cuánticas desarrolladas. En el Capítulo 6 se introduce una nueva métrica para estados cuánticos. La propuesta está fuertemente inspirada en la divergencia de Jensen-Shannon cuántica y surge de manera natural en el contexto de la Teoría de la Información Cuántica. Se estudian algunas de sus propiedades y se ilustra su computabilidad con el cálculo explícito de la distancia para dos estados mixtos de un qubit.

Para finalizar y, a modo de énfasis, en el Capítulo 7 se presentan las conclusiones generales del trabajo.

My greatest concern was what to call it. I thought of calling it “information”, but the word was overly used, so I decided to call it “uncertainty”. When I discussed it with John von Neumann, he had a better idea. Von Neumann told me, “You should call it entropy, for two reasons.

*In the first place your uncertainty function has been used in statistical mechanics under that name, so it already has a name. In the second place, and more important, no one knows what entropy really is, so in a debate you will always have the advantage”*¹

— Claude E. Shannon²

En este capítulo se presentan y desarrollan algunas de las cantidades fundamentales de Teoría de la Información. Tiene un propósito doble, por un lado introducir los conceptos utilizados en este trabajo de tesis o necesarios para entenderlo y por otro, debido a que los tópicos de Teoría de la Información no forman parte de la formación estándar de un grado de física, constituir una introducción breve para el lector que se enfrenta a los conceptos por primera vez. Por este motivo, en la Sección 2.6 hay un compendio de bibliografía comentada que detalla las fuentes en las que se basa el contenido o las ideas de las distintas partes de este capítulo.

Comenzamos introduciendo la entropía de Shannon y algunos ejemplos para enfatizar algunas de sus propiedades en la Sección 2.1. Posteriormente, en la Sección 2.2 se definen los conceptos de entropía conjunta y condicional y en la Sección 2.3, el de información mutua. Finalizamos el capítulo presentando dos medidas de disimilitud entre distribuciones de probabilidad. En la Sección 2.4 se introduce la divergencia de Kullback-Leibler y algunas de sus propiedades, entre ellas su relación con la información mutua. En la Sección 2.5 se define la divergencia de Jensen-Shannon y se discuten algunas de sus propiedades. Por ser ésta, una cantidad clave en el desarrollo de la tesis

¹ Mi mayor preocupación era cómo llamarla. Pensé en llamarla “información”, pero la palabra estaba sobreutilizada, así que me decidí por “incertidumbre”. Cuando discutí este tema con John von Neumann, él tenía una idea mejor. Von Neumann me dijo, “Deberías llamarla entropía, por dos razones. En primer lugar, tu función incertidumbre ha sido utilizada en mecánica estadística bajo ese nombre, así que ya tiene un nombre. En segundo lugar, y más importante aún, nadie sabe que es realmente la entropía, así que en cualquier discusión siempre vas a tener ventaja”

² Conversación privada con Myron Tribus en 1961, como aparece citada en la Referencia [68]. El hecho de que la conversación entre Shannon y von Neumann haya efectivamente ocurrido de la manera en que Tribus lo cuenta ha sido puesto en duda a partir de una entrevista que Robert Price le hizo a Shannon en el año 1982. Ver por ejemplo la Referencia [24].

se incluye en la Sección 2.5.3 una profundización de sus interpretaciones en diversos contextos: en física estadística, en la misma Teoría de la Información, en estadística y en inferencia bayesiana.

2.1 ENTROPÍA DE SHANNON

En un trabajo pionero del año 1948, Claude E. Shannon [64] sienta las bases de lo que vendría posteriormente a llamarse Teoría de la Información (TI). La piedra fundamental de dicho trabajo es la *entropía*, con la cual Shannon pretende cuantificar la *incertidumbre* que tenemos acerca del resultado de un experimento o la *información* asociada a un dado proceso.

Vamos a hacer una construcción heurística de la definición de entropía con el propósito de convencernos en el camino de que la propuesta de Shannon tiene varias propiedades que están de acuerdo con la noción intuitiva de lo que una medida de información debería ser.

Para ello, consideremos una variable aleatoria X , con un rango discreto, que puede tomar los valores x_1, x_2, \dots, x_N . Denotamos por $p_i \equiv p(x_i) = P(X = x_i) \equiv P(x_i)$, a la probabilidad de que la variable aleatoria X tome un valor particular x_i . Con los requerimientos usuales para que la distribución de probabilidad esté bien definida:

$$p_i \geq 0 \quad \forall i = 1, 2, \dots, N; \quad \sum_{i=1}^N p_i = 1.$$

Antes de seguir avanzando hacia la definición recordemos cómo se calculan los promedios asociados a la variable aleatoria X si conocemos su distribución de probabilidad $P(X)$.

El *primer momento* (también llamado *valor medio* o sencillamente *promedio*) se define como:

$$\langle X \rangle = \sum_{i=1}^N x_i p(x_i), \quad (1)$$

es decir, sumando sobre todos los posibles resultados de la variable aleatoria X con un peso igual a la correspondiente probabilidad de que cada resultado ocurra.

Del mismo modo, el *segundo momento* se define del siguiente modo:

$$\langle X^2 \rangle = \sum_{i=1}^N x_i^2 p(x_i), \quad (2)$$

y la *varianza* (cuadrado de la *desviación estándar*) toma la forma:

$$\sigma^2 = \langle X^2 \rangle - \langle X \rangle^2. \quad (3)$$

Más generalmente, el promedio de cualquier función de la variable aleatoria, $f(X)$, es:

$$\langle f(X) \rangle = \sum_{i=1}^N f(x_i) p(x_i). \quad (4)$$

La pregunta que aquí nos interesa es entonces: ¿cuál es la *información promedio* que se obtiene de observar la variable aleatoria X ?

Para intentar responderla conviene quizá pensar primero en una pregunta ligeramente más sencilla: ¿cuán *informativo* es medir algún valor particular x_i ? ó ¿cuán *sorprendente* es saber que la variable aleatoria tomó el valor x_i ?

La respuesta dependerá de qué tan probable es el resultado x_i . Si la probabilidad $p(x_i)$ es grande no es muy informativo saber que sucedió, ¡porque pasa la mayoría de las veces! Por otro lado, si el resultado x_i tiene asociada una probabilidad pequeña, medir X y obtener el valor x_i es bastante informativo porque ocurre muy raramente.

Puede resultar más claro si pensamos en términos de que tan sorprendente es un resultado. Si la probabilidad $p(x_i)$ es alta el resultado x_i no es muy sorprendente como sí lo es cuando la probabilidad es baja.

En las vacaciones, jugando a los dados con parte de mi familia cuando en un tiro hice *generalá servida*³, además de alegrarme desmedidamente por el triunfo instantáneo, le saqué una foto y se la mandé a mi hermano para contarle. A mi hermano, a pesar de estar ocupado con sus amigos, no sólo no le molestó recibirla sino que le pareció una noticia interesante, ¡y ya sabemos cómo son los adolescentes cuando están con sus amigos! Mi (humilde) éxito comunicacional se debió simplemente a que una generalá servida es un evento extremadamente improbable, distinta hubiera sido su reacción si le avisaba cada vez que hacía un *full*⁴.

Otro ejemplo que podemos considerar es el de la variable aleatoria *resultado de un partido del Barcelona de Messi*. Esta variable podría tomar los valores *ganó*, *perdió* o *empató*. No es muy informativo o sorprendente enterarse de que *ganó*, en cambio es mucho más informativo saber que *empató* o que *perdió*⁵.

³ Obtener de un solo tiro los cinco dados iguales.

⁴ La probabilidad de hacer un *full servido* (tres y dos dados iguales) es 50 veces más que la de hacer una *generalá servida*. Conforme a las reglas del juego uno tiene tres tiros para armar una figura, pudiendo elegir qué dados deja en la mesa y cuáles vuelve a arrojar, por lo que el cálculo de la probabilidad de obtener un *full* es bastante más complicado, pero es (permítaseme la vaguedad) muchas veces más probable que el de la generalá servida. Sin ser rigurosos pero a los fines de hacernos una idea, vale la pena mencionar que en una partida típica es común que todos los jugadores hagan el *full*. Sin buscar una partida excepcional yo recuerdo haber hecho *full* 5 o 6 veces en un mismo juego, en cambio, la *generalá servida* la hice una única vez en toda mi vida.

⁵ Finalizada la primera rueda de la temporada 2012/2013 de la liga española de fútbol, sobre 19 partidos jugados el Barcelona ganó 18, empató 1 y no perdió ninguno, ver por ejemplo [1].

Resumiendo, entonces:

- $p(x_i)$ grande: “poco informativo” (“no muy sorprendente”)
- $p(x_i)$ chico: “bastante informativo” (“sorprendente”)

De este modo, podemos pensar que la información asociada a x_i va como la inversa de su probabilidad: $\sim \frac{1}{p(x_i)}$.

Otra propiedad que nos gustaría que una medida cuantitativa de la información tuviera es que la información de dos eventos independientes sea la suma de las informaciones individuales.

Si cierta cantidad de información está asociada al resultado de mi tirada de dados y cierta otra cantidad de información al resultado de un partido del Barcelona, no esperamos que una persona que obtenga la información de ambos resultados obtenga ni más ni menos información que la suma de los valores individuales. Mi tirada de dados y el resultado de un partido del Barcelona son eventos intuitivamente independientes. Distinta sería una hipotética situación en la que yo fuera el técnico del Barcelona y decidiera la formación del equipo tirando los dados, en ese caso los eventos no serían independientes, la información no sería la suma y yo no duraría mucho tiempo en mi cargo, seguramente.

Si consideramos dos variables aleatorias X e Y con distribuciones de probabilidad $P(X)$ y $P(Y)$ respectivamente, diremos que son *independientes* si la distribución de probabilidad conjunta $P(X, Y)$ se factoriza en la distribuciones de probabilidad correspondientes a cada variable, es decir si $P(X, Y) = P(X)P(Y)$.

Si recordamos la propiedad del logaritmo de *convertir productos en sumas*:

$$\log(a \cdot b) = \log(a) + \log(b),$$

podemos comprobar fácilmente que

$$\log\left(\frac{1}{P(x_i, y_j)}\right) = \log\left(\frac{1}{P(x_i)P(y_j)}\right) = \log\left(\frac{1}{P(x_i)}\right) + \log\left(\frac{1}{P(y_j)}\right),$$

tal como queríamos.

Por lo tanto, asociando al resultado x_i una *información*: $\log\left(\frac{1}{p(x_i)}\right)$ se satisfacen nuestros dos requisitos intuitivos que son: que la información sea grande (chica) cuando la probabilidad es chica (grande) y que la información asociada a dos eventos independientes sea la suma de la información de cada evento.

De esta manera, obtendremos la *información promedio* de observar la variable aleatoria X con distribución de probabilidad $P(X)$ si tomamos el valor medio de $\log\left(\frac{1}{p(x_i)}\right)$. De acuerdo a la Ecuación (4) es:

$$H(X) = \sum_{i=1}^N p(x_i) \log\left(\frac{1}{p(x_i)}\right) = - \sum_{i=1}^N p(x_i) \log(p(x_i)), \quad (5)$$

y esta expresión es la que se conoce como *entropía de Shannon* o información de Shannon.

Nos referiremos entonces indistintamente a la información asociada a la variable aleatoria X , a la entropía de la distribución $P(X)$, o a la incertidumbre asociada a un evento X , siempre haciendo referencia a esta cantidad y queriendo significar la información promedio que se obtiene al observar un gran número de veces el valor que toma la variable aleatoria X .

Como la entropía es un funcional de la distribución de probabilidad de X (no depende de los valores explícitos que la variable aleatoria puede tomar sino de sus probabilidades), en ocasiones puede resultar conveniente utilizar la notación $H(P)$ haciendo referencia explícita a la distribución de probabilidad P a la cual se encuentra asociada. Utilizaremos tanto los nombres como la notación de la forma que sea más conveniente en cada caso.

En TI es usual utilizar el logaritmo en base 2 y de esta manera la entropía se mide en *bits* (en el Ejemplo 2 de la Sección 2.1.2 discutiremos brevemente sobre el significado de un bit). En la Ecuación (5) hemos utilizado la notación $\log \equiv \log_2$ y lo mismo haremos de aquí en adelante salvo que se especifique lo contrario.

2.1.1 Breve digresión sobre la deducción formal de la entropía

La construcción que hicimos de la definición de entropía usó algunos conceptos intuitivos que tenemos sobre información, pero esto no debe confundirnos, la presentación de Shannon en su artículo de 1948 [64] tiene la rigurosidad de un teorema⁶. Shannon demostró que el único funcional $H(p_1, p_2, \dots, p_N)$ de las probabilidades p_1, p_2, \dots, p_N que satisface las siguientes tres asunciones:

1. H debe ser continua en los p_i ,
2. si todos los p_i son iguales, $p_i = \frac{1}{N} \forall i$, H debe ser una función monótonamente creciente de N ,
3. si un posible resultado se divide en dos alternativas sucesivas, el H original debe ser igual a la suma pesada de los valores individuales de los H ,

⁶ Es interesante reflexionar acerca del hecho de que lo que Shannon hizo es bastante diferente a la forma en que estamos acostumbrados a trabajar en la física para describir fenómenos naturales. Normalmente tenemos observaciones que pueden hacerse cuantitativas y buscamos estructuras matemáticas que puedan explicar de manera unificada esas observaciones. En contraste, Shannon toma un fenómeno cotidiano para el cual tenemos un lenguaje coloquial y se pregunta si ese lenguaje se puede hacer matemáticamente preciso sin referencia a datos cuantitativos. Ver Referencia [5] para una discusión más detallada de esta idea.

es:

$$H = -k \sum_{i=1}^N p_i \log(p_i),$$

que es exactamente lo mismo que la Ecuación (5), excepto ligeramente por la notación (debido a que Shannon no asocia la distribución de probabilidad a ninguna variable aleatoria como sí hicimos nosotros) y por el factor k . En la deducción de Shannon k es una constante arbitraria que fija las unidades en que se mide la entropía (la elección que hicimos en la Ecuación (5), es la que mide la entropía en bits).

2.1.2 Algunos ejemplos sencillos

En esta sección mostraremos algunos ejemplos que ilustran ciertas propiedades e interpretaciones de la entropía.

EJEMPLO 1: Distribución uniforme.

Consideremos el caso en que todos los posibles resultados de la variable aleatoria X tienen la misma probabilidad, $p(x_i) = \frac{1}{N} \forall i$.

De acuerdo a la Ecuación (5) tenemos que:

$$H(X) = \sum_{i=1}^N \frac{1}{N} \log(N) = \log(N).$$

Como el logaritmo es una función creciente, la entropía resulta una función creciente del número de resultados posibles. Esto es consistente con la noción intuitiva de que, si todos los resultados son igualmente probables, mientras mayor es el número de resultados posibles más información obtenemos al saber cuál de todos efectivamente ocurrió.

El hecho de que en este caso la entropía resulte una función creciente del número de resultados posibles es, como se mencionó en la Sección 2.1.1, una de las asunciones originales de Shannon.

EJEMPLO 2: Distribución uniforme: caso binario.

Consideremos el caso especial en que $N = 2$, conocido como caso binario, en el cual la variable aleatoria X sólo tiene dos posibles resultados, por ejemplo 0 y 1. Si la distribución es uniforme, es decir $P(X = 0) = P(X = 1) = \frac{1}{2}$, entonces por el resultado del ejemplo anterior tenemos que:

$$H(X) = \log(2) = 1 \text{ bit.}$$

Por lo tanto, un bit es la información contenida en la elección entre dos alternativas igualmente probables.

EJEMPLO 3: Certidumbre.

En el otro extremo del Ejemplo 1 consideremos el caso en que tenemos certeza respecto al resultado de la variable X , es decir, el caso en que $p(x_j) = 1$ y $p(x_i) = 0 \quad \forall i \neq j$.

A partir de la Ecuación (5) tenemos en este caso (con ayuda de la regla de L'Hôpital para calcular el límite):

$$H(X) = 0 \text{ bit.}$$

Esto significa que la información promedio que obtenemos al observar el resultado de una variable que siempre da x_j es 0 bit. ¿Para qué habríamos de tomarnos la molestia de observar una variable cuyo resultado ya conocemos?⁷ Eso es claramente no-informativo y la definición de Shannon es consistente.

EJEMPLO 4: Caso binario general.

Consideremos el caso binario $N = 2$. La distribución más general en este caso depende de un único parámetro (por la condición de normalización de la distribución): $P(X = 0) = p$ y $P(X = 1) = 1 - p$.

En este caso la Ecuación (5) da lugar a :

$$H(X) = -p \log(p) - (1 - p) \log(1 - p),$$

cuyo gráfico podemos ver en la Figura 1.

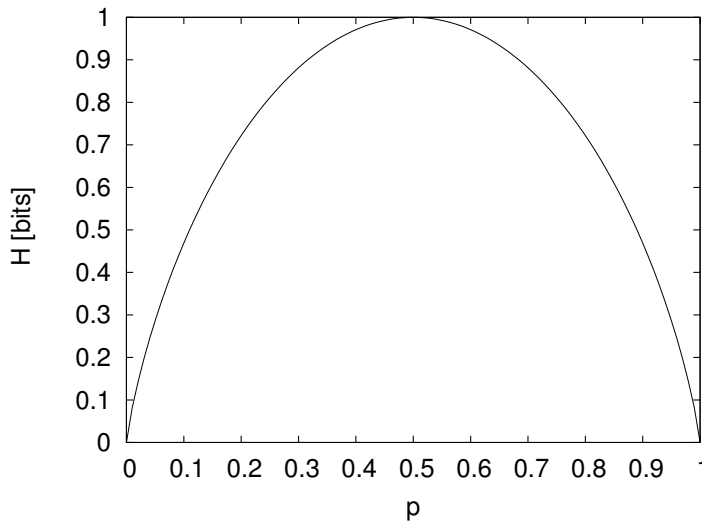


Figura 1: Entropía para una variable aleatoria binaria.

Los dos casos en los cuales tenemos certeza acerca del resultado (como vimos en el Ejemplo 3) son los correspondientes a $p = 0$ y

⁷ Evitar toda tentación epistemológica relacionada a las consecuencias de la falsación de un argumento. Claramente es informativo medir la velocidad de la luz con la idea de contrastar la hipótesis de su constancia en el contexto de la relatividad especial, por ejemplo. Aquí nos referimos a una situación ideal en que se sabe con certeza la distribución de probabilidad.

$p = 1$ que dan ambos la entropía mínima $H = 0$ bit. Por otro lado, el caso equiprobable, correspondiente a $p = \frac{1}{2}$ da la entropía máxima $H = 1$ bit (como vimos en el Ejemplo 2). Cualquier otro valor de p da un valor intermedio para la entropía. Notar además la simetría de la curva respecto al valor $p = \frac{1}{2}$ que refleja el hecho de que dado un valor de p cualquiera, la información (entropía) del sistema es la misma sin importar cuál de los dos resultados posibles (0 o 1) tiene esa probabilidad. Por ejemplo, el caso $p = \frac{1}{3}$ equivale en términos de información al caso $p = \frac{2}{3}$; en el primero, $x = 1$ es el resultado que tiene probabilidad $\frac{1}{3}$ y en el segundo, $x = 0$.

EJEMPLO 5: Preguntas binarias.

Consideremos una variable aleatoria X que puede tomar los valores $\{a, b, c, d\}$ con las siguientes probabilidades:

$$X = \begin{cases} a, & p(x = a) = 1/2 \\ b, & p(x = b) = 1/4 \\ c, & p(x = c) = 1/8 \\ d, & p(x = d) = 1/8 \end{cases}$$

Siguiendo la prescripción de la Ecuación (5), se puede ver que en este caso, la entropía es $H(X) = 1,75$ bits.

Supongamos que queremos conocer el resultado de la variable aleatoria y que tenemos la posibilidad de hacer preguntas binarias, esto es, preguntas cuya respuesta es sí/no. Un esquema eficiente para hacerlo es el que se muestra en la Figura 2.

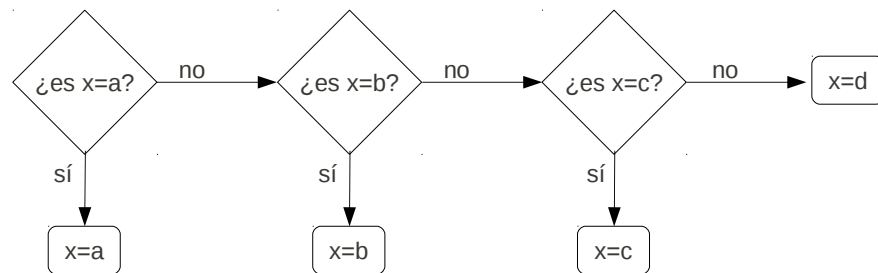


Figura 2: Esquema de preguntas binarias para determinar el valor de la variable aleatoria X .

Así, si el resultado es $x = a$ hará falta una única pregunta, si el resultado es $x = b$ harán falta dos preguntas y si el resultado es $x = c$ o $x = d$ serán necesarias tres preguntas. Por lo tanto, el número

promedio de preguntas binarias necesarias para conocer el valor de la variable aleatoria X con el esquema que propusimos es:

$$\begin{aligned}\langle \mathcal{N} \rangle &= 1 \cdot P(x = a) + 2 \cdot P(x = b) + 3 \cdot (P(x = c) + P(x = d)) \\ &= 1 \cdot \frac{1}{2} + 2 \cdot \frac{1}{4} + 3 \cdot \left(\frac{1}{8} + \frac{1}{8} \right) \\ &= 1,75.\end{aligned}$$

¡Exactamente el mismo valor que la entropía! Esto no es una casualidad sino que se puede demostrar [16] que el *mínimo número de preguntas binarias esperado* para determinar el valor de X , $\langle \mathcal{N}_{\min} \rangle$, tiene las siguientes cotas en términos de la entropía:

$$H(X) \leq \langle \mathcal{N}_{\min} \rangle \leq H(X) + 1. \quad (6)$$

En el ejemplo, hemos elegido un esquema óptimo en el cual se cumple la igualdad $H(X) = \langle \mathcal{N}_{\min} \rangle = \langle \mathcal{N} \rangle$.

Si consideramos un ejemplo ligeramente diferente en el que todos los estados tienen la misma probabilidad de ocurrir ($1/4$), la entropía vale $H(X) = 2$ bits, cualquier orden de preguntas es equivalente y el mínimo número de preguntas esperadas es igual a $\langle \mathcal{N}_{\min} \rangle = 2,25$ (que, por supuesto, satisface también la Ecuación (6)).

El punto importante del ejemplo es que la entropía da una cota inferior al número de preguntas binarias esperadas para determinar el valor de X .

En esta misma línea de consideraciones vale la pena mencionar que la entropía también da una cota a la longitud promedio necesaria de un mensaje que describe los resultados de X . Entrar en los detalles de esta afirmación implicaría introducir qué son las palabras de códigos, cómo construir códigos eficientes, etcétera. Aquí se pretende simplemente destacar que la entropía en sí misma está asociada a una *representación eficiente*. Es bastante impresionante que la única forma de cuantificar cuánto aprendemos (información que ganamos al oír la respuesta) sea medir cuánto espacio es necesario para escribirlo (representar la respuesta en el menor espacio posible). El lector interesado puede profundizar estas ideas leyendo la Referencia [16].

2.2 ENTROPÍA CONJUNTA Y ENTROPÍA CONDICIONAL

En ocasiones resulta necesario conocer la información promedio asociada no a una sino a un par de variables aleatorias. Para ello, debemos generalizar la Ecuación (5). Si $P(X, Y)$ es la distribución conjunta asociada al par de variables aleatorias X e Y , por analogía con el razonamiento que hicimos en la Sección 2.1, resulta natural definirla del siguiente modo:

$$H(X, Y) = - \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log(p(x, y)), \quad (7)$$

en donde \mathcal{X} es el rango de X , \mathcal{Y} es el rango de Y y las sumatorias se extienden sobre todo el rango de cada variable.

La expresión de la Ecuación (7) se conoce como *entropía conjunta* y cuantifica la incertidumbre asociada al par de variables aleatorias o, como se dijo, la información promedio asociada a observar las variables simultáneamente. Esta última interpretación queda más clara si escribimos la entropía conjunta del siguiente modo:

$$H(X, Y) = \left\langle \log \left(\frac{1}{P(X, Y)} \right) \right\rangle,$$

en donde el valor medio se calcula ahora utilizando la distribución conjunta $P(X, Y)$.

Una pregunta relacionada pero diferente se puede plantear al suponer que tenemos dos variables aleatorias y conocemos el resultado de una, ¿cuánta incertidumbre permanece respecto a la variable restante? o dicho de otro modo ¿cuánta información promedio obtenemos de observar una variable *dado que* ya conocemos el resultado de otra?.

La respuesta la provee la *entropía condicional* $H(Y|X)$ que se define del siguiente modo:

$$\begin{aligned} H(Y|X) &= \sum_{x \in \mathcal{X}} p(x) H(Y|X = x) & (8) \\ &= \sum_{x \in \mathcal{X}} p(x) \cdot \left(- \sum_{y \in \mathcal{Y}} p(y|x) \log(p(y|x)) \right) \\ &= - \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x) p(y|x) \log(p(y|x)) \\ &= - \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log(p(y|x)) & (9) \\ &= \left\langle \log \left(\frac{1}{P(Y|X)} \right) \right\rangle. \end{aligned}$$

La Ecuación (9) da la información promedio de observar Y dado que ya conocemos X , pero sin importar el valor particular que X tomó. Notar que en la definición, Ecuación (8), se promedia sobre todos los posibles valores de X (cada valor está pesado con su probabilidad de ocurrencia, $p(x)$).

La naturalidad de las definiciones de esta sección se hace más evidente a la luz del siguiente resultado: la entropía conjunta de dos variables aleatorias X e Y es igual a la entropía de una más la entropía condicional de la otra, es decir:

$$\begin{aligned} H(X, Y) &= H(X) + H(Y|X) & (10) \\ &= H(Y) + H(X|Y). \end{aligned}$$

La prueba, que es bastante directa a partir de las definiciones, puede verse en el Apéndice A.1.

Podemos parafrasear la afirmación anterior diciendo que la información promedio que obtenemos al observar un par de variables aleatorias es igual a la información promedio de observar una de ellas sumada a la información promedio de observar la segunda condicionada a la primera.

En el caso particular en que las variables aleatorias son independientes, $P(X, Y) = P(X)P(Y)$ y de la Ecuación (9) se puede ver que en este caso $H(Y|X) = H(Y)$, con lo que la Ecuación (10) se reduce a $H(X, Y) = H(X) + H(Y)$. Esto significa, tal como insistimos en la Sección 2.1, que la información asociada a dos variables aleatorias independientes es la suma de las informaciones individuales.

2.3 INFORMACIÓN MUTUA

La igualdad para la entropía conjunta dada en la Ecuación (10) se puede reescribir del siguiente modo:

$$H(X) - H(X|Y) = H(Y) - H(Y|X). \quad (11)$$

Como $H(X|Y)$ representa la incertidumbre que queda en X luego de conocer Y (la información promedio que queda por conocer acerca de X dado que conocemos la información asociada a Y), y $H(X)$ representa la información promedio total asociada a X , la resta del lado izquierdo de la igualdad nos da la información que X e Y comparten. Otra forma de decirlo es que el lado izquierdo da la resta de la información total de X menos todo lo que tiene que ver sólo con X y no con Y , resultando en la información que comparten. Un razonamiento similar se puede utilizar para el lado derecho de la igualdad, obteniéndose nuevamente la información que comparten X e Y .

Esto sugiere la introducción de una nueva cantidad, llamada *información mutua*, $I(X, Y)$, asociada a las variables aleatorias X e Y y que cuantifica la información promedio que las variables comparten:

$$\begin{aligned} I(X, Y) &= H(X) - H(X|Y) \\ &= H(Y) - H(Y|X), \end{aligned} \quad (12)$$

esta cantidad es, por definición, una cantidad simétrica ante intercambios de X e Y . La información que X comparte con Y es la misma que la que Y comparte con X .

2.4 DIVERGENCIA DE KULLBACK-LEIBLER

Regresemos al caso de una única variable aleatoria X con distribución de probabilidad $P(X)$, y supongamos que por alguna razón (bien sea por error o porque disponemos de una versión aproximada) asociamos la distribución $Q(X)$ a la variable, en lugar de su distribución verdadera $P(X)$. En esta sección presentaremos la *divergencia de*

Kullback-Leibler o *entropía relativa* que es una medida de la ineficiencia de asumir una distribución de probabilidad incorrecta o aproximada en lugar de la distribución verdadera.

En la Sección 2.1 vimos que la información asociada a un resultado particular x de la variable aleatoria X con distribución de probabilidad $P(X)$ era: $\log\left(\frac{1}{p(x)}\right)$. La diferencia de información al considerar dos distribuciones $P(X)$ y $Q(X)$ para el mismo resultado x será pues:

$$\log\left(\frac{1}{q(x)}\right) - \log\left(\frac{1}{p(x)}\right) = \log\left(\frac{p(x)}{q(x)}\right),$$

por lo tanto, si la verdadera distribución de X es $P(X)$ la diferencia de información promedio será:

$$D_{\text{KL}}(P, Q) = \left\langle \log\left(\frac{P(X)}{Q(X)}\right) \right\rangle_P = \sum_{x \in \mathcal{X}} p(x) \log\left(\frac{p(x)}{q(x)}\right), \quad (13)$$

y se conoce como divergencia de Kullback-Leibler (**DKL**).

A fines de ilustrar el significado de la **DKL** podemos mencionar que, si conocemos la distribución verdadera, P , de la variable aleatoria, podríamos construir un código para los posibles resultados de X con una longitud de palabra promedio igual a $H(P)$. Si, en cambio, utilizáramos un código basado en la distribución Q , la longitud de palabra promedio sería igual a $H(P) + D_{\text{KL}}(P, Q)$. Eso significa que si transmitiéramos los resultados de la variable X con un código construido a partir de Q estaríamos enviando en promedio $D_{\text{KL}}(P, Q)$ bits de más.

2.4.1 Algunas propiedades de la divergencia de Kullback-Leibler

Revisemos algunas de las propiedades más importantes de la **DKL**.

Si $P(X)$ y $Q(X)$ son dos distribuciones posibles para la variable aleatoria X entonces se cumple que:

$$D_{\text{KL}}(P(X), Q(X)) \geq 0, \quad (14)$$

con $D_{\text{KL}}(P(X), Q(X)) = 0$ si y sólo si $P(x) = Q(x) \forall x \in \mathcal{X}$.

Por esta propiedad es que a menudo se utiliza la **DKL** como una medida de disimilitud entre distribuciones de probabilidad, sin embargo, hay que tener presente que no constituye una verdadera métrica en el espacio de las probabilidades, de hecho:

1. no es simétrica respecto a sus argumentos:

$$D_{\text{KL}}(P(X), Q(X)) \neq D_{\text{KL}}(Q(X), P(X)), \quad (15)$$

2. no cumple la desigualdad triangular,
3. no es acotada.

Otra limitación importante que tiene la [DKL](#) es que requiere que la distribución $P(X)$ sea absolutamente continua respecto a la distribución $Q(x)$ [39]. Esto significa que se debe cumplir que $P(X = x) = 0$ siempre que $Q(X = x) = 0$. En caso contrario, la divergencia no está definida (diverge).

2.4.2 La información mutua como caso particular de la divergencia de Kullback-Leibler

Escribiendo explícitamente la definición de la información mutua, Ecuación (12), podemos reinterpretarla a la luz de la [DKL](#). Haciendo uso de la definición de entropía, Ecuación (5), y de entropía condicional, Ecuación (9), podemos escribir:

$$\begin{aligned} I(X, Y) &= H(Y) - H(Y|X) \\ &= - \sum_{y \in \mathcal{Y}} p(y) \log(p(y)) + \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log(p(y|x)). \end{aligned}$$

Si ahora utilizamos que $\sum_{x \in \mathcal{X}} p(x, y) = p(y)$ y juntamos los sumandos, tenemos que:

$$\begin{aligned} I(X, Y) &= - \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log(p(y)) + \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log(p(y|x)) \\ &= \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log\left(\frac{p(y|x)}{p(y)}\right). \end{aligned}$$

Por último utilicemos que $p(x, y) = p(y|x)p(x)$ para obtener:

$$\begin{aligned} I(X, Y) &= \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log\left(\frac{p(x, y)}{p(x)p(y)}\right) \\ &= D_{\text{KL}}(P(X, Y), P(X)P(Y)). \end{aligned} \tag{16}$$

Esto significa que podemos pensar que la información mutua cuantifica cuán lejos está la distribución conjunta verdadera $P(X, Y)$ del caso en que las variables aleatorias X e Y son independientes.

En el caso en que la [DKL](#) se anula, eso significa que $P(X, Y) = P(X)P(Y)$, es decir que las variables aleatorias son en realidad independientes, como es claro igualando a cero la definición de la información mutua (no comparten nada de información, $H(Y|X) = H(Y)$).

La no-negatividad de la divergencia de Kullback-Leibler que vimos en la Sección 2.4.1 se demuestra como una aplicación de la desigualdad de Jensen (ver Apéndice A.2). Un corolario de esta propiedad de la divergencia, a la luz de la Ecuación (16), es la no-negatividad de la información mutua:

$$I(X, Y) \geq 0.$$

La igualdad se cumple sí y sólo si X e Y son independientes.

Carece de sentido (tanto intuitivo como formal) hablar de que la información compartida por dos variables aleatorias es negativa.

2.5 DIVERGENCIA DE JENSEN-SHANNON

Si bien la *DKL* como medida de disimilitud aparece de manera natural en el contexto de la *TI*, tiene también importantes limitaciones como discutimos en la Sección 2.4.1. Es por eso que J. Lin [39] introduce una nueva cantidad que está íntimamente relacionada con la *DKL* y la bautiza con el nombre de *divergencia de Jensen-Shannon*.

Lin propone la medida:

$$D_{JS}^{1/2}(P, Q) = \frac{1}{2} \left(D_{KL} \left(P, \frac{P+Q}{2} \right) + D_{KL} \left(Q, \frac{P+Q}{2} \right) \right). \quad (17)$$

Cada término de esta expresión da el valor de la *DKL* entre una de las distribuciones y el promedio de ambas. Esto soluciona el problema de la continuidad absoluta respecto a la segunda distribución pues la distribución promedio no se anula salvo que se anulen ambas distribuciones. Además, hace un promedio entre estas dos divergencias para garantizar la simetría (cambiando P por Q cambiaríamos simplemente el orden de los sumandos).

Utilizando la definición de la *DKL*, Ecuación (13), es posible reescribir la Ecuación (17) en término de la entropía, Ecuación (5), del siguiente modo:

$$D_{JS}^{1/2}(P, Q) = H \left(\frac{P+Q}{2} \right) - \frac{1}{2} (H(P) + H(Q)). \quad (18)$$

Acto seguido, Lin generaliza la expresión para proponer la divergencia de Jensen-Shannon (*DJS*) entre dos distribuciones P y Q de la siguiente forma:

$$D_{JS}(P, Q) = H(\pi_1 P + \pi_2 Q) - (\pi_1 H(P) + \pi_2 H(Q)), \quad (19)$$

en donde π_1 y π_2 son los pesos asociados a las distribuciones P y Q , respectivamente. Estos pesos son números reales no negativos tales que $\pi_1 + \pi_2 = 1$. Con esta propuesta, es posible asignar pesos diferentes a las distribuciones de acuerdo a su importancia⁸.

La *DJS* propuesta por Lin está fuertemente inspirada en un trabajo previo de C.R. Rao [57], en el cual propone una cantidad llamada *diferencia de Jensen* para una entropía general. En el caso particular en el que se utiliza la entropía de Shannon, la diferencia de Jensen, propuesta por Rao, coincide con la *DJS*.

2.5.1 Algunas propiedades de la divergencia de Jensen-Shannon

Repasemos alguna de las propiedades más importantes que tiene la *DJS*, a saber:

⁸ A criterio del usuario. Más adelante, cuando discutamos algunas interpretaciones de la *DJS* veremos que en ocasiones la elección de los pesos aparece de forma bastante natural.

1. si $P(X)$ y $Q(X)$ son dos distribuciones posibles para la variable aleatoria X entonces se cumple que:

$$D_{JS}(P(X), Q(X)) \geq 0, \quad (20)$$

con $D_{JS}(P(X), Q(X)) = 0$ si y sólo si $P(x) = Q(x) \forall x \in \mathcal{X}$;

2. siempre está bien definida (no requiere continuidad absoluta para las distribuciones a comparar);
3. es acotada:

$$0 \leq D_{JS}(P, Q) \leq 1.$$

Por otro lado la **DJS** en su versión con los pesos iguales, Ecuación (17), tiene dos propiedades adicionales a las del caso general, que son:

1. es simétrica respecto a sus argumentos:

$$D_{JS}^{1/2}(P, Q) = D_{JS}^{1/2}(Q, P); \quad (21)$$

2. es el cuadrado de una métrica [21], esto significa que:

$$d_{JS}(P, Q) = \sqrt{D_{JS}^{1/2}(P, Q)}$$

satisface, además de las propiedades antes mencionadas, la desigualdad triangular: si $P(X)$, $Q(X)$ y $R(X)$ son distribuciones de probabilidad, entonces

$$d_{JS}(P, Q) \leq d_{JS}(P, R) + d_{JS}(R, Q).$$

Estas propiedades hacen que la **DJS** sea una buena medida de disimilitud entre distribuciones de probabilidad y que d_{JS} sea una verdadera distancia⁹.

2.5.2 Divergencia de Jensen-Shannon generalizada

Una de las características más interesantes de la **DJS** es que puede ser generalizada para comparar un número arbitrario de distribuciones. Consideremos m distribuciones de probabilidad $P^{(1)}, P^{(2)}, \dots, P^{(m)}$ con pesos asociados $\pi_1, \pi_2, \dots, \pi_m$, con las restricciones habituales tanto para las distribuciones como para los pesos¹⁰. La *divergencia de Jensen-Shannon generalizada* se define entonces de la siguiente manera:

$$D_{JS}(P^{(1)}, \dots, P^{(m)}) = H\left(\sum_{i=1}^m \pi_i P^{(i)}\right) - \left(\sum_{i=1}^m \pi_i H(P^{(i)})\right). \quad (22)$$

⁹ En el sentido matemático: es definida positiva, simétrica, se anula sólo si los elementos que está comparando son iguales y cumple la desigualdad triangular.

¹⁰ No negatividad y normalización.

Notablemente, las propiedades mencionadas en la Sección 2.5.1 para el caso de dos distribuciones se verifican también para esta versión generalizada. Así la DJS es no negativa y se anula si y sólo si todas las distribuciones son iguales, y está siempre bien definida (no requiere distribuciones absolutamente continuas). Además es “simétrica” ante el intercambio de dos distribuciones cualesquiera, siempre y cuando inter cambiemos junto con la distribuciones sus respectivos pesos.

2.5.3 Interpretaciones de la divergencia de Jensen-Shannon

La DJS juega un rol crucial en este trabajo de tesis por lo que vale la pena discutir un poco acerca de algunas de las interpretaciones de la DJS en diversos contextos de aplicación.

2.5.3.1 En física estadística

En el contexto de la física estadística la DJS puede interpretarse como la *entropía de mezclas por partícula*¹¹ [23].

Consideremos m recipientes, cada uno de los cuales contiene una mezcla de k gases ideales y denotemos $f^{(j)} = (f_1^{(j)}, f_2^{(j)}, \dots, f_k^{(j)})$ al vector que da la fracción molar de los k gases en el recipiente j -ésimo, para $j = 1, 2, \dots, m$.

La fracción molar es $f_i^{(j)} = \frac{n_i^{(j)}}{n^{(j)}}$ donde $n_i^{(j)}$ es el número de moléculas del gas i -ésimo en el recipiente j -ésimo y $n^{(j)}$ es el número total de moléculas en el recipiente j -ésimo.

Se cumple que $n^{(j)} = \sum_{i=1}^k n_i^{(j)}$, $\forall j = 1, \dots, m$.

Consideremos la situación en que cada recipiente está a la misma temperatura, T , y tiene la misma densidad, $\frac{1}{v}$, (v es el volumen por partícula). Como todos los gases son ideales se cumplirá:

$$P_j V_j = n^{(j)} k_B T \quad \forall j = 1, \dots, m$$

donde k_B es la constante de Boltzmann, P_j y V_j son la presión y el volumen del recipiente j -ésimo, respectivamente.

Como el volumen por partícula, v , y la temperatura, T , son los mismos en todos los recipientes, la presión también será la misma. Esto se puede ver reescribiendo la expresión anterior del siguiente modo:

$$P_j = \frac{n^{(j)}}{V_j} k_B T = \frac{k_B T}{v} \equiv P \quad \forall j = 1, \dots, m.$$

Como todos son gases ideales, la entropía del recipiente j -ésimo será la suma de la de cada uno de los gases que contiene, por lo tanto:

$$S^{(j)} = \sum_{i=1}^k \left(n_i^{(j)} s_{i0} + n_i^{(j)} c_i k_B \ln \left(\frac{T}{T_0} \right) + n_i^{(j)} k_B \ln \left(\frac{V_j}{n_i^{(j)} v_0} \right) \right),$$

¹¹ En realidad sería una *entropía de la mezcla de mezclas*.

donde T_0 y v_0 son una temperatura y un volumen por partícula de referencia, c_i es una constante que depende del tipo de gas (si es monoatómico, diatómico, etcétera) y s_{i0} es un valor que depende de la referencias elegidas y del valor de c_i ¹².

Utilizando que $V_j = n^{(j)}v$, entonces:

$$S^{(j)} = \sum_{i=1}^k \left(n_i^{(j)} s_{i0} + n_i^{(j)} c_i k_B \ln \left(\frac{T}{T_0} \right) + n_i^{(j)} k_B \ln \left(\frac{n^{(j)}}{n_i^{(j)}} \right) \right) + k_B n^{(j)} \ln \left(\frac{v}{v_0} \right).$$

Como nuestro sistema está constituido por m recipientes, la entropía total será la suma de la entropía de cada uno de ellos, por lo tanto:

$$\begin{aligned} S_\alpha &= \sum_{j=1}^m S^{(j)} \\ &= \sum_{j=1}^m \sum_{i=1}^k \left(n_i^{(j)} s_{i0} + n_i^{(j)} c_i k_B \ln \left(\frac{T}{T_0} \right) + n_i^{(j)} k_B \ln \left(\frac{n^{(j)}}{n_i^{(j)}} \right) \right) + \\ &\quad + \sum_{j=1}^m k_B n^{(j)} \ln \left(\frac{v}{v_0} \right). \end{aligned}$$

Utilizando que $f_i^{(j)} = \frac{n_i^{(j)}}{n^{(j)}}$ y $\sum_{j=1}^m n^{(j)} = N$, finalmente:

$$\begin{aligned} S_\alpha &= \sum_{j=1}^m \sum_{i=1}^k \left(n_i^{(j)} s_{i0} + n_i^{(j)} c_i k_B \ln \left(\frac{T}{T_0} \right) - n_i^{(j)} k_B \ln \left(f_i^{(j)} \right) \right) + \\ &\quad + k_B N \ln \left(\frac{v}{v_0} \right). \end{aligned} \quad (23)$$

Esta es la entropía total de los m recipientes donde cada uno tiene su contenido separado de los demás. Si ahora removemos las particiones entre los recipientes de modo que todos los gases de todos los recipientes puedan explorar el volumen total, la nueva expresión diferirá de la que acabamos de encontrar. De hecho, la segunda ley de la termodinámica nos dice que la entropía total aumentará.

Calculemos la nueva expresión. Ahora, cada tipo de gas explora el volumen total, V , sin interactuar con los demás gases (todos son gases ideales). Por lo tanto, cada tipo de gas aportará un término a la entropía sin importar de que recipiente provino (las moléculas son indistinguibles).

El volumen por partícula de la mezcla será el mismo, puesto que:

$$V = \sum_{j=1}^m V_j = \sum_{j=1}^m n^{(j)}v = Nv.$$

¹² Por la expresión de la entropía de un gas ideal que aquí se da por conocida, ver por ejemplo la Referencia [10].

La temperatura, T , se mantiene igual y por lo tanto también se mantendrá igual la presión, ya que se cumple: $PV = Nk_B T$.

Por lo tanto, la entropía total después de remover las particiones será:

$$S_d = \sum_{i=1}^k \left(n_i s_{i0} + n_i c_i k_B \ln \left(\frac{T}{T_0} \right) + n_i k_B \ln \left(\frac{V}{n_i v_0} \right) \right),$$

donde, como dijimos, sumamos sobre cada tipo de gas y $n_i = \sum_{j=1}^m n_i^{(j)}$ es la suma de todas las moléculas del gas i -ésimo que originalmente estaban separadas en los recipientes.

Trabajemos algebraicamente con esa expresión:

$$S_d = \sum_{i=1}^k \left(\left(\sum_{j=1}^m n_i^{(j)} \right) s_{i0} + \left(\sum_{j=1}^m n_i^{(j)} \right) c_i k_B \ln \left(\frac{T}{T_0} \right) \right) + \sum_{i=1}^k \left(\left(\sum_{j=1}^m n_i^{(j)} \right) k_B \ln \left(\frac{V}{\left(\sum_{l=1}^m n_i^{(l)} \right) v_0} \right) \right).$$

Reordenando términos y utilizando que $\sum_{i=1}^k \sum_{j=1}^m n_i^{(j)} = N$ y $V = Nv$, tenemos que:

$$\begin{aligned} S_d &= \sum_{i=1}^k \sum_{j=1}^m \left(n_i^{(j)} s_{i0} + n_i^{(j)} c_i k_B \ln \left(\frac{T}{T_0} \right) \right) + \\ &+ \sum_{i=1}^k \sum_{j=1}^m n_i^{(j)} k_B \ln \left(\frac{Nv}{\left(\sum_{l=1}^m n_i^{(l)} \right) v_0} \right) \\ &= \sum_{i=1}^k \sum_{j=1}^m \left(n_i^{(j)} s_{i0} + n_i^{(j)} c_i k_B \ln \left(\frac{T}{T_0} \right) \right) + \\ &+ Nk_B \ln \left(\frac{v}{v_0} \right) - \sum_{i=1}^k \sum_{j=1}^m n_i^{(j)} k_B \ln \left(\frac{\sum_{l=1}^m n_i^{(l)}}{N} \right). \end{aligned} \quad (24)$$

La entropía de mezclas será la diferencia entre la entropía después de abrir las compuertas, Ecuación (24), y antes de abrirlas, Ecuación (23):

$$\begin{aligned} S_{mezcla} &= S_d - S_a \\ &= - \sum_{i=1}^k \sum_{j=1}^m n_i^{(j)} k_B \ln \left(\frac{\sum_{l=1}^m n_i^{(l)}}{N} \right) + \sum_{i=1}^k \sum_{j=1}^m n_i^{(j)} k_B \ln \left(f_i^{(j)} \right). \end{aligned}$$

En la primera sumatoria la única dependencia en j está en los $n_i^{(j)}$ por lo que podemos reescribir:

$$\sum_{j=1}^m n_i^{(j)} = N \sum_{j=1}^m \frac{n_i^{(j)}}{n^{(j)}} \frac{n^{(j)}}{N} = N \sum_{j=1}^m f_i^{(j)} \frac{n^{(j)}}{N} \equiv N f_i, \quad (25)$$

lo mismo puede utilizarse en el argumento del logaritmo del primer sumando, con l en el rol de j .

Si además utilizamos que $n_i^{(j)} = n^{(j)} f_i^{(j)}$, podemos reescribir la entropía de mezclas del siguiente modo:

$$\begin{aligned} S_{mezcla} &= - \sum_{i=1}^k N f_i k_B \ln(f_i) + \sum_{i=1}^k \sum_{j=1}^m n^{(j)} f_i^{(j)} k_B \ln(f_i^{(j)}) \\ &= N k_B \left(- \sum_{i=1}^k f_i \ln(f_i) \right) - N k_B \sum_{j=1}^m \frac{n^{(j)}}{N} \left(- \sum_{i=1}^k f_i^{(j)} \ln(f_i^{(j)}) \right) \\ &= N k_B \ln(2) \left(H(f) - \sum_{j=1}^m \frac{n^{(j)}}{N} H(f^{(j)}) \right), \end{aligned}$$

donde H es la entropía de Shannon.

El vector de fracciones molares de la mezcla, f , cuyas componentes definimos en la Ecuación (25), tiene la forma:

$$f = \sum_{j=1}^m \frac{n^{(j)}}{N} f^{(j)},$$

por lo tanto, recordando la Ecuación (22), podemos identificar en la entropía de mezcla la **DJS** entre las fracciones molares de los recipientes para obtener:

$$S_{mezcla} = N k_B \ln(2) D_{JS} \left(f^{(1)}, f^{(2)}, \dots, f^{(m)} \right), \quad (26)$$

si elegimos los pesos como $\pi_j = \frac{n^{(j)}}{N}$.

Por lo tanto, la **DJS** puede interpretarse como la entropía de mezclas por partícula en unidades de $k_B \ln(2)$. Esto significa que podemos calcular la entropía de la mezcla calculando directamente la **DJS** entre las fracciones molares de cada recipiente, antes de mezclar.

Algo que vale la pena enfatizar es que este cálculo de la entropía de la mezcla (de mezclas) es bien general e incluye como caso particular el ejemplo típico de los libros de termodinámica (ver por ejemplo la Referencia [10]) en el que se presenta la entropía de mezcla. En dicho caso se trabaja con recipientes que tienen, cada uno, un único gas ideal en su interior.

2.5.3.2 En teoría de la información

Dentro de la misma **TI**, la **DJS** puede interpretarse como información mutua [23].

Consideremos una secuencia S de N símbolos escogidos de un alfabeto $\mathcal{A} = \{a_1, a_2, \dots, a_k\}$ y llamemos p_i a la probabilidad de encontrar el símbolo a_i en una dada posición fija pero arbitraria de la secuencia.

Dividamos la secuencia S en m subsecuencias $S^{(1)}, S^{(2)}, \dots, S^{(m)}$ de longitudes $n^{(1)}, n^{(2)}, \dots, n^{(m)}$ y denotemos por $p_i^{(j)}$ a la probabilidad de encontrar el símbolo a_i en una posición fija pero arbitraria de la subsecuencia $S^{(j)}$.

Definamos dos variables aleatorias $a \in \mathcal{A}$ y $s \in \{S^{(1)}, S^{(2)}, \dots, S^{(m)}\}$. Sorteemos un número aleatorio con distribución uniforme para que nos dé una posición n de la secuencia S y llamemos a al símbolo que se encuentra en dicha posición y s a la subsecuencia en la cual se encuentra.

Repitiendo este procedimiento podemos construir la distribución de probabilidad conjunta, en la cual p_{ij} nos da la probabilidad de que $a = a_i$ y $s = S^{(j)}$ simultáneamente.

Entonces, se cumple que la probabilidad de que el símbolo sorteado (sin importar en cuál subsecuencia) sea a_i es $p_i = \sum_{j=1}^m p_{ij}$.

De igual modo, la probabilidad de que la posición sorteada pertenezca a la subsecuencia j -ésima, es decir $s = S^{(j)}$ (sin importar cuál símbolo), será igual a $\pi_j = \sum_{i=1}^k p_{ij}$. Por otro lado, sabemos que $\pi_j = \frac{n^{(j)}}{N}$ pues del total de N posiciones, $n^{(j)}$ pertenecen a la subsecuencia $S^{(j)}$.

Hecho ya el trabalenguas de la introducción y definición de cantidades, supongamos que alguien sortea una posición de la secuencia S , lee el valor de a y s , nos comunica el valor de a y nuestra tarea es adivinar el valor de s . Es decir, nos dicen cuál es el símbolo y con ello debemos adivinar cuál es la subsecuencia a la que pertenece la posición sorteada. Nuestro único conocimiento es la identidad del símbolo a , ¿podemos decir algo? ¿tenemos algo de información acerca de s por conocer a ?

La respuesta a esta pregunta en el contexto de la TI la da, como vimos en la Sección 2.3, la información mutua, que cuantifica la información que comparten dos variables aleatorias. En este caso, siguiendo la Ecuación (16), la información mutua se puede escribir del siguiente modo:

$$I(s, a) = \sum_{i=1}^k \sum_{j=1}^m p_{ij} \log \left(\frac{p_{ij}}{p_i \pi_j} \right). \quad (27)$$

Como $p_i^{(j)}$ es la probabilidad condicional de obtener el símbolo a_i dado que estamos en la secuencia $S^{(j)}$, se sigue que $p_{ij} = \pi_j p_i^{(j)}$. Con esto podemos reescribir la Ecuación (27) para obtener:

$$\begin{aligned} I(s, a) &= \sum_{i=1}^k \sum_{j=1}^m \pi_j p_i^{(j)} \log \left(\frac{p_i^{(j)}}{p_i} \right) \\ &= \sum_{j=1}^m \pi_j \sum_{i=1}^k p_i^{(j)} \log \left(p_i^{(j)} \right) - \sum_{i=1}^k \left(\sum_{j=1}^m \pi_j p_i^{(j)} \right) \log (p_i). \end{aligned}$$

Como último paso, usemos que $p_i = \sum_{j=1}^m p_{ij} = \sum_{j=1}^m \pi_j p_i^{(j)}$ con lo cual:

$$\begin{aligned} I(s, a) &= - \sum_{j=1}^m \pi_j H(p^{(j)}) + H\left(\sum_{j=1}^m \pi_j p^{(j)}\right) \\ &= D_{JS}(p^{(1)}, p^{(2)}, \dots, p^{(m)}), \end{aligned} \quad (28)$$

donde $p^{(1)}, p^{(2)}, \dots, p^{(m)}$ son las distribuciones de probabilidad asociadas a cada una de las subsecuencias $s^{(1)}, s^{(2)}, \dots, s^{(m)}$.

Vale la pena enfatizar que, en el cálculo de la **DJS** los pesos asignados a cada distribución son $\pi_j = \frac{n^{(j)}}{N}$, que es la fracción de caracteres en cada subsecuencia (y también, como se dijo, la probabilidad de que sorteando una posición cualesquiera de la secuencia, ésta pertenezca a la subsecuencia j -ésima).

Por lo tanto, la Ecuación (28) nos dice que calcular la **DJS** entre las distribuciones correspondientes a cada subsecuencia es equivalente a calcular la información mutua entre la identidad del símbolo y la subsecuencia a la que pertenece.

Lo hemos planteado como la información que obtenemos acerca de la identidad de la subsecuencia al conocer la identidad del símbolo pero como sabemos la información mutua es simétrica y el problema inverso es equivalente (que se sortee la posición, nos digan la subsecuencia y adivinemos el símbolo).

La información mutua o, como acabamos de ver, la **DJS** cuantifican la información compartida entre la variable a y la variable s , cómo extraer esa información para diseñar una estrategia para efectivamente adivinar de la mejor forma posible la subsecuencia conociendo el símbolo no es obvio y puede no ser fácil.

Como un caso límite ilustrativo, si todas las subsecuencias son generadas con la misma distribución de probabilidad, saber la identidad del símbolo no nos dice nada acerca de la subsecuencia a la que pertenece, intuitivamente la información mutua es cero y como $p^{(1)} = p^{(2)} = \dots = p^{(m)}$ la **DJS** también es cero.

2.5.3.3 En estadística

La **DJS** se puede interpretar en un contexto estadístico como el cociente de log-probabilidad (que definiremos en seguida) [23].

Consideremos una secuencia simbólica S formada por N caracteres independientes e idénticamente distribuidos, escogidos de un alfabeto $\mathcal{A} = \{a_1, a_2, \dots, a_k\}$.

Supongamos que queremos encontrar la distribución de probabilidad $P = (p_1, p_2, \dots, p_k)$ que mejor representa la secuencia S . Cada p_i representa la probabilidad de ocurrencia del símbolo a_i en una posición arbitraria pero fija de la secuencia, para $i = 1, 2, \dots, k$.

Una manera de hacerlo sería buscar la distribución que maximice la probabilidad de reobtener la secuencia completa al sortear N caracteres. Partiendo de una distribución cualesquiera P , como estamos asumiendo que cada caracter es independiente y es generado por la misma distribución, la probabilidad de reobtener la secuencia es el producto de las probabilidades de reobtener cada uno de los símbolos de la secuencia en el orden preciso en que aparecen, por lo tanto es:

$$\mathcal{L}(S|P) = \prod_{i=1}^k p_i^{f_i}, \quad (29)$$

donde f_i es el número de ocurrencias del símbolo a_i en la secuencia S .

Entonces, la mejor distribución será aquella que maximice la Ecuación (29). Para encontrarla introducimos un multiplicador de Lagrange para la restricción que proviene de la normalización de la distribución $\sum_{i=1}^k p_i = 1$ y procedemos en la manera habitual del cálculo de variaciones. Para que el cálculo sea más sencillo conviene aprovechar el hecho de que el logaritmo es una función monótonamente creciente, razón por la cuál es equivalente (¡pero más fácil!) maximizar $\ln(\mathcal{L})$. En este caso, se obtiene que la distribución que maximiza $\ln(\mathcal{L})$ está dada por:

$$p_i = \frac{f_i}{N} \quad i = 1, 2, \dots, k \quad (30)$$

y la log-probabilidad máxima es:

$$\ln(\mathcal{L}_{\max}) = -N \ln(2) H(P), \quad (31)$$

donde H es la entropía de Shannon en unidades de bits (esto es, como vimos en la Sección 2.1, utilizando el logaritmo en base 2). Este resultado es el que justifica el procedimiento habitual para estimar la distribución de probabilidad asociada a una dada secuencia, que consiste en contar el número de ocurrencias de cada símbolo y dividirlo por el total de caracteres.

Consideremos ahora el problema ligeramente más complicado de caracterizar una secuencia S compuesta de N caracteres escogidos del mismo alfabeto \mathcal{A} pero formada concatenando m subsecuencias $s^{(1)}, s^{(2)}, \dots, s^{(m)}$ con longitudes $n^{(1)}, n^{(2)}, \dots, n^{(m)}$, respectivamente ($\sum_{i=1}^m n^{(i)} = N$).

Si tenemos *a priori* el conocimiento de que la secuencia S está formada como concatenación de m subsecuencias y conocemos además las longitudes de cada subsecuencia, podemos construir las m distribuciones de probabilidades $P^{(1)}, P^{(2)}, \dots, P^{(m)}$ que maximizan la probabilidad de reobtener cada una de las subsecuencias y utilizarlas para reobtener la secuencia completa.

Llamemos $p_i^{(j)}$ a la probabilidad asociada al símbolo a_i en la subsecuencia $\mathcal{S}^{(j)}$ para $i = 1, 2, \dots, k$ y $j = 1, 2, \dots, m$.

Cada una de las m subsecuencias presenta un problema equivalente al que acabamos de tratar, por lo tanto ya sabemos la respuesta. La mejor distribución (de acuerdo con la Ecuación (30)) para cada subsecuencia será:

$$p_i^{(j)} = \frac{f_i^{(j)}}{n^{(j)}} \quad i = 1, 2, \dots, k \quad (32)$$

donde $f_i^{(j)}$ es el número de ocurrencias del símbolo a_i en la subsecuencia $\mathcal{S}^{(j)}$.

Esta distribución maximiza $\ln(\mathcal{L}^{(j)})$, que de acuerdo a (31) asume un valor máximo igual a:

$$\ln(\mathcal{L}_{\max}^{(j)}) = -n^{(j)} \ln(2) H(P^{(j)}), \quad (33)$$

Como cada subsecuencia es independiente, la probabilidad de reobtener la secuencia completa en este caso, $\tilde{\mathcal{L}}$, cumple que:

$$\tilde{\mathcal{L}} = \prod_{j=1}^m \mathcal{L}^{(j)}.$$

Por lo tanto tenemos que:

$$\ln(\tilde{\mathcal{L}}_{\max}) = \sum_{j=1}^m \ln(\mathcal{L}_{\max}^{(j)}) = -\ln(2) \sum_{j=1}^m n^{(j)} H[P^{(j)}]. \quad (34)$$

Si desconociéramos el hecho de que \mathcal{S} se construyó a partir de m subsecuencias, lo más sencillo sería suponer que la totalidad de los símbolos de la secuencia fueron generados por una única distribución P . Bajo esta suposición la mejor distribución sería (de nuevo) aquella que maximiza la probabilidad $\mathcal{L} = \mathcal{L}(\mathcal{S}|P)$:

$$p_i = \frac{f_i}{N} \quad i = 1, 2, \dots, k$$

que da

$$\ln(\mathcal{L}_{\max}) = -N \ln(2) H(P). \quad (35)$$

Los resultados en ambos casos (sabiendo que la secuencia está construida como concatenación de subsecuencias o sin saberlo) son diferentes y nos interesa cuantificar esa diferencia.

Para eso definimos el *cociente de log-probabilidad*, $\Delta\mathcal{L}$, como la diferencia en log-probabilidad de reobtener la secuencia a partir de m distribuciones¹³ y la de reobtenerla a partir de una única distribución, esto es:

$$\Delta\mathcal{L} = \ln(\tilde{\mathcal{L}}_{\max}) - \ln(\mathcal{L}_{\max}). \quad (36)$$

¹³ Con las longitudes de las subsecuencias también previamente especificadas.

A partir de las Ecuaciones (34) y (35) podemos reescribir esta cantidad del siguiente modo:

$$\Delta\mathcal{L} = -\ln(2) \sum_{j=1}^m n^{(j)} H(p^{(j)}) + N \ln(2) H(P) \quad (37)$$

$$= N \ln(2) \left(H(P) - \sum_{j=1}^m \frac{n^{(j)}}{N} H(p^{(j)}) \right). \quad (38)$$

Ahora bien, si llamamos $\pi_j = \frac{n^{(j)}}{N}$, se cumple que $p_i = \sum_{j=1}^m \pi_j p_i^{(j)}$ (la probabilidad de ocurrencia del símbolo a_i en la secuencia completa es la suma de las probabilidades en cada una de las subsecuencias, $p_i^{(j)}$, pesado por la fracción de veces que un caracter de esa subsecuencia aparece, π_j), por lo tanto:

$$\Delta\mathcal{L} = N \ln(2) \left(H \left(\sum_{j=1}^m \pi^{(j)} p^{(j)} \right) - \sum_{j=1}^m \pi^{(j)} H(p^{(j)}) \right) \quad (39)$$

$$= N \ln(2) D_{JS}(p^{(1)}, p^{(2)}, \dots, p^{(m)}). \quad (40)$$

Con eso hemos demostrado que la **DJS** es proporcional al cociente de log-probabilidad. Es decir, calcular la **DJS** entre m distribuciones de probabilidad con pesos asociados $\pi^{(j)} = \frac{n^{(j)}}{N}$, es equivalente a calcular la diferencia en log-probabilidad. Provee, por lo tanto, una manera directa de cuantificar la diferencia en la probabilidad de reobtener la secuencia describiéndola con m distribuciones de probabilidad o con una sola.

En la práctica es común que desconozcamos si una secuencia es estacionaria, es decir, si es sensato suponer que todos los símbolos fueron generados con una misma distribución, como veremos en el Capítulo 3, el hecho de que la **DJS** sea proporcional al cociente de log-probabilidad sugiere un método para decidir si una secuencia es estacionaria y para segmentar secuencias no-estacionarias en subsecuencias estacionarias.

2.5.3.4 En inferencia bayesiana

En un contexto de inferencia bayesiana la **DJS** puede interpretarse como la “ganancia” de información promedio (esto será más claro en seguida) [21].

Supongamos que tenemos dos distribuciones de probabilidad conocidas $Q_1(x)$ y $Q_2(x)$ asociadas a la variable aleatoria X y que se nos dice que una de ellas es la distribución verdadera, nuestra tarea es averiguar cuál de ambas es, teniendo acceso a la observación de la variable aleatoria.

Definamos una nueva variable aleatoria, s , para cuantificar nuestra capacidad de discernir cuál es la distribución verdadera de X . Así,

s toma dos valores $\{1, 2\}$ que hacen referencia a que la distribución verdadera sea Q_1 o Q_2 respectivamente. Si, por ejemplo, $P(s = 1) = 1$, esto significa que sabemos con certeza que la distribución verdadera es Q_1 .

Antes de observar una realización de X no tenemos ningún otro dato, por lo que cualquiera de las dos distribuciones puede ser la verdadera y la distribución *a priori* para s será:

$$P_{\text{prio}} = \begin{cases} P(s = 1) = \frac{1}{2} \\ P(s = 2) = \frac{1}{2} \end{cases} \quad (41)$$

Si ahora observamos el resultado de la variable aleatoria X y es igual a \tilde{x} , podemos actualizar la distribución de probabilidad siguiendo lo que nos dice el Teorema de Bayes¹⁴:

$$P_{\text{post}} = \begin{cases} P(s = 1|X = \tilde{x}) = \frac{P(X=\tilde{x}|s=1)P(s=1)}{P(X=\tilde{x})} = \frac{Q_1(\tilde{x})}{Q_1(\tilde{x})+Q_2(\tilde{x})} \\ P(s = 2|X = \tilde{x}) = \frac{P(X=\tilde{x}|s=2)P(s=2)}{P(X=\tilde{x})} = \frac{Q_2(\tilde{x})}{Q_1(\tilde{x})+Q_2(\tilde{x})} \end{cases}, \quad (42)$$

donde hemos utilizado que $P(X = \tilde{x}|s = 1) = Q_1(\tilde{x})$, $P(X = \tilde{x}|s = 2) = Q_2(\tilde{x})$, $P(X = \tilde{x}) = P(X = \tilde{x}|s = 1)P(s = 1) + P(X = \tilde{x}|s = 2)P(s = 2)$ y utilizado la distribución *a priori* (41) para $P(s = 1)$ y $P(s = 2)$.

La ganancia de información, ΔI , la podemos cuantificar como la *DKL* entre las distribuciones *a priori* y *a posteriori*, como vimos en la Sección 2.4. Por lo tanto, tenemos que:

$$\begin{aligned} \Delta I &= D_{\text{KL}}(P_{\text{post}}, P_{\text{prio}}) \\ &= P(s = 1|X = \tilde{x}) \log \left(\frac{P(s = 1|X = \tilde{x})}{P(s = 1)} \right) + \\ &\quad + P(s = 2|X = \tilde{x}) \log \left(\frac{P(s = 2|X = \tilde{x})}{P(s = 2)} \right) \\ &= \frac{Q_1(\tilde{x})}{Q_1(\tilde{x}) + Q_2(\tilde{x})} \log \left(\frac{2Q_1(\tilde{x})}{Q_1(\tilde{x}) + Q_2(\tilde{x})} \right) + \\ &\quad + \frac{Q_2(\tilde{x})}{Q_1(\tilde{x}) + Q_2(\tilde{x})} \log \left(\frac{2Q_2(\tilde{x})}{Q_1(\tilde{x}) + Q_2(\tilde{x})} \right). \end{aligned}$$

Ahora bien, esa es la ganancia de información dado el resultado particular \tilde{x} , la ganancia de información promedio, la obtendremos

¹⁴ Ver, por ejemplo, la Referencia [19].

promediando sobre todos los valores posibles de la variable aleatoria X :

$$\begin{aligned}
\langle \Delta I \rangle &= \sum_{\tilde{x} \in \mathcal{X}} P(\tilde{x}) \Delta I \\
&= \sum_{\tilde{x} \in \mathcal{X}} \frac{Q_1(\tilde{x}) + Q_2(\tilde{x})}{2} \left(\frac{Q_1(\tilde{x})}{Q_1(\tilde{x}) + Q_2(\tilde{x})} \log \left(\frac{2Q_1(\tilde{x})}{Q_1(\tilde{x}) + Q_2(\tilde{x})} \right) + \right. \\
&\quad \left. + \frac{Q_2(\tilde{x})}{Q_1(\tilde{x}) + Q_2(\tilde{x})} \log \left(\frac{2Q_2(\tilde{x})}{Q_1(\tilde{x}) + Q_2(\tilde{x})} \right) \right) \\
&= \sum_{\tilde{x} \in \mathcal{X}} \frac{Q_1(\tilde{x})}{2} \log \left(\frac{Q_1(\tilde{x})}{Q_1(\tilde{x}) + Q_2(\tilde{x})} \right) + \\
&\quad + \sum_{\tilde{x} \in \mathcal{X}} \frac{Q_2(\tilde{x})}{2} \log \left(\frac{Q_2(\tilde{x})}{Q_1(\tilde{x}) + Q_2(\tilde{x})} \right) \\
&= \frac{1}{2} D_{\text{KL}} \left(Q_1, \frac{Q_1 + Q_2}{2} \right) + \frac{1}{2} D_{\text{KL}} \left(Q_2, \frac{Q_1 + Q_2}{2} \right) \\
&= D_{\text{JS}}^{1/2} (Q_1, Q_2). \tag{43}
\end{aligned}$$

Por lo tanto, la **DJS** (en su versión con pesos $\frac{1}{2}$) entre dos distribuciones de probabilidad Q_1 y Q_2 nos cuantifica la ganancia de información promedio que se obtiene al observar el resultado de X , relacionada a la posibilidad de distinguir cuál de las dos distribuciones es la verdadera.

2.6 REFERENCIAS COMENTADAS

La bibliografía clásica en **TI** es el libro de Cover y Thomas, “Elements of Information Theory” [16], de ahí se extrajeron la mayoría de las definiciones y algunos ejemplos. El concepto de información como sorpresa esta presentado de manera breve pero amena en el libro “Bayesian Brain” editado por Doya *et al.* [19]. La definición y algunas propiedades de la **DJS** están discutidas con considerable detalle en el artículo de Grosse *et al.* [23]. En particular se presenta su interpretación en física estadística, **TI** y estadística. La demostración de que la **DJS** es el cuadrado de una métrica está en el artículo de Endres *et al.* [21]. En él también se presenta su interpretación en el contexto de la inferencia bayesiana. Una mirada histórica sobre los pormenores de la propuesta de la **DJS** puede obtenerse leyendo el artículo original de Lin [39]. En el recientemente publicado libro de Bialek, “Biophysics: Searching for Principles” [5], se presentan muchos de los conceptos básicos de **TI** de una forma original, con una perspectiva fuertemente biológica y con énfasis en la *representación eficiente*.

LA DIVERGENCIA DE JENSEN-SHANNON COMO HERRAMIENTA PARA SEGMENTAR SECUENCIAS NO-ESTACIONARIAS

En este capítulo presentamos un método de segmentación de secuencias simbólicas no-estacionarias basado en la *DJS*. Este método es parte fundamental de este trabajo de tesis, en el Capítulo 4 se discutirá en detalle su adaptación para la localización de paredes de dominio en la dinámica de los motores moleculares KIF1A.

En la Sección 3.1 motivamos el método a partir de una de las propiedades de la *DJS* discutidas en el Capítulo 2. En la Sección 3.2 presentamos la significancia que es parte clave del criterio para decidir si se segmenta o no. En la Sección 3.3 se dan los lineamientos generales del método de segmentación y en la Sección 3.4 se explicitan los pasos a seguir para utilizarlo. Por último, en la Sección 3.5 se presenta un ejemplo concreto para mostrar su funcionamiento e ilustrar la forma recursiva en que se utiliza el método de segmentación.

3.1 REPASO Y CONSIDERACIONES PREVIAS

Al trabajar con secuencias simbólicas de cualquier tipo uno se enfrenta a menudo con el problema de determinar si la secuencia bajo consideración es estacionaria. Nos referimos a una secuencia estacionaria como aquella cuyos símbolos fueron generados a partir de una misma distribución de probabilidad, es decir, a partir de una fuente que no cambió durante la generación de la totalidad de los caracteres que constituyen la secuencia.

Como veremos en esta sección, la *DJS* provee una forma directa de afrontar este problema haciendo uso de una de sus propiedades, ya discutida en la Sección 2.5.3. Concretamente, el hecho de que sea proporcional al cociente de log-probabilidad, $\Delta\mathcal{L}$:

$$\Delta\mathcal{L} = N \ln(2) D_{JS} \left(p^{(1)}, p^{(2)}, \dots, p^{(m)} \right). \quad (44)$$

Como se indicó en la Sección 2.5.3.3, el cociente de log-probabilidad es la diferencia en (log-)probabilidad entre reobtener la secuencia a partir de m distribuciones y la de reobtenerla a partir de una sola distribución.

Podríamos pensar, ingenuamente, que la respuesta correcta daría una probabilidad más alta de reobtener la secuencia y que esto nos puede proveer de un método para decidir si la secuencia fue generada por una única distribución o por m . En principio podríamos

comparar el valor para todos los m posibles y de esa forma averiguar si la secuencia es estacionaria.

Dos golpes a la ilusión de este procedimiento saltan a la vista: por un lado, si asumimos m distribuciones nada impide que todas ellas resulten iguales, de lo que se deduce inmediatamente que la probabilidad de reobtener la secuencia con m distribuciones va a ser siempre mayor o igual que la de reobtenerla con una. Por otro lado, parece una empresa absurda tener que comparar dicha probabilidad para todos los valores posibles de m .

La visión ingenua que presentamos es incorrecta pero a pesar de eso el camino que señala no lo es. La solución al primer inconveniente la daremos a continuación y la solución al segundo la pospondremos un poco pero podemos adelantar la idea: si realizamos un procedimiento recursivo que nos permita ir comparando de a dos subsecuencias no es necesario elegir *a priori* ningún m .

Debido a la no-negatividad de la DJS (ver Sección 2.5.1), el cociente de log-probabilidad es siempre mayor o igual que cero: $\Delta\mathcal{L} \geq 0$. Esto significa, como dijimos, que la probabilidad de reobtener la secuencia con m distribuciones es siempre mayor o igual que la probabilidad de reobtenerla con una única distribución.

Intuitivamente, mientras más distribuciones utilicemos para describir la secuencia, mejor podremos captar posibles diferencias locales que ésta posea y mayor será la probabilidad de reobtenerla. En el caso extremo en el que utilicemos una distribución diferente para cada carácter de la secuencia está claro que la secuencia completa se reobtendría con probabilidad 1 (estas distribuciones consistirían en una probabilidad igual a 1 para el símbolo correcto y 0 para los restantes).

Estas consideraciones sugieren que no es tan relevante entonces que $\Delta\mathcal{L} \geq 0$ pues sucede siempre, pero en cambio sí lo es establecer si es *significativamente* mayor que cero. Ya construiremos un criterio preciso para cuantificar esta expresión pero siguiendo de un modo un poco laxo, si $\Delta\mathcal{L}$ es mucho mayor que cero, describir la secuencia con m distribuciones es conveniente, en cambio si $\Delta\mathcal{L}$ es chico no resulta conveniente.

Como conclusión provisoria podemos decir que calcular el valor de $\Delta\mathcal{L}$ y establecer si es significativamente mayor que cero es un camino posible para decidir si una secuencia es estacionaria y, a la luz de la Ecuación (44), una forma directa de hacerlo es calculando el valor de la DJS para las distribuciones correspondientes a las candidatas a subsecuencias.

3.2 ¿QUÉ ES SIGNIFICATIVAMENTE MAYOR QUE CERO?

Supongamos que queremos decidir si un dado valor de $\Delta\mathcal{L}$ es significativamente mayor que cero. Como ya establecimos el paralelismo

con los valores de la **DJS** es equivalente establecer si el valor de la **DJS** es significativamente mayor que cero.

Para decidirlo, utilizaremos el criterio que se describe a continuación. Supongamos que observamos un valor z para la **DJS**. Consideraremos la probabilidad de obtener dicho valor o cualquiera inferior, por azar en una secuencia generada a partir de una única distribución y llamaremos a dicha cantidad *significancia* $s(z)$:

$$s(z) = \text{Prob} [D_{JS} \leq z \mid \text{una distribución}]. \quad (45)$$

Supongamos que consideramos un caso en que la secuencia fue generada a partir de una única distribución: idealmente las m subsecuencias serían iguales y la **DJS** valdría cero. En la práctica esto no es así pues construimos estimadores para las distribuciones de probabilidad correspondientes a las subsecuencias y por lo tanto las fluctuaciones propias de esta construcción arrojarán valores de la **DJS** mayores que cero.

La significancia, como la definimos, es una manera de cuantificar estas fluctuaciones y de intentar responder a la pregunta ¿es posible que la secuencia haya sido generada por una única distribución y que el valor z observado sea resultado de las fluctuaciones? Por supuesto que no podemos dar una respuesta taxativa y debemos conformarnos con calcular la probabilidad de que dicho valor (o cualquiera inferior) haya sido obtenido con una única distribución.

Dicho de otro modo, la significancia da la probabilidad de obtener el valor z al calcular el valor de la **DJS** entre m distribuciones de probabilidad asociadas a particiones de una secuencia que fue generada con una única distribución.

Como la significancia acumula la probabilidad de todos los valores inferiores a z , si el valor de la significancia está muy cerca de 1, eso significa que es muy poco probable encontrar un valor mayor o igual que z con una única distribución y entonces, el valor obtenido es significativo. Ya discutiremos en términos más concretos esta aseveración un poco más adelante.

Para calcular la significancia es necesario conocer la distribución de probabilidad de la **DJS**. Grosse y colaboradores [23] hicieron un desarrollo de Taylor de orden 2 de la **DJS** y encontraron que hasta ese orden es igual al *estadístico* χ^2 . Éste converge para valores asintóticamente grandes de N a la *distribución* χ^2 , de modo que se tiene:

$$s(z) \simeq \frac{\gamma\left[\frac{\nu}{2}, N \ln(2)z\right]}{\Gamma\left(\frac{\nu}{2}\right)}, \quad (46)$$

donde $\Gamma(a)$ es la función gamma, $\gamma(a, x)$ es la función gamma incompleta, $\nu = (k - 1)(m - 1)$ y k es el número de símbolos en el alfabeto.

Ver el Apéndice B.1.1 por información acerca de las funciones gamma completa e incompleta, el Apéndice B.2 por algunas considera-

ciones breves acerca del estadístico χ^2 y el Apéndice B.3 por una explicación de los pasos necesarios para llegar a la Ecuación (46).

3.3 MÉTODO DE SEGMENTACIÓN

Tratemos de juntar los ingredientes que venimos desarrollando para describir de que forma vamos a atacar el problema que nos interesa, es decir, cómo decidir si una dada secuencia es o no estacionaria.

En el caso de que sea no-estacionaria, desconocemos tanto el número de subsecuencias que la constituyen (el valor de m) como las longitudes de cada una de ellas (los *puntos de corte* de la secuencia) y nos gustaría poder identificarlos.

La idea del método de segmentación es trabajar con $m = 2$ de forma recursiva, es decir comenzar analizando si la secuencia completa se describe apropiadamente con una única distribución o son necesarias dos distribuciones y repetir este procedimiento de ser necesario.

Si una distribución es suficiente para describirla, la secuencia es estacionaria (al nivel de significancia al que estamos trabajando), si en cambio son necesarias dos distribuciones, el proceso se repite analizando ahora cada una de las subsecuencias hasta que todos los segmentos resulten estacionarios.

Establecido el hecho de que vamos a trabajar de forma recursiva con dos distribuciones, es decir, fijado el valor de m , nos queda el problema de averiguar cuál es la longitud de cada subsecuencia, es decir, cuál es el posible punto de corte.

Recordemos que la Ecuación (44) se cumple para longitudes de subsecuencia predeterminadas. Como desconocemos el punto de corte pero sabemos que la DJS es proporcional a $\Delta\mathcal{L}$, la posición en la secuencia que es la mejor candidata a punto de corte será aquella en la cual la DJS se maximice, pues allí la diferencia en log-probabilidad será también máxima (será por lo tanto la posición en donde la diferencia de probabilidad entre reobtener la secuencia completa mediante dos distribuciones y reobtenerla con una sola, es mayor).

De este modo, moveremos un cursor a lo largo de todas las posiciones de la secuencia e iremos calculando la DJS entre las distribuciones que la posición del cursor define. La posición en la cual la DJS toma su valor máximo será la mejor candidata a punto de corte. Para decidir si el valor observado $D_{JS_{\max}} = z$ es estadísticamente significativo (esto es, si es lo suficientemente grande como para proceder a segmentar) calculamos la probabilidad de obtener ese valor o cualquiera inferior por azar con una secuencia generada por una única distribución, es decir calcularemos la significancia máxima:

$$s_{\max}(z) = \text{Prob} [D_{JS_{\max}} \leq z \mid \text{una distribución}]. \quad (47)$$

Es importante enfatizar la diferencia entre la significancia máxima y la significancia (a secas) que presentamos en la Ecuación (45). La

significancia cuantificaba la probabilidad de obtener el valor z en una posición predeterminada, por azar, en una secuencia generada con una única distribución. Por otro lado, la significancia máxima cuantifica la probabilidad de obtener un valor máximo para la *DJS* igual a z , en cualquier posición de una secuencia generada a partir de una única distribución.

En la sección siguiente discutiremos cómo calcular la significancia máxima, pero para cerrar la idea veamos antes cómo decidimos si se segmenta o no.

El criterio de corte, es decir, de segmentación, va a ser que la significancia máxima supere algún valor *umbral* s_0 . Así, si $s_{\max}(z) > s_0$ el valor observado es significativo y la secuencia se trunca. Esto significa que la probabilidad de obtener por azar un valor tan grande como el observado (en verdad mayor o igual al observado) es menor que $1 - s_0$. Esta probabilidad puede hacerse tan chica como deseemos, eligiendo s_0 tan cerca de 1 como haga falta.

3.3.1 Significancia máxima

Como ya mencionamos $s_{\max}(z) \neq s(z)$, por lo que no podemos utilizar la Ecuación (46) para calcularla.

Grosse y colaboradores [23] propusieron calcular la significancia máxima con una expresión fenomenológica de la siguiente forma:

$$s_{\max}(z) = [s(\beta z)]^{N_{ef}}, \quad (48)$$

donde $s(z)$ es la expresión de la Ecuación (46) y β y N_{ef} son dos parámetros a ajustar.

Los valores de los parámetros para el caso de un alfabeto binario ($k = 2$) que es el único en el que nosotros estamos interesados son:

$$\beta = 0,80, \quad (49)$$

$$N_{ef} = 2,96 \ln(N) - 7,88, \quad (50)$$

donde N es la longitud total de la secuencia.

Más aún, en el caso particular de un alfabeto binario ($k = 2$) y dos subsecuencias ($m = 2$), la Ecuación (48) puede simplificarse hasta quedar en estos términos:

$$s_{\max}(z) = \left(\operatorname{erf} \left(\sqrt{N \ln(2)} \beta z \right) \right)^{N_{ef}}, \quad (51)$$

donde erf es la función error.

Ver el Apéndice B.3.1 por los detalles que llevan a las Ecuaciones (48) y (51), y por la forma en que se obtuvieron los valores de los parámetros. Ver el Apéndice B.1.2 por la definición de la función error.

3.4 Y CON TODO ESO, ¿CÓMO SE HACE PARA SEGMENTAR SECUENCIAS NO ESTACIONARIAS?

A modo de énfasis y esperando aportar algo de claridad, resumamos el método de segmentación basado en la **DJS** que presentamos.

Consideremos una secuencia simbólica S , de longitud N , que consta de caracteres escogidos de un alfabeto binario $A = \{0, 1\}$. Para determinar si la secuencia es estacionaria y, si no lo es, para segmentarla en subsecuencias estacionarias realizamos los siguientes pasos:

1. Utilizar un cursor para definir una posición en la secuencia, definiendo una subsecuencia hacia la izquierda del cursor, $S^{(1)}$, con longitud $n^{(1)}$ y otra subsecuencia hacia la derecha del cursor, $S^{(2)}$, con longitud $n^{(2)}$. Esta claro que $n^{(1)} + n^{(2)} = N$.
2. Construir a partir de la frecuencia de ocurrencia de los símbolos del alfabeto los estimadores para las distribuciones de probabilidad que la posición del cursor define, $\hat{P}^{(1)}$ y $\hat{P}^{(2)}$.

Si $n_0^{(1)}$ es el número de ceros a la izquierda del cursor y $n_1^{(1)}$ es el número de unos a la izquierda del cursor, entonces:

$$\hat{P}^{(1)} = \begin{cases} \hat{P}^{(1)}(0) = \frac{n_0^{(1)}}{n^{(1)}} \\ \hat{P}^{(1)}(1) = \frac{n_1^{(1)}}{n^{(1)}} \end{cases}.$$

De igual modo, si $n_0^{(2)}$ es el número de ceros a la derecha del cursor y $n_1^{(2)}$ es el número de unos a la derecha del cursor, entonces:

$$\hat{P}^{(2)} = \begin{cases} \hat{P}^{(2)}(0) = \frac{n_0^{(2)}}{n^{(2)}} \\ \hat{P}^{(2)}(1) = \frac{n_1^{(2)}}{n^{(2)}} \end{cases}.$$

3. Calcular el valor de la **DJS** entre las distribuciones $\hat{P}^{(1)}$ y $\hat{P}^{(2)}$ utilizando para cada distribución un peso proporcional a la longitud de la subsecuencia que la posición del cursor define, es decir, $\pi^{(1)} = \frac{n^{(1)}}{N}$ y $\pi^{(2)} = \frac{n^{(2)}}{N}$.
4. Repetir para todas las posiciones de la secuencia.
5. Buscar la posición para la cual la **DJS** toma su valor máximo, llamémoslo $d_{JS_{max}}$.
6. Evaluar el valor de la significancia máxima allí, $s_{max}(d_{JS_{max}})$ utilizando la Ecuación (51).
7. Si el valor de la significancia máxima es menor que el valor del umbral s_0 , la secuencia es estacionaria. Si, por el contrario, el

valor de la significancia máxima supera el valor del umbral, procedemos a segmentar, esto es, a trincar la secuencia en dos subsecuencias en la posición en la que se obtuvo el valor máximo. En este caso, repetimos el procedimiento con cada subsecuencia hasta que todos los pedazos resultantes sean estacionarios a significancia- s_0 .

NOTA 1: la aproximación para la significancia funciona bien para una longitud de secuencia $N \geq 15$. No utilizaremos el método de segmentación para secuencias de longitud menor que 15.

NOTA 2: para un alfabeto cualesquiera los pasos son los mismos con la expresión adecuada para la significancia máxima. Aquí se enfatizó el caso binario porque es el que nos interesa en el resto de la tesis.

NOTA 3: la expresión para la significancia máxima queda dependiente de el valor que se obtenga para $D_{JS_{max}}$ y de la longitud de la secuencia bajo consideración. Notar que esta longitud va a ir cambiando conforme el procedimiento avanza recursivamente.

3.5 EJEMPLO DE APLICACIÓN

Haremos aquí un ejemplo concreto de aplicación del método de segmentación para ilustrar los pasos a seguir y para convencernos de su utilidad “viendo” como funciona.

Construimos un ensamble de 100 secuencias binarias (alfabeto de dos símbolos, $k = 2$) de $N = 1500$ caracteres cada una, constituidas concatenando cinco subsecuencias, cada una con la longitud $n^{(i)}$ y la distribución de probabilidad $P^{(i)}$ que se detallan en la Tabla 1. La distribución se da en la siguiente notación $P^{(i)}(x = 0) = p_i$ (y por lo tanto, $P^{(i)}(x = 1) = 1 - p_i$). Trabajaremos con un nivel de significancia $s_0 = 0,95$.

SUBSECUENCIA $\mathcal{S}^{(i)}$	1	2	3	4	5
$n^{(i)}$	200	350	50	300	600
p_i	0.8	0.5	0.1	0.6	0.8

Tabla 1: Longitudes y distribuciones de las subsecuencias del ejemplo.

La idea es utilizar el método de segmentación y chequear si segmentamos el ensamble de secuencias en las mismas posiciones en que cambian las distribuciones de probabilidad que generan los símbolos, esto es, ver si recuperamos las subsecuencias tal como las construimos.

Veamos los resultados que obtuvimos, siguiendo paso a paso el método recursivo de segmentación.

Paso 1

Comenzamos analizando la secuencia completa S . En la Figura 3 se muestran los resultados para la DJS promediada en el ensamble. Se obtuvo un valor máximo $\langle D_{JS_{max}} \rangle = 0,04235$ para la DJS en la posición $x_{max} = 900$, calculando la significancia máxima allí, obtuvimos $s_{max} (N = 1500, \langle D_{JS_{max}} \rangle = 0,04235) = 1 > 0,95$ entonces se trunca en las subsecuencias S_i con una longitud $n_i = 900$ y la subsecuencia S_d con una longitud $n_d = 600$. Veamos aquí un esquema (a escala) de la situación:

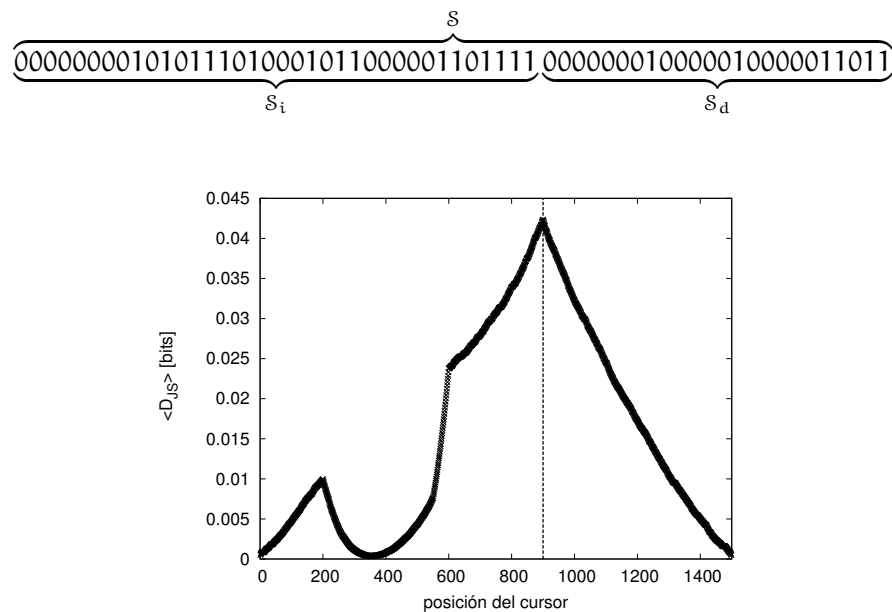


Figura 3: Promedio en el ensamble de la DJS para cada posición del cursor. Paso 1 del método de segmentación. Con línea de puntos se indica la posición 900 en la cual se segmenta la secuencia.

Paso 2

Como en el paso anterior encontramos que la secuencia no es estacionaria (se segmentó en dos subsecuencias) debemos continuar repitiendo el mismo procedimiento en cada uno de los segmentos.

Por lo tanto, este paso consta de dos partes: i) analizar S_i y ii) analizar S_d .

- i) En la Figura 4 se puede ver el resultado de analizar S_i , se encontró que el valor máximo es $\langle D_{JS_{max}} \rangle = 0,04589$, en la posición $x_{max} = 200$. La significancia máxima toma el valor

$s_{\max} (N = 900, \langle D_{J_{S_{\max}}} \rangle = 0,04589) = 1 > 0,95$, por lo tanto la secuencia se trunca en las subsecuencias S_{ii} con longitud $n_{ii} = 200$ y S_{id} con longitud $n_{id} = 700$.

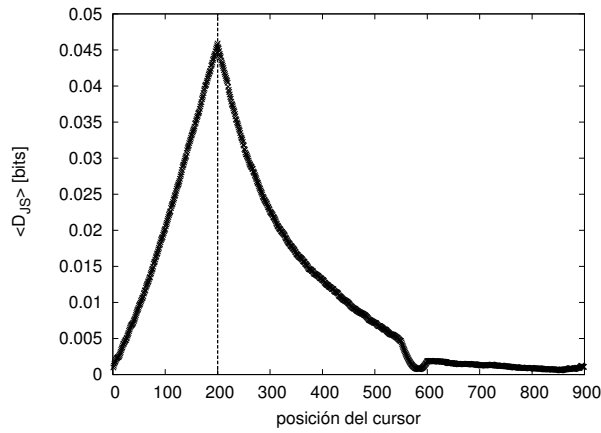


Figura 4: Promedio en el ensamble de la DJS para cada posición del cursor. Primera parte del paso 2 del método de segmentación. Con línea de puntos se indica la posición 200 en la cual se segmenta la secuencia.

- ii) En la Figura 5 se muestra el resultado de analizar S_d . El valor máximo de la DJS se encontró en la posición $x_{\max} = 7$, tomando el valor $\langle D_{J_{S_{\max}}} \rangle = 0,001631$, por lo tanto la significancia máxima vale $s_{\max} (N = 600, \langle D_{J_{S_{\max}}} \rangle = 0,001631) = 0,020 < 0,95$, y así, este segmento es estacionario y no se trunca.

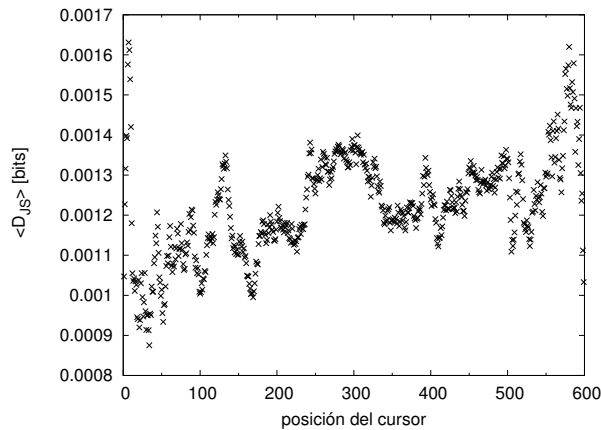
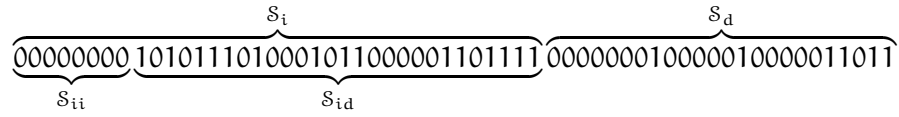


Figura 5: Promedio en el ensamble de la DJS para cada posición del cursor. Segunda parte del paso 2 del método de segmentación.

Por lo tanto luego del paso 2, el esquema se actualiza del siguiente modo:



Paso 3

Como en el paso anterior, \mathcal{S}_i se segmentó en \mathcal{S}_{ii} y \mathcal{S}_{id} pero \mathcal{S}_d no se segmentó, este paso consistirá en dos partes: i) analizar \mathcal{S}_{ii} y ii) analizar \mathcal{S}_{id} .

- i) En la Figura 6 analizamos la secuencia \mathcal{S}_{ii} . Obtuvimos un valor máximo de $\langle D_{JS_{max}} \rangle = 0,005454$, en la posición $x_{max} = 179$. La significancia máxima es s_{max} ($N = 200$, $\langle D_{JS_{max}} \rangle = 0,005454$) = $0,084 < 0,95$, por lo tanto no se trunca.

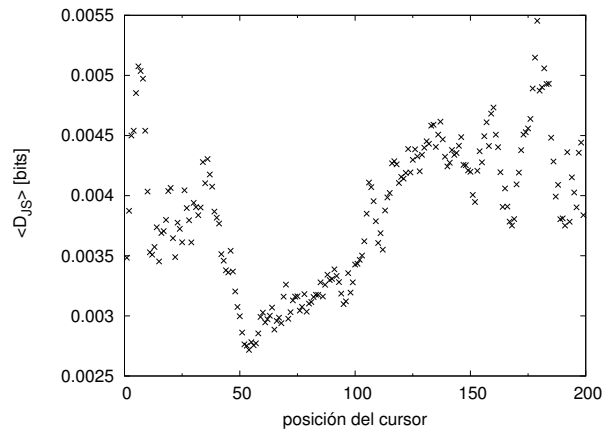
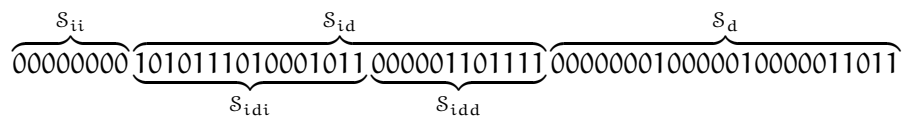


Figura 6: Promedio en el ensamble de la DJS para cada posición del cursor. Primera parte del paso 3 del método de segmentación.

- ii) En la Figura 7 vemos el resultado de analizar la secuencia \mathcal{S}_{id} . El valor máximo es $\langle D_{JS_{max}} \rangle = 0,01885$, en la posición $x_{max} = 400$, la significancia máxima es s_{max} ($N = 700$, $\langle D_{JS_{max}} \rangle = 0,01885$) = $0,998 > 0,95$, entonces se trunca en una subsecuencia \mathcal{S}_{idi} con longitud $n_{idi} = 400$ y una subsecuencia \mathcal{S}_{idd} con longitud $n_{idd} = 300$.

Y el esquema, luego de este paso se actualiza para quedar:



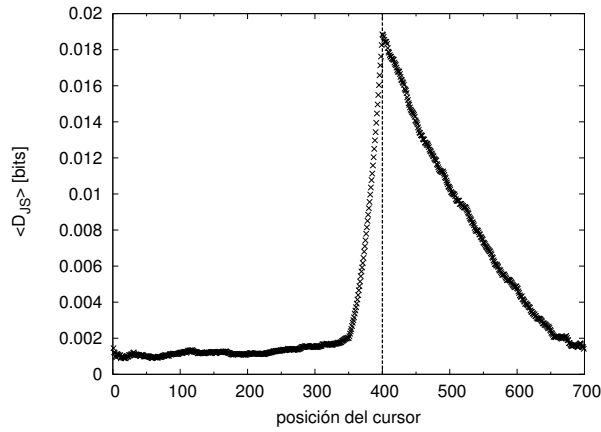


Figura 7: Promedio en el ensamble de la DJS para cada posición del cursor. Segunda parte del paso 3 del método de segmentación. Con línea de puntos se indica la posición 400 en la cual se segmenta la secuencia.

Paso 4

En el paso 3, \mathcal{S}_{ii} resultó estacionaria pero \mathcal{S}_{id} se segmentó en \mathcal{S}_{idi} y \mathcal{S}_{idd} , por lo tanto, este paso consistirá en dos partes: i) analizar \mathcal{S}_{idi} y ii) analizar \mathcal{S}_{idd} .

- i) En la Figura 8 se muestra el análisis de la secuencia \mathcal{S}_{idi} . Obtenemos un valor máximo $\langle D_{JS_{max}} \rangle = 0,06061$, en la posición $x_{max} = 350$, que arroja un valor para la significancia máxima de $s_{max} (N = 400, \langle D_{JS_{max}} \rangle = 0,06061) = 1 > 0,95$ por lo que se trunca en la subsecuencia \mathcal{S}_{idii} con una longitud $n_{idii} = 350$ y la subsecuencia \mathcal{S}_{idid} con una longitud $n_{idid} = 50$.

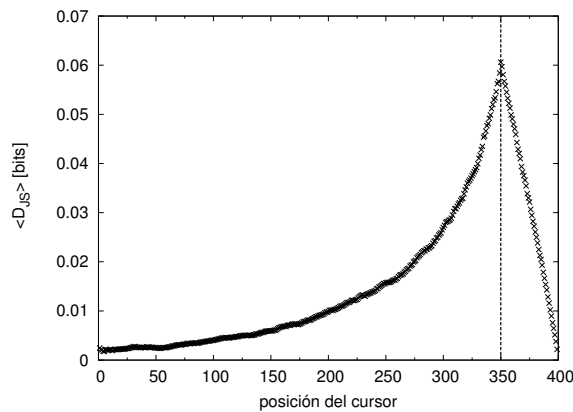


Figura 8: Promedio en el ensamble de la DJS para cada posición del cursor. Primera parte del paso 4 del método de segmentación. Con línea de puntos se indica la posición 350 en la cual se segmenta la secuencia.

- ii) En la Figura 9 podemos ver el resultado de analizar \mathcal{S}_{idd} . El valor máximo $\langle D_{JS_{max}} \rangle = 0,00372$ se obtiene en la posición $x_{max} = 294$, y la significancia máxima toma allí el siguiente valor $s_{max}(N = 300, \langle D_{JS_{max}} \rangle = 0,00372) = 0,062 < 0,95$, por lo tanto la secuencia es estacionaria.

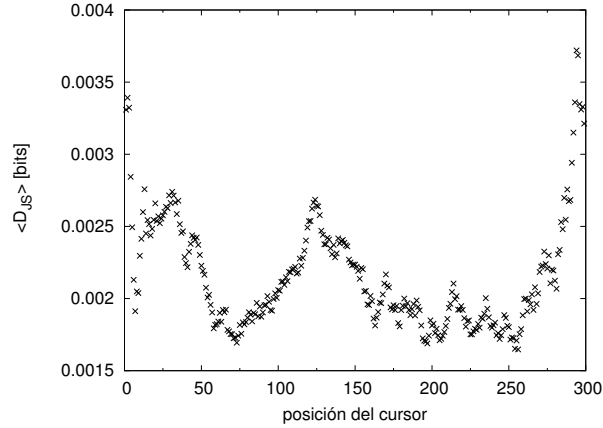


Figura 9: Promedio en el ensamble de la DJS para cada posición del cursor. Segunda parte del paso 4 del método de segmentación.

El esquema, luego del paso 4 queda:

$$\overbrace{00000000}^{S_{ii}} \overbrace{10101110100010}^{S_{idi}} \overbrace{11}^{S_{idd}} \overbrace{000001101111}^{S_d} \overbrace{000000010000010000011011}^{S_d}$$

$$\underbrace{10101110100010}_{S_{idii}} \underbrace{11}_{S_{idid}}$$

Paso 5

En el paso anterior, \mathcal{S}_{idi} se segmentó en \mathcal{S}_{idii} y \mathcal{S}_{idid} pero \mathcal{S}_{idd} resultó estacionaria. Entonces, este paso consistirá en: i) analizar \mathcal{S}_{idii} y ii) analizar \mathcal{S}_{idid} .

- i) En la Figura 10 analizamos la secuencia \mathcal{S}_{idii} . En la posición $x_{max} = 347$ se encuentra $\langle D_{JS_{max}} \rangle = 0,00317$ y la significancia máxima queda $s_{max}(N = 350, \langle D_{JS_{max}} \rangle = 0,00317) = 0,0527 < 0,95$, por lo tanto no se trunca.
- ii) En la Figura 11 vemos el resultado de analizar la secuencia \mathcal{S}_{idid} . Se obtiene $\langle D_{JS_{max}} \rangle = 0,02081$, en la posición $x_{max} = 17$ y la significancia máxima da $s_{max}(N = 50, \langle D_{JS_{max}} \rangle = 0,02081) = 0,292 < 0,95$, entonces es estacionaria.

Aquí termina el procedimiento recursivo puesto que ninguno de las dos subsecuencias vuelve a segmentarse. El esquema luego de finalizada la segmentación queda:

$$\overbrace{00000000}^{S_{ii}} \overbrace{10101110100010}^{S_{idii}} \overbrace{11}^{S_{idid}} \overbrace{000001101111}^{S_{idd}} \overbrace{000000010000010000011011}^{S_d}$$

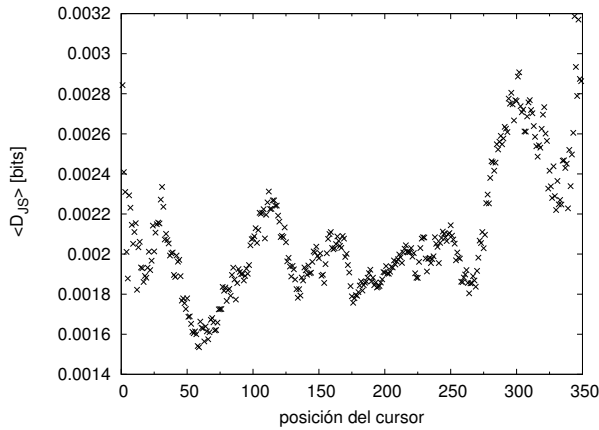


Figura 10: Promedio en el ensamble de la DJS para cada posición del cursor. Primera parte del paso 5 del método de segmentación.

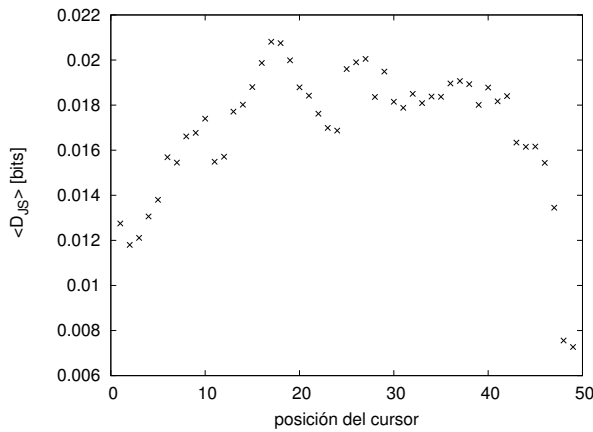


Figura 11: Promedio en el ensamble de la DJS para cada posición del cursor. Segunda parte del paso 5 del método de segmentación.

Resumen

Ordenando lo que obtuvimos, podemos reconstruir las subsecuencias con sus respectivas longitudes, como se muestra en la Tabla 2.

Estos valores coinciden exactamente con aquellos con los cuales generamos el ensamble de secuencias (comparar con la Tabla 1).

De este modo, hemos visto funcionar el método de segmentación y nos podemos convencer que la localización del máximo de la DJS permite separar regiones de diferente composición en una secuencia simbólica.

Quizá merece un breve comentario el hecho de que hayamos trabajado promediado un ensamble de 100 secuencias. Como construimos las distribuciones a partir de estimadores, los valores de la DJS fluc-

$\mathcal{S}^{(1)} = \mathcal{S}_{ii}$	$\mathcal{S}^{(2)} = \mathcal{S}_{idii}$	$\mathcal{S}^{(3)} = \mathcal{S}_{idid}$	$\mathcal{S}^{(4)} = \mathcal{S}_{idd}$	$\mathcal{S}^{(5)} = \mathcal{S}_d$
$n^{(1)} = 200$	$n^{(2)} = 350$	$n^{(3)} = 50$	$n^{(4)} = 300$	$n^{(5)} = 600$

Tabla 2: Subsecuencias reconstruidas.

túan de una secuencia a otra por más que éstas estén generadas con la(s) misma(s) distribución(ones) de probabilidad.

Por lo tanto, si trabajamos con una única secuencia (en lugar de trabajar con 100) el método funciona pero la detección de los bordes entre las subsecuencias no es tan precisa.

Si podemos disponer de un ensamble de secuencias, las curvas para la [DJS](#) se suavizan y la detección mejora sustancialmente. Pero mientras más grande es el número de secuencias que consideremos, mayor será el costo computacional de calcular la [DJS](#). Tenemos pues un compromiso en la elección del número de secuencias con el que trabajar¹.

Analizando varios valores posibles para el número de secuencias encontramos que 100 permitía una detección precisa sin aumentar tanto el costo de almacenamiento y procesamiento de las secuencias. Es por este motivo que trabajamos este ejemplo directamente con 100 secuencias y es también la razón por la que se utiliza de este modo en el [Capítulo 4](#) en el que adaptamos el método de segmentación para detectar paredes de dominio en la dinámica del motor molecular KIF1A.

¹ ¡Suponiendo que podemos disponer de un ensamble!, si sólo tenemos una secuencia, no hay nada que discutir.

Parte II

APLICACIONES BIOLÓGICAS

(...) es decir, las ideas que podrían surgir en la mente del físico que, después de haber aprendido su física y, más especialmente, la fundamentación estadística de su ciencia, empieza a pensar en los organismos y en cómo funcionan y se comportan, llegando a preguntarse conscientemente si él, basándose en lo que ha aprendido, puede hacer alguna contribución relevante al problema desde el punto de vista de su ciencia, tan simple, clara y modesta en comparación.

— Erwin Schrödinger ².

² Extraído del libro “¿Qué es la vida?” [62].

DETECCIÓN DE PAREDES DE DOMINIO EN LA DINÁMICA DEL MOTOR MOLECULAR KIF_{1A}

*The operative industry of Nature is so prolific that machines will be eventually found not only unknown to us but also unimaginable by our mind.*¹

— *Marcello Malpighi (1666)*²

KIF_{1A} es el nombre de un motor molecular citoesquelético³. Esto significa que se encuentra en el interior de las células eucariotas y que se mueve a lo largo de alguno de los filamentos que componen la estructura de estas células.

Este motor molecular pertenece a la familia kinesina y, como los demás miembros de la familia, se mueve de forma dirigida a lo largo de los microtúbulos de manera altamente procesiva: cuando un motor se une al filamento da cientos de pasos antes de desligarse. Esta característica es muy importante para su función biológica que es la de transportar cargas de una parte a otra de la célula.

Un mismo filamento puede ser utilizado simultáneamente por un gran número de motores, para una concentración suficientemente grande eso da lugar a fenómenos colectivos, uno de los más importantes es la formación de *shocks* o *paredes de dominio*. Una pared de dominio es la interfase que separa regiones de alta y baja densidad de ocupación de motores.

En este capítulo presentamos el desarrollo de un método de análisis basado en métodos entrópicos que permite estudiar la dinámica de los motores moleculares KIF_{1A}. A partir de la divergencia de Jensen-Shannon estudiamos secuencias simbólicas binarias que representan el estado estacionario de un modelo estocástico que describe el tráfico de la proteína motora KIF_{1A}.

Más específicamente, las secuencias binarias construidas representan el estado de un protofilamento del microtúbulo (MT) para diferentes concentraciones de adenosín trifosfato (ATP) y motores KIF_{1A} en el citosol. El modelo presenta algunos regímenes estacionarios con coexistencia de fase. El método de análisis desarrollado permite identificar los casos en que se produce coexistencia de fase y en tales

¹ La industria operativa de la Naturaleza es tan prolífica que eventualmente se descubrirán máquinas no sólo desconocidas para nosotros sino también inimaginables para nuestra mente.

² Como se cita en la Referencia [9].

³ En la notación estandarizada introducida por Lawrence *et al.* en el año 2004 este motor se denomina *kinesina-3* [35].

casos localizar, de manera automática, la posición en el filamento de la interfase que separa las regiones con diferente fase [65].

En la Sección 4.1 se dan algunas nociones generales sobre los motores moleculares KIF1A incluyendo qué son, cómo se mueven, y los problemas que se originan cuando no funcionan de manera apropiada. En la Sección 4.2 se presenta el modelo para el tráfico de motores KIF1A utilizado y se describe la simulación desarrollada para implementarlo. En la Sección 4.3 se propone un método de localización de paredes de dominio basado en la DJS y se discuten los resultados encontrados. Por último, en la Sección 4.4, se resumen los resultados presentados en este capítulo a modo de conclusión parcial.

Este capítulo hace uso de denominaciones y conceptos propios de la biología. Por este motivo se incluyó, en el Apéndice C, una introducción elemental a la biología de la célula, en la cual se presentan de forma breve los conceptos que se utilizan en este capítulo. Se sugiere al lector no familiarizado con ciertas nociones básicas de biología referirse al mencionado apéndice antes de adentrarse en la lectura del presente capítulo.

4.1 GENERALIDADES ACERCA DEL MOTOR MOLECULAR KIF1A

4.1.1 Motivación: ¿qué? y ¿para qué?

En cualquier célula eucariota y en particular en las neuronas, como la que se muestra en la Figura 12, hay decenas de motores moleculares citoesqueléticos.

Las células eucariotas tienen una especie de esqueleto que se denomina *citoesqueleto*, que consiste en un conjunto de filamentos. El citoesqueleto es responsable de su forma, es clave para su locomoción, provee resistencia para soportar tensiones y provee las sendas a partir de la cual se posicionan diversas organelas y se produce el transporte intracelular [2].

En la Figura 12, los microtúbulos (MTs), uno de los tipos de filamentos, están representados como varillas. A estos filamentos se ligan motores moleculares citoesqueléticos, unas proteínas elongadas que se desplazan a lo largo de ellos transportando cargas.

Hay tres grandes familias de motores moleculares: miosina, kinesina y dineína. Estos motores se ligan a los filamentos del citoesqueleto y usan la energía derivada de la hidrólisis de ATP para moverse a lo largo de estas sendas. Las kinesinas y dineínas usan los MTs como senderos a lo largo de los cuales transportan cargas, mientras que las miosinas se mueven a lo largo de los microfilamentos y generan la fuerza para la contracción de los músculos [2, 29].

Los motores citoesqueléticos son responsables del transporte intracelular de cargas en las células eucariotas, especialmente en las células polarizadas estructuralmente (células asimétricas) como las neuro-

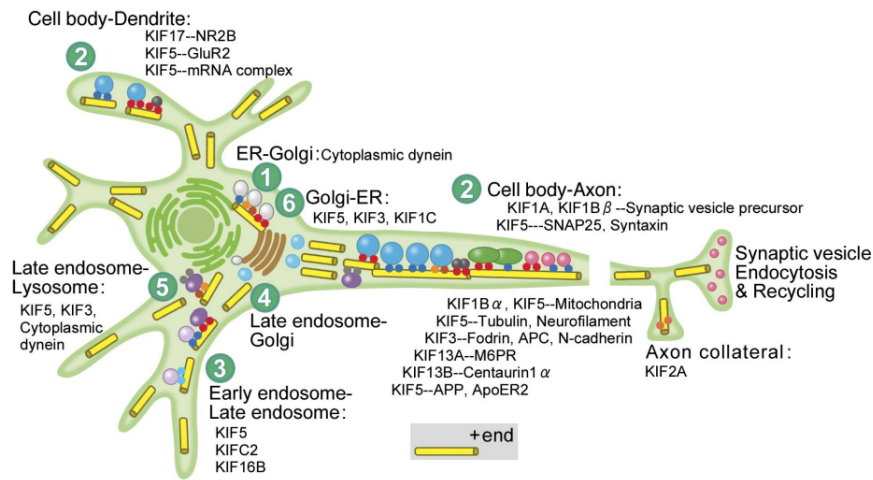


Figura 12: Representación esquemática de una neurona y del transporte intracelular mediado por motores moleculares que en ella ocurre. Los MTs se representan como varillas, el extremo en que se observa la sección de la varilla representa el extremo (+). Se muestran diversos motores moleculares y se especifica la carga que transportan. KIF1A puede observarse en el axón, a la derecha, arriba. Figura extraída de la Referencia [26].

nas y las células epiteliales, en las cuales la difusión en el citoplasma de las proteínas requeridas para su normal funcionamiento está severamente limitada por el tamaño y la morfología de la célula.

Estos motores moleculares tienen funciones biológicas muy importantes, que incluyen la segregación de cromosomas durante la división celular, el transporte axonal en las neuronas y el transporte en el interior de los cilios y flagelos. Estudios recientes han demostrado que tienen también roles esenciales en procesos fisiológicos como la determinación de izquierda-derecha, la supresión de la génesis tumoral, el cableado cerebral, la supervivencia neuronal y algunas funciones superiores del cerebro [27].

Los motores KIF1A pertenecen a la familia kinesina y se mueven sobre los MTs, hacia el extremo (+). En la representación de la Figura 12, el extremo (+) corresponde a aquel en el cual se ve la sección de la varilla. En el axón de las neuronas los MTs están orientados con el extremo (+) apuntando hacia el terminal del axón (donde hace contacto sináptico con la neurona siguiente) y están desordenados en las dendritas, como se puede observar en la figura.

En general, en una célula eucariota, los MTs están orientados con los extremos (+) hacia la periferia de la célula.

Ligándose a los MTs hay una gran diversidad de motores moleculares que se mueven transportando organelas hacia su ubicación apropiada en la célula y transportando diversos componentes que la célula necesita para funcionar adecuadamente.

Hay una alta especificidad entre el tipo de motor y la carga que liga y transporta, y eso define su función biológica. La función principal del motor molecular KIF1A, que puede verse en la Figura 12 en la parte superior derecha, es el transporte axonal, transporta vesículas con precursores sinápticos hacia el terminal del axón [28]. Estos precursores sinápticos son neurotransmisores que la neurona libera cada vez que se produce una sinapsis.

4.1.1.1 Problemas en el transporte intracelular.

Hay evidencia de que disfunciones y defectos en el transporte intracelular están asociados a varias enfermedades neurodegenerativas [11]. Por ejemplo, defectos en el transporte de mitocondrias ha sido relacionado a la Esclerosis Lateral Amiotrófica, mientras que defectos en el transporte de ARN, a la Atrofia Muscular Espinal.

En pacientes con enfermedades de la neurona motora se encontró una hinchazón axonal que muestra acumulación de kinesina [67]. Es decir, los motores no estaban transportando las cargas a su destino y estaban acumulándose en un lugar del axón.

Esta es una motivación fuerte para intentar comprender el funcionamiento de los motores moleculares y las posibles causas de los defectos en el transporte intracelular que llevan a cabo.

En la Figura 13 se muestran resultados experimentales *in vitro* realizados con motores moleculares KIF1A. En el experimento se marcaron por fluorescencia el MT, de color verde, y los motores KIF1A, de color rojo.

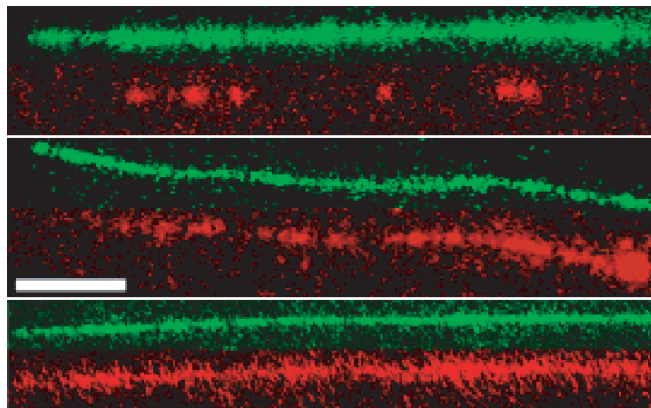


Figura 13: Acumulación de kinesina en el extremo del microtúbulo. KIF1A se observa marcado por fluorescencia (rojo) y el microtúbulo también (verde). En la figura superior la concentración de KIF1A es de 10pM, en la figura central de 100pM y en la figura inferior de 1000pM. En todos los casos la concentración de ATP es de 2mM. La longitud de la barra blanca es de 2 μ m. Figura extraída de la Referencia [47].

Se muestran tres imágenes con distintas concentraciones de motores. En la imagen superior hay algunos pocos motores ligados en distintas partes del MT. En la imagen del medio se ve una región embotellada en el extremo (+) y el resto del filamento con motores ligados de forma dispersa. En la imagen inferior la concentración de motores es tan grande que todo el filamento se encuentra completamente embotellado.

Este experimento, realizado *in vitro* con motores moleculares KIF1A únicamente, ya presenta evidencia de embotellamiento.

4.1.2 ¿Qué son los motores moleculares KIF1A?

En la Figura 14 vemos una representación del motor molecular KIF1A. Lo que se ve en color gris representa un protofilamento del MT, que se forma ensamblando de forma ordenada subunidades de la proteína tubulina.

La tubulina es un dímero formado por dos proteínas globulares: α -y β -tubulina.

El MT es un cilindro hueco compuesto por 13 protofilamentos paralelos, presenta una polaridad estructural que tiene su origen en el hecho de que los dímeros de tubulina se ensamblan de forma ordenada. El extremo en que α -tubulina queda expuesta se denomina extremo (-) y aquel en que queda expuesta β -tubulina se denomina extremo (+).

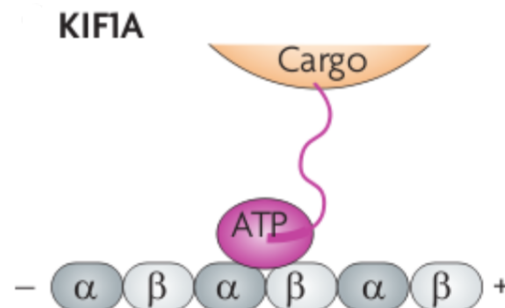


Figura 14: Esquema de un protofilamento del microtúbulo en el cual se encuentra ligado un motor molecular KIF1A. En la representación, el motor tiene ligada una molécula de ATP en su dominio motor y en el otro extremo posee una carga adherida. Figura extraída de la Referencia [28].

El motor KIF1A se representa de color rosa en la Figura 14. La parte globular que está en contacto con el protofilamento se denomina *dominio motor* ó *cabeza*. KIF1A es un monómero, lo que significa que tiene un único dominio motor, a diferencia de otros miembros de la familia kinesina que tienen dos, entre ellos, la kinesina convencional.

El dominio motor es la región con la que KIF1A se relaciona con el MT y también la región en la cual liga moléculas de ATP, de las que obtiene la energía que utiliza para moverse de forma dirigida.

Del dominio motor sale una región elongada, que se denomina *cola*, en el extremo de la cual el motor molecular liga las cargas que va a transportar.

KIF1A es una proteína que convierte energía química en energía mecánica. Una vez que liga una molécula de ATP, la hidroliza y utiliza la energía que obtiene de la hidrólisis para avanzar a lo largo del MT.

Debido a que convierte energía química en energía mecánica se lo conoce como *motor molecular*.

También nos referiremos a él como *motor citoesquelético*, *proteína motora*, *motor KIF1A* ó simplemente *motor*. A todos estos términos los utilizaremos de forma casi indistinta, si bien motor molecular, proteína motora o motor citoesquelético pueden hacer referencia a otras proteínas en otro contexto.

Motor molecular es el término más genérico de todos y se utiliza para todas las proteínas que convierten energía química en energía mecánica, entre ellas las polimerasas de ácido desoxirribonucleico (ADN) (que se mueven a lo largo de las hebras de ADN para replicarla), las polimerasas de ácido ribonucleico (ARN) (que se mueven a lo largo de las hebras de ADN para transcribirla a una molécula de ARN) y las bombas iónicas (que transportan iones y pequeñas moléculas a través de la membrana celular).

Motor citoesquelético hace referencia a todos los motores moleculares que se ligan y mueven sobre alguno de los filamentos del citoesqueleto, esto incluye todos los miembros de la familia kinesina, miosina y dineína.

Proteína motora se refiere a todas las proteínas cuya función principal es generar fuerza mecánica y movimiento, esto incluye a los motores citoesqueléticos y también a las helicasas de ADN (separan las hebras durante la replicación).

Al igual que las máquinas inventadas por los hombres para funcionar de forma eficiente en el mundo macroscópico, estas proteínas contienen partes móviles altamente coordinadas. Subyaciendo esta coordinación hay una serie ordenada de cambios conformacionales llevados adelante a través de la hidrólisis de ATP. Como hay disipación de energía libre, estos cambios proceden en una única dirección.

4.1.3 Mecanismo de motilidad

Para subir una escalera se comienza por levantar esa parte del cuerpo situada a la derecha abajo, envuelta casi siempre en cuero o gamuza, y que salvo excepciones cabe exactamente en el escalón. Puesta en el primer peldaño dicha parte, que para abreviar llamaremos pie, se recoge la parte equivalente de la izquierda (también llamada pie, pero que no ha de

confundirse con el pie antes citado), y llevándola a la altura del pie, se le hace seguir hasta colocarla en el segundo peldaño, con lo cual en éste descansará el pie, y en el primero descansará el pie. (Los primeros peldaños son siempre los más difíciles, hasta adquirir la coordinación necesaria. La coincidencia de nombre entre el pie y el pie hace difícil la explicación. Cuídese especialmente de no levantar al mismo tiempo el pie y el pie.)

— Julio Cortázar ⁴

Una de las características más importantes de los motores KIF1A es que son altamente procesivos, pueden dar cientos de pasos sin desligarse del filamento. De esta manera, a pesar que la distancia entre dos sitios de ligadura del MT es de 8nm (tamaño de un dímero de tubulina), se pueden desplazar distancias de micrómetros de longitud en una sola corrida [49].

¿Cómo hacen los motores KIF1A para moverse? ¿cuál es el mecanismo que les permite dar cientos de pasos sin desligarse?

Históricamente, la procesividad se entendió primero para otro miembro de la familia kinesina, la kinesina convencional [29].

Este motor molecular, a diferencia de KIF1A, es un dímero, es decir que posee dos cabezas en vez de una. Lo que se encontró fue que existe una coordinación entre ambas cabezas de modo que siempre hay una cabeza ligada al MT. Esto se logra a través de una alternancia entre las cabezas en un mecanismo que se denominó “mano sobre mano” y que consiste en que la cabeza delantera permanece ligada al MT hasta que la cabeza trasera se mueve y liga en la posición siguiente del MT, momento en el cual intercambian roles.

Este movimiento es similar al caminar de los seres humanos, en el cual un pie permanece en el suelo hasta que el otro se adelanta y hace contacto con el suelo (y distinto a correr, para lo que resulta necesario despegar ambos pies del suelo).

Pero, ¡KIF1A es un monómero! Por lo tanto, no puede utilizar el mismo mecanismo. A pesar de esto, la evidencia experimental demostró que es un motor procesivo [49].

Se tardó un par de años más en entenderlo pero finalmente se encontró que KIF1A avanza alternando entre dos estados de diferente afinidad con el MT [50, 51, 48]:

ESTADO 1: el motor se encuentra fuertemente ligado a un sitio particular del filamento.

ESTADO 2: el motor se encuentra desligado del filamento pero anclado a él mediante una atracción electrostática que evita que difunda hacia cualquier parte del interior celular. Esta atracción electrostática restringe el movimiento del motor mientras se encuentra en este estado, a un movimiento difusivo unidimensional, paralelo al MT.

⁴ Fragmento de “Instrucciones para subir una escalera” [15].

En ocasiones el estado 1 se denomina *estado fuerte* y el estado 2, *estado débil*. Utilizaremos indistintamente, ambas denominaciones.

La procesividad se debe a que estos estados alternan permitiendo que el motor se desplace a lo largo del MT sin “soltarse” completamente de él.

Esta alternancia está regulada a partir de la hidrólisis de ATP, es decir, hay un acoplamiento entre su movimiento mecánico y su ciclo bioquímico. Ver Figura 15.

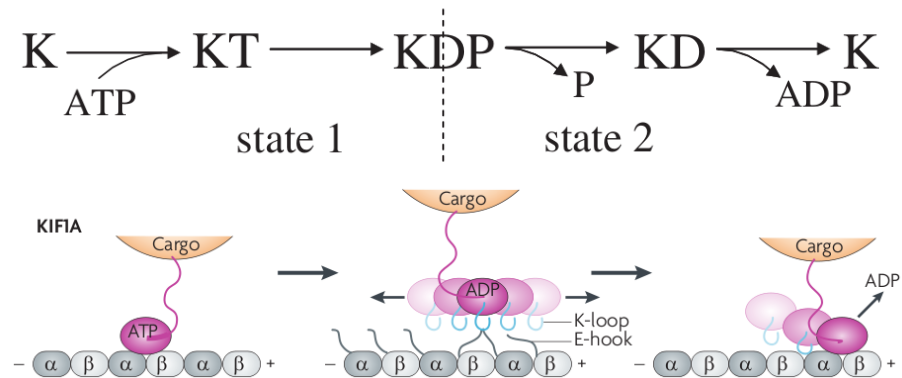


Figura 15: Acoplamiento mecano-químico del motor KIF1A. La figura superior, extraída de la Referencia [47], muestra los diferentes estados en que se puede encontrar al motor molecular en su relación al ciclo de hidrólisis y sus productos. La figura inferior, extraída de la Referencia [28], muestra el acoplamiento mecánico del motor al MT en los diferentes estados. Ver el texto principal por más detalles.

Los alcances de este acoplamiento no son sólo la provisión de energía sino también la direccionalidad del ciclo de cambios conformacionales. Al acoplar el cambio conformacional a la hidrólisis de ATP el mecanismo inverso implicaría volver a crear el enlace covalente del grupo fosfato que se rompió, utilizando agua, adenosín difosfato (ADP) y P_i para producir una molécula de ATP, lo cuál es energéticamente muy costoso y por lo tanto (casi) nunca sucede.

Imaginemos que tenemos un motor KIF1A ligado a un sitio del filamento. En la parte superior de la Figura 15, lo representamos con la letra K.

Cuando una molécula de ATP se liga al motor molecular, lo representamos como KT.

Con la molécula de ATP ligada puede producirse la hidrólisis. Esto es, se rompe el enlace del grupo fosfato terminal, obteniéndose una molécula de ADP y fosfato inorgánico (en este proceso se incorpora también una molécula de agua, de ahí el nombre de hidrólisis). La hidrólisis de la molécula de ATP libera energía que provoca un cambio conformacional en el motor.

Hay un estado intermedio, de tiempo de vida muy corto en el cual el motor ya realizó la hidrólisis pero tiene ligado el ADP y el fosfato, esto lo representamos en la figura como KDP.

En primer término se libera el fosfato por lo que sólo la molécula de ADP permanece ligada al motor, esto es lo que representamos como KD.

El cambio conformacional que sufre el motor lo suelta del filamento y lo acerca al sitio de ligadura siguiente, allí permanece anclado por atracción electrostática pero con la posibilidad de difundir unidimensionalmente. Como el cambio conformacional lo acercó al sitio siguiente lo más probable es que al liberar el ADP se ligue a este sitio, volviendo al comienzo del ciclo en estado K.

Los estados K y KT tienen la misma afinidad con el filamento y constituyen lo que llamamos estado 1. El estado KD es el que llamamos estado 2. Al estado KDP no tiene sentido clasificarlo porque, como dijimos, tiene un tiempo de vida mucho más corto que los otros estados, así que podemos pensarlo simplemente como un estado de transición.

En la parte inferior de la Figura 15 vemos una representación de este proceso. En primer lugar, en la imagen de la izquierda, el motor está ligado a un sitio del filamento y tiene en su dominio motor ligada una molécula de ATP. Al hidrolizarla sufre un cambio conformacional que lo suelta del filamento y lo acerca al siguiente sitio de ligadura, como se ve en la imagen central. Desligado del filamento difunde en una dimensión, paralela al MT por una atracción electrostática que le impide alejarse. Por último, como muestra la imagen de la derecha, al liberar el ADP el motor se liga al sitio más cercano que la mayoría de las veces es el sitio siguiente, a veces es el mismo sitio donde comenzó y ocasionalmente es el sitio anterior.

Un artículo que revisa los mecanismos de motilidad de KIF1A puede encontrarse en la Referencia [28].

4.2 MODELO PARA LA DINÁMICA DE KIF1A

El proceso de exclusión simple asimétrico (ASEP) es un modelo que consiste en partículas que saltan a lo largo de una red unidimensional con exclusión mutua, esto significa que no puede haber más de una partícula por sitio y por lo tanto para que una partícula se mueva debe hacerlo a un sitio vacío [17].

En un trabajo de Parmeggiani y colaboradores [55] se introdujo un modelo tipo ASEP que relaja la restricción de la conservación de partículas en el *bulk*, permitiendo la ligadura y desligadura de partículas en cualquier sitio. Este modelo matemático, inspirado en los motores moleculares citoesqueléticos, ha mostrado coexistencia de fase en las que fases de alta y baja densidad están separadas por disconti-

nidades estables en el perfil de densidad, lo que llamamos *shocks* o paredes de dominio.

En dos artículos recientes de Nishinari *et al.* [47] y Greulich *et al.* [22], se introdujo un modelo para los motores KIF1A, fuertemente inspirado en el modelo de Parmeggiani pero enriquecido con dos contribuciones significativas.

Por un lado, incorporaron de manera explícita el mecanismo *ratchet* para cada uno de los motores KIF1A. Esto es, incluyeron el acoplamiento mecano-químico involucrado en el movimiento dirigido de KIF1A que tiene en cuenta el ciclo de hidrólisis de ATP por el cual el motor obtiene la energía para avanzar.

Por otro lado, el modelo está construido a partir de parámetros que tienen correspondencia directa con cantidades controlables experimentalmente, siendo los dos más importantes la concentración de ATP y la concentración de motores KIF1A.

A continuación presentaremos el modelo de Nishinari, Greulich y colaboradores, y discutiremos la simulación que desarrollamos para implementar dicho modelo, posteriormente se presentará el método de análisis que permite la localización de paredes de dominio en el estado estacionario de dicho modelo.

4.2.1 Ecuaciones maestras

Un protofilamento del MT está representado por una red unidimensional de 600 sitios de ligadura con dos estados internos, que representan las dos formas en que un motor se puede relacionar con él.

Como ya se mencionó, la tubulina es la subunidad básica que conforma el MT y, por lo tanto, su longitud determina el período estructural del MT. La distancia entre dos sitios de ligadura consecutivos en el MT es igual a la longitud de la tubulina, 8 nm y por lo tanto ese es el valor que se elige para el parámetro de red.

De esta manera la longitud del filamento es $600 \times 8 \text{ nm} = 4,8 \text{ }\mu\text{m}$, que es del orden de magnitud del tamaño típico de una célula eucariota.

Un motor molecular KIF1A está representado por una partícula que salta a lo largo de la red unidimensional con exclusión mutua. Esto significa que si un motor está ocupando un dado sitio de la red, no puede otro motor ligarse a ese sitio.

El modelo queda especificado a partir de las ecuaciones maestras para las probabilidades de encontrar un motor ligado al filamento en un dado sitio en alguno de los dos estados posibles. Estas ecuaciones, que se presentarán en seguida, definen completamente la dinámica del sistema para todo tiempo.

Denotemos por $r_i(t)$ y $q_i(t)$ a las probabilidades de encontrar un motor KIF1A en el sitio de red i a tiempo t en el estado 1 y 2, respecti-

vamente. La probabilidad de que el sitio i esté desocupado a tiempo t será: $1 - r_i(t) - q_i(t)$.

Las ecuaciones maestras para la dinámica de los motores que se encuentren en un sitio del *bulk* vienen dadas por:

$$\begin{aligned} \frac{dr_i}{dt} = & \omega_a (1 - r_i - q_i) - (\omega_d + \omega_h) r_i + \omega_s q_i \\ & + \omega_f q_{i-1} (1 - r_i - q_i), \end{aligned} \quad (52)$$

$$\begin{aligned} \frac{dq_i}{dt} = & \omega_h r_i - \omega_s q_i - \omega_f q_i (1 - r_{i+1} - q_{i+1}) \\ & - \omega_b q_i (2 - r_{i-1} - q_{i-1} - r_{i+1} - q_{i+1}) \\ & + \omega_b (q_{i-1} + q_{i+1}) (1 - r_i - q_i). \end{aligned} \quad (53)$$

Las ecuaciones maestras para el extremo (-) son:

$$\frac{dr_1}{dt} = \alpha (1 - r_1 - q_1) - (\gamma_1 + \omega_h) r_1 + \omega_s q_1, \quad (54)$$

$$\begin{aligned} \frac{dq_1}{dt} = & \omega_h r_1 - \omega_s q_1 - \omega_f q_1 (1 - r_2 - q_2) - \omega_b q_1 (1 - r_2 - q_2) \\ & - \gamma_2 q_1 + \omega_b q_2 (1 - r_1 - q_1). \end{aligned} \quad (55)$$

Y las ecuaciones maestras para el extremo (+), son:

$$\begin{aligned} \frac{dr_L}{dt} = & \delta (1 - r_L - q_L) - (\beta_1 + \omega_h) r_L + \omega_s q_L \\ & + \omega_f q_{L-1} (1 - r_L - q_L), \end{aligned} \quad (56)$$

$$\begin{aligned} \frac{dq_L}{dt} = & \omega_h r_L - \omega_s q_L - \beta_2 q_L - \omega_b q_L (1 - r_{L-1} - q_{L-1}) \\ & + \omega_b q_{L-1} (1 - r_L - q_L). \end{aligned} \quad (57)$$

Los parámetros de estas ecuaciones representan las tasas de transición entre los diferentes estados. El significado de cada uno de ellos se discutirá en detalle en la próxima sección.

4.2.2 Simulación

Para implementar el modelo, realizamos simulaciones Monte Carlo que actualizan el estado de todos los sitios del filamento en un orden secuencial aleatorio.

Esto significa, en primer lugar, que no seguimos la trayectoria de cada motor KIF1A en el citosol. Como nos interesa caracterizar el transporte intracelular y sus posibles defectos, estamos interesados sólo en los motores ligados y por lo tanto actualizamos el estado de ocupación de los sitios del filamento.

Por otro lado decir que el orden es secuencial aleatorio significa que vamos actualizando de a un sitio por vez, eligiendo el sitio al azar, hasta actualizar todos los sitios del filamento en cada paso Monte Carlo (MCS). El orden de actualización de los sitios se elige al azar

en cada MCS de manera de no sesgar a través de la simulación la interacción de los motores.

Recordemos que cada motor se puede mover sólo si su sitio destino está desocupado, por lo tanto el hecho de que otro motor se mueva o permanezca en su sitio limita los movimientos de los restantes.

Actualizar de forma aleatoria significa que en un dado MCS cualquier motor puede tomar la iniciativa de moverse primero.

En la simulación, el estado de ocupación de cada sitio podrá tomar tres valores:

- 0: si no hay motor ligado,
- 1: si hay un motor ligado en estado 1,
- 2: si hay un motor ligado en estado 2.

La evolución dinámica se describe a partir de *tasas de transición* que reflejan la naturaleza estocástica del movimiento de los motores moleculares.

En la Figura 16 se puede ver la única posibilidad de actualización del estado de ocupación de un sitio del filamento que se encuentra desocupado. En este caso, lo único que puede suceder es que un motor se ligue al sitio, en estado 1, lo cuál ocurre con una tasa ω_a .

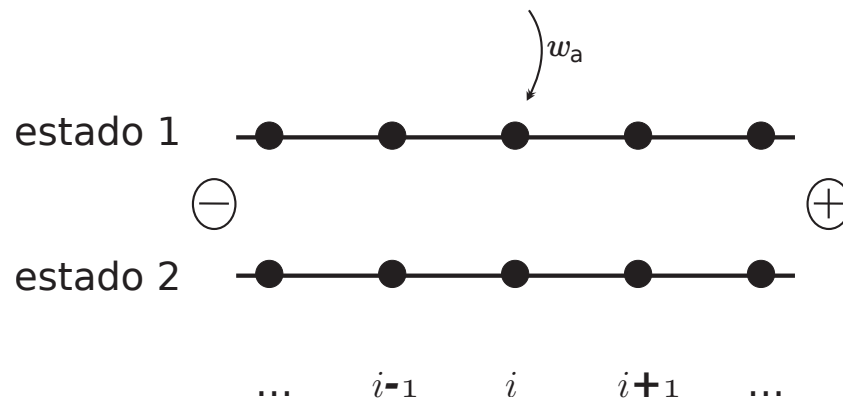


Figura 16: Esquema de las posibilidades de actualización para un sitio que se encuentra desocupado. Ver texto principal por más detalles.

En la figura se representa cada estado como una red diferente. Esto es así para visualizar mejor las transiciones pero no debe perderse de vista que hay un único filamento real y que los estados internos representan las diferentes formas en que el motor se relaciona con el filamento. La misma forma de representación se utiliza en las figuras que siguen.

Las posibilidades de actualización para un sitio en estado 1 se muestran en la Figura 17. El motor puede desligarse, con una tasa ω_d , quedando el sitio en el estado 0. También puede ligar una molécula de ATP, hidrolizarla y cambiar por lo tanto, al estado 2, con una tasa ω_h .

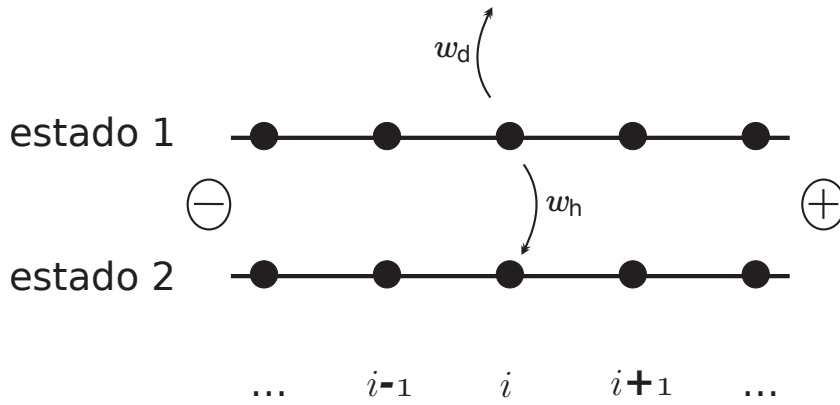


Figura 17: Esquema de las posibilidades de actualización para un sitio que se encuentra ocupado en estado 1. Ver texto principal por más detalles.

En la Figura 18 se muestran las posibilidades de actualización para un sitio que tiene ligado un motor en estado 2. Un motor ligado en estado 2 puede difundir hacia cualquiera de sus primeros vecinos, con una tasa ω_b . Puede también liberar el ADP y, o bien avanzar de forma activa con una tasa ω_f (ligándose al sitio siguiente en estado 1), o permanecer en el mismo sitio (cambiando al estado 1) con una tasa ω_s .

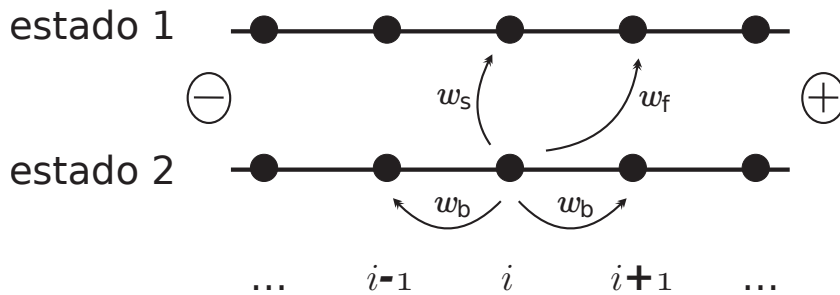


Figura 18: Esquema de las posibilidades de actualización para un sitio que se encuentra ocupado en estado 2. Ver texto principal por más detalles.

La dinámica de los sitios ubicados en los extremos debe tratarse de forma separada. Tomamos α y δ en lugar de ω_a como las tasas de ligadura en los extremos (-) y (+), respectivamente. Del mismo modo, γ_1 y β_1 en lugar de ω_d son las tasas de desligadura, y γ_2 y β_2 en lugar de ω_b son las tasas de salida de los motores debido a movimiento browniano en los extremos (-) y (+), respectivamente.

En la Figura 19 se muestra un esquema completo, que incluye las posibilidades de actualización de un sitio perteneciente al *bulk*⁵ y tam-

⁵ Significa que es un sitio que no está en los bordes del filamento, es decir que no es ninguno de los extremos.

bién las posibilidades de actualización de los extremos. Se incluyen en la misma figura las posibles transiciones entre todos los estados.

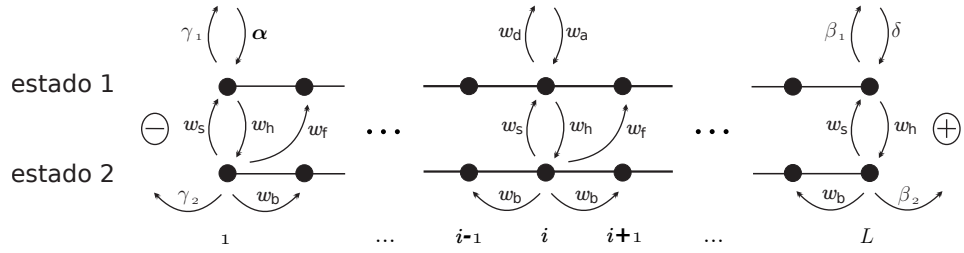


Figura 19: Esquema del modelo mostrando las posibilidades de actualización y las tasas de transición asociadas tanto para un sitio del *bulk* como para los dos extremos. La actualización del estado de ocupación de un dado sitio no sólo depende de su propio estado sino también del estado de sus primeros vecinos. Ver el texto principal para más detalles.

Es importante enfatizar que cualquier actualización que implique un movimiento de un motor a alguno de los sitios vecinos puede ocurrir solamente cuando el sitio destino está desocupado.

Las detalles de las relaciones de las tasas con las cantidades controlables experimentalmente y con los procesos físicos correspondientes a un mecanismo *ratchet*, así también como la elección de los valores de los parámetros puede verse en el Apéndice D.1.

Aquí nos concentramos en el estudio del estado estacionario que se obtiene para diferentes concentraciones de ATP y motores en el citosol. Por lo tanto, es importante enfatizar que ω_a es directamente proporcional a la concentración de motores, C_{KIF1A} , y que ω_h aumenta desde cero con concentración de ATP creciente hasta que llega a un valor de saturación de $0,25\text{ms}^{-1}$ si la concentración de ATP, C_{ATP} , es suficientemente grande.

4.2.2.1 ¿Cuánto tiempo es un paso Monte Carlo?

El intervalo de tiempo, dt , asociado a un MCS debe escogerse cuidadosamente: la suma de las probabilidades de todos los cambio de estados posibles debe ser menor o igual a uno.

Los distintos estados de ocupación posibles de un sitio y sus vecinos ofrecen distintas posibilidades de cambio de estado para el sitio a actualizar pero en todos lo casos debe cumplirse:

$$1 = \sum_i P_i + \xi,$$

donde ξ es la probabilidad de que el sitio permanezca en el mismo estado en que se encontraba y P_i es la probabilidad de ejecutar uno de los cambios permitidos según el estado de ocupación, i recorre todos los cambios permitidos.

En la notación con la que venimos trabajando: $P_i = \omega_i dt$. En este caso, ω_i es una tasa de transición permitida de acuerdo al estado de ocupación y dt es el intervalo de tiempo que representará un paso de la simulación. Por lo tanto, se cumple que:

$$1 = dt \sum_i \omega_i + \xi.$$

El intervalo de tiempo debe elegirse con cuidado ya que tenemos una relación de compromiso entre escogerlo suficientemente chico para que “pasen cosas” (que no tengamos muchos pasos de simulación idénticos porque las probabilidades de cambiar de estado sean muy pequeñas) y por otro, que no sea demasiado chico de modo de “perdernos cosas” (que el motor pueda hacer más de un cambio de estado en el intervalo de tiempo escogido).

En la práctica si mantenemos el intervalo de tiempo, dt , fijo durante toda la simulación tendremos que considerar cual de todas las combinaciones de estados de ocupación (del sitio y de sus primeros vecinos) da el caso en el cual el valor de $\sum_i \omega_i$ es máximo y entonces elegir dt de modo que cumpla la restricción:

$$dt \text{ MAX} \left[\sum_i \omega_i \right] \leq 1. \quad (58)$$

Elegimos:

$$dt \simeq \frac{1}{\text{MAX}[\sum_i \omega_i]}$$

pero ligeramente más chico, de modo de satisfacer la Ecuación (58). Con esa elección, para el caso máximo se cumple $\xi \simeq 0$, y todos los demás casos quedan bien definidos (en cada caso el valor de ξ será distinto pero en todos será mayor igual que cero, lo que garantiza que la simulación está bien definida).

Al correr las simulaciones en todos los casos se utilizó un intervalo de tiempo $dt = 0,71\text{ms}$ para un MCS, que es una buena elección con el criterio antes mencionado.

NOTA: podría suceder para algún caso que la diferencia entre $\sum_i \omega_i$ y el caso correspondiente al máximo $\text{MAX}[\sum_i \omega_i]$ sea muy grande, de modo que en dicha situación ξ es grande y un sitio puede permanecer muchos pasos de simulación en el mismo estado. Estas diferencias tienen que ver con la dinámica real del sistema y con la diferencia entre las tasas de transición de cada uno de los estados y no representan un problema de la simulación en si misma.

4.2.2.2 Detalles de la implementación

Como ya se mencionó, en un MCS actualizamos el estado del filamento completo siguiendo un orden de actualización secuencial aleatorio,

esto es, dando a todos los sitios, en orden aleatorio, la oportunidad de cambiar su estado de ocupación.

Para todas las combinaciones de parámetros con las que trabajamos y que se detallan más adelante, corremos una simulación que comienza sin motores ligados al filamento y dejamos que el sistema evolucione hasta alcanzar el estado estacionario.

Almacenamos el estado del filamento en cien pasos temporales una vez alcanzado el estado estacionario y construimos una secuencia binaria para cada uno de estos tiempos (pasos).

En el Apéndice D.2 se puede ver el núcleo alrededor del cual fue pensada la simulación desarrollada, y en el Apéndice D.3, el código fuente completo.

4.2.2.3 Construcción de las secuencias

La idea central es construir secuencias simbólicas que nos permitan identificar si se produjo un *shock* en el filamento y en caso afirmativo localizar su posición. Por eso decidimos construir secuencias binarias que tengan una relación uno-a-uno con los sitios de ligadura del filamento de modo que podamos *ir y venir* entre el filamento y las secuencias.

Como dijimos, se almacena el estado del filamento para cien instantes temporales una vez alcanzado el estado estacionario. A cada uno de esos tiempos le asociaremos una secuencia binaria que tendrá tantos caracteres como sitios hay en el filamento.

Ponemos un 0 en el primer carácter de la secuencia si el primer sitio del filamento está desocupado en ese instante temporal y un 1 si está ocupado con un motor en cualquiera de los dos estados. Repetimos para todos los sitios del filamento, de forma ordenada. Así, el carácter i -ésimo de la secuencia será un 0 si el sitio de ligadura i -ésimo del filamento estaba vacío en ese tiempo, y será un 1 si el sitio estaba ocupado (en cualquiera de los dos estados).

Construyendo las secuencias de esta forma distinguimos si el sitio está ocupado o desocupado, que es lo que nos interesa para caracterizar la producción de embotellamientos, y además logramos una referencia inmediata entre la posición en la secuencia y la posición en el filamento.

Otras formas de construir las secuencias fueron analizadas con objeto de distinguir de forma separada ambos estados. Consideraciones sobre estos resultados pueden verse en el Apéndice D.4.

4.3 LOCALIZACIÓN DE PAREDES DE DOMINIO

Realizamos simulaciones para diferentes combinaciones de los parámetros ω_a y ω_h , dentro de los valores biológicamente relevantes para estos parámetros (ver Apéndice D.1), y esperamos a que el sistema alcance el estado estacionario.

Lo que se observa es que, dependiendo de la concentración de ATP y de motores, el estado estacionario presenta diferentes características. Hay regímenes en los que hay baja densidad de ocupación de motores en todo el filamento. Otros en los cuales hay una única densidad de ocupación, alta. Y también hay algunos que presentan coexistencia de fase, es decir, en los cuales una región del filamento cercana al extremo (+) tiene una alta densidad de ocupación de motores, mientras que el resto del filamento que se extiende hacia el extremo (-) presenta una baja densidad de ocupación.

Una fase con baja densidad implica que los motores pueden desplazarse más o menos libremente a lo largo del filamento y por lo tanto, el transporte es normal. Una fase de alta densidad, en cambio, implica que no hay transporte porque hay una región embotellada que impide el flujo de motores. Los sitios están casi todos ocupados y, debido a la exclusión mutua, los motores están atascados sin poder avanzar.

En los casos en que hay una coexistencia de fases estable en el estado estacionario, diremos que se ha producido un *shock* o *pared de domino*. La posición de la interfase entre ambas regiones determina la posición de la pared de dominio.

A continuación se describe el procedimiento desarrollado, que es esencialmente una adaptación del método de segmentación descrito en el Capítulo 3. El método, basado en la DJS, permite distinguir de manera automática los casos en los que el estado estacionario presenta una única fase, de aquellos en que se produce una coexistencia de fases. Para estos últimos casos se localiza la posición del filamento en la cual se produce la pared de dominio.

4.3.1 Procedimiento

1. Correr la simulación del modelo para alguna combinación de parámetros convenientemente elegida. Una vez alcanzado el estado estacionario, grabar el estado del filamento en los últimos cien instantes temporales.
2. Construir para cada uno de los cien instantes temporales una secuencia binaria que represente el estado del filamento.
3. Evaluar la DJS para cada posición del cursor e ir almacenando su valor para cada uno de las cien secuencias.
4. Promediar el valor de la DJS en las cien secuencias para cada posición del cursor.
5. Buscar la posición del cursor para la cual la DJS promedio toma su valor máximo. Ese será nuestro candidato a punto de corte.
6. Evaluar la significancia máxima en la posición correspondiente al candidato a punto de corte. Si supera el valor del umbral,

truncar y repetir para cada una de las subsecuencias hasta que ninguna se segmente.

Diremos que hemos detectado un *shock* si el procedimiento detecta un punto de segmentación.

El concepto detrás del método es que cuando el estado estacionario presenta coexistencia de fases, las secuencias que construimos serán composicionalmente inhomogéneas. La región del filamento con baja densidad dará lugar a un segmento de la secuencia composicionalmente diferente a aquella correspondiente a la región de alta densidad y la máxima disimilitud se dará en la posición de la interfase. De esta manera, las secuencias construidas pueden analizarse de la forma discutida en el Capítulo 3, asociando ahora una segmentación en la secuencia a la presencia de una pared de dominio en el filamento. La posición del filamento en la cual se produjo la pared de dominio puede recuperarse porque construimos la secuencia en relación directa con los sitios de ligadura del MT, cada carácter de la secuencia representa una posición en el filamento.

Si, en cambio, el estado estacionario presenta una única fase (sin importar si es alta o baja), la secuencia asociada será composicionalmente homogénea y no habrá segmentación.

4.3.2 Resultados y comentarios acerca del procedimiento

Los resultados obtenidos a partir del esquema presentado anteriormente para una dada combinación de parámetros (ver Tabla 3), biológicamente plausibles⁶, se muestran en la Figura 20.

$\omega_a \left[\frac{1}{\text{ms}} \right]$	$\omega_d \left[\frac{1}{\text{ms}} \right]$	$\omega_h \left[\frac{1}{\text{ms}} \right]$	$\omega_s \left[\frac{1}{\text{ms}} \right]$	$\omega_f \left[\frac{1}{\text{ms}} \right]$	$\omega_b \left[\frac{1}{\text{ms}} \right]$
0.00001	0.0001	0.1	0.145	0.055	0.6

Tabla 3: Parámetros del modelo para el caso de la Figura 20 (y por lo tanto también para las Figuras 21a y 21b). El número de sitios de la red es $L = 600$ y los parámetros correspondientes a los extremos se utilizan de la siguiente forma: $\beta_1 = \beta_2 = \omega_d$, $\gamma_1 = \gamma_2 = \delta = 0$ y $\alpha = \omega_a$.

En la parte de abajo de la figura se muestra un gráfico espacial del estado estacionario del filamento para cien MCSs.

Una línea horizontal representa el estado del filamento completo a un dado instante temporal. Hay un punto blanco por cada sitio vacío, un punto azul por cada sitio ocupado con un motor en estado 1, y un punto rojo por cada sitio ocupado con un motor en estado 2.

A partir de esta imagen espacial del filamento esta claro que se ha producido un *shock* para esta combinación de parámetros. Hay una

⁶ Ver Apéndice D.1 por una discusión de este punto.

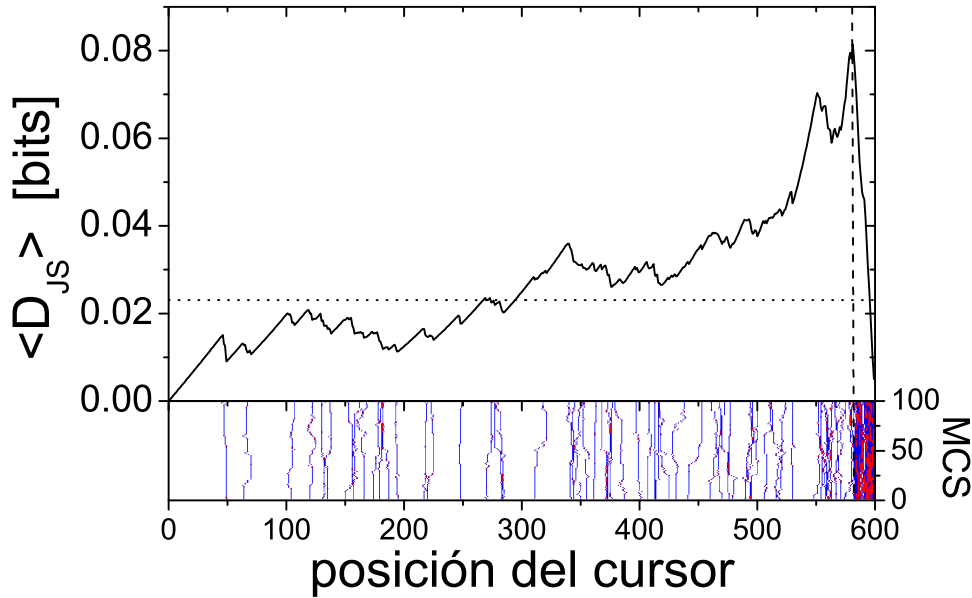


Figura 20: Valores de la DJS promedio para cada posición del cursor correspondientes a un caso en el cual el estado estacionario presenta coexistencia de fases. Los parámetros del modelo para este caso se explicitan en la Tabla 3. Se muestra también el valor crítico que corresponde al umbral de segmentación para este caso. La línea de puntos vertical corresponde a la posición 581 en la cual se detectó un *shock*. En la parte inferior de la figura se puede ver un gráfico espacial en el que cada fila representa el estado del filamento en un dado instante temporal. Se muestran cien MCS una vez que el sistema alcanzó el estado estacionario. Un punto blanco representa un sitio vacío para ese tiempo, un punto azul, un sitio ocupado con un motor en estado 1 y un punto rojo, un sitio ocupado con un motor en estado 2. Ver el texto principal para más detalles.

pequeña región de alta densidad a la derecha, cerca del extremo (+) del filamento, mientras que el resto del filamento se encuentra en una fase de baja densidad con la mayoría de los sitios desocupados.

En la parte superior de la figura se pueden observar los valores obtenidos para la DJS promedio para cada posición del cursor a lo largo de la secuencia⁷ y el valor para el umbral correspondiente a la secuencia completa.

El valor umbral es el valor crítico de la DJS promedio, es decir el valor de la DJS promedio para el cual la significancia toma el valor del umbral $s_0 = 0,999$ y se muestra en la gráfica con el único objetivo

⁷ De ahora en adelante, utilizaremos el término “secuencia” para designar el conjunto completo de cien secuencias. Lo mismo vale para “subsecuencia”.

de ayudar en el entendimiento y visualización del método. Ver el Apéndice B.3.2 por más información.

Como la DJS promedio toma valores más grandes que el umbral, la secuencia debe ser truncada. El máximo de la DJS promedio se da para la posición del cursor 581 y este es, por lo tanto, el punto de corte.

En la Figura 21 se muestra la DJS promedio para las dos subsecuencias con sus respectivos valores para el umbral. En la Figura 21a se muestran los resultados para la subsecuencia uno (subsecuencia de la izquierda) y en la Figura 21b los resultados para la subsecuencia dos (subsecuencia de la derecha).

Es importante enfatizar que el valor del umbral depende de la longitud de la secuencia bajo consideración y es mayor para secuencias más cortas. Ver Apéndice B.3.2, en particular la Figura 42.

Ninguna de las dos subsecuencias tienen valores para la DJS promedio que superen el valor de sus respectivos umbrales y por lo tanto no volvemos a segmentar, y podemos concluir que tenemos un caso en que se produjo una pared de dominio en la posición 581 del filamento. Esta pared de dominio separa una fase de baja densidad que se extiende hacia el extremo (-) de una fase de alta densidad que se extiende hacia el extremo (+).

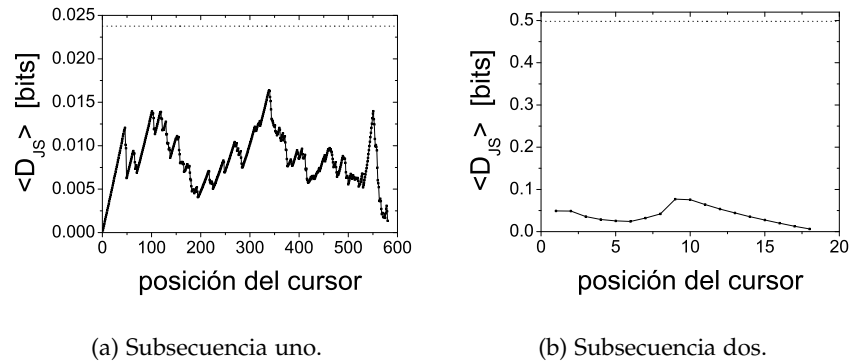


Figura 21: Valores de la DJS promedio para cada posición del cursor correspondientes a las dos subsecuencias en que se divide el caso de la Figura 20. Se muestra también el valor crítico que corresponde al umbral de segmentación para cada una de las subsecuencias. En ninguno de los dos casos el valor de la DJS promedio supera el umbral por lo tanto el procedimiento finaliza en este paso. Ver el texto principal para más detalles.

En efecto, para la secuencia completa se obtuvo un valor máximo $\langle D_{JS_{max}} \rangle = 0,08145$ en la posición $x_{max} = 581$, calculando la significancia máxima allí, obtuvimos $s_{max}(N = 600, \langle D_{JS_{max}} \rangle = 0,08145) = 1 > 0,999$, entonces se trunca en las subsecuencias S_1 con una longitud $N_1 = 581$ y S_2 con una longitud $N_2 = 19$.

Luego de realizada la segmentación, para la subsecuencia uno se encontró que el valor máximo de la DJS promedio es $\langle D_{JS_{max}} \rangle = 0,01637$, en la posición $x_{max} = 339$. La significancia máxima toma el valor $s_{max} (N = 581, \langle D_{JS_{max}} \rangle = 0,01637) = 0,9873 < 0,999$ y por lo tanto la subsecuencia no se trunca.

Por otro lado, para la subsecuencia dos, se encontró que el valor máximo es $\langle D_{JS_{max}} \rangle = 0,07684$, en la posición $x_{max} = 9$. La significancia máxima toma el valor $s_{max} (N = 19, \langle D_{JS_{max}} \rangle = 0,07684) = 0,8271 < 0,999$ y por lo tanto esta subsecuencia tampoco se trunca.

Vale la pena mencionar que la morfología y los valores de la DJS promedio cambian cuando segmentamos la secuencia, con resultados que no son equivalentes para la misma posición del cursor antes y después de segmentar.

Esto está claro a partir de la altitud de los picos. Por ejemplo, el segundo pico en importancia de la secuencia completa en la Figura 20 no coincide con el pico principal de la subsecuencia de la izquierda que se puede ver en la Figura 21a.

De este modo, la posición donde debemos calcular el valor de la significancia para determinar si la subsecuencia uno debe o no ser segmentada no es la posición que esperaríamos al observar los valores de la secuencia completa. La razón para esta diferencia es que la región de la secuencia que acabamos de truncar no forma parte de los cálculos de la subsecuencia y por lo tanto cambia los valores de la DJS promedio para todas las posiciones del cursor.

Así, queda claro que la DJS debe ser recalculada para cada una de las subsecuencias de modo de poder establecer si vuelven a truncarse o no, tal como se discutió en el Capítulo 3.

La elección del umbral de significancia es crucial para que el método funcione apropiadamente. Encontramos que un valor de $s_0 = 0,999$ es una buena elección cuando comparamos los resultados del procedimiento con los gráficos espaciales del estado del filamento para distintas combinaciones de concentración de ATP y concentración de motores, como puede observarse en la Figura 22.

La concentración de motores, que es proporcional a ω_a , aumenta conforme nos movemos hacia la derecha en la gráfica, mientras que la concentración de ATP aumenta con ω_h conforme nos movemos hacia arriba. El resto de los parámetros se utiliza como se detalla en la Tabla 4.

$\omega_d \left[\frac{1}{ms} \right]$	$\omega_s \left[\frac{1}{ms} \right]$	$\omega_f \left[\frac{1}{ms} \right]$	$\omega_b \left[\frac{1}{ms} \right]$
0.0001	0.145	0.055	0.6

Tabla 4: Parámetros del modelo para el caso de la Figura 22. El número de sitios de la red es $L = 600$ y los parámetros correspondientes a los extremos se utilizan de la siguiente forma: $\beta_1 = \beta_2 = \omega_d$, $\gamma_1 = \gamma_2 = \delta = 0$ y $\alpha = \omega_a$.

Lo primero que podemos notar en la Figura 22 es que dependiendo de la combinación de parámetros se produce o no un *shock*.

Si la concentración de ATP es muy baja, como en los tres casos de la fila inferior (Figuras 22(g)- 22(i)), la dinámica de los motores es esencialmente de ligadura y desligadura del filamento. El ATP disponible es escaso y el movimiento dirigido de los motores es prácticamente inexistente (los motores obtienen la energía para avanzar de la hidrólisis de ATP), provocando que el estado estacionario presente una única fase.

Los casos de las Figuras 22(g) y 22(h) presentan una única fase de baja densidad.

En cambio, el caso de la Figura 22(i) presenta una única fase de alta densidad. Si bien la concentración de ATP es baja, la concentración de motores es lo suficientemente alta como para que el filamento esté completamente ocupado con motores ligados.

Una situación similar ocurre para los casos de las Figuras 22(c) y 22(f) en los cuales la dinámica está esencialmente regulada por la alta concentración de motores que produce una única fase de alta densidad. El papel de la concentración de ATP se reduce a regular el número de motores que se encuentra en estado 1 y el número de motores que se encuentra en estado 2, pero no juega un rol crucial en la formación de *shocks* en estos casos.

En los restantes cuatro casos, que corresponden a las Figuras 22(a), 22(b), 22(d) and 22(e), las secuencias no son homogéneas y hay *shocks* en todos esos casos.

La posición del *shock*, que se ha podido localizar con el método, se marca con una línea de puntos vertical en cada una de las gráficas.

En la Figura 22(a) hay claramente dos fases separadas por una pared de dominio. Aunque la concentración de motores es baja, la concentración de ATP es suficientemente alta como para que el régimen estacionario presente un *shock*.

En la Figura 22(d), se produce un *shock* en una posición más cercana al extremo (+) que en el caso de la Figura 22(a). Esto puede entenderse por el hecho de que la concentración de ATP es más baja que en el caso anterior, lo cual ralentiza la dinámica de los motores provocando una disminución en el tamaño de la región embotellada y desplazando la posición de la pared de dominio hacia la derecha, más cerca del extremo (+).

En los casos correspondientes a las Figuras 22(b) y 22(e), la dinámica es más ruidosa teniendo regiones intermedias que son difíciles de clasificar pero presentando una clara coexistencia de fases (la región de la izquierda del filamento con baja densidad y la región derecha, con alta densidad). En ambos casos el análisis da lugar a la segmentación de la secuencia.

En el caso de la Figura 22(b) el filamento se divide en tres regiones: la región más cercana al extremo (+) con concentración alta de

motores y las otras dos con concentración baja. Este caso es útil para enfatizar los pasos del procedimiento recursivo descrito previamente. Luego de la primera segmentación de la secuencia trabajamos con cada subsecuencia separadamente, esto es, repetimos los pasos 3 al 6 de la Sección 4.3.1 para ambas subsecuencias.

De esta forma, encontramos que la subsecuencia de la derecha no vuelve a dividirse pero la subsecuencia de la izquierda sí. Por lo tanto, procedimos a truncar nuevamente la subsecuencia de la izquierda y repetimos el procedimiento para las dos nuevas subsubsecuencias. El procedimiento termina allí puesto que estas dos subsubsecuencias resultan homogéneas a significancia s_0 . Así, el resultado final, como muestra la Figura 22(b) es que el filamento se divide en tres regiones.

Tomamos a este como un caso con pared de dominio porque presenta una coexistencia de fase con baja concentración de motores cerca del extremo (-) y concentración alta cerca del extremo (+).

En el caso de la Figura 22(e) el filamento se divide en dos regiones.

Calculamos, para los casos de la Figura 22 que presentan *shock*, exactamente el mismo procedimiento en 1050 simulaciones donde todos los parámetros eran iguales excepto por la semilla del generador de números pseudo-aleatorios.

Las desviaciones estándar correspondientes a la posición del *shock* para cada uno de los casos puede verse en la Tabla 5. Estos valores dan el orden de magnitud del error en el método de detección de *shocks*.

Queda claro de estos resultados que la dinámica es mucho más ruidosa en los casos 22(b) y 22(e), como ya fue mencionado.

FIGURA 22(A)	FIGURA 22(B)	FIGURA 22 (D)	FIGURA 22(E)
12	38	10	92

Tabla 5: Desviación estándar de la posición del *shock* en unidades del parámetro de red para diferentes casos. Los parámetros para cada uno de los casos tienen los mismos valores que en las Figuras 22(a), 22(b), 22(c) y 22(d). Los valores que se muestran fueron obtenidos de 1050 simulaciones con diferentes semillas para el generador de números pseudo-aleatorios.

Analizando la columna de la izquierda en la Figura 22, comenzando desde la Figura 22(a) y bajando hacia las Figuras 22(d) y 22(g), para concentraciones de ATP cada vez más bajas, pareciera que la posición del *shock* se mueve hacia la derecha hasta que eventualmente desaparece.

Esto sugiere analizar cómo cambia la posición del *shock* cuando modificamos la concentración de ATP en el citosol (el valor de ω_h).

Estos resultados se muestran en la Figura 23, en donde se utilizaron los parámetros que se detallan en la Tabla 6.

En efecto, se observa que conforme la concentración de ATP aumenta la posición del *shock* se mueve hacia el extremo (-). Esto se debe a que al aumentar la concentración de ATP, la dinámica de los motores en el filamento se acelera debido a que los tiempos de espera disminuyen (estamos aumentando la disponibilidad de combustible) y los motores son más propensos a embotellarse, formando una región de alta densidad de mayor tamaño y, por lo tanto, desplazando la posición de la pared de dominio más cerca del extremo (-).

Al observar la Figura 23 conviene recordar que el extremo (+) del filamento corresponde a la posición 600 y el extremo (-), a la posición 1 (lo mismo vale para la Figura 24).

ω_a [ms^{-1}]	ω_d [$\frac{1}{\text{ms}}$]	ω_s [$\frac{1}{\text{ms}}$]	ω_f [$\frac{1}{\text{ms}}$]	ω_b [$\frac{1}{\text{ms}}$]
0.00001	0.0001	0.055	0.145	0.6

Tabla 6: Parámetros del modelo para el caso de la Figura 23. El número de sitios de la red es $L = 600$ y los parámetros correspondientes a los extremos se utilizan de la siguiente forma: $\beta_1 = \beta_2 = \omega_d$, $\gamma_1 = \gamma_2 = \delta = 0$ y $\alpha = \omega_a$.

También analizamos el comportamiento de la posición del *shock* cuando modificamos la concentración de motores (y consecuentemente ω_a). Esto se muestra en la Figura 24, en la cual se utilizaron los parámetros que se detallan en la Tabla 7.

En este caso, la posición del *shock* cambia más dramáticamente, con casos en los cuales la región embotellada cubre el filamento casi hasta el extremo (-).

Como los motores pueden ligarse al filamento en cualquier posición resulta razonable que, para una concentración de motores suficientemente alta, el filamento completo se encuentre completamente ocupado con motores ligados.

Para mejorar la visualización de lo que está sucediendo, se muestran en la Figura 25 los mismos resultados pero agregando un gráfico espacial para cada uno de los casos. Para poder incluirlos fue necesario invertir la posición de los ejes de coordenadas respecto a la Figura 24. En los gráficos espaciales se puede observar la posición del *shock*, lo que facilita la interpretación de la variación de su posición descrita en la parte superior de la figura.

ω_d [$\frac{1}{\text{ms}}$]	ω_h [$\frac{1}{\text{ms}}$]	ω_s [$\frac{1}{\text{ms}}$]	ω_f [$\frac{1}{\text{ms}}$]	ω_b [$\frac{1}{\text{ms}}$]
0.0001	0.2	0.055	0.145	0.6

Tabla 7: Parámetros del modelo para el caso de las Figuras 24 y 25. El número de sitios de la red es $L = 600$ y los parámetros correspondientes a los extremos se utilizan de la siguiente forma: $\beta_1 = \beta_2 = \omega_d$, $\gamma_1 = \gamma_2 = \delta = 0$ y $\alpha = \omega_a$.

4.4 CONCLUSIONES DEL CAPÍTULO

La *DJS* entre distribuciones de probabilidad se utilizó para decidir si el estado estacionario de un modelo para el motor molecular KIF_{1A} presenta coexistencia de fases y para localizar la posición de la pared de dominio en los casos en que se produce. Para ello fue necesario desarrollar previamente una simulación para el tráfico de los motores KIF_{1A}.

El procedimiento para la localización de *shocks* es un método recursivo basado en las diferencias composicionales de las secuencias que describen el estado estacionario en los casos con coexistencia de fase. La región de alta densidad cercana al extremo (+) es composicionalmente diferente a la región de baja densidad cercana al extremo (-), cuando ambas fases están presentes en el estado estacionario del sistema y la *DJS* toma un valor máximo en la posición de la pared de dominio.

La elección de los pesos utilizados para el cálculo de la *DJS* (tomados proporcionales a la longitud de las subsecuencia bajo análisis) nos permite evitar efectos de bordes no deseados que estarían presentes si utilizáramos los mismos pesos para ambas subsecuencias, sin importar su longitud. Esto, a su vez, evita la detección de falsos positivos en la búsqueda de paredes de dominio en nuestro problema.

La *DJS* se promedia sobre cien secuencias en el estado estacionario para minimizar el efecto del ruido, mejorando la detección de los *shocks*.

El valor del umbral de significancia es importante para que el método recursivo funcione apropiadamente. Un valor de umbral inferior podría dar lugar a la detección de inhomogeneidades más sutiles en la secuencia, y por lo tanto en el perfil de densidad de los motores.

Las secuencias binarias fueron construidas con el propósito de detectar los *shocks* que separan las regiones de densidad alta de las regiones de densidad baja y por lo tanto, se mapeó un 0 a todo sitio vacío y un 1 a todo sitio ocupado (sin importar si el motor se encontraba en el estado 1 o el estado 2).

Hay otras formas posibles de construir las secuencias binarias, exploramos otras dos. Primero, mapear un cero a todo sitio vacío u ocupado con un motor en estado 2 y un 1 a todo sitio ocupado con un motor en estado 1. Segundo, mapeando un 0 a todo sitio vacío u ocupado con un motor en estado 1 y un 1 a todo sitio ocupado con un motor en estado 2. Estas construcciones permiten la visualización de los perfiles de densidad de cada uno de los posibles estados de forma separada. Encontramos los mismos resultados que previamente. En los casos con *shock*, se encontraron en las mismas posiciones, pero los resultados son mucho más ruidosos (especialmente en los casos en que la mayoría de los sitios ocupados tienen motores en el mismo estado). Cuando ambos perfiles de densidad individuales pre-

sentaron *shock*, la posición fue la misma, consistente con el hecho de que en un modelo con dos estados internos, si hay un *shock* en una componente y también en la otra, deben coincidir (si no es así, el perfil global presentaría dos *shocks*). Por lo tanto, podemos concluir que el mapeo utilizado en la construcción de las secuencias es adecuado para la localización de las paredes de dominio en el filamento.

Como un comentario final vale la pena destacar que un enfoque similar al presentado en este capítulo para la detección de *shocks* en la dinámica de KIF1A puede utilizarse para detectar embotellamientos en cualquier otro modelo de motores moleculares (o para otros modelos de tráfico) si se mapea la dinámica a secuencias simbólicas de la misma manera que hicimos en este capítulo.

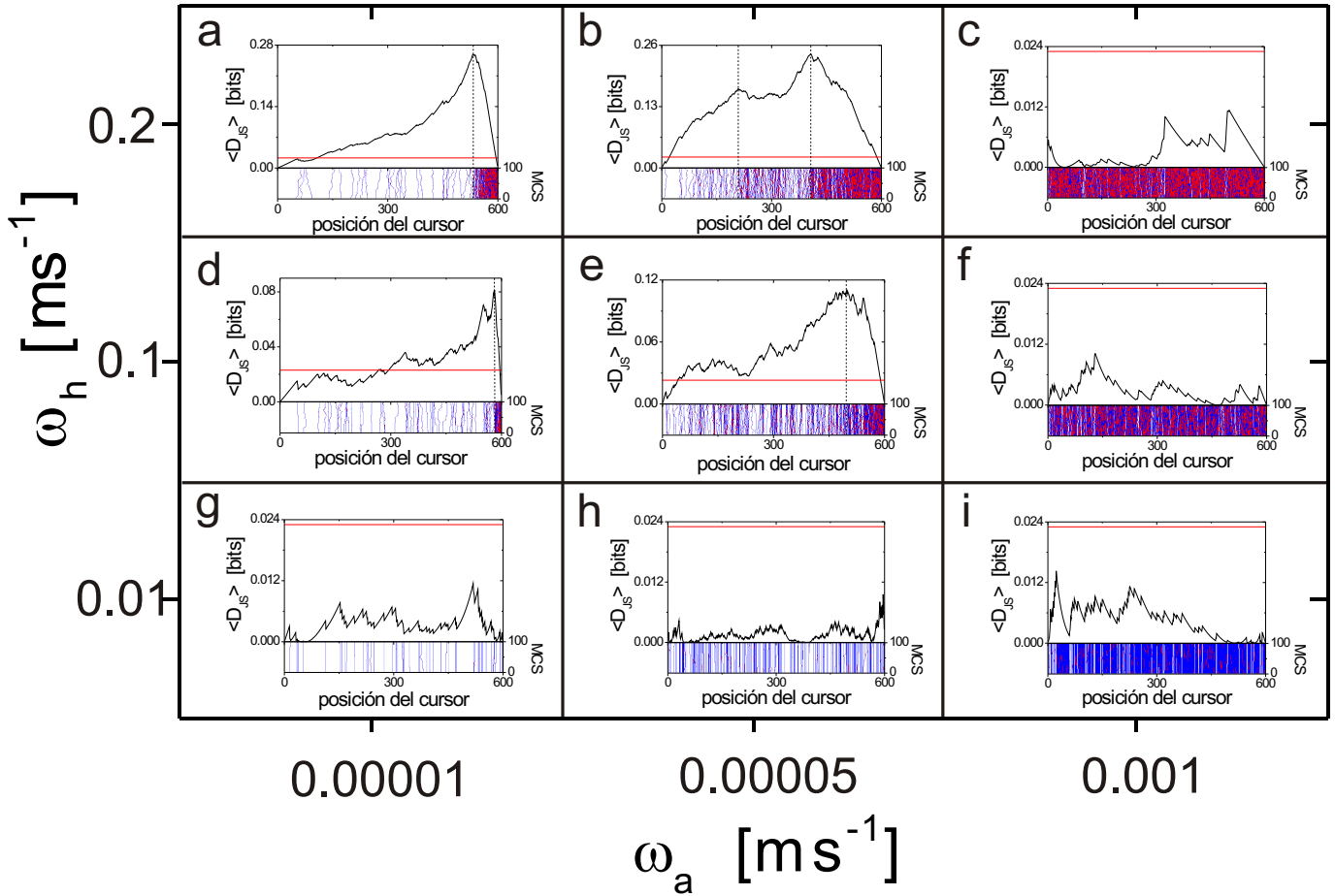


Figura 22: Detalles del procedimiento de localización de *shocks* para diferentes combinaciones de parámetros ω_a (relacionado a la concentración de KIF1A) y ω_h (relacionado a la concentración de ATP). Se muestra la *DJS* promedio para cada posición del cursor para cada uno de los casos. También se muestra el valor del umbral, es la función constante mostrada en cada uno de los gráficos y tiene el mismo valor porque depende únicamente de la longitud de la secuencia. En las Figuras (a), (b), (d) y (e) los valores de la *DJS* promedio son mayores que el del umbral y las secuencias se truncan en la posición que se muestra con línea punteada vertical. Esa corresponde, por lo tanto, a la posición de la pared de dominio. Ver el texto principal para una explicación detallada. En los casos restantes: (c), (f), (g), (h) y (i), la *DJS* promedio toma siempre valores más pequeños que el umbral y por lo tanto no hay *shock*. Para cada combinación de parámetros ω_a y ω_h se muestra, en la parte inferior de la correspondiente figura, un gráfico espacial del mismo modo que en la Figura 20.

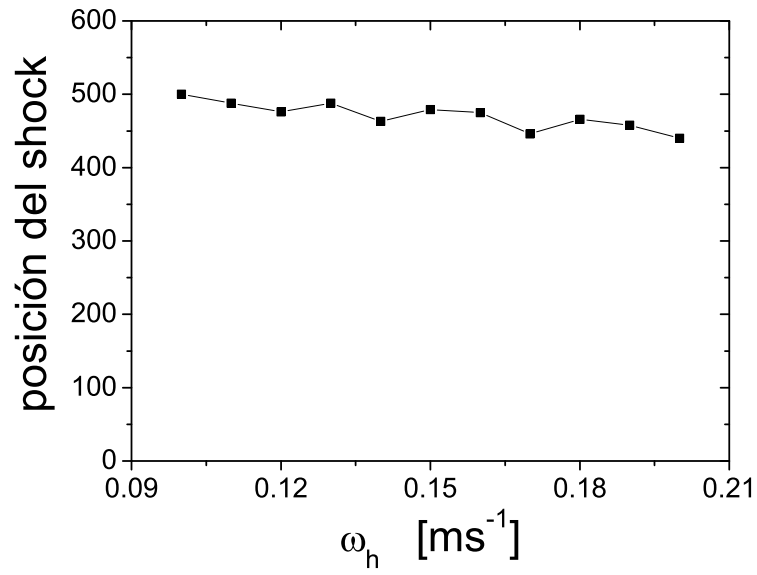


Figura 23: Posición del *shock* para diferentes valores de ω_h (relacionado a la concentración de ATP en el citosol). Los parámetros del modelo para este caso se explicitan en la Tabla 6.

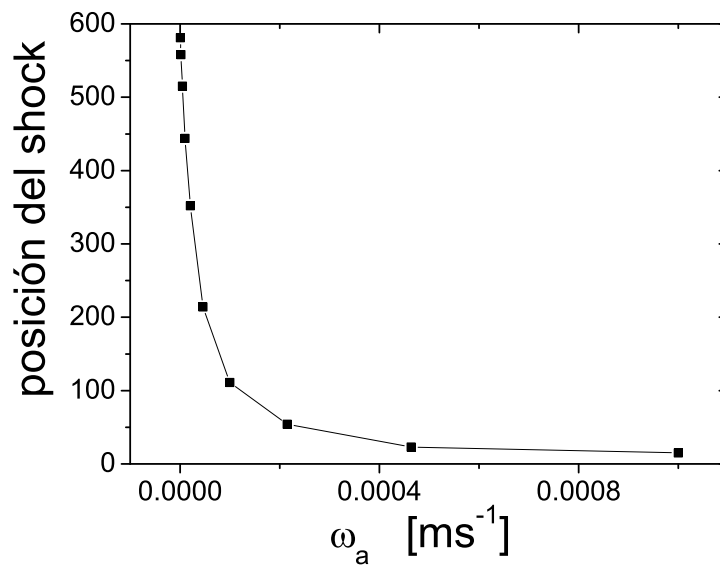


Figura 24: Posición del *shock* para diferentes valores de ω_a (relacionado a la concentración de motores KIF1A en el citosol). Los parámetros del modelo para este caso se explicitan en la Tabla 7.

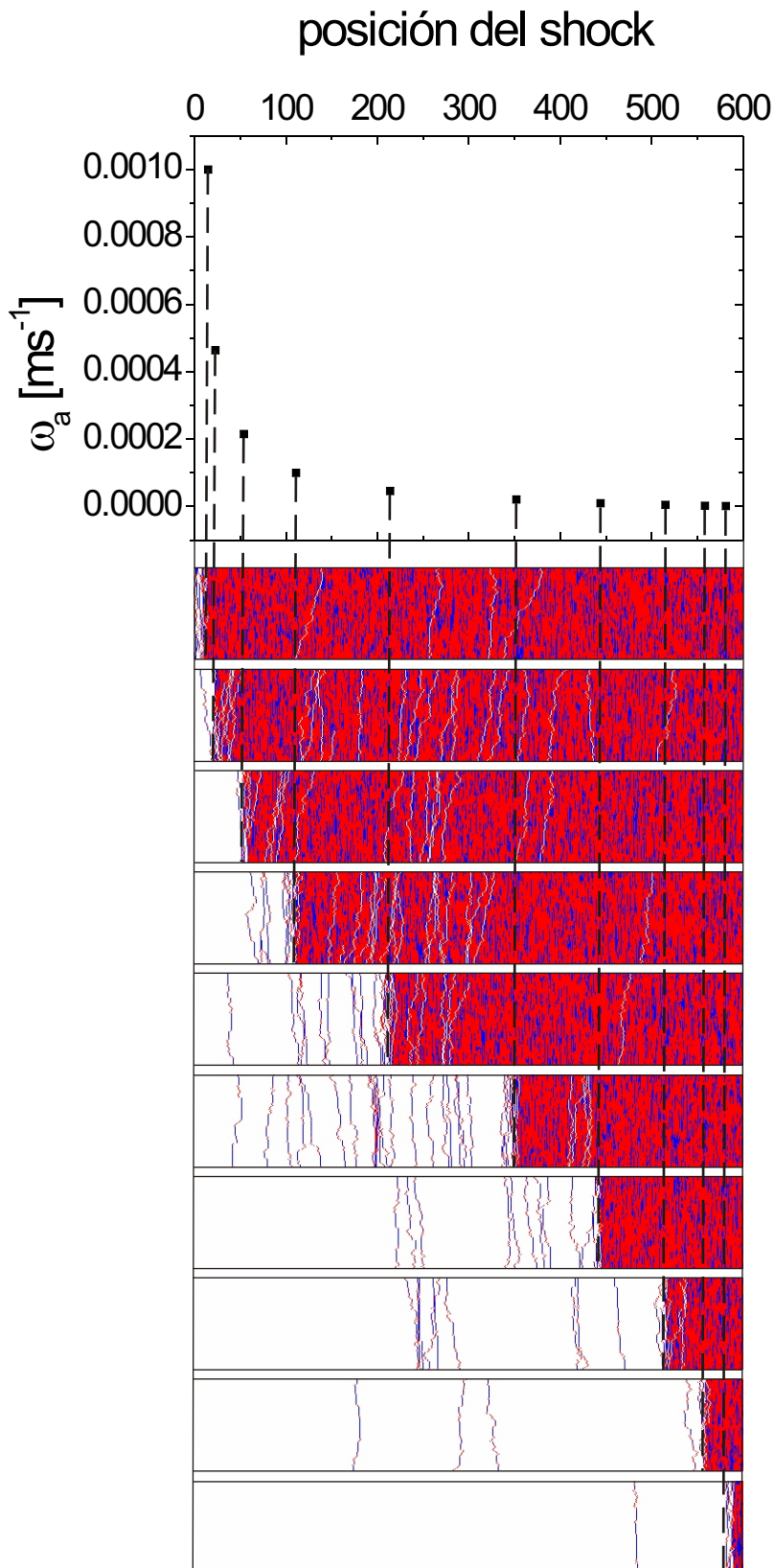


Figura 25: Se presentan los mismos datos que en la Figura 24 con los ejes de coordenadas invertidos de modo de mostrar el gráfico espacial correspondiente a cada uno de los casos.

EFEECTO DE LA PROTEÍNA ASOCIADA AL MICROTÚBULO TAU EN LA DINÁMICA DE KIF_{1A}

El transporte intracelular basado en motores moleculares y su regulación son cruciales para el funcionamiento de las células. Los filamentos en los cuales se desplazan los motores citoesqueléticos tienen ligadas proteínas no-móviles conocidas como *proteínas asociadas al microtúbulo* (MAPs) que interfieren con los motores moleculares.

Motivados por experimentos en los cuales se estudió la interacción entre la kinesina y la MAP tau desarrollamos un modelo estocástico para los motores moleculares KIF_{1A} que tiene en cuenta las interacciones entre éstos y las moléculas de tau [66].

En este capítulo presentamos el modelo propuesto y discutimos algunos de sus resultados. En la Sección 5.1 se dan algunas nociones básicas sobre las proteínas asociadas al microtúbulo, en general, y sobre la proteína tau, en particular. Además se discuten los detalles de la interacción entre tau y las kinesinas, presentando no sólo resultados experimentales de la literatura sino también cuáles son los efectos que consideramos más importantes para incluir en un modelo. En la Sección 5.2 se introduce el modelo estocástico propuesto para la dinámica de los motores KIF_{1A} en presencia de moléculas de tau y se describe la forma de implementarlo en una simulación Monte Carlo. También se presenta una solución analítica aproximada en el límite de alta densidad de motores y se compara con los resultados de las simulaciones. Por último se comenta brevemente sobre el límite del modelo cuando no hay moléculas de tau presentes en el filamento. En la Sección 5.3 se presentan algunos resultados del modelo propuesto, en particular se discuten los comportamientos del tiempo de ligadura y de la longitud de corrida de los motores KIF_{1A}, que son observables muy importantes para caracterizar la dinámica de las proteínas motoras y que se utilizan para analizar los efectos de la interacción con tau. En la Sección 5.4 se discute la localización de paredes de dominio en presencia de moléculas de tau. Por último en la Sección 5.5 se hace una revisión de los resultados del capítulo a modo de conclusión parcial.

5.1 GENERALIDADES ACERCA DE TAU

5.1.1 Motivación: ¿por qué es importante incluir a tau en el modelo?

Los filamentos en los cuales se desplazan los motores citoesqueléticos están abundantemente decorados de proteínas no-móviles conocidas

como proteínas asociadas al microtúbulo (MAPs), tales como MAP2, tau y MAP4, entre otras [36, 25].

Estas proteínas se ligan al MT y tienen un rol importante en la organización del citoesqueleto. En particular, promueven el crecimiento de neuritas y controlan el agrupamiento y espaciado de los MTs *in vivo* [13].

Por otro lado, las MAPs pueden afectar el transporte intracelular debido a motores moleculares. Como ya se mencionó en el Capítulo 4 el transporte intracelular es fundamental para el funcionamiento celular y sus disrupciones o defectos están asociados a ciertas patologías neurodegenerativas (ver Sección 4.1.1.1).

Las MAPs son capaces de inhibir el transporte activo de material citoplasmático al interferir con los motores moleculares de la familia kinesina. Esta interferencia se da de dos formas, por un lado, las kinesinas compiten con las MAPs por la ligadura a la superficie del MT y, por otro, MAPs ya ligadas al MT pueden bloquear el paso de proteínas motoras que se están desplazando a lo largo de éste.

Tau es una proteína asociada al microtúbulo (MAP) principalmente neuronal, que se encuentra de forma abundante en los axones de las neuronas, en donde es la MAP más abundantemente expresada [53].

Entre otras funciones, tau se asocia a la estabilización del MT, a la generación de procesos celulares y al crecimiento y agrupación de los MTs.

Estructuralmente, la proteína tau puede dividirse en un dominio con el cuál se liga a los MTs y un dominio terminal que se proyecta más allá de la superficie del MT e interactúa con otras componentes del citoesqueleto [52].

Ha sido demostrado que tau puede inhibir el transporte de vesículas mediado por las kinesinas [20]. La presencia de tau reduce no sólo la frecuencia de ligadura de la kinesina a los MTs sino también la distancia que la kinesina es capaz de recorrer a lo largo del MT en una sola corrida [69].

También se ha observado que cuando motores de kinesina se encuentran con una acumulación de moléculas de tau, la mayoría de los motores se desliga del MT [18].

5.1.2 Interacción kinesina-tau

En un trabajo del año 2008, Dixit y colaboradores [18] estudiaron la interacción entre las kinesinas y las moléculas de tau con técnicas de molécula individual.

Marcaron por fluorescencia las kinesinas y las moléculas de tau y observaron de manera directa, a partir de un microscopio de fluorescencia de reflexión total interna, los encuentros individuales entre kinesina y tau. Analizaron los efectos de tau tanto en la motilidad de

la kinesina a lo largo del MT como en la dinámica de ligadura/desligadura del filamento.

Los principales resultados de Dixit *et al.* son:

- cuando se produce un encuentro entre una kinesina y un tau, la kinesina se desliga, se detiene o atraviesa. Aproximadamente la mitad de las veces se desliga, un tercio de las veces atraviesa y en las restantes ocasiones se detiene.
- la frecuencia de ligadura de la kinesina al MT depende de la concentración local de tau.
- la frecuencia de ligadura promedio de la kinesina en presencia de tau en el citosol se reduce a aproximadamente un tercio del valor que tiene en ausencia de tau.
- la presencia de tau disminuye la longitud de corrida de la kinesina¹.
- cuando una kinesina encuentra acumulación de moléculas de tau, se desliga. La posición más probable para la desligadura se da en el pico de la intensidad de fluorescencia del tau. Por lo tanto, la probabilidad de desligadura es una función de la concentración local de tau.

Podemos agrupar estas observaciones en dos efectos diferentes, por un lado tau afecta la frecuencia de ligadura de la kinesina al filamento y, por otro lado, tau también afecta a motores que están ligados y desplazándose por el filamento.

Además, otro hecho a destacar de las observaciones de Dixit y colaboradores es que ambos efectos dependen de la concentración local de tau.

A partir de estas consideraciones, modelamos el efecto de tau, en su interacción individual con una molécula de kinesina a través de dos parámetros p_a y p_m que cuantifican los dos efectos mencionados.

El parámetro p_a corresponderá a la probabilidad de que una molécula de tau impida la ligadura al filamento de una kinesina; y el parámetro p_m corresponderá a la probabilidad de que una kinesina que se está moviendo a lo largo del filamento se desligue al encontrarse en su camino con una molécula de tau.

Si bien las observaciones experimentales muestran que, al producirse un encuentro entre una kinesina moviéndose a lo largo del filamento y una molécula de tau, la kinesina puede desligarse, atravesar o detenerse, lo más importante es que la kinesina se desliga o permanece en el filamento. De esta manera, simplificamos la descripción considerando que existe una probabilidad p_m de que, al producirse el encuentro, la molécula de tau obligue a la kinesina a desligarse,

¹ Por longitud de corrida se entiende la distancia promedio que una kinesina recorre en el filamento antes de desligarse.

dejando una probabilidad $1 - p_m$ de que la kinesina atravesase dicha posición. Como experimentalmente se observó que aproximadamente la mitad de las veces la kinesina se desliga, elegimos:

$$p_m = 0,5.$$

Respecto a la frecuencia de ligadura, como se observó que en presencia de tau decrece a aproximadamente un tercio del valor que tiene cuando no hay moléculas de tau, elegimos la probabilidad de que tau impida la ligadura igual a $2/3$ (de este modo se ligarán en promedio $1/3$ de las que se ligaban sin tau). De este modo, tomamos:

$$p_a = 0,67.$$

El hecho de modelar el efecto de tau mediante las interacciones individuales kinesina-tau permitirá reobtener las observaciones experimentales respecto a la concentración local de tau de forma automática a partir de la acumulación de los efectos individuales.

5.2 MODELO PARA LA DINÁMICA DE KIF1A EN PRESENCIA DE TAU

El modelo que proponemos para la dinámica de KIF1A en presencia de tau, es una extensión del modelo de Nishinari *et al.* y Greulich *et al.* que presentamos en la Sección 4.2. Las características básicas del modelo y la motilidad propia de los motores KIF1A se mantienen y se agrega una concentración de moléculas de tau en el filamento y la interacción entre los motores y tau, como se presentó en la sección anterior.

Al igual que en el Capítulo 4, trabajamos con una red unidimensional de L sitios que representan uno de los protofilamentos del MT.

Cada sitio en la red corresponde a uno de los sitios de ligadura que tiene el protofilamento y el parámetro de red se toma igual a 8nm , que es la separación entre dos sitios de ligadura adyacentes en el MT.

Cada motor KIF1A se representa por una partícula que, cuando está ligada al filamento, puede saltar hacia cualquiera de sus primeros vecinos (es decir sitios adyacentes). La proteína motora puede también ligarse a (desligarse de) el MT.

Cada sitio del filamento tiene dos estados internos que representan las dos formas posibles en las cuales un motor KIF1A puede relacionarse con el MT. En el estado 1, KIF1A está fuertemente ligado a un sitio del filamento, mientras que en el estado 2 está débilmente ligado (permanece anclado al MT mediante una atracción electrostática que evita que difunda alejándose del filamento).

Por lo tanto, tenemos tres estados posibles para cada sitio:

0: si el sitio está desocupado

1: si el sitio está ocupado con un motor en estado 1

2: si el sitio está ocupado con un motor en estado 2

Hasta aquí, estamos simplemente recordando las características principales del modelo para la dinámica de KIF1A, lo novedoso es que consideramos a su vez, una distribución de moléculas de tau decorando el filamento. Cada sitio de ligadura del filamento podrá o no tener una molécula de tau ligada.

Para caracterizar la cantidad de tau presente en el MT nos referiremos a la *concentración de moléculas de tau* y la denotaremos C_{tau} . A esta concentración la definimos como el número de moléculas de tau por sitio del filamento. Esto es, si hay n moléculas de tau presentes en un filamento con L sitios, diremos que éste tiene una concentración de tau: $C_{\text{tau}} = \frac{n}{L}$. Por lo tanto, los valores de la concentración de tau estarán siempre entre 0 y 1. Los casos extremos, 0 y 1, corresponden a que no haya ninguna molécula de tau en el filamento y que haya una molécula de tau en cada uno de los sitios del filamento, respectivamente.

Como Dixit y colaboradores encontraron que tau permanece estable por varios minutos, mientras que las longitudes de corrida de los motores duran del orden de segundos [18], consideraremos una distribución estable de moléculas de tau en el filamento. Esto significa que si un dado sitio tiene ligada una molécula de tau, permanecerá ligada a él para todo tiempo, y si un sitio no tiene tau ligada, permanecerá sin tau para todo tiempo.

Avancemos en la escritura de las ecuaciones maestras del sistema para ir agregando elementos que nos permitan describir el modelo. Y en particular para ver como aparecen los parámetros p_a y p_m que caracterizan el efecto de tau.

5.2.1 Ecuaciones maestras

Denotemos por $r_i(t)$ y $q_i(t)$ a las probabilidades de encontrar un motor KIF1A en el sitio de red i a tiempo t en el estado 1 y 2, respectivamente. La probabilidad de que el sitio i esté desocupado a tiempo t será: $1 - r_i(t) - q_i(t)$.

Las ecuaciones maestras para la dinámica de los motores que se encuentren en un sitio del *bulk* vienen dadas por:

$$\begin{aligned} \frac{dr_i}{dt} = & (1 - \tau_a(i)) \omega_a (1 - r_i - q_i) - (\omega_d + \omega_h) r_i + \omega_s q_i \\ & + (1 - \tau_m(i)) \omega_f q_{i-1} (1 - r_i - q_i), \end{aligned} \quad (59)$$

$$\begin{aligned} \frac{dq_i}{dt} = & \omega_h r_i - \omega_s q_i - \omega_f q_i (1 - r_{i+1} - q_{i+1}) \\ & - \omega_b q_i (2 - r_{i-1} - q_{i-1} - r_{i+1} - q_{i+1}) \\ & + (1 - \tau_m(i)) \omega_b (q_{i-1} + q_{i+1}) (1 - r_i - q_i), \end{aligned} \quad (60)$$

donde

$$\tau_a(i) = \begin{cases} p_a & \text{si hay una molécula de tau en el sitio } i \\ 0 & \text{si no hay} \end{cases}, \quad (61)$$

y

$$\tau_m(i) = \begin{cases} p_m & \text{si hay una molécula de tau en el sitio } i \\ 0 & \text{si no hay} \end{cases}. \quad (62)$$

Como ya se mencionó, p_a es la probabilidad de que tau impida un intento de ligadura al **MT** de una kinesina, y p_m es la probabilidad de que tau obligue a una kinesina que se está moviendo a lo largo del **MT** a desligarse. Los parámetros intermedios τ_a y τ_m aparecen para diferenciar los sitios en que los que hay tau de los sitios en los que no hay. Así, una dada molécula de tau ligada en un dado sitio del **MT** puede afectar la ligadura de la kinesina sólo en ese sitio y, de igual modo, sólo puede obligar a desligarse a una kinesina que se mueve atravesando el sitio en que la molécula de tau se encuentra ligada.

El resto de los parámetros son equivalentes a los del modelo descrito en la Sección 4.2. En la próxima sección volveremos a discutir el significado de cada uno de ellos.

Las ecuaciones maestras para el extremo (-) son:

$$\frac{dr_1}{dt} = (1 - \tau_a(1)) \alpha (1 - r_1 - q_1) - (\gamma_1 + \omega_h)r_1 + \omega_s q_1, \quad (63)$$

$$\begin{aligned} \frac{dq_1}{dt} = & \omega_h r_1 - \omega_s q_1 - \omega_f q_1 (1 - r_2 - q_2) - \omega_b q_1 (1 - r_2 - q_2) \\ & - \gamma_2 q_1 + (1 - \tau_m(1)) \omega_b q_2 (1 - r_1 - q_1). \end{aligned} \quad (64)$$

Y las ecuaciones maestras para el extremo (+), son:

$$\begin{aligned} \frac{dr_L}{dt} = & (1 - \tau_a(L)) \delta (1 - r_L - q_L) - (\beta_1 + \omega_h)r_L + \omega_s q_L \\ & + (1 - \tau_m(L)) \omega_f q_{L-1} (1 - r_L - q_L), \end{aligned} \quad (65)$$

$$\begin{aligned} \frac{dq_L}{dt} = & \omega_h r_L - \omega_s q_L - \beta_2 q_L - \omega_b q_L (1 - r_{L-1} - q_{L-1}) \\ & + (1 - \tau_m(L)) \omega_b q_{L-1} (1 - r_L - q_L). \end{aligned} \quad (66)$$

Los parámetros para los extremos son iguales que para los sitios del *bulk*, excepto que tomamos α y δ en lugar de ω_a como las tasas de ligadura en los extremos (-) y (+), respectivamente. Y además, γ_1 y β_1 en lugar de ω_d son las tasas de desligadura, y γ_2 y β_2 en lugar de ω_b son las tasas de salida de los motores debido a movimiento browniano en los extremos (-) y (+), respectivamente.

5.2.2 Simulación

La evolución dinámica del sistema se describe a partir de tasas de transición que reflejan la naturaleza estocástica del movimiento de las proteínas motoras y de su interacción con las moléculas de tau. Esta evolución queda determinada por las ecuaciones maestras que se especificaron en la sección anterior, aquí iremos explicando las distintas transiciones posibles entre los estados a la vez que discutimos cómo se realizó la simulación.

Para implementar el modelo realizamos simulaciones Monte Carlo que actualizan el estado de todos los sitios del filamento siguiendo un orden secuencial aleatorio. En un MCS le damos a todos los sitios, en orden aleatorio, la oportunidad de cambiar su estado de ocupación al igual que hacíamos en el caso sin tau.

Ahora el cambio de estado de un dado sitio estará regulado por tasas de transición que dependen del estado de ocupación del sitio, del estado de ocupación de sus primeros vecinos y también de la presencia (o no) de tau en los sitios en consideración.

La Figura 26 muestra un esquema con las distintas posibilidades de actualización para un sitio perteneciente al *bulk*. Este esquema permite visualizar la dinámica del modelo definido a partir de las Ecuaciones (59) y (60). En la representación se muestra cada estado interno como una red diferente, del mismo modo que se hizo en el Capítulo 4. Es importante recordar que los estados internos corresponden a las dos formas en que el motor se puede relacionar con el MT, pero el filamento real es uno sólo.

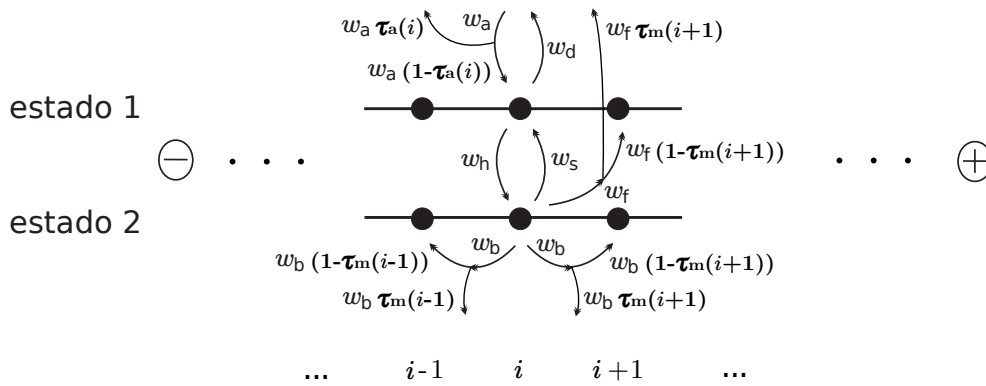


Figura 26: Esquema del modelo mostrando las posibilidades de actualización para un sitio del bulk, con las tasas de probabilidad asociadas. El efecto de tau, evitando que un motor se ligue al filamento o forzando que un motor que se está moviendo a lo largo del filamento se desligue, es representado como una flecha que se divide.

Consideremos el sitio i -ésimo: si el sitio está vacío, el único cambio de estado posible es que un motor se ligue en estado 1, esto se produce con una tasa $\omega_a(1 - \tau_a(i))$.

La tasa de ligadura ω_a es proporcional a la concentración de motores KIF1A y es la misma tasa de ligadura que consideramos en el modelo sin tau. Pero ahora, para que la ligadura se complete de forma exitosa deberemos considerar también la posible interacción con una molécula de tau que se encuentre ligada en ese sitio. En efecto, $(1 - \tau_a(i))$ es la probabilidad de que tau no impida la ligadura. Si no hay molécula de tau ligada al sitio i -ésimo, entonces $\tau_a(i) = 0$ y la ligadura no se impide nunca. Si hay una molécula de tau ligada al sitio i -ésimo, entonces $\tau_a(i) = p_a$ es la probabilidad de que tau impida la ligadura del motor. Por lo tanto, $(1 - \tau_a(i)) = (1 - p_a)$ será la probabilidad de que no la impida.

Si el sitio i -ésimo está ocupado por un motor en estado 1, dos cosas pueden suceder. En primer lugar, el motor puede desligarse del sitio, con una tasa ω_d , y en segundo lugar, puede hidrolizar una molécula de ATP y permanecer en el sitio (ahora en estado 2), con una tasa ω_h . En estas transiciones tau no juega ningún rol directo.

Por último, si el sitio está ocupado con un motor en estado 2 varias posibilidades aparecen. El motor puede difundir a uno de sus sitios vecinos, bien sea a la izquierda con una tasa $\omega_b(1 - \tau_m(i - 1))$ ó a la derecha con una tasa $\omega_b(1 - \tau_m(i + 1))$ permaneciendo en cualquiera de los dos casos en el estado 2 (en el sitio destino). Puede también liberar el ADP y avanzar activamente al sitio siguiente con una tasa $\omega_f(1 - \tau_m(i + 1))$, ligándose al sitio siguiente en estado 1. Finalmente puede liberar el ADP pero permanecer en el mismo sitio con una tasa ω_s , cambiando al estado 1.

Notar que para que cualquiera de los movimientos a lo largo del filamento se concrete exitosamente, la interacción con tau no debe forzar al motor a desligarse. Eso se tiene en cuenta a partir del factor $(1 - \tau_m(j))$ que da la probabilidad de que una molécula de tau en el sitio j -ésimo no obligue al motor, que intenta moverse a ese sitio, a desligarse. Si no hay molécula de tau en el sitio j , entonces $\tau_m(j) = 0$ y los movimientos se dan con las tasas habituales (las que teníamos en el caso sin tau), pero si en el sitio j hay una molécula de tau ligada, $\tau_m(j) = p_m$ y la probabilidad de que el movimiento del motor se concrete exitosamente (es decir la probabilidad de que tau no obligue al motor a desligarse) es $(1 - \tau_m(j)) = (1 - p_m)$.

Vale la pena enfatizar que, igual que en el caso sin tau, cualquier cambio de sitio del motor está permitido sólo si el sitio destino no está ocupado por otro motor.

En la Figura 27 se puede ver el esquema con las posibilidades de actualización del extremo (-), y en la Figura 28, el esquema para el extremo (+).

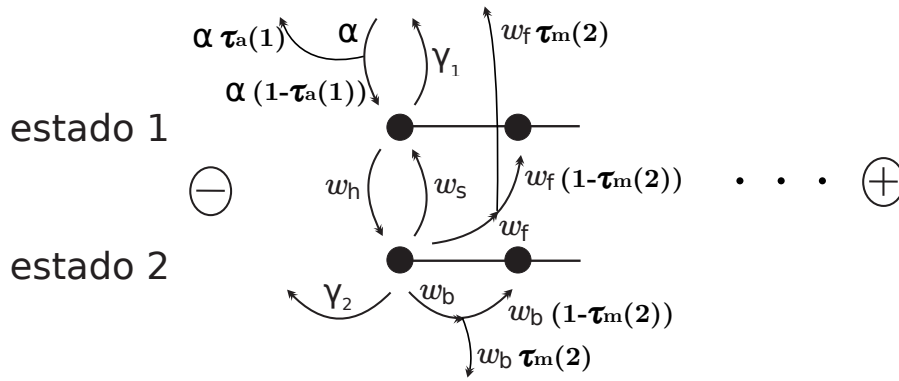


Figura 27: Esquema del modelo mostrando las posibilidades de actualización para el extremo (-), con las tasas de probabilidad asociadas. El efecto de tau, evitando que un motor se ligue al filamento o forzando un que un motor que se está moviendo a lo largo del filamento se desligue, es representado como una flecha que se divide.

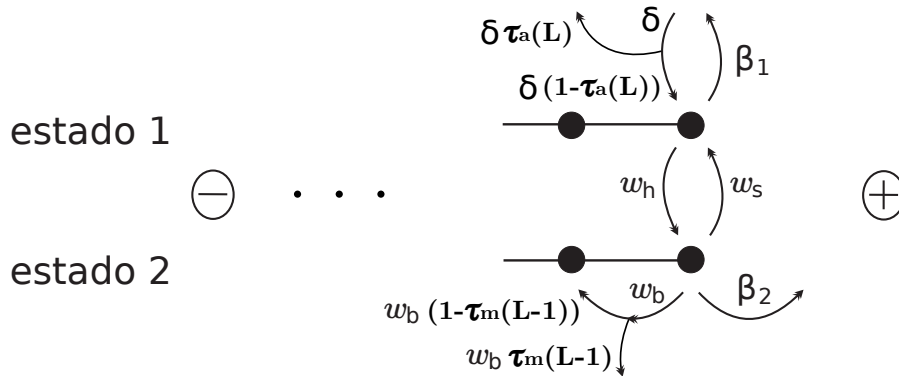


Figura 28: Esquema del modelo mostrando las posibilidades de actualización para el extremo (+), con las tasas de probabilidad asociadas. El efecto de tau, evitando que un motor se ligue al filamento o forzando un que un motor que se está moviendo a lo largo del filamento se desligue, es representado como una flecha que se divide.

Para la dinámica en los extremos, como ya se mencionó, tomamos α y δ en lugar de ω_a como la tasa de ligadura en el extremo (-) y (+), respectivamente. De la misma manera, γ_1 y β_1 (en lugar de ω_d) son las tasas de desligadura, y γ_2 y β_2 (en lugar de ω_b) son las tasas a las cuales un motor sale del filamento por movimiento browniano en el extremo (-) y (+), respectivamente. El efecto de tau se interpreta de la misma manera que la mencionada para un sitio del *bulk*, con los cambios de nombre de estos parámetros.

5.2.2.1 *Detalles de la implementación*

De igual modo que en la simulación para el modelo sin tau, en un MCS actualizamos el estado del filamento completo siguiendo un orden de actualización secuencial aleatorio, esto es, dando a todos los sitios, en orden aleatorio, la oportunidad de cambiar su estado de ocupación.

Para todas las combinaciones de parámetros con las que trabajamos y que se detallan más adelante, corremos una simulación que tiene una concentración fija de moléculas de tau y comienza sin motores ligados al filamento y dejamos que el sistema evolucione hasta alcanzar el estado estacionario.

Al comenzar la simulación, construimos la distribución de moléculas de tau en el filamento. Para obtener la concentración deseada, le damos a cada sitio del filamento la oportunidad de tener una molécula de tau con probabilidad C_{tau} , de esta manera tendremos en todo el filamento $\sim L C_{\text{tau}}$ moléculas de tau, tal como queremos. La disposición de las moléculas de tau permanece fija en el curso de toda la simulación.

El intervalo de tiempo de un MCS se elige igual a 0,71ms que es un intervalo de tiempo apropiado con los criterios establecidos en la Sección 4.2.2.1.

En el Apéndice E.1 puede verse el código fuente completo de la simulación.

5.2.3 *Límite de alta densidad*

Es posible obtener una solución analítica aproximada en el estado estacionario (para las Ecuaciones (59) y (60) que definen el modelo) en el límite de alta densidad de motores. Es decir en el límite de ω_a alto, que corresponde a una región embotellada cubriendo todo el filamento².

Esta aproximación analítica resulta útil principalmente para comparar con las simulaciones en dicho límite.

El límite de alta densidad de motores permite buscar soluciones en las que todos los sitios del filamento son equivalentes, debido justamente al hecho de que todo el filamento se encuentra embotellado. Así, si se toman condiciones de contorno periódicas, las soluciones $(r_i, q_i) \equiv (r, q)$ de las ecuaciones maestras en el estado estacionario

² La ambigüedad con la que nos referimos al valor de ω_a como “alto” es en algún sentido inevitable a fines de ser concisos y está bien definida en la práctica. Lo que queremos decir es que la concentración de motores toma un valor tan grande que el filamento está completamente ocupado por motores. Eso, como ya vimos en la Sección 4.3.2, depende también del valor de ω_h y es, de algún modo, la raíz de la ambigüedad lingüística.

rio, con la suposición adicional de que $\tau_a(i) = p_a C_{\text{tau}} \equiv \tilde{\tau}_a$ y $\tau_m(i) = p_m C_{\text{tau}} \equiv \tilde{\tau}_m$, son:

$$r = \frac{1 + (K_s(1 - \Omega_f) - 1) q}{1 + K_d + K_h(1 - \Omega_f)}, \quad (67)$$

$$q = \frac{-A - \sqrt{A^2 - 4\Omega_h(K_d + (K_h + K_s)(1 - \Omega_f))}}{2(K_d + (K_h + K_s)(1 - \Omega_f))}, \quad (68)$$

en donde:

$$\begin{aligned} A &= K_h(\Omega_f - 1) - K_d(\Omega_s + 1) - (\Omega_s + \Omega_h), \\ K_s &= \frac{\omega_s}{(1 - \tilde{\tau}_a)\omega_a}, \\ K_h &= \frac{\omega_h}{(1 - \tilde{\tau}_a)\omega_a}, \\ K_d &= \frac{\omega_s}{(1 - \tilde{\tau}_a)\omega_a}, \\ \Omega_s &= \frac{\omega_s}{\omega_f + 2\omega_b\tilde{\tau}_m}, \\ \Omega_h &= \frac{\omega_h}{\omega_f + 2\omega_b\tilde{\tau}_m}, \\ \Omega_f &= \frac{(1 - \tilde{\tau}_m)\omega_f}{\omega_f + 2\omega_b\tilde{\tau}_m}. \end{aligned}$$

Si bien un sistema con condiciones de contorno periódicas no es realista, provee una buena aproximación en el límite de alta concentración de motores para el caso equivalente (con los mismos parámetros) pero con condiciones de contorno abiertas que se puede obtener a partir de la simulación. La Figura 29 muestra una comparación entre los resultados de la aproximación analítica y las simulaciones numéricas para tres casos correspondientes a diferentes magnitudes de la interacción tau-kinesina. Los parámetros utilizados se detallan en la Tabla 8.

ω_a [$\frac{1}{\text{ms}}$]	ω_d [$\frac{1}{\text{ms}}$]	ω_h [$\frac{1}{\text{ms}}$]	ω_s [$\frac{1}{\text{ms}}$]	ω_f [$\frac{1}{\text{ms}}$]	ω_b [$\frac{1}{\text{ms}}$]
0.001	0.0001	0.01	0.145	0.055	0.6

Tabla 8: Parámetros del modelo para el caso de la Figura 29. El número de sitios de la red es $L = 600$, la concentración de tau es $C_{\text{tau}} = 0,01$ y los parámetros correspondientes a los extremos se utilizan de la siguiente forma: $\beta_1 = \beta_2 = \omega_d$, $\gamma_1 = \gamma_2 = \delta = 0$ y $\alpha = \omega_a$.

5.2.4 Reducción al caso sin tau

Para considerar el límite en el cual no hay moléculas de tau presentes en el citosol en las expresiones obtenidas en la sección anterior (Ecuaciones (67) y (68)) debe tomarse $\tilde{\tau}_a = 0$ y $\tilde{\tau}_m = 0$ ³.

³ Esto surge de tomar $C_{\text{tau}} = 0$.

En dicho límite, las expresiones de las Ecuaciones (67) y (68) se reducen a las que encontraron Nishinari y colaboradores para el régimen de alta densidad en el modelo para la dinámica de KIF1A sin moléculas de tau (ver Referencia [47]).

Más aún, cuando no hay moléculas de tau presentes en el citosol las ecuaciones maestras del modelo que propusimos se reducen a las ecuaciones maestras del modelo de KIF1A sin tau (ver Sección 4.2.1).

5.3 TIEMPOS DE LIGADURA Y LONGITUDES DE CORRIDA

En esta sección presentamos algunos de los resultados que pueden obtenerse con el modelo propuesto. Principalmente nos centramos en la discusión de cantidad dinámicas relevantes para caracterizar el efecto de tau en los motores KIF1A. Los observables que discutiremos serán el tiempo promedio que los motores están ligados al MT y la longitud de corrida.

Para los casos mostrados en las Figuras 30 y 31 los parámetros del modelo fueron tomados como se muestra en la Tabla 9.

$\omega_a \left[\frac{1}{\text{ms}} \right]$	$\omega_d \left[\frac{1}{\text{ms}} \right]$	$\omega_s \left[\frac{1}{\text{ms}} \right]$	$\omega_f \left[\frac{1}{\text{ms}} \right]$	$\omega_b \left[\frac{1}{\text{ms}} \right]$
0.00001	0.001	0.145	0.055	0.6

Tabla 9: Parámetros del modelo para los casos de las Figuras 30 y 31. El número de sitios de la red es $L = 600$ y los parámetros correspondientes a los extremos se utilizan de la siguiente forma: $\beta_1 = \beta_2 = \omega_d$, $\gamma_1 = \gamma_2 = \delta = 0$ y $\alpha = \omega_a$. Los parámetros que caracterizan la interacción con las moléculas de tau se toman: $p_a = 0,67$ y $p_m = 0,5$.

Los observables que se muestran se obtuvieron promediando los resultados de 1000 simulaciones con exactamente los mismos parámetros, pero con distinta semilla para el generador de números aleatorios. El error asociado a los observables corresponde a la desviación estándar.

Cada una de las simulaciones genera una nueva distribución de moléculas de tau (con la misma concentración) y se deja evolucionar por 2×10^6 MCSs luego de una termalización de 2×10^5 MCSs que garantiza que el sistema alcanzó el estado estacionario.

La Figura 30 muestra el comportamiento del tiempo de ligadura como función de ω_h . Vale la pena recordar que ω_h es una función monótona creciente de la concentración de ATP ($\omega_h = 0,25 \text{ms}^{-1}$ corresponde a una concentración saturante de ATP) [47].

En una dada simulación el tiempo de ligadura es calculado promediando los intervalos entre ligadura y desligadura de cada KIF1A. Eso da lugar a un valor t_i para cada simulación. El valor que se

muestra en la figura corresponde al promedio de estos valores t_i en las 1000 simulaciones⁴.

Para $C_{\text{tau}} = 0$ no hay moléculas de tau presentes en el citosol (en este caso, los valores de p_a y p_m son irrelevantes) y el tiempo de ligadura crece con los valores de ω_h , desde cerca de 15s para baja concentración de ATP hasta casi 25s para condiciones saturantes de ATP.

Cuando no hay moléculas de tau presentes en el MT, los motores moleculares pueden desligarse del filamento sólo cuando están en el estado 1. Así, conforme ω_h aumenta, la concentración de ATP aumenta y el tiempo que el motor pasa en el estado 2 se hace más grande (si hay suficiente ATP disponible, el motor hidroliza alguna de las moléculas y cambia al estado 2 casi inmediatamente). De esta manera, la duración temporal de las corridas de los motores en el filamento también aumenta.

Por otro lado, en presencia de moléculas de tau, los motores KIF1A pueden ser forzados a desligarse aún si están en el estado 2. Cuando una proteína motora que se está moviendo (ya sea avanzando activamente o moviéndose difusivamente hacia cualquiera de los lados) se encuentra con una molécula de tau en su sitio destino, tiene una probabilidad p_m de desligarse del filamento debido a su interacción con la molécula de tau. De este modo, la presencia de tau acorta los tiempos de ligadura promedio de los motores.

Mientras más grandes son los valores de concentración de tau mayor es el número de encuentros KIF1A-tau y más significativo será el efecto de tau en el valor del tiempo de ligadura promedio.

Esto puede observarse en la Figura 30, en la cuál los valores del tiempo de ligadura promedio están ordenados de arriba a abajo de forma creciente en los valores de la concentración de tau.

Los resultados para $C_{\text{tau}} = 0,005$, en la cual la concentración de tau es muy baja son similares a los obtenidos en el caso sin tau, esto es, el tiempo de ligadura promedio aumenta con ω_h . Para $C_{\text{tau}} = 0,01$ y $C_{\text{tau}} = 0,1$, en cambio, la concentración de tau es lo suficientemente grande para cancelar el efecto de una alta concentración de ATP y el tiempo de ligadura promedio permanece esencialmente constante para todos los valores de ω_h . Más aún, los valores para $C_{\text{tau}} = 0,1$ presentan una tendencia ligeramente decreciente conforme ω_h aumenta (tiempo de ligadura promedio disminuye aproximadamente de 0,22s a 0,13s) lo cual se podría asociar a un comportamiento regulado por tau. Una concentración de ATP alta acelera la dinámica de los motores favoreciendo los encuentros con moléculas de tau lo que aumenta las chances de desligadura mediada por tau.

El efecto de tau en la dinámica de KIF1A considerando la *longitud de corrida promedio* puede establecerse en términos similares a los mencionados en relación al tiempo de ligadura.

⁴ Por lo tanto es un promedio de promedios.

La longitud de corrida en cada simulación se calcula promediando la distancia que cada motor recorre a lo largo del filamento antes de desligarse, esto da lugar a un valor l_i que a su vez se promedia en las 1000 simulaciones.

En la Figura 31 se muestran los resultados de la longitud de corrida promedio en función de ω_h para distintas concentraciones de tau. En primer lugar, podemos ver que en ausencia de tau la longitud de corrida promedio aumenta con ω_h .

Las moléculas de ATP son el combustible que los motores KIF1A utilizan para moverse activamente hacia el extremo (+) del MT, por lo tanto una concentración mayor de ATP acelera la dinámica de los motores, extendiendo las distancias que los motores recorren antes de desligarse del MT.

En presencia de moléculas de tau, el movimiento de KIF1A a lo largo del MT da lugar a encuentros que favorecen su desligadura, por lo tanto al aumentar la concentración de ATP se da una competencia entre dos efectos. Por un lado, como ya mencionamos, la disponibilidad de moléculas de ATP favorece longitudes de corrida más largas, pero por otro lado, también favorece un mayor número de encuentros kinesina-tau que propician recorridos más cortos.

Así, para $C_{\text{tau}} = 0,005$ y $C_{\text{tau}} = 0,01$ las longitudes de corrida son funciones crecientes de ω_h pero para $C_{\text{tau}} = 0,1$ ésta permanece esencialmente constante en todo el rango de ω_h . En este caso, la concentración de tau es lo suficientemente grande como para compensar la mayor disponibilidad de combustible (ATP) con un mayor número de desligaduras de motores.

Para un mismo valor de ω_h (concentración de ATP) la longitud de corrida es siempre más pequeña para valores más grandes de C_{tau} , es decir, mientras más moléculas de tau hay en el filamento menor es la distancia que los motores pueden viajar a lo largo de éste.

La Figura 32 muestra el efecto que tiene tau en la distribución de longitudes de corrida de KIF1A, para diferentes concentraciones. Los parámetros utilizados se detallan en la Tabla 10. Conforme la concentración de tau aumenta las longitudes de corrida disminuyen. En los histogramas eso queda claro a partir del hecho que el caso sin tau corresponde a una distribución con "cola" más larga, mientras que las longitudes menores van teniendo un peso cada vez mayor conforme la concentración de tau aumenta. Resulta pertinente destacar que estos resultados concuerdan bien con las observaciones experimentales de Dixit y colaboradores publicados en la Referencia [18].

5.4 LOCALIZACIÓN DE PAREDES DE DOMINIO EN PRESENCIA DE TAU

En esta sección se describen los resultados de aplicar el método de localización de paredes de dominio desarrollado (ver Sección 4.3) para

$\omega_a \left[\frac{1}{\text{ms}} \right]$	$\omega_d \left[\frac{1}{\text{ms}} \right]$	$\omega_h \left[\frac{1}{\text{ms}} \right]$	$\omega_s \left[\frac{1}{\text{ms}} \right]$	$\omega_f \left[\frac{1}{\text{ms}} \right]$	$\omega_b \left[\frac{1}{\text{ms}} \right]$
0.00001	0.0001	0.2	0.145	0.055	0.6

Tabla 10: Parámetros del modelo para el caso de la Figura 32. El número de sitios de la red es $L = 600$ y los parámetros correspondientes a los extremos se utilizan de la siguiente forma: $\beta_1 = \beta_2 = \omega_d$, $\gamma_1 = \gamma_2 = \delta = 0$ y $\alpha = \omega_a$. Los parámetros que caracterizan la interacción con las moléculas de tau se toman: $p_a = 0,67$ y $p_m = 0,5$. Número total de MCSs = 2×10^6 , después de una termalización de 2×10^5 MCSs.

estudiar el efecto que tiene la proteína tau en la formación estas paredes de dominio. Utilizamos un umbral de segmentación $s_0 = 0,999$, al igual que en el Capítulo 4.

Se analiza un caso particular, cuya combinación de parámetros se puede ver en la Tabla 10 y se discuten los resultados para varias concentraciones de tau en el filamento. Se comienza mostrando los resultados para el caso sin tau y después se va subiendo progresivamente la concentración de tau.

CASO $C_{\text{tau}} = 0$: en la Figura 33 se puede ver el resultado de analizar el estado estacionario. Se encontró que el valor máximo es $\langle D_{JS_{\text{max}}} \rangle = 0,2830$, en la posición $x_{\text{max}} = 541$. La significancia máxima toma el valor $s_{\text{max}} (N = 600, \langle D_{JS_{\text{max}}} \rangle = 0,2830) = 1 > 0,999$, por lo tanto este caso presenta una pared de dominio.

Los resultados para la densidad de motores en estado 1, ρ_1 , para la densidad de motores en estado 2, ρ_2 , y para la densidad total de motores, ρ_T , se muestran en la Figura 34⁵.

CASO $C_{\text{tau}} = 0,005$: en la Figura 35 se puede ver el resultado de analizar el estado estacionario. Se encontró que el valor máximo es $\langle D_{JS_{\text{max}}} \rangle = 0,0835$, en la posición $x_{\text{max}} = 570$. La significancia máxima toma el valor $s_{\text{max}} (N = 600, \langle D_{JS_{\text{max}}} \rangle = 0,0835) = 1 > 0,999$, por lo tanto hay una pared de dominio.

Los perfiles de densidad ρ_1 (en estado 1), ρ_2 (en estado 2) y ρ_T (total) se muestran en la Figura 36.

CASO $C_{\text{tau}} = 0,01$: en la Figura 37 se puede ver el resultado de analizar el estado estacionario. Se encontró que el valor máximo es $\langle D_{JS_{\text{max}}} \rangle = 0,0294$, en la posición $x_{\text{max}} = 567$. La significancia máxima toma el valor $s_{\text{max}} (N = 600, \langle D_{JS_{\text{max}}} \rangle = 0,0294) =$

⁵ Las perfiles de densidad se construyen, una vez alcanzado el estado estacionario, contando en cuántos pasos cada sitio se encontró en cada estado. Así, si de N pasos, el sitio i -ésimo se encontró n_1 veces en el estado 1, n_2 veces en el estado 2 y $n_0 = N - n_1 - n_2$ veces en estado 0, tendremos que $\rho_1(i) = \frac{n_1}{N}$, $\rho_2(i) = \frac{n_2}{N}$ y $\rho_T(i) = \rho_1(i) + \rho_2(i)$.

$0,99989 > 0,999$, por lo tanto este caso también presenta una pared de dominio.

Los resultados para la densidad de motores en estado 1, ρ_1 , estado 2, ρ_2 , y total, ρ_T , se muestran en la Figura 38.

CASO $C_{\text{tau}} = 0,1$: en la Figura 39 se puede ver el resultado de analizar el estado estacionario. Se encontró que el valor máximo es $\langle D_{JS_{\text{max}}} \rangle = 0,0017$, en la posición $x_{\text{max}} = 296$. La significancia máxima toma el valor $s_{\text{max}}(N = 600, \langle D_{JS_{\text{max}}} \rangle = 0,0017) = 0,02357 < 0,999$, por lo tanto no hay pared de dominio.

Los perfiles de densidad en estado 1, ρ_1 , en estado 2, ρ_2 , y la densidad total, ρ_T , se muestran en la Figura 40.

Los resultados muestran que tau juega un rol importante en la posición en que se produce la pared de dominio. Para esta combinación de parámetros y en ausencia de tau, la región embotellada se extiende hasta la posición 541. La presencia de tau encoge la región embotellada, cuando la concentración de tau en el filamento es de $C_{\text{tau}} = 0,005$ la pared de dominio se encuentra en la posición 570, y para $C_{\text{tau}} = 0,01$, en la posición 567. Para una concentración de $C_{\text{tau}} = 0,1$ la pared de dominio directamente desaparece.

Como el caso aquí estudiado tiene los mismos parámetros que los de la Figura 22(a) del caso sin tau, podemos estimar el error de la localización del *shock* de acuerdo a lo detallado en la Tabla 5, por lo tanto corresponde un error de 12 sitios de red. Con esto en mente podemos decir que las posiciones de la pared de dominio en los casos $C_{\text{tau}} = 0,005$ y $C_{\text{tau}} = 0,01$ son indistinguibles entre sí, pero son distinguibles de los otros dos casos. Y la tendencia de disminución de la región embotellada se hace más clara.

El caso de $C_{\text{tau}} = 0,1$ merece una mención especial: para la combinación de parámetros estudiada, esta concentración de tau es tan alta que casi todos los motores KIF1A son obligados a desligarse del filamento, como se puede ver en el gráfico espacial de la Figura 39 y también en los perfiles de densidad de la Figura 40. Por lo tanto, si bien no se detecta un *shock*, el transporte intracelular también se ve interrumpido.

Estos resultados sugieren que tau puede jugar un rol clave en la formación de paredes de dominio y que una concentración apropiada de moléculas de tau puede favorecer un transporte intracelular adecuado.

5.5 CONCLUSIONES DEL CAPÍTULO

En este capítulo presentamos un modelo estocástico para el transporte intracelular de las proteínas motoras KIF1A que incluye su interacción con la MAP tau.

Nuestro modelo es una extensión del modelo de Nishinari y colaboradores para el tráfico de KIF1A en ausencia de tau e intenta preservar una de sus principales virtudes, la de relacionar las propiedades de transporte con cantidades controlables experimentalmente.

El modelo tiene en cuenta los efectos de tau en los motores KIF1A, a partir de dos parámetros p_a y p_m . El efecto que una molécula de tau tiene evitando que una kinesina se ligue al MT se caracteriza con p_a , y el efecto que tiene obligando a una kinesina que se está moviendo a desligarse cuando se da un encuentro, se caracteriza con p_m .

Los valores de estos parámetros han sido fijados basándonos en resultados experimentales obtenidos por Dixit *et al.* [18]. Por otro lado, hemos probado que los resultados obtenidos para los observables son robustos desde el punto de vista de la simulación. Esto es, pequeños cambios en los valores de estos parámetros no modifican el comportamiento cualitativo de los observables discutidos. Más aún, cambios en órdenes de magnitud en los parámetros p_a y p_m , cambian el valor de los observables en el mismo orden de magnitud.

Nuestro modelo coincide exactamente con el modelo de Nishinari *et al.* [47] y Greulich *et al.* [22] cuando no hay moléculas de tau presentes en el filamento. Lo mismo se verifica para los cálculos aproximados en el límite de densidad alta de motores. Las soluciones analíticas aproximadas obtenidas para nuestro modelo, se reducen a las encontradas por Nishinari *et al.* [47] cuando no hay moléculas de tau en el filamento.

La procesividad de los motores molecular, que es un aspecto crucial para el transporte intracelular, puede definirse de varias formas [54], dos de ellas son el tiempo de vida de ligadura de las proteínas motoras (tiempo de ligadura) y la longitud media recorrida por un motor en una única travesía a lo largo del filamento (longitud de corrida). Encontramos efectos significativos debidos a la presencia de tau tanto en el tiempo de ligadura como en la longitud de corrida de los motores KIF1A.

También se observaron efectos significativos de tau en la formación de paredes de dominio, desplazándose su posición hacia el extremo (+) con concentración de tau creciente, esto es, disminuyendo la región embotellada.

Los resultados, en conjunto, sugieren que tau podría jugar un papel importante en la regulación del transporte intracelular debido a las kinesinas. Más aún, la regulación local de la concentración de tau en el citosol podría constituir un mecanismo para evitar la formación de embotellamientos. Esto podría darse, por ejemplo, con una concentración baja de tau cerca del extremo (-) y una concentración alta cerca del extremo (+).

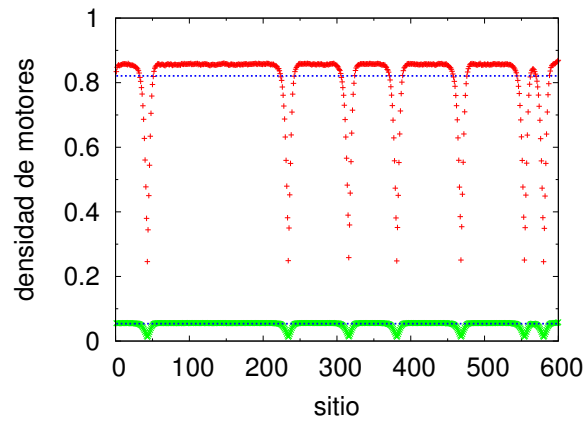
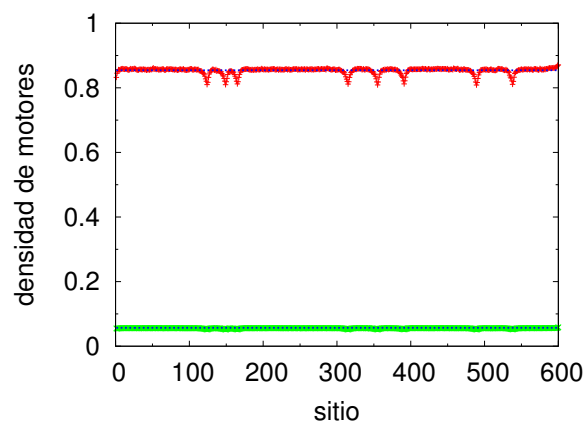
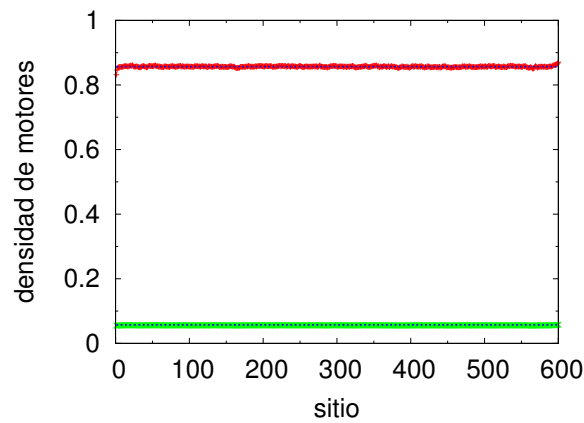
(a) $p_\alpha = 0,67$, $p_m = 0,5$.(b) $p_\alpha = 0,067$, $p_m = 0,05$.(c) $p_\alpha = 0,0067$, $p_m = 0,005$.

Figura 29: Perfiles de densidad en el estado estacionario. Los parámetros del modelo para este caso se explicitan en la Tabla 8. El valor elegido para ω_α permite la realización del límite de alta densidad de motores. Comparación de las simulaciones y la aproximación analítica. En cada uno de los casos, los puntos rojos de arriba (verdes de abajo) corresponden a los resultados de las simulaciones para la densidad de motores en el estado 1 (estado 2). Las líneas punteadas son las predicciones de la aproximación analítica para sistemas con condiciones de contorno periódicas pero con los mismos parámetros. Ver el texto principal para más detalles.

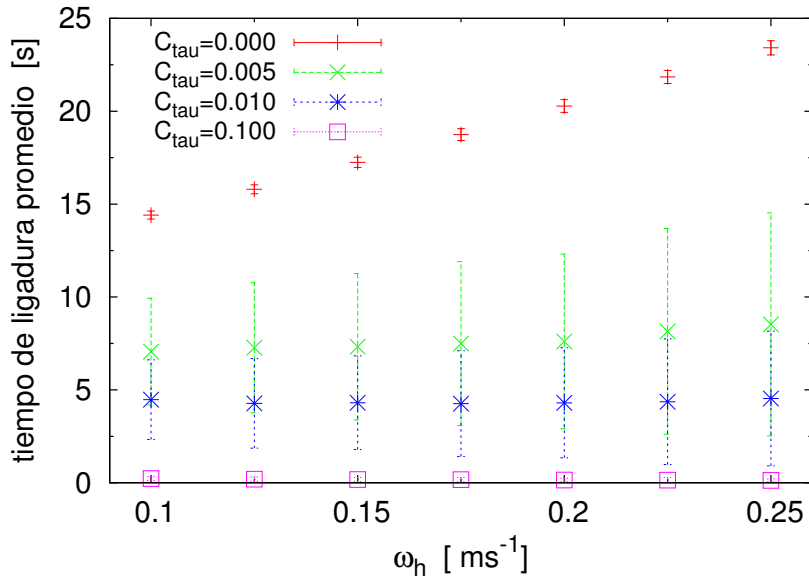


Figura 30: Tiempo promedio entre ligadura y desligadura como función de la tasa de hidrólisis para diferentes concentraciones de tau. Los parámetros del modelo para este caso se explicitan en la Tabla 9.

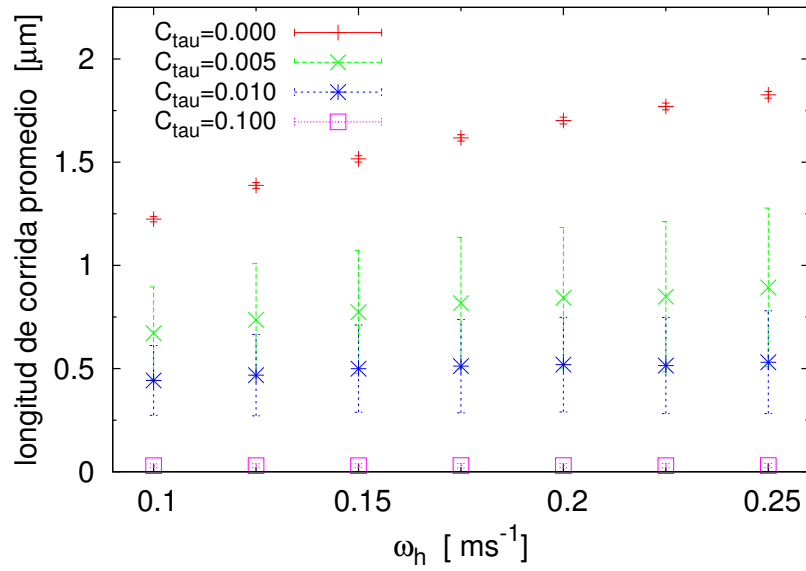


Figura 31: Longitud de corrida promedio que recorre un motor KIF1A antes de desligarse como función de la tasa de hidrólisis para diferentes concentraciones de tau. Los parámetros del modelo para este caso se explicitan en la Tabla 9.

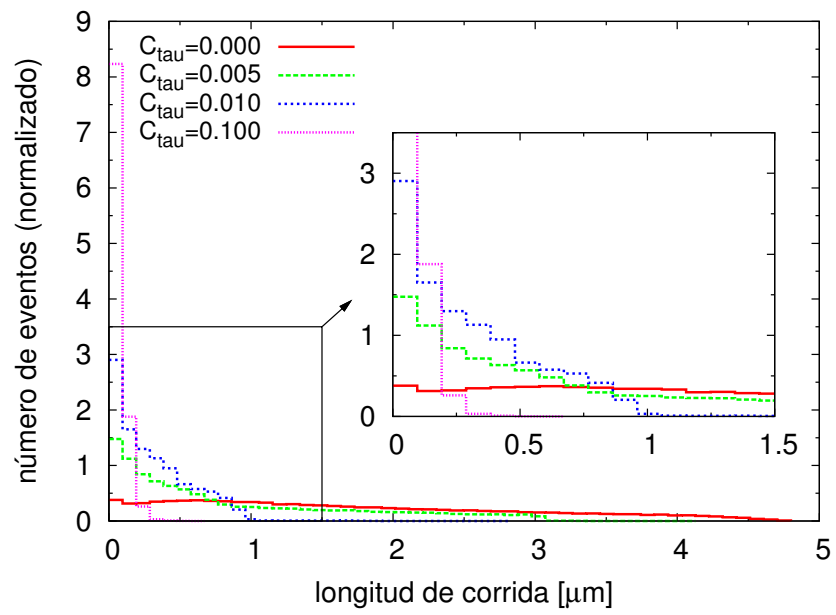


Figura 32: Histogramas que muestran el número de eventos (normalizado) para la longitud de corrida de KIF1A para diferentes concentraciones de tau. Los parámetros del modelo para este caso se explicitan en la Tabla 10.

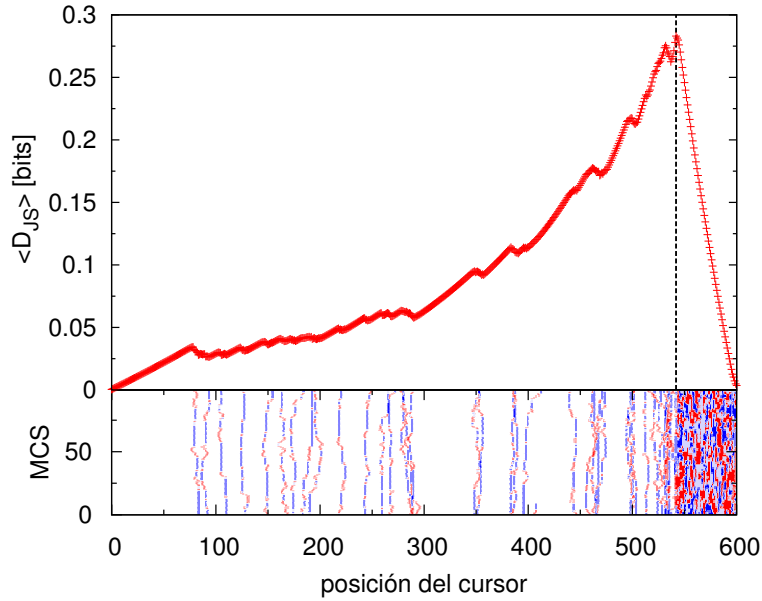


Figura 33: Valores de la *DJS* promedio para cada posición del cursor correspondientes a un caso con $C_{tau} = 0$ y el resto de los parámetros del modelo que se explicitan en la Tabla 10. La línea de puntos vertical corresponde a la posición 541 en la cual se detectó un *shock*. En la parte inferior de la figura se puede ver un gráfico espacial en el que cada fila representa el estado del filamento en un dado instante temporal. Se muestran cien MCSs una vez que el sistema alcanzó el estado estacionario. Un punto blanco representa un sitio vacío para ese tiempo, un punto azul, un sitio ocupado con un motor en estado 1 y un punto rojo, un sitio ocupado con un motor en estado 2.

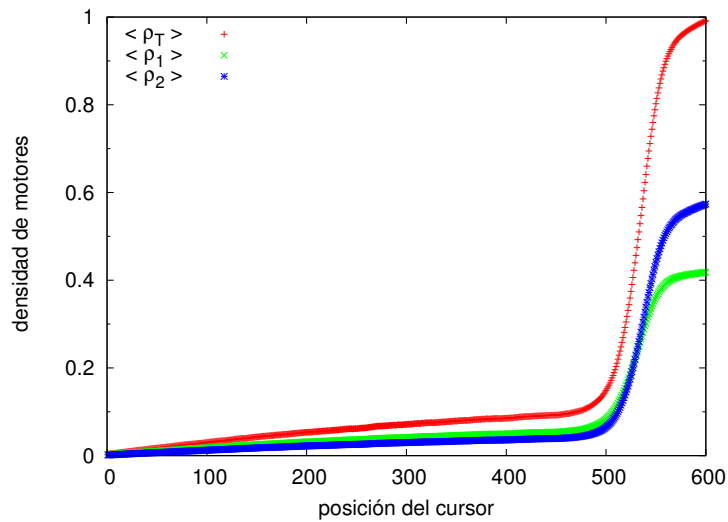


Figura 34: Densidad de motores en estado 1, en estado 2 y densidad total para un caso con $C_{tau} = 0$ y el resto de los parámetros del modelo que se explicitan en la Tabla 10.

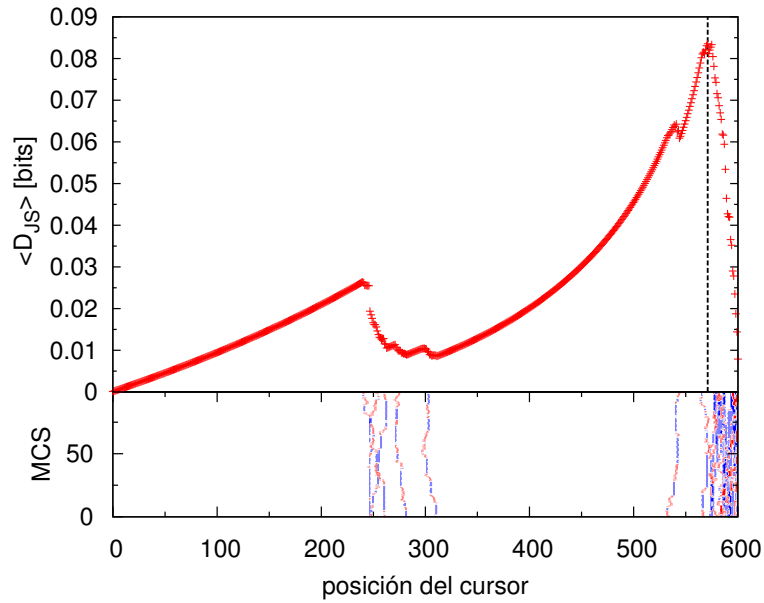


Figura 35: Valores de la *DJS* promedio para cada posición del cursor correspondientes a un caso con $C_{\text{tau}} = 0,005$ y el resto de los parámetros del modelo que se explicitan en la Tabla 10. La línea de puntos vertical corresponde a la posición 570 en la cual se detectó un *shock*. En la parte inferior de la figura se puede ver un gráfico espacial en el que cada fila representa el estado del filamento en un dado instante temporal. Se muestran cien MCSs una vez que el sistema alcanzó el estado estacionario. Un punto blanco representa un sitio vacío para ese tiempo, un punto azul, un sitio ocupado con un motor en estado 1 y un punto rojo, un sitio ocupado con un motor en estado 2.

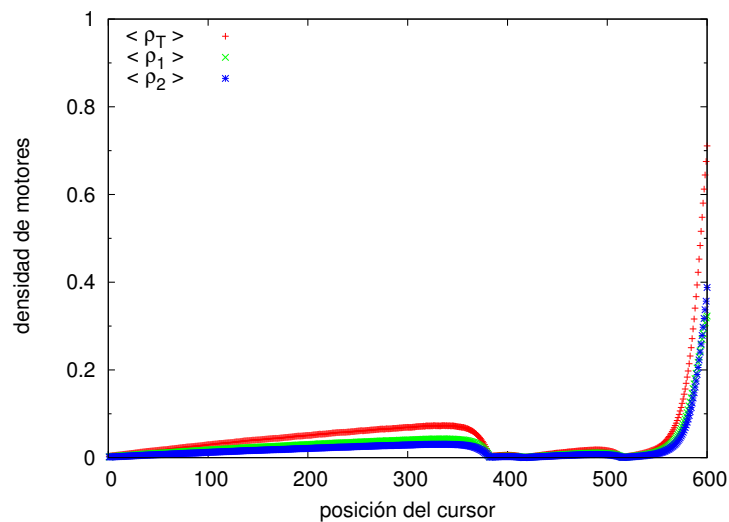


Figura 36: Densidad de motores en estado 1, en estado 2 y densidad total para un caso con $C_{\text{tau}} = 0,005$ y el resto de los parámetros del modelo que se explicitan en la Tabla 10.

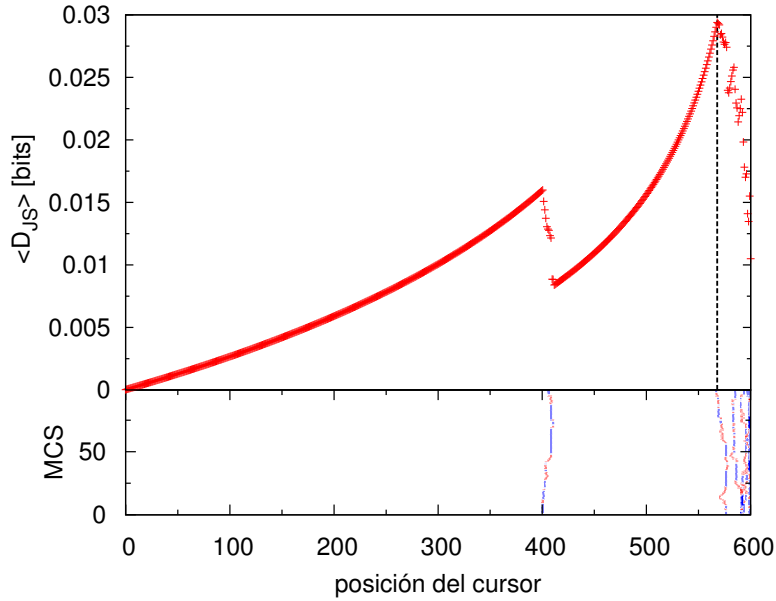


Figura 37: Valores de la *DJS* promedio para cada posición del cursor correspondientes a un caso con $C_{tau} = 0,01$ y el resto de los parámetros del modelo que se explicitan en la Tabla 10. La línea de puntos vertical corresponde a la posición 567 en la cual se detectó un *shock*. En la parte inferior de la figura se puede ver un gráfico espacial en el que cada fila representa el estado del filamento en un dado instante temporal. Se muestran cien MCSs una vez que el sistema alcanzó el estado estacionario. Un punto blanco representa un sitio vacío para ese tiempo, un punto azul, un sitio ocupado con un motor en estado 1 y un punto rojo, un sitio ocupado con un motor en estado 2.

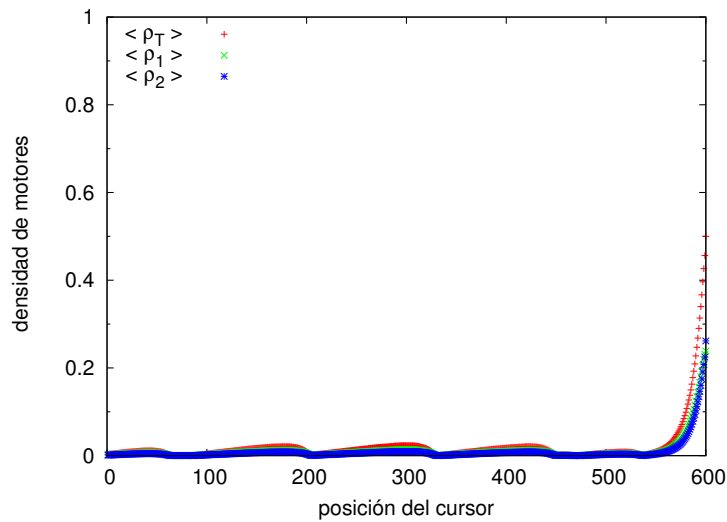


Figura 38: Densidad de motores en estado 1, en estado 2 y densidad total para un caso con $C_{tau} = 0,01$ y el resto de los parámetros del modelo que se explicitan en la Tabla 10.

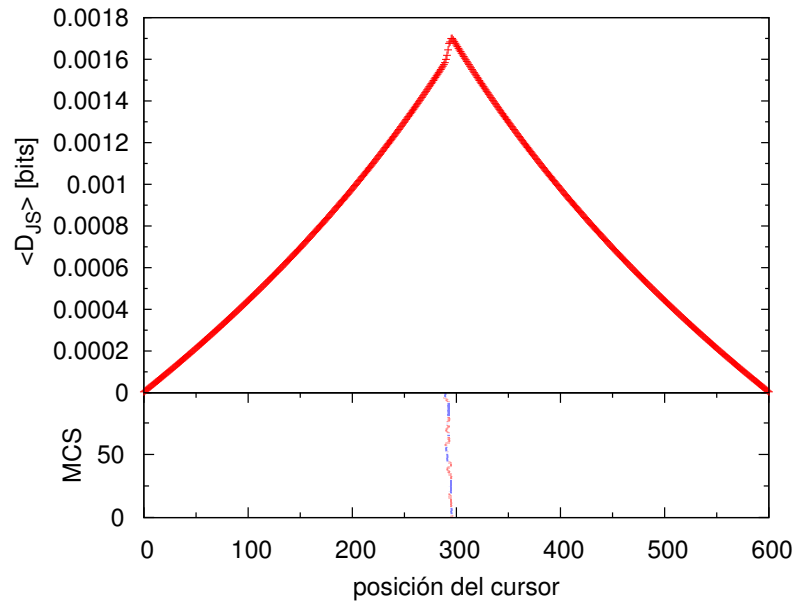


Figura 39: Valores de la DJS promedio para cada posición del cursor correspondientes a un caso con $C_{tau} = 0,1$ y el resto de los parámetros del modelo que se explicitan en la Tabla 10. En la parte inferior de la figura se puede ver un gráfico espacial en el que cada fila representa el estado del filamento en un dado instante temporal. Se muestran cien MCSs una vez que el sistema alcanzó el estado estacionario. Un punto blanco representa un sitio vacío para ese tiempo, un punto azul, un sitio ocupado con un motor en estado 1 y un punto rojo, un sitio ocupado con un motor en estado 2.

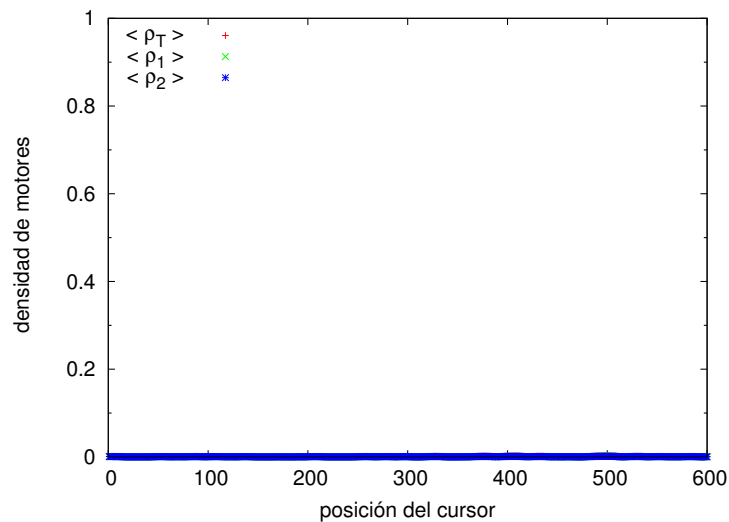


Figura 40: Densidad de motores en estado 1, en estado 2 y densidad total para un caso con $C_{tau} = 0,1$ y el resto de los parámetros del modelo que se explicitan en la Tabla 10.

Parte III

APLICACIONES CUÁNTICAS

UNA NUEVA MÉTRICA PARA TEORÍA DE LA INFORMACIÓN CUÁNTICA

Como parte de la investigación llevada a cabo con la divergencia de Jensen–Shannon, realizamos una propuesta que utiliza un derivado de ésta como métrica para estados cuánticos. Este capítulo presenta los resultados encontrados sobre este tema que amplían el marco de estudio realizado hasta aquí para incluir también sistemas cuánticos.

Se propone una nueva métrica para estados cuánticos a partir de los conceptos de entropía y purificación [34]. La métrica para dos estados mixtos se define como la raíz cuadrada de la entropía del promedio de las purificaciones representativas de dichos estados. Se analizan algunas propiedades básicas y se investiga su relación con otras distancias. Como una aplicación ilustrativa, se evalúa la métrica propuesta para estados mixtos de 1-qubit.

Además de los resultados encontrados en sí mismos, este capítulo enfatiza y realza la robustez y versatilidad de los métodos y herramientas de TI.

El capítulo se organiza de la siguiente manera: en la Sección 6.1 presentamos brevemente el problema de las distancias en mecánica cuántica. En la Sección 6.3 consideramos una medida de distancia para estados puros definida usando la entropía de von Neumann y estudiamos sus propiedades. La Sección 6.4 se dedica a la nueva propuesta de distancia entre estados mixtos recurriendo al concepto de purificación. En la Sección 6.5 presentamos una aplicación para el caso de estados mixtos de 1-qubit con un ejemplo de cálculo explícito. Finalmente, en la Sección 6.6 se presentan las conclusiones de esta parte del trabajo.

6.1 DISTANCIAS EN MECÁNICA CUÁNTICA

El estudio de distancias y métricas entre estados cuánticos es un área de permanente interés, últimamente ha cobrado gran impulso a causa de problemas que emergen de la Teoría de la Información Cuántica (TIC) [46]. Las distancias son utilizadas como medidas de distinguibilidad entre estados cuánticos [71] y en la definición del grado de entrelazamiento [72], para mencionar dos ejemplos muy relevantes. También caracterizan la estructura geométrica del espacio de los estados cuánticos [4].

En el formalismo matemático de la mecánica cuántica los estados de un sistema físico \mathcal{S} son representados por operadores actuando en un espacio de Hilbert \mathcal{H} . Más precisamente, los estados del sistema

\mathcal{S} son representados por los elementos de $\mathcal{B}(\mathcal{H})_1^+$, esto es, el conjunto de operadores positivos de traza uno sobre \mathcal{H} .

La noción de un estado como vector unitario de \mathcal{H} se refiere a los elementos extremos de $\mathcal{B}(\mathcal{H})_1^+$. Un estado $\rho \in \mathcal{B}(\mathcal{H})_1^+$ es extremo si y solo si es idempotente, es decir si $\rho^2 = \rho$. En este caso ρ es de la forma $|\psi\rangle\langle\psi|$ para algún vector unitario $|\psi\rangle \in \mathcal{H}$, y es llamado un estado *puro*. Si ρ no es idempotente, el correspondiente estado es llamado *mixto*.

Diferentes autores han tomado distintos caminos para introducir distancias entre estados cuánticos [70, 74, 7, 30, 37, 41, 45]. Un ejemplo importante es el de Wootters que, analizando las fluctuaciones estadísticas en los resultados de las mediciones [74], arribó a la siguiente distancia:

$$d_W(|\psi\rangle, |\varphi\rangle) = \arccos(|\langle\psi|\varphi\rangle|), \quad (69)$$

donde $\langle\psi|\varphi\rangle$ representa el producto interno entre los estados puros $|\psi\rangle$ y $|\varphi\rangle$, y por eso d_W da el “ángulo” entre estos dos estados.

Un enfoque alternativo para abordar el problema de introducir distancias entre estados cuánticos es generalizar la noción de distancia definida en un espacio de distribuciones de probabilidad clásico. Es posible, por ejemplo, generalizar la entropía relativa (DKL) de la teoría de la información clásica, que discutimos en la Sección 2.4.

La entropía relativa de un operador ρ con respecto a un operador σ , ambos pertenecientes a $\mathcal{B}(\mathcal{H})_1^+$, es:

$$S(\rho, \sigma) = \text{Tr}[\rho(\log \rho - \log \sigma)], \quad (70)$$

donde \log representa, como en el resto de la tesis, el logaritmo en base dos.

La entropía relativa no es una métrica porque no es simétrica y no satisface la desigualdad triangular. Más aún, puede no ser acotada y está bien definida sólo cuando el soporte de σ es mayor o igual que el de ρ [40]. El soporte de un operador es el subespacio generado por los autovectores del operador con autovalores no nulos. Esta restricción en el soporte es fuerte y puede ser violada en algunas situaciones físicas relevantes, como por ejemplo cuando σ es un estado puro de referencia.

Recientemente se ha investigado una distancia entre estados cuánticos mixtos bautizada como divergencia de Jensen-Shannon cuántica (DJSC), que es una versión simetrizada de la entropía relativa, Ecuación (70), y que se define del siguiente modo:

$$\hat{D}_{JS}(\rho, \sigma) = \frac{1}{2} \left[S\left(\rho, \frac{\rho + \sigma}{2}\right) + S\left(\sigma, \frac{\rho + \sigma}{2}\right) \right]. \quad (71)$$

Utilizaremos la notación \hat{D}_{JS} para identificar la DJSC, de modo de no confundirla con la DJS clásica (a la versión clásica general la denotamos D_{JS} y a la versión clásica con pesos $\frac{1}{2}$, la denotamos $D_{JS}^{1/2}$, ésta

última constituye el equivalente clásico de la que estamos presentando aquí).

Las principales propiedades de la DJSC como una medida de distinguibilidad son las siguientes:

- es simétrica y siempre bien definida.
- $0 \leq \hat{D}_{JS}(\rho, \sigma) \leq 1$, con $\hat{D}_{JS}(\rho, \sigma) = 0 \iff \rho = \sigma$.
- es invariante ante transformaciones unitarias, esto es:

$$\hat{D}_{JS}(U\rho U^\dagger, U\sigma U^\dagger) = \hat{D}_{JS}(\rho, \sigma),$$

donde U es un operador unitario. Una transformación unitaria representa una rotación en el espacio de Hilbert, por lo tanto, resulta natural que la distancia entre los estados rotados permanezca invariante.

- verifica el teorema H generalizado, esto es, para cualquier operador evolución Φ (no necesariamente unitario) dado por $\Phi\rho = \sum_i V_i \rho V_i^\dagger$ y $\sum_i V_i^\dagger V_i = I$, se satisface:

$$\hat{D}_{JS}(\Phi\rho, \Phi\sigma) \leq \hat{D}_{JS}(\rho, \sigma).$$

Esto significa que una operación no unitaria decrece la distinguibilidad entre los estados.

- es el cuadrado de una métrica, esto es:

$$\hat{d}_{JS}(\rho, \sigma) = \sqrt{\hat{D}_{JS}(\rho, \sigma)},$$

satisface además la desigualdad triangular.

Para la demostración de estas propiedades ver la Referencia [43], con excepción de la última que puede verse en las Referencias [33, 8].

Además, la DJSC tiene muchas interpretaciones interesantes en el ámbito de la TIC. En particular, puede identificarse con la cota Holevo. En efecto, la DJSC puede generalizarse como una medida de “distancia” entre los elementos de un ensamble cuántico $\{q_i, \rho_i\}$ con $\sum_i q_i = 1$, del siguiente modo:

$$\hat{D}_{JS}(\{q_i, \rho_i\}) = H_N\left(\sum_i q_i \rho_i\right) - \sum_i q_i H_N(\rho_i), \quad (72)$$

donde $H_N(\rho) = -\text{Tr}(\rho \log \rho)$ es la entropía de von Neumann que discutiremos en detalle en la siguiente sección. En el contexto de los procesos cuánticos de comunicación la DJSC, dada por la Ecuación 72, representa la cantidad Holevo, la cual acota la información mutua entre el emisor de un mensaje clásico codificado en estados cuánticos y su receptor.

A su vez, la DJSC ha sido aplicada como una medida del grado de entrelazamiento [44].

Como veremos en la Sección 6.3 la propuesta de métrica puede pensarse como un derivado de la DJSC. La propuesta gira en torno a dos conceptos centrales de la TIC, entropía y purificación, que discutiremos a continuación.

6.2 ENTROPÍA Y PURIFICACIÓN

La entropía es una noción fundamental en la TIC, al igual que en la TI clásica, como vimos en el Capítulo 2.

En el contexto cuántico, la expresión para la entropía difiere de la entropía de Shannon que presentamos en la Sección 2.1, Ecuación (5). Para un estado mixto descrito por el operador densidad ρ , von Neumann definió la entropía como:

$$H_N(\rho) = -\text{Tr}(\rho \log \rho) = -\sum_i \lambda_i \log \lambda_i, \quad (73)$$

donde $\{\lambda_i\}$ es el conjunto de autovalores del operador ρ .

La entropía de von Neumann tiene muchas propiedades interesantes [73]. Entre ellas podemos destacar que da una cota para una combinación convexa $\rho = \sum_i p_i \rho_i$, con $\sum_i p_i = 1$:

$$H_N(\rho) \leq \sum_i p_i H_N(\rho_i) + H(\{p_i\}), \quad (74)$$

donde H representa la entropía de Shannon. La igualdad se verifica si y solo si los estados ρ_i tienen soportes en subespacios ortogonales.

Aunque las expresiones de la entropía de Shannon y von Neumann parecen similares, son bastante diferentes. Estas diferencias son particularmente relevantes para la TIC. Por ejemplo, si a es un mensaje tomado de la fuente \mathcal{A} y $p(a)$ es la probabilidad del mensaje a , la entropía de Shannon de la fuente es

$$H[\mathcal{A}] = -\sum_a p(a) \log p(a).$$

Supongamos que tenemos un conjunto de estados cuánticos que codifican los mensajes tomados de la fuente \mathcal{A} , es decir al mensaje clásico a se le asigna un estado cuántico $|a_S\rangle\langle a_S|$. El ensamble de estados posibles (asociados a los mensajes de la fuente) será representado por el operador densidad:

$$\mu = \sum_a p(a) |a_S\rangle\langle a_S|.$$

Si los estados de las señales $|a_S\rangle$ no son ortogonales, a partir de la Ecuación (74), se satisface la siguiente desigualdad:

$$H_N(\mu) < H(\{p(a)\}).$$

No sólo queda claro que ambas entropías son diferentes sino también que la entropía de Shannon de la fuente da una cota para la entropía de von Neumann de los estados cuánticos que codifican los mensajes [31].

Otra propiedad importante de la entropía de von Neumann es que da el número mínimo de qubits necesarios para representar de forma fiel una señal cuántica.

Supongamos que Alicia tiene una fuente que produce una secuencia de estados puros de un qubit $|\psi\rangle$ y $|\varphi\rangle$. Cada estado que la fuente emite puede ser $|\psi\rangle$ o $|\varphi\rangle$ con igual probabilidad a priori (cada una será un medio). En este caso la matriz densidad de la fuente es $\frac{1}{2}(|\psi\rangle\langle\psi| + |\varphi\rangle\langle\varphi|)$ y la entropía de von Neumann:

$$H_N \left(\frac{|\psi\rangle\langle\psi| + |\varphi\rangle\langle\varphi|}{2} \right). \quad (75)$$

Para comunicar a Bob de manera fiel la secuencia de estados que la fuente va emitiendo, Alicia podría enviar un qubit por estado emitido. Sin embargo, el teorema de codificación de la fuente cuántica establece que la forma óptima de comunicarlos (con fidelidad arbitrariamente alta) es enviando, H_N (Ecuación 75) qubits por estado [63].

El segundo concepto clave que necesitamos es el de purificación. La denominación *purificación* hace referencia al hecho de que cada estado cuántico mixto puede ser considerado como un estado puro en un espacio de Hilbert de dimensión más alta.

Veamos la definición formal: sea ρ cualquier estado mixto en el espacio de Hilbert \mathcal{H} , una purificación de ρ es cualquier estado puro $|\psi\rangle$ en un espacio de Hilbert extendido $\mathcal{H} \otimes \mathcal{H}_{aux}$ que cumpla que:

$$\text{Tr}_{aux} (|\psi\rangle\langle\psi|) = \rho,$$

donde Tr_{aux} representa la traza parcial sobre el espacio de Hilbert \mathcal{H}_{aux} . En otras palabras, una purificación es cualquier estado puro en $\mathcal{H} \otimes \mathcal{H}_{aux}$ que tiene a ρ como estado reducido en el subsistema \mathcal{H} [30].

Esto es un procedimiento puramente matemático que permite asociar estados puros a estados mixtos. Sin embargo, vale la pena destacar que una realización experimental de purificación para un estado de un qubit ha sido propuesta en la Referencia [12] y llevada a cabo exitosamente, como se puede ver en la Referencia [59].

Si ρ admite la descomposición $\rho = \sum_i p_i |e_i\rangle\langle e_i|$ donde $p_i \geq 0$ y $\sum_i p_i = 1$, un ejemplo de purificación de ρ esta dado por:

$$|\psi\rangle = \sum_i \sqrt{p_i} |e_i\rangle \otimes |a_i\rangle, \quad (76)$$

donde los estados auxiliares $|a_i\rangle$ son mutuamente ortogonales.

En efecto, se cumple que:

$$\begin{aligned}
 \text{Tr}_{\text{aux}} (|\psi\rangle\langle\psi|) &= \text{Tr}_{\text{aux}} \left(\sum_{i,j} \sqrt{p_i p_j} |e_i\rangle\langle e_j| \otimes |a_i\rangle\langle a_j| \right) \\
 &= \sum_{i,j} \sqrt{p_i p_j} |e_i\rangle\langle e_j| \text{Tr}_{\text{aux}} (|a_i\rangle\langle a_j|) \\
 &= \sum_{i,j} \sqrt{p_i p_j} |e_i\rangle\langle e_j| \delta_{ij} \\
 &= \sum_i p_i |e_i\rangle\langle e_i| \\
 &= \rho,
 \end{aligned}$$

tal como requerimos para que constituya una purificación.

Completamos esta sección destacando dos propiedades de la purificación que vamos a utilizar más adelante:

1. para dos purificaciones $|\psi_1\rangle$ y $|\psi_2\rangle$ del estado ρ , existe una transformación unitaria U que actúa en \mathcal{H}_{aux} tal que:

$$|\psi_1\rangle = (I \otimes U)|\psi_2\rangle, \quad (77)$$

donde I es el operador identidad en el espacio \mathcal{H} [46].

2. si uno está interesado en las purificaciones de dos estados, se puede suponer, sin pérdida de generalidad, que ambas pertenecen al mismo espacio de Hilbert extendido [30].

En efecto, dadas $|\phi_1\rangle \in \mathcal{H} \otimes \mathcal{H}'$ y $|\phi_2\rangle \in \mathcal{H} \otimes \mathcal{H}''$, siempre podemos considerar a ambas purificaciones en el espacio extendido $\mathcal{H} \otimes (\mathcal{H}' \oplus \mathcal{H}'')$.

Desde un punto de vista físico la idea de purificación resulta útil para comprender el mecanismo de la decoherencia cuántica, como se puede ver en la Referencia [75].

6.3 ENTROPÍA DE VON NEUMANN COMO UNA MÉTRICA PARA ESTADOS PUROS

Sean $|\psi\rangle$ y $|\varphi\rangle$ dos estados puros en un dado espacio de Hilbert \mathcal{H} . Definimos la distancia entre estos estados en la forma:

$$D_N(|\psi\rangle, |\varphi\rangle) \equiv \sqrt{H_N \left(\frac{|\psi\rangle\langle\psi| + |\varphi\rangle\langle\varphi|}{2} \right)}. \quad (78)$$

Notemos que esta expresión es equivalente a la raíz cuadrada de la DJSC si la evaluamos entre estados puros. De hecho, la DJSC, Ecuación (71), puede reescribirse en términos de la entropía de von Neumann, Ecuación (73), de la siguiente forma:

$$D_{JS}(\rho, \sigma) = H_N \left(\frac{\rho + \sigma}{2} \right) - \frac{1}{2} H_N(\rho) - \frac{1}{2} H_N(\sigma).$$

Como la entropía de von Neumann se anula para todo estado puro, para los correspondientes operadores $\rho = |\psi\rangle\langle\psi|$ y $\sigma = |\varphi\rangle\langle\varphi|$ tendremos que:

$$D_{JS}(|\psi\rangle\langle\psi|, |\varphi\rangle\langle\varphi|) = D_N^2(|\psi\rangle, |\varphi\rangle).$$

Luego de realizar algunos pasos algebraicos, podemos reescribir el cuadrado de la Ecuación (78), en términos del producto interno entre $|\psi\rangle$ y $|\varphi\rangle$:

$$D_N^2(|\psi\rangle, |\varphi\rangle) = \Phi(|\langle\psi|\varphi\rangle|), \quad (79)$$

donde

$$\Phi(x) \equiv -\left(\frac{1-x}{2}\right) \log\left(\frac{1-x}{2}\right) - \left(\frac{1+x}{2}\right) \log\left(\frac{1+x}{2}\right). \quad (80)$$

Las principales propiedades de $D_N(|\psi\rangle, |\varphi\rangle)$, dada por la expresión de la Ecuación (78), son:

- es simétrica.
- se anula si y solo si $|\psi\rangle = e^{i\alpha}|\varphi\rangle$. Esto es, si ambos estados pertenecen al mismo rayo.
- está acotada: $0 \leq D_N(|\psi\rangle, |\varphi\rangle) \leq 1$. Esto se verifica a partir de la desigualdad de la Ecuación (74), utilizando el hecho de que la entropía de von Neumann se anula para todo estado puro.
- verifica la desigualdad triangular. Notar que $H_N\left(\frac{|\psi\rangle\langle\psi|+|\varphi\rangle\langle\varphi|}{2}\right)$ no la verifica.

Todas estas propiedades, que se extienden de las propiedades de la DJSC antes mencionadas, dan a D_N el carácter de métrica.

La distancia D_N entre dos estados puros vecinos $|\psi\rangle = \sum_j \sqrt{p_j} e^{i\phi_j}|j\rangle$ y $|\tilde{\psi}\rangle = |\psi\rangle + |\delta\psi\rangle = \sum_j \sqrt{p_j + \delta p_j} e^{i(\phi_j + \delta\phi_j)}|j\rangle$, donde $\{|j\rangle\}$ es una base ortonormal, esta dado por:

$$D_N^2(|\psi\rangle, |\tilde{\psi}\rangle) \simeq \sum_j \left(\frac{1}{8} \frac{dp_j^2}{p_j} - \frac{1}{16} \frac{dp_j^3}{p_j^2} \right). \quad (81)$$

Con este desarrollo y haciendo un desarrollo similar al cuadrado de la Ecuación (69) se puede demostrar que la distancia de Wootters y la distancia D_N (dada en la Ecuación (78)) son equivalentes hasta tercer orden en dp_j , de hecho, satisfacen la siguiente relación:

$$d_W(|\psi\rangle, |\tilde{\psi}\rangle) = \sqrt{2} D_N(|\psi\rangle, |\tilde{\psi}\rangle).$$

Una relación similar pero hasta segundo orden es válida con la métrica de Fubini–Study, como puede verse en la Referencia [42]. Así, el comportamiento de la métrica para estados vecinos es similar a la de otras métricas de la literatura.

6.4 EXTENSIÓN DE LA DEFINICIÓN DE LA MÉTRICA PARA ESTADOS MIXTOS

En la sección previa hemos utilizado la raíz cuadrada de la entropía de von Neumann del promedio $\frac{1}{2}(|\psi\rangle\langle\psi| + |\varphi\rangle\langle\varphi|)$ como una métrica entre los dos estados puros $|\psi\rangle$ y $|\varphi\rangle$. Aquí discutiremos su extensión de modo que sea posible considerar también estados mixtos.

Comencemos recordando que la distancia de Bures entre dos estados mixtos, esto es, para dos operadores ρ y σ pertenecientes a $\mathcal{B}(\mathcal{H})_1^+$, está dada por

$$D_B(\rho, \sigma) = \sqrt{2 - 2F(\rho, \sigma)},$$

donde $F(\rho, \sigma) = \text{Tr}\sqrt{\rho^{1/2}\sigma\rho^{1/2}}$ es la función fidelidad. Ver, por ejemplo, la Referencia [46].

El teorema de Uhlmann [70, 30] afirma que la fidelidad $F(\rho, \sigma)$ puede ser expresada en la forma:

$$F(\rho, \sigma) = \max_{|\psi\rangle, |\varphi\rangle} |\langle\psi|\varphi\rangle|, \quad (82)$$

donde la maximización se realiza sobre todas las purificaciones $|\psi\rangle$ de ρ y todas las purificaciones $|\varphi\rangle$ de σ . Luego, la métrica de Bures puede ser reescrita de la siguiente forma:

$$D_B(\rho, \sigma) = \min_{|\psi\rangle, |\varphi\rangle} \sqrt{2 - 2|\langle\psi|\varphi\rangle|}, \quad (83)$$

donde el mínimo se toma sobre todas las purificaciones de ρ y σ .

Por analogía con esta última expresión, podemos definir a partir de la Ecuación (78), una métrica para estados mixtos arbitrarios ρ y σ pertenecientes a $\mathcal{B}(\mathcal{H})_1^+$ de la siguiente manera:

$$D_N(\rho, \sigma) \equiv \min_{|\psi\rangle, |\varphi\rangle} \sqrt{H_N\left(\frac{|\psi\rangle\langle\psi| + |\varphi\rangle\langle\varphi|}{2}\right)}, \quad (84)$$

donde, nuevamente, el mínimo es tomado sobre todas las purificaciones $|\psi\rangle$ de ρ y todas las purificaciones $|\varphi\rangle$ de σ .

A partir de la Ecuación (79) y teniendo en cuenta que $\Phi(x)$ es una función decreciente¹ de $x = |\langle\psi|\varphi\rangle| \in [0, 1]$, buscar el mínimo en nuestra propuesta, Ecuación (84), es equivalente a buscar las purificaciones que maximicen la superposición $|\langle\psi|\varphi\rangle|$.

El carácter de métrica de $D_N(\rho, \sigma)$ para estados mixtos se mantiene por construcción. Otra ventaja importante es que está siempre bien definida (no hay restricciones en el soporte).

Desde un punto de vista conceptual, lo que la propuesta plantea es que resulta equivalente reemplazar el problema de medir la distancia entre dos estados mixtos arbitrarios, ρ y σ , por el problema de

¹ Esto se puede corroborar haciendo uso de la Ecuación (80).

distinguibilidad entre todos los ensambles $\{|\psi\rangle, |\varphi\rangle; p_1 = \frac{1}{2}, p_2 = \frac{1}{2}\}$ contruidos de las purificaciones de ρ y σ respectivamente [31].

De acuerdo a la identidad de la Ecuación (82), la métrica $D_N(\rho, \sigma)$ puede ser expresada en términos de la fidelidad de la siguiente forma:

$$D_N(\rho, \sigma) = \sqrt{\Phi(F(\rho, \sigma))}. \quad (85)$$

Usando esta representación, podemos derivar las propiedades principales de la métrica $D_N(\rho, \sigma)$. A saber:

- D_N es invariante bajo transformaciones unitarias:

$$D_N(U\rho U^\dagger, U\sigma U^\dagger) = D_N(\rho, \sigma),$$

- si \mathcal{Q} es una operación cuántica que preserva traza, se cumple:

$$D_N(\mathcal{Q}(\rho), \mathcal{Q}(\sigma)) \leq D_N(\rho, \sigma).$$

Por otro lado, es posible derivar una expresión alternativa para la fidelidad [46]:

$$F(\rho, \sigma) = \max_{|\varphi\rangle} |\langle \psi | \varphi \rangle|,$$

donde $|\psi\rangle$ es cualquier purificación *fija* de ρ , y la maximización se realiza sobre todas las purificaciones $|\varphi\rangle$ de σ .

Con esta expresión para la fidelidad podemos reescribir la métrica propuesta, a partir de la Ecuación (85), de modo que puede ser expresada en una manera más conveniente:

$$D_N(\rho, \sigma) = \min_{|\varphi\rangle} \sqrt{H_N\left(\frac{|\psi\rangle\langle\psi| + |\varphi\rangle\langle\varphi|}{2}\right)}, \quad (86)$$

donde $|\psi\rangle$ es cualquier purificación fija de ρ , y la minimización se toma sobre todas las purificaciones $|\varphi\rangle$ de σ .

Antes de pasar a considerar el caso de dos estados mixtos de 1-qubit en la próxima sección, vale la pena hacer dos consideraciones respecto a la definición de distancia entre estados cuánticos dado por la Ecuación (84).

En primer lugar, sobre su computabilidad. Como ya mencionamos, las purificaciones de ρ y σ pueden pensarse como pertenecientes al mismo espacio de Hilbert extendido. Además, de acuerdo a la Ecuación (77), cada purificación se relaciona con otra por una transformación unitaria. Por lo tanto, la tarea técnica de encontrar el mínimo en la Ecuación (86) es equivalente a encontrar la operación unitaria que actúa en el espacio de Hilbert auxiliar que maximiza la superposición $|\langle \psi | \varphi \rangle|$. Para ilustrar este punto, en la próxima sección calcularemos la distancia para un ejemplo particular.

En segundo lugar, la importancia de la métrica $D_N(\rho, \sigma)$ a la luz de la decoherencia. La decoherencia puede ser pensada como la evolución unitaria de un sistema compuesto que consiste en el sistema bajo estudio y su ambiente, bajo el cual ambas componentes se convierten en entrelazadas [75]. En este contexto, el espacio auxiliar \mathcal{H}_{aux} correspondería al ambiente.

6.5 EVALUACIÓN DE LA MÉTRICA PARA ESTADOS MIXTOS DE 1-QUBIT

Como aplicación ilustrativa, vamos a evaluar la métrica D_N dada en la Ecuación (84), para dos estados de 1-qubit. En este caso es conveniente usar la representación de Bloch para un estado mixto de 1-qubit:

$$\rho = \frac{I_2 + \vec{r} \cdot \vec{\sigma}}{2}, \quad (87)$$

donde I_n denota la matriz identidad de $n \times n$, el vector de Bloch \vec{r} es un vector 3-dimensional tal que $\|\vec{r}\| \leq 1$, y el vector $\vec{\sigma}$ tiene como componentes las matrices de Pauli. La matriz ρ pertenece al plano complejo C^2 .

En la Referencia [14] se muestra que una purificación de una matriz densidad arbitraria ρ actuando sobre $C^2 \otimes C^2$ (esto es, una matriz 4×4) esta dada por:

$$P_\rho = \frac{1}{4} \left(I_4 + \sum_{i=1}^3 r_i \sigma_i \otimes I_2 + \sum_{i=1}^3 \gamma_i I_2 \otimes \sigma_i + \sum_{i=1}^3 \sum_{j=1}^3 A_{ij} \sigma_i \otimes \sigma_j \right), \quad (88)$$

donde r_i son las componentes del vector \vec{r} , γ_i son las componentes del vector $\vec{\gamma} = A^T \vec{r}$, y la matriz real 3×3 , A , es una solución del sistema de ecuaciones:

$$AA^T = (1 - \|\vec{r}\|) I_3 + \vec{r} \vec{r}^T \quad (89)$$

$$\det(A) = \|\vec{r}\|^2 - 1. \quad (90)$$

El resultado de Constantinescu *et al.* [14] establece además que todas las purificaciones de ρ son de la forma dada en la Ecuación (88), con la matriz $A = \tilde{A}S$, siendo S un elemento del grupo de Lie $SO(3, \mathbb{R}^3)$ y \tilde{A} una solución particular del sistema dado en las Ecuaciones (89) y (90).

Por lo tanto, evaluar el mínimo en la Ecuación (86) para obtener D_N en este caso da lugar a un problema de extremización sobre el espacio de parámetros del grupo de Lie $SO(3, \mathbb{R}^3)$.

Para ver como funciona el método consideremos el ejemplo particular de un canal de depolarización y veamos cómo encontrar este

mínimo evaluando la distancia $D_N(\rho, \mathcal{E}_p(\rho))$ entre un estado de entrada, ρ , y su estado emergente, $\mathcal{E}_p(\rho)$, en dicho canal:

$$\mathcal{E}_p(\rho) = \frac{p I_2}{2} + (1-p)\rho, \quad 0 \leq p \leq 1. \quad (91)$$

El canal de depolarización puede verse como un proceso que puede llevar al sistema a un estado completamente mixto con probabilidad p , o dejarlo sin ningún cambio con probabilidad $1-p$. El efecto del canal de depolarización, Ecuación (91), sobre el estado ρ es el de “contraer” el vector \vec{r} en un factor $(1-p)$. Esto significa que si ρ está dado por la Ecuación (87) con un vector \vec{r} , entonces:

$$\mathcal{E}_p(\rho) = \frac{I_2 + \vec{r}' \cdot \vec{\sigma}}{2}, \quad \vec{r}' = (1-p)\vec{r}.$$

Una estrategia para evaluar la métrica $D_N(\rho, \mathcal{E}_p(\rho))$ se puede establecer de la siguiente manera: sea A una solución del sistema dado en las Ecuaciones (89) y (90), correspondiente al vector de Bloch \vec{r} , y sea A_p la solución correspondiente a $(1-p)\vec{r}$. Se cumple que:

$$A_p A_p^T = A A^T + f(p) \Omega \quad (92)$$

$$\det A_p = \det A - f(p) \|\vec{r}\|^2, \quad (93)$$

donde $f(p) = 1 - (1-p)^2$ y $\Omega = \|\vec{r}\|^2 I_3 - \vec{r} \vec{r}^T$.

Ahora tomemos una purificación fija ρ caracterizada por la matriz A (solución del sistema dado en las Ecuaciones (89) y (90)) y busquemos una matriz \tilde{A}_p que satisfice las Ecuaciones (92) y (93) (asociada con la purificación $P_{\mathcal{E}_p(\rho)}$). Luego, cada purificación de $\mathcal{E}_p(\rho)$ puede ser expresada en la forma $A_p = \tilde{A}_p S$ donde S es un elemento de $SO(3, \mathbb{R}^3)$.

Por esto, encontrar $S \in SO(3, \mathbb{R}^3)$ que minimice nuestra propuesta, dada por la Ecuación (86), es equivalente a encontrar $S \in SO(3, \mathbb{R}^3)$ que minimice la norma $\|\tilde{A}_p S - A\|_{HS}$ [14]. Aquí $\|\cdot\|_{HS}$ representa la norma de Hilbert-Schmidt.

Este es un problema clásico en la literatura que se conoce como problema de Procusto. Una solución algorítmica para el problema de Procusto se puede encontrar en la Referencia [61].

Resolviendo el sistema dado en las Ecuaciones (89), (90) y el de las Ecuaciones (92) y (93), y aplicando el algoritmo para resolver el problema de Procusto, podemos encontrar la purificación $P_{\mathcal{E}_p(\rho)}$ que extremiza la métrica dada por la Ecuación (86). Luego, calculando los autovalores del “promedio” $\frac{P_\rho + P_{\mathcal{E}_p(\rho)}}{2}$ podemos evaluar la distancia $D_N(\rho, \mathcal{E}_p(\rho))$.

La Figura 41 muestra los valores obtenidos como función del parámetro p que caracteriza el canal de depolarización, para diferentes estados de 1-qubit, ρ , cada uno con un valor diferente de la norma del vector de Bloch asociado, $\|\vec{r}\| = r$.

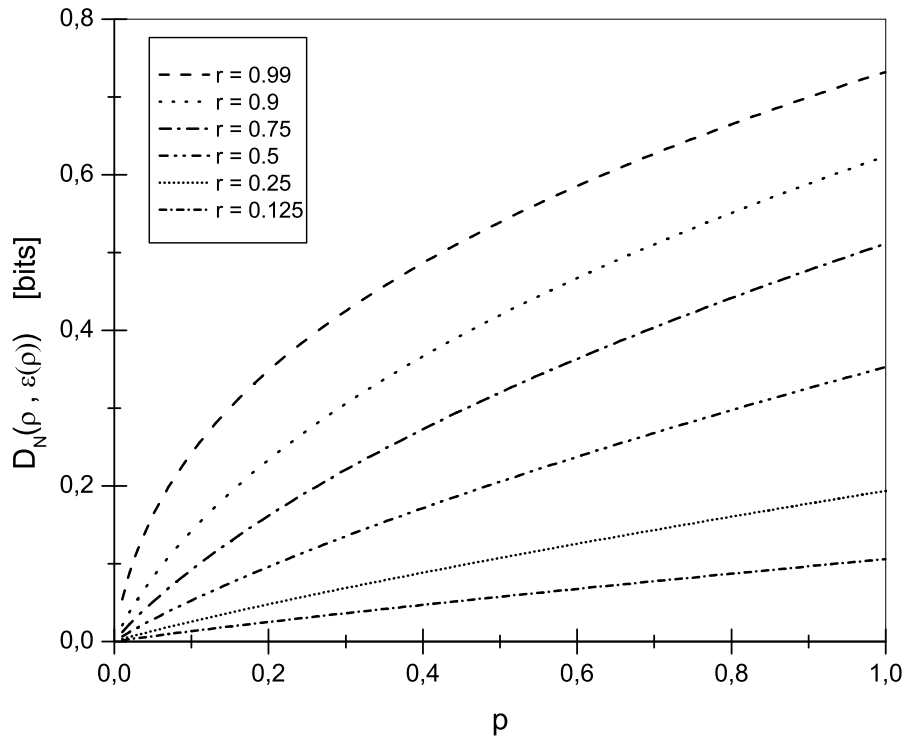


Figura 41: Distancia con la métrica propuesta entre un estado de 1-qubit y el estado correspondiente que emerge de un canal de depolarización. Se muestran los resultados para diferentes estados, caracterizados por la norma del vector de Bloch, en función del parámetro p que caracteriza el canal de depolarización. Ver texto principal por más detalles.

Podemos notar que en todos los casos la distancia se anula para $p = 0$ que corresponde al caso en que el canal de depolarización deja al estado intacto, y crecen monótonamente hasta tomar el valor máximo en $p = 1$, valor para el cual el efecto del canal es el de llevar el estado a uno completamente mixto. Sin embargo, no debemos perder de vista que el punto más importante era mostrar a través del cómputo de un ejemplo concreto el procedimiento para evaluar la distancia.

6.6 CONCLUSIONES DEL CAPÍTULO

Hemos introducido una nueva propuesta de métrica para estados cuánticos y estudiado algunas de sus propiedades.

En el caso de dos estados puros $|\psi\rangle$ y $|\varphi\rangle$, la métrica $D_N(|\psi\rangle, |\varphi\rangle)$ dada en la Ecuación (78) corresponde a la raíz cuadrada de la entropía

de von Neumann del promedio de los estados, o más precisamente a la raíz cuadrada de la DJSC entre ellos.

En el caso de dos estados mixtos ρ y σ , la métrica $D_N(\rho, \sigma)$ dada en la Ecuación (84) se define como una extensión natural de ésta última, y viene dada por el valor mínimo de la entropía del promedio de las purificaciones correspondientes a cada uno de los estados.

Desde un punto de vista físico hemos reemplazado el problema de evaluar la distancia entre dos estados mixtos por el problema de distinguibilidad de ensambles construidos a partir de purificaciones de esos estados.

A través de un ejemplo particular mostramos la computabilidad de la distancia propuesta.

Por último, es importante destacar que Życzkowski y colaboradores demostraron recientemente [60] que el cuadrado de la distancia aquí propuesta provee una cota más fina para la cantidad de Holevo para el ensamble $\{\frac{1}{2}, \frac{1}{2}, \rho_1, \rho_2\}$. Esto le da, a la distancia aquí propuesta, un significado físico muy importante.

Parte IV

EPÍLOGO

Si bien al final de los Capítulos 4, 5 y 6 se incluyeron conclusiones parciales de los temas que se fueron discutiendo en cada uno, en este capítulo se presenta, a modo de cierre y para enfatizar los resultados obtenidos, un compendio de las conclusiones generales del trabajo de tesis.

El estudio de la dinámica del motor molecular KIF1A comenzó con el desarrollo de una simulación que implementa el modelo de Nishinari *et al.* [47] y Greulich *et al.* [22].

Se estableció un método de análisis para estudiar la dinámica del estado estacionario de este modelo pero que puede ser utilizado para analizar cualquier otro modelo de tráfico.

El procedimiento involucra la construcción de secuencias simbólicas que caracterizan el estado del filamento y su posterior análisis a partir de la DJS.

Como parte de este análisis se adaptó un método de segmentación de secuencias no estacionarias. En este contexto y debido a la forma en que las secuencias simbólicas se construyeron, la presencia de una pared de dominio en el filamento se traduce en diferencias composicionales en las secuencias.

El método permite la localización automática de la posición en el filamento de esta pared de dominio.

La elección de los pesos asociados a las distribuciones de probabilidad en el cálculo de la DJS, del valor del umbral de significancia y del número de secuencias utilizados para realizar el análisis son importantes para que el método de localización funcione de manera adecuada.

Como paso previo al análisis con la DJS resulta clave el mapeo que se hace de la dinámica a secuencias simbólicas. La construcción adoptada demostró ser eficiente en la detección de las paredes de dominio.

Posteriormente, introducimos un modelo estocástico para el transporte intracelular de los motores moleculares KIF1A que incluye la interacción con la proteína tau.

El modelo fue construido como una extensión del modelo de Nishinari y colaboradores con el que se venía trabajando y se desarrolló de modo de preservar una de sus principales virtudes: que los parámetros se relacionan directamente con cantidades controlables experimentalmente.

La interacción de los motores con la proteína tau se modela a través de dos parámetros p_a y p_m que surgen de los efectos observados en los experimentos de Dixit y colaboradores [18]. El efecto que una

molécula de tau tiene en la frecuencia de ligadura de las kinesinas al filamento aparece en el modelo a través de p_a , y el efecto favoreciendo la desligadura de kinesinas que se mueven a lo largo del filamento aparece a través de p_m .

Los valores de estos parámetros han sido fijados basándonos en los resultados experimentales de Dixit *et al.* pero hemos probado que los resultados obtenidos para los observables son robustos ante modificaciones de estos parámetros desde el punto de vista de la simulación.

Nuestro modelo coincide con el modelo de Nishinari y colaboradores cuando no hay moléculas de tau presentes en el filamento, y en este sentido es apropiado referirlo como una extensión de éste.

Los resultados que encontramos tanto para el tiempo de ligadura y la longitud de corrida de los motores KIF1A como para la formación de paredes de dominios, bajo la presencia de tau, sugieren que esta proteína podría jugar un papel importante en la regulación del transporte intracelular debido a las kinesinas.

Paralelamente hemos introducido una nueva propuesta de métrica para estados cuánticos y estudiado algunas de sus propiedades.

Para el caso de estados puros, la métrica corresponde a la raíz cuadrada de la entropía de von Neumann del promedio de los estados y, en el caso de dos estados mixtos, se calcula a partir de las purificaciones de los estados, y viene dada por el valor mínimo de la entropía del promedio de las purificaciones correspondientes a cada uno de los estados.

Para mostrar la computabilidad de la métrica propuesta, se calculó explícitamente la distancia entre dos estados de 1-qubit: el estado de entrada a un canal de depolarización y el estado emergente.

Parte V

APÉNDICES

EXTRA, EXTRA, . . . INFORMACIÓN DE ÚLTIMO MOMENTO

Este apéndice contiene algunas consideraciones de Teoría de la Información que completan o complementan las definiciones y propiedades presentadas en el Capítulo 2.

A.1 ENTROPÍA CONJUNTA

En la Sección 2.2 presentamos la entropía conjunta y la entropía condicional y enunciamos el siguiente resultado:

$$H(X, Y) = H(X) + H(Y|X) \quad (94)$$

$$= H(Y) + H(X|Y), \quad (95)$$

que dice que la información asociada a un par de variables aleatorias X e Y es igual a la información de una más la información de la otra condicionada a la primera.

Veamos la demostración de este resultado. Por la definición de entropía conjunta, Ecuación (7), tenemos que:

$$H(X, Y) = - \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log(p(x, y)).$$

Como $p(x, y) = p(x)p(y|x)$ podemos reescribir la expresión del siguiente modo:

$$\begin{aligned} H(X, Y) &= - \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log(p(x)p(y|x)) \\ &= - \sum_{x \in \mathcal{X}} \left(\sum_{y \in \mathcal{Y}} p(x, y) \right) \log(p(x)) - \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log(p(y|x)) \\ &= - \sum_{x \in \mathcal{X}} p(x) \log(p(x)) - \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log(p(y|x)). \end{aligned}$$

De acuerdo a la Ecuación (5) podemos identificar el primer sumando como $H(X)$ y de acuerdo a la Ecuación (9) podemos identificar el segundo sumando como $H(Y|X)$, demostrando así la Ecuación (94). Demostrar la Ecuación (95) es exactamente lo mismo cambiando X por Y en lo que acabamos de hacer.

A.2 LA INFORMACIÓN NO HACE DAÑO

Presentamos en esta sección un resultado muy importante en TI que se conoce como desigualdad de Jensen y algunas de sus consecuen-

cias, entre ellas, la no-negatividad de la [DKL](#) como prometimos en la Sección [2.4.2](#).

A.2.1 Desigualdad de Jensen

Una función $f(x)$ se dice *convexa* en un intervalo (a, b) si dado λ , $0 \leq \lambda \leq 1$, se cumple $\forall x_1, x_2 \in (a, b)$ que:

$$f(\lambda x_1 + (1 - \lambda)x_2) \leq \lambda f(x_1) + (1 - \lambda)f(x_2). \quad (96)$$

En una función convexa toda secante queda por encima de la gráfica.

Una función f se dice *cóncava* si $-f$ es convexa.

TEOREMA (DESIGUALDAD DE JENSEN): Si f es una función convexa y X es una variable aleatoria entonces:

$$f(\langle X \rangle) \leq \langle f(X) \rangle. \quad (97)$$

Demostraremos este resultado por inducción en el rango de la variable aleatoria.

Si $N = 2$, la desigualdad de Jensen (97) toma la forma:

$$f(px_1 + (1 - p)x_2) \leq pf(x_1) + (1 - p)f(x_2). \quad (98)$$

Si elegimos $\lambda = p$ en la definición de convexidad (96), la Ecuación (98) se cumple inmediatamente.

Sigamos entonces con nuestra demostración por inducción, asumiendo que la desigualdad de Jensen se verifica para $N = k - 1$ y mostrando que esto implica que se cumple para $N = k$. Es decir nuestra hipótesis inductiva es:

$$f\left(\sum_{i=1}^{k-1} p_i x_i\right) \leq \sum_{i=1}^{k-1} p_i f(x_i) \quad (99)$$

y queremos ver que eso implica que se cumple:

$$f\left(\sum_{i=1}^k p_i x_i\right) \leq \sum_{i=1}^k p_i f(x_i). \quad (100)$$

Vale la pena mencionar que los " p_i " en las Ecuaciones (99) y (100) no son "los mismos", en el sentido de que cada uno constituye una distribución bien definida para su rango.

Comencemos trabajando el lado derecho de la Ecuación (100):

$$\sum_{i=1}^k p_i f(x_i) = p_k f(x_k) + \sum_{i=1}^{k-1} p_i f(x_i).$$

Como $\sum_{i=1}^k p_i = 1$, entonces $\sum_{i=1}^{k-1} p_i = 1 - p_k$.

Si definimos $\tilde{p}_i = \frac{p_i}{1-p_k} \forall i = 1, 2, \dots, k-1$, se cumple que $\sum_{i=1}^{k-1} \tilde{p}_i = \frac{1}{1-p_k} \sum_{i=1}^{k-1} p_i = 1$, por lo tanto los \tilde{p}_i constituyen una distribución bien definida para el rango $k-1$ y se verifica para ellos la Ecuación (99). Entonces podemos escribir:

$$\begin{aligned} \sum_{i=1}^k p_i f(x_i) &= p_k f(x_k) + (1-p_k) \sum_{i=1}^{k-1} \tilde{p}_i f(x_i) \\ &\geq p_k f(x_k) + (1-p_k) f\left(\sum_{i=1}^{k-1} \tilde{p}_i x_i\right), \end{aligned} \quad (101)$$

donde hemos utilizado justamente la Ecuación (99) para el último paso.

Si ahora utilizamos la definición de convexidad, Ecuación (96), con $\lambda = p_k$, tendremos que:

$$\begin{aligned} p_k f(x_k) + (1-p_k) f\left(\sum_{i=1}^{k-1} \tilde{p}_i x_i\right) &\geq f\left(p_k x_k + (1-p_k) \sum_{i=1}^{k-1} \tilde{p}_i x_i\right) \\ &= f\left(p_k x_k + \sum_{i=1}^{k-1} p_i x_i\right) \\ &= f\left(\sum_{i=1}^k p_i x_i\right), \end{aligned} \quad (102)$$

y juntando las Ecuaciones (101) y (102) el teorema queda demostrado.

A.2.2 Aplicaciones de la desigualdad de Jensen

A.2.2.1 No-negatividad de la divergencia de Kullback-Leibler

Como una aplicación de la desigualdad de Jensen se puede demostrar la no-negatividad de la divergencia de Kullback-Leibler (ver Secciones 2.4.1 y 2.4.2), conocida a veces con nombre propio como *desigualdad de Información*.

TEOREMA (DESIGUALDAD DE INFORMACIÓN): Si $P(X)$ y $Q(X)$ son dos distribuciones posibles para la variable aleatoria X entonces se cumple que:

$$D_{KL}(P(X), Q(X)) \geq 0, \quad (103)$$

con $D_{KL}(P(X), Q(X)) = 0$ si y sólo si $P(x) = Q(x) \forall x \in \mathcal{X}$.

Demostración:

$$\begin{aligned}
 -D_{\text{KL}}(P, Q) &= -\sum_{x \in \mathcal{X}} p(x) \log \left(\frac{p(x)}{q(x)} \right) \\
 &= \sum_{x \in \mathcal{X}} p(x) \log \left(\frac{q(x)}{p(x)} \right) \\
 &= \left\langle \log \left(\frac{Q(X)}{P(X)} \right) \right\rangle.
 \end{aligned} \tag{104}$$

Como la función logaritmo es cóncava cumple la desigualdad de Jensen con el signo cambiado (es lo mismo pensar que la función menos-logaritmo es convexa y después multiplicar por -1). Entonces:

$$\begin{aligned}
 -D_{\text{KL}}(P, Q) &= \left\langle \log \left(\frac{Q(X)}{P(X)} \right) \right\rangle \\
 &\leq \log \left(\sum_{x \in \mathcal{X}} p(x) \frac{q(x)}{p(x)} \right) \\
 &\leq \log \left(\sum_{x \in \mathcal{X}} q(x) \right) \\
 &= \log(1) = 0.
 \end{aligned} \tag{105}$$

Por lo tanto, $D_{\text{KL}}(P, Q) \geq 0$.

Para entender el caso de la igualdad conviene antes hacer una pequeña digresión para comentar que sucede en un caso general de la desigualdad de Jensen cuando se da una igualdad. Si $\langle f(X) \rangle = f(\langle X \rangle)$, se cumple que $\langle X \rangle = X$, es decir que X es una constante.

En este caso entonces, la igualdad $\sum_{x \in \mathcal{X}} p(x) \frac{q(x)}{p(x)} = \frac{q(x)}{p(x)}$ implica que $\frac{q(x)}{p(x)} = c$ pero $c = 1$ es la única constante que permite que ambas distribuciones estén normalizadas.

Por lo tanto, $D_{\text{KL}}(P, Q) = 0 \Leftrightarrow p(x) = q(x) \forall x$.

A.2.2.2 Cota a la entropía

Otra consecuencia de la desigualdad de Información es que una variable aleatoria tendrá la entropía máxima cuando su distribución sea uniforme, como establece el siguiente teorema.

TEOREMA: Sea X una variable aleatoria que puede tomar alguno de los siguientes valores x_1, x_2, \dots, x_N , la entropía asociada cumple que:

$$H(X) \leq \log(N),$$

donde N denota el número de elementos en el rango de X . La igualdad se da si y sólo si X tiene una distribución uniforme sobre su rango.

El caso $N = 2$ de este resultado se puede contemplar en el Ejemplo 3 de la Sección 2.1.2 (y en particular en la Figura 1).

Demostración:

Llamemos $P(X)$ a una distribución cualquiera de X y $U(X) = \frac{1}{N}$ a la distribución uniforme.

La no-negatividad de la DKL implica que:

$$\begin{aligned} 0 \leq D_{KL}(P, U) &= \sum_{x \in \mathcal{X}} p(x) \log \left(\frac{p(x)}{u(x)} \right) \\ &= \sum_{x \in \mathcal{X}} p(x) \log(p(x)) + \sum_{x \in \mathcal{X}} p(x) \log(N) \\ &= -H(X) + \log(N). \end{aligned} \tag{106}$$

Por lo tanto queda demostrado que $H(X) \leq \log(N)$.

Por otro lado como $D_{KL}(P, U) = 0 \Leftrightarrow P = U$, tenemos que el valor máximo se alcanza si y sólo si X tiene una distribución uniforme.

A.2.2.3 La entropía condicional

Por último, a partir de la interpretación que hicimos en la Sección 2.4.2 de la información mutua como un caso particular de la DKL, podemos demostrar otro resultado que es consecuencia de la desigualdad de información (y por ende, de la desigualdad de Jensen).

TEOREMA:

$$H(X|Y) \leq H(X),$$

la igualdad se da si y sólo si X e Y son independientes.

Este teorema dice que conocer otra variable aleatoria Y puede sólo disminuir la incertidumbre acerca de X o parafraseando, que la información no hace daño.

Sin embargo, debe interpretarse con la precaución de que es cierto en promedio. Así, mientras $H(X|Y = y)$ puede ser mayor, igual o menor que $H(X)$, el promedio $H(X|Y)$ es siempre menor o igual que $H(X)$.

La demostración es directa con los siguientes ingredientes: la definición de información mutua (Ecuación (12)), la información mutua es un caso particular de la DKL (Ecuación (16)) y la DKL es no-negativa (Ecuación (103)). Con todo eso:

$$0 \leq I(X, Y) = H(X) - H(X|Y),$$

de donde se deduce la desigualdad del teorema.

FUNCIONES ESPECIALES Y CONSIDERACIONES ESTADÍSTICAS

En este capítulo se incluyen definiciones y algunas propiedades de funciones especiales, así también como breves comentarios acerca del estadístico chi-cuadrado. Todos ellos aparecen en el desarrollo del método de segmentación basado en la [DJS](#) del Capítulo 3. También se explicitan los pasos para llegar a expresiones fundamentales del método de segmentación como son la significancia y la significancia máxima. Estas deducciones son importantes para tener claro el significado de las cantidades que utilizamos y las aproximaciones que se esconden detrás de las expresiones finales pero no se incluyeron en el Capítulo 3 para no oscurecer la presentación del método.

B.1 FUNCIONES ESPECIALES

En esta sección presentaremos la definición de algunas funciones especiales que son de utilidad en este trabajo: las funciones gamma completa e incompleta y la función error. Por más información acerca de estas funciones, ver por ejemplo la Referencia [3].

B.1.1 Función gamma

La función gamma se define del siguiente modo :

$$\Gamma(a) = \int_0^{\infty} e^{-t} t^{a-1} dt, \quad \text{Re}(a) > 0.$$

Satisface la siguiente propiedad:

$$\Gamma(a + 1) = a \Gamma(a).$$

Cuando el argumento es entero, la función gamma se reduce a la función factorial (desplazada una unidad):

$$\Gamma(n + 1) = n! .$$

La función gamma incompleta se define como sigue:

$$\gamma(a, x) = \int_0^x e^{-t} t^{a-1} dt, \quad \text{Re}(a) > 0. \quad (107)$$

B.1.2 Función error

La función error se define del siguiente modo:

$$\text{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt.$$

De acuerdo a la Ecuación (107), la función error se puede ver como un caso especial de la función gamma incompleta con $\alpha = \frac{1}{2}$, de hecho, se cumple:

$$\operatorname{erf}(z) = \frac{\gamma(\frac{1}{2}, z^2)}{\sqrt{\pi}}. \quad (108)$$

B.2 ¿SON DOS DISTRIBUCIONES DIFERENTES?

En ocasiones nos enfrentamos al siguiente problema estadístico: ¿podemos desechar, a un dado nivel de significancia, la hipótesis nula de que dos conjuntos de datos fueron generados con la misma distribución de probabilidad? Si podemos desechar la hipótesis nula habremos probado que las distribuciones son diferentes, sin embargo no desechar la hipótesis nula significa solamente que los datos pueden ser consistentes con una única distribución (¡no que efectivamente lo son!).

En esta sección discutiremos brevemente uno de los métodos para responder esa pregunta, el test chi-cuadrado. Por más información consultar la Referencia [56].

B.2.1 Chi-cuadrado

Supongamos que queremos comparar datos correspondientes a distribuciones discretas o discretizadas, de modo que podemos agrupar los resultados en intervalos. Si en el intervalo i -ésimo observamos N_i eventos y de acuerdo a alguna distribución conocida esperábamos n_i eventos, el estadístico chi-cuadrado es:

$$\chi^2 = \sum_i \frac{(N_i - n_i)^2}{n_i}, \quad (109)$$

donde la suma se extiende sobre todos los intervalos.

Un valor grande de χ^2 indica que la hipótesis nula (de que los N_i fueron obtenidos de la distribución representada por los n_i) es improbable.

Cualquier término j en el cual $0 = n_j = N_j$ debe omitirse de la suma. Un término con $n_j = 0$ y $N_j \neq 0$ da un valor infinito para χ^2 , como debería ser puesto que en este caso es imposible que los N_i hayan sido obtenidos de la distribución de los n_i .

La distribución de probabilidad chi-cuadrado, denotada $P(\chi^2|\nu)$, donde ν es un entero que representa el número de grados de libertad, es una función gamma incompleta, en efecto, se cumple:

$$P(\chi^2|\nu) = \frac{\gamma(\frac{\nu}{2}, \frac{\chi^2}{2})}{\Gamma(\frac{\nu}{2})}. \quad (110)$$

Estrictamente hablando, $P(\chi^2|\nu)$ es la probabilidad de que la suma de ν variables aleatorias con distribución normal (de media cero y varianza uno) sea menor igual a χ^2 .

Los términos de la suma del estadístico chi-cuadrado, Ecuación (109), no tienen cada uno una distribución normal. Sin embargo, si el número de intervalos es muy grande ($\gg 1$) o el número de eventos en cada intervalo es muy grande ($\gg 1$), la distribución chi-cuadrado es una buena aproximación de la distribución del estadístico chi-cuadrado, Ecuación (109), en el caso de la hipótesis nula. Es común su uso para estimar la significancia del test chi-cuadrado.

El valor de ν es igual al número de intervalos menos el número de parámetros libres que fueron ajustados en los n_i para hacerlos comparables a los N_i .

B.3 SIGNIFICANCIA

En esta sección demostraremos algunos de los resultados enunciados en el Capítulo 3 relacionados tanto a la significancia como a la significancia máxima.

Una de las cosas que nos interesaba en ese capítulo era cuantificar la probabilidad de que la DJS tomara cierto valor z o cualquier inferior y utilizamos el siguiente resultado:

$$s(z) = \text{Prob}[D_{JS} \leq z \mid \text{una distribución}] \simeq \frac{\gamma\left[\frac{\nu}{2}, N \ln(2)z\right]}{\Gamma\left(\frac{\nu}{2}\right)}. \quad (111)$$

Demostremos esa expresión siguiendo los lineamientos que Grosse y colaboradores describen en la Referencia [23].

Comenzemos reescribiendo la expresión para la DJS, Ecuación (22), de la siguiente forma:

$$\begin{aligned} D_{JS}(P^{(1)}, \dots, P^{(m)}) &= H\left(\sum_{j=1}^m \pi_j P^{(j)}\right) - \left(\sum_{j=1}^m \pi_j H(P^{(j)})\right) \\ &= -\sum_{i=1}^k p_i \log(p_i) + \sum_{j=1}^m \pi_j \sum_{i=1}^k p_i^{(j)} \log(p_i^{(j)}). \end{aligned}$$

Utilizando que $p_i = \sum_{j=1}^m \pi_j p_i^{(j)}$, tenemos que:

$$\begin{aligned} D_{JS}(P^{(1)}, \dots, P^{(m)}) &= -\sum_{i=1}^k \sum_{j=1}^m \pi_j p_i^{(j)} \log(p_i) + \sum_{j=1}^m \sum_{i=1}^k \pi_j p_i^{(j)} \log(p_i^{(j)}) \\ &= \sum_{i=1}^k \sum_{j=1}^m \pi_j p_i^{(j)} \log\left(\frac{\pi_j p_i^{(j)}}{\pi_j p_i}\right). \end{aligned}$$

Vamos a hacer un desarrollo de Taylor de esa expresión, para ello notemos que si desarrollamos hasta orden 2 la función $f(x) = x \log\left(\frac{x}{a}\right)$ alrededor de $x = a$, obtenemos:

$$x \log\left(\frac{x}{a}\right) = \frac{x-a}{\ln(2)} + \frac{(x-a)^2}{2\ln(2)a} + \mathcal{O}((x-a)^3).$$

Utilizando esa expresión con $x = \pi_j p_i^{(j)}$ y $a = \pi_j p_i$ podemos escribir:

$$D_{JS}(P^{(1)}, \dots, P^{(m)}) \simeq \sum_{i=1}^k \sum_{j=1}^m \left(\frac{\pi_j p_i^{(j)} - \pi_j p_i}{\ln(2)} + \frac{(\pi_j p_i^{(j)} - \pi_j p_i)^2}{2\ln(2)\pi_j p_i} \right).$$

Debido a que $\sum_{j=1}^m \pi_j p_i^{(j)} = p_i$ y $\sum_{j=1}^m \pi_j = 1$ el primer sumando se anula y obtenemos:

$$D_{JS}(P^{(1)}, \dots, P^{(m)}) \simeq \sum_{i=1}^k \sum_{j=1}^m \frac{(\pi_j p_i^{(j)} - \pi_j p_i)^2}{2\ln(2)\pi_j p_i}. \quad (112)$$

Por otro lado, si utilizamos el estadístico chi-cuadrado presentada en la Sección B.2 para comparar el modelo “m-distribuciones” con el modelo “una-distribución” (la hipótesis nula es que todas las distribuciones son iguales), definiendo un evento como la ocurrencia del símbolo i -ésimo en la subsecuencia j -ésima, tenemos que realizar una suma doble con:

$$N_{ij} = p_i^{(j)} n^{(j)}, \quad (113)$$

$$n_{ij} = p_i n^{(j)}, \quad (114)$$

donde $n^{(j)}$ es la longitud de la subsecuencia j -ésima y hemos utilizando una notación ligeramente generalizada de la presentada en la Sección B.2.

Una forma de entender la expresión (113) dentro del modelo de “m-distribuciones” es pensar que si observamos al azar N símbolos, $n^{(j)}$ pertenecerán a la subsecuencia j -ésima y, de esos, una fracción $p_i^{(j)}$ corresponderán al símbolo i -ésimo.

De igual modo, en la expresión (114), dentro del modelo de “una-distribución”, $n^{(j)}$ pertenecerán a la subsecuencia j -ésima pero sin importar cuál es la identidad de esta subsecuencia, la fracción de símbolos i será p_i .

Por lo tanto, podemos escribir:

$$\begin{aligned} \chi^2 &= \sum_{i=1}^k \sum_{j=1}^m \frac{(p_i^{(j)} n^{(j)} - p_i n^{(j)})^2}{p_i^{(j)} n^{(j)}} \\ &= N \sum_{i=1}^k \sum_{j=1}^m \frac{(p_i^{(j)} \pi_j - p_i \pi_j)^2}{p_i^{(j)} \pi_j}, \end{aligned} \quad (115)$$

en donde $\pi_j = \frac{n^{(j)}}{N}$.

Comparando la Ecuaciones (112) y (115), encontramos que:

$$D_{JS} \left(P^{(1)}, \dots, P^{(m)} \right) \simeq \frac{\chi^2}{2N \ln(2)}. \quad (116)$$

Es decir que hasta segundo orden la *DJS* es igual al estadístico chi-cuadrado, que sabemos que para valores asintóticamente grandes de N sigue la distribución chi-cuadrado, como discutimos en la Sección

B.2. Allí vimos que en dicho límite $P(\chi^2 | \nu) = \frac{\gamma\left(\frac{\nu}{2}, \frac{\chi^2}{2}\right)}{\Gamma\left(\frac{\nu}{2}\right)}$ da la probabilidad de que el estadístico chi-cuadrado tome un valor menor o igual que χ^2 .

A la luz de la Ecuación (116), la probabilidad de que la *DJS* tome un valor menor igual a z , lo que definimos como significancia, será:

$$s(z) = \text{Prob}[D_{JS} \leq z \mid \text{una distribución}] \simeq \frac{\gamma\left[\frac{\nu}{2}, N \ln(2)z\right]}{\Gamma\left(\frac{\nu}{2}\right)}, \quad (117)$$

con $\nu = (k-1)(m-1)$. Y eso es justamente lo que queríamos demostrar.

B.3.1 Significancia máxima

Para el método de segmentación, más que la significancia, nos interesa la significancia máxima, como vimos en la Sección 3.3.1. Ahora veremos como llegar a las Ecuaciones (48) y (51) que allí enunciamos para la significancia máxima en un caso general y para el caso de un alfabeto binario, respectivamente.

La significancia máxima no es igual a la significancia que discutimos en la sección anterior. De hecho, si el valor de la *DJS* en cada posición de la secuencia fuera independiente se cumpliría:

$$s_{\max}(z) = (s(z))^N, \quad (118)$$

pues tendríamos que tener un valor menor o igual que z en cada una de las N posiciones de la secuencia.

Claramente los valores de la *DJS* no son estadísticamente independientes, de hecho dos posiciones adyacentes son casi idénticas. Pensemos en que movemos el cursor una posición hacia la derecha, las distribuciones en la nueva posición difieren de las de la posición anterior solo en un símbolo que deja de pertenecer a la de la izquierda para pertenecer a la de la derecha.

Grosse y colaboradores [23] realizaron simulaciones Monte Carlo para estudiar el comportamiento de la s_{\max} . Generaron para un dado alfabeto de tamaño k y una dada longitud de secuencia N , un ensamble de 10^5 secuencias aleatorias independientes idénticamente distribuidas y construyeron empíricamente el valor de $\hat{s}_{\max}(z)$. Es

decir, calcularon la DJS para cada una de las posiciones de una secuencia, buscaron el valor máximo $D_{JS_{max}}$, repitieron para todas las secuencias y fueron construyendo el histograma que aproxima la distribución, del cual se pueden estimar los valores de $\hat{s}_{max}(z)$. Estudiaron valores de k entre 2 y 12, y valores de N entre 10^2 y 10^8 , y encontraron que la forma funcional de $\hat{s}_{max}(z)$ era parecida a la de la Ecuación (118), con unos ligeros cambios. Introducen una longitud efectiva N_{ef} y un factor de escala β de modo que la expresión para la significancia máxima queda:

$$s_{max}(z) = (s(\beta z))^{N_{ef}}, \quad (119)$$

donde $s(z)$ es la significancia como está dada en la Ecuación (117). Ésta es una expresión fenomenológica en la cual podemos interpretar a N_{ef} como el número efectivo de posiciones independientes y a β como un factor de escala que cuantifica el hecho de que la varianza de $D_{JS_{max}}$ se reduce por las correlaciones entre los valores de D_{JS} calculados en las distintas posiciones.

En principio tanto N_{ef} como β pueden depender tanto del número de símbolos en el alfabeto, k , como de la longitud de la secuencia, N .

Grosse et al. ajustaron los parámetros N_{ef} y β de forma de minimizar la distancia de Kolmogorov-Smirnov¹ $|\hat{s}_{max}(z) - s_{max}(z)|$ para los diferentes valores de k y N que analizaron y encontraron que:

- la expresión dada en la Ecuación (119) ajusta bien los datos de las simulaciones Monte Carlo,
- β es independiente de N ,
- fijado el número de símbolos del alfabeto, se cumple que:

$$N_{ef} = a \ln(N) + b.$$

Los valores de β , a y b que Grosse y colaboradores encontraron para alfabetos de diferentes tamaños pueden verse en la Tabla 11.

NÚMERO DE SÍMBOLOS EN EL ALFABETO (k)	a	b	β
2	2,96	-7,88	0,80
4	2,44	-6,15	0,80
12	2,32	-4,32	0,85

Tabla 11: Parámetros para la expresión fenomenológica de la significancia máxima. Extraídos de la Referencia [23].

Nosotros estamos particularmente interesados en el caso de un alfabeto binario $k = 2$, en cuyo caso tenemos que los valores de los

¹ Ver por ejemplo, la Referencia [56].

parámetros son: $\beta = 0,80$ y $N_{ef} = 2,96 \ln(N) - 7,88$. Además nos interesa el caso en que comparamos dos distribuciones $m = 2$, por lo tanto $\nu = 1$ y la expresión de la significancia máxima, Ecuación (119), se puede escribir del siguiente modo:

$$s_{\max}(z) = \left(\frac{\gamma\left(\frac{1}{2}, N \ln(2) \beta z\right)}{\Gamma\left(\frac{1}{2}\right)} \right)^{N_{ef}}.$$

Utilizando la Ecuación (108) y el hecho de que $\Gamma\left(\frac{1}{2}\right) = \sqrt{\pi}$, tenemos que:

$$s_{\max}(z) = \left(\operatorname{erf}\left(\sqrt{N \ln(2) \beta z}\right) \right)^{N_{ef}}, \quad (120)$$

que no es otra cosa que la Ecuación (51), de la Sección 3.3.1.

B.3.2 Valor crítico de la DJS

En el método de segmentación, como explicamos en la Sección 3.4, dada una secuencia de longitud N se busca el valor máximo de la DJS promedio, $\langle D_{JS} \rangle_{\max} = z$, y se calcula la significancia máxima utilizando la Ecuación (120) para compararla con el umbral s_0 elegido y decidir si se produce la segmentación.

Sin embargo, para entender el método o para visualizar un caso concreto (como se hizo por ejemplo en la Sección 4.3.2) resulta útil invertir la expresión de la Ecuación (120) de modo de obtener el valor crítico de segmentación para la DJS promedio.

Esto significa considerar, dada una longitud de secuencia N , cuál es el valor de z tal que la significancia es igual al valor del umbral, s_0 .

Invertiendo la Ecuación (120), se obtiene:

$$z = \frac{\left(\operatorname{erf}^{-1}\left(s_0^{\frac{1}{N_{ef}}}\right) \right)^2}{N \ln(2) \beta}, \quad (121)$$

donde erf^{-1} es la inversa de la función error.

El gráfico de esta expresión para el caso de un alfabeto binario (esto es, usando los parámetros que se detallaron en la sección previa) para un valor de umbral de $s_0 = 0,999$ (el que se utiliza en el Capítulo 4 para la detección de shock en la dinámica de KIF1A) se puede ver en la Figura 42.

Queda claro que el valor crítico de la DJS depende de la longitud de la secuencia bajo consideración y es más grande para longitudes de secuencia más cortas. Esto es consistente con el hecho de que mientras más corta es la secuencia mayores son las fluctuaciones que esperamos y por lo tanto mayor tendrá que ser la disimilitud entre las subsecuencias para que sea significativa.

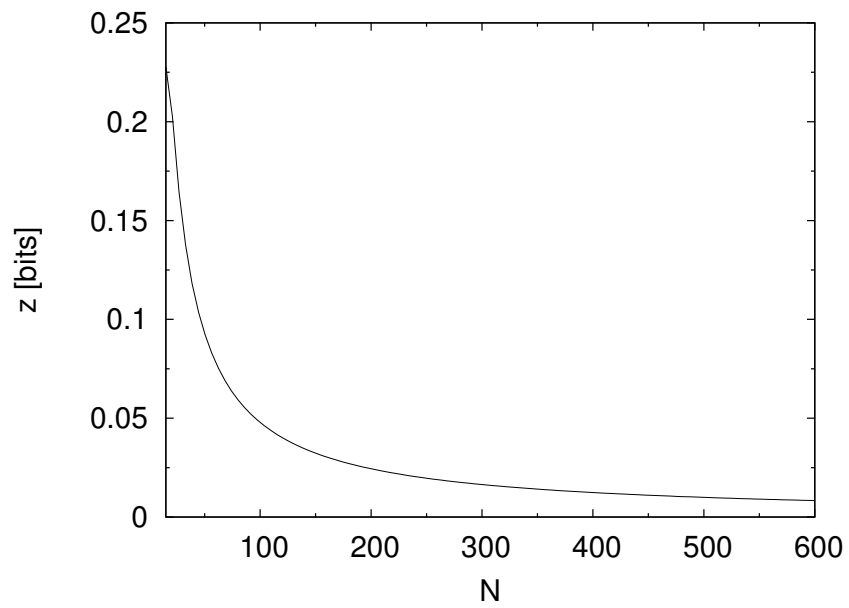


Figura 42: Valor crítico para la [DJS](#) en función de la longitud de secuencia para un alfabeto binario y un umbral $s_0 = 0,999$. Notar que el valor crítico es mayor cuando la longitud de la secuencia es más corta.

BREVE INTRODUCCIÓN A LA CÉLULA

The whole of biology is a counterpoint between the two themes: astonishing variety in individual particulars; astonishing constancy in fundamental mechanisms^{1,2}.

Este apéndice es una versión libre y caprichosamente resumida de los Capítulos 1, 2 y 3 del libro “Molecular Biology of the Cell” de Alberts *et al.* [2]. Ese completísimo libro es la referencia por excelencia de biología molecular.

La intención de este apéndice es la de proveer una revisión breve de algunas cuestiones básicas de la célula de modo de situar en contexto la parte biológica de este trabajo de tesis y familiarizar al lector con terminología que es inevitable utilizar al discutir nuestros resultados sobre el motor molecular KIF1A.

Además de constituir una introducción quizá logre, por qué no, estimular al lector a acercarse a algunas de las abrumadoramente numerosas pero maravillosas 1600 páginas del “Alberts”.

C.1 CÉLULAS

Todas las cosas vivas están hechas de células y todas estas unidades de materia viva comparten la misma maquinaria para sus funciones más básicas.

En todos los casos, así discutamos acerca de una bacteria o del agregado de más de 10^{13} células que forman el cuerpo humano, todo el organismo ha sido generado por divisiones celulares de una única célula. La célula es, por lo tanto, el vehículo para la información hereditaria que define la especie.

Todas las células vivas de la Tierra almacenan su información hereditaria en moléculas de ADN: dos cadenas poliméricas emparejadas que están formadas por los mismos cuatro tipos de monómeros: A, T, C, G.

En un proceso llamado *transcripción* segmentos de la secuencia de ADN se usan como plantilla para guiar la síntesis de un tipo diferentes de moléculas, que además son más cortas, moléculas de ARN. Luego, en un proceso más complejo llamado *traducción* muchas de estas moléculas de ARN sirven para la síntesis de polímeros muy diferentes químicamente: las proteínas.

¹ Toda la biología es un contrapunto entre dos temas: una asombrosa variedad en los individuos; una asombrosa constancia en los mecanismos fundamentales.

² Extraído de la Referencia [2], página 3.

Las *proteínas* son cadenas poliméricas (sin ramificaciones) formadas colocando monómeros uno detrás de otro.

Hay veinte tipos de monómeros con los cuales se construyen las proteínas de todas las células vivas, los *aminoácidos*. Cada aminoácido se construye alrededor de la misma estructura central, a través de la cuál puede unirse de una manera estándar a cualquier otro aminoácido. Ligado a este núcleo hay un grupo secundario que le da a cada aminoácido un carácter químico distintivo.

Cada una de las proteínas creada juntando aminoácidos en una secuencia particular, se pliega en una forma tridimensional precisa.

Estos polímeros de aminoácidos se ligan con alta especificidad a otras moléculas y actúan como *enzimas* que catalizan reacciones en las cuales se forman y rompen ligaduras covalentes. De esta forma dirigen la gran mayoría de los procesos químicos que ocurren en la célula.

Las proteínas tienen una gran cantidad de otras funciones: mantener estructuras, generar movimientos, censar señales, etcétera. Cada proteína realiza una función específica dependiendo de su propia secuencia de aminoácidos, la cual queda completamente especificada por su información genética.

Las proteínas, por sobre todo, son las moléculas que ponen la información genética de la célula en acción.

C.1.1 *Procariotas y eucariotas*

Los organismos pueden clasificarse de acuerdo a su estructura celular en dos grupos: *eucariotas* y *procariotas*.

Las eucariotas mantienen su **ADN** en un compartimiento intracelular recubierto de membrana llamado *núcleo*. Las procariotas, en cambio, no tienen un compartimiento especial para almacenar su **ADN**.

La mayoría de las procariotas son pequeñas y simples en apariencia y la mayoría vive como individuos independientes más que como organismos multicelulares. Las células eucariotas son, en general, más grandes y más elaboradas que las células procariotas y su genoma es más grande y elaborado también. El mayor tamaño está acompañado por grandes diferencias en la estructura y función celular. Más aún, muchas células eucariotas forman organismos multicelulares que alcanzan grados de complejidad no comparables a ninguna célula procariota.

Las células eucariotas son típicamente 10 veces más grandes que las procariotas en dimensión lineal (1000 veces en volumen). Tienen un *citoesqueleto*, que es un sistema de filamentos proteicos que atraviesan el citoplasma y forma, junto a un gran número de proteínas que se adhieren a él, un sistema de vigas, cuerdas y motores que le dan a la célula fortaleza mecánica, permiten controlar su forma y guían su movimiento.

El núcleo es uno de un conjunto de membranas internas, cada una estructuralmente similar a la membrana plasmática y encerrando diferentes tipos de espacios dentro de la célula, muchos de los cuales están involucrados en procesos relacionados a la digestión y a la secreción.

Casi todas las células eucariotas contienen *mitocondrias*. Éstos son pequeños compartimientos en el citoplasma, encerrados por una doble capa de membrana, que consumen oxígeno y obtienen energía de la oxidación de moléculas como los azúcares para producir la mayoría del ATP que es utilizado por la célula para realizar sus distintas actividades. Las mitocondrias tienen su propio genoma y su propia maquinaria diferente de aquella del resto de la célula eucariota.

Un esquema con las principales componentes de una célula eucariota se puede ver en la Figura 43.

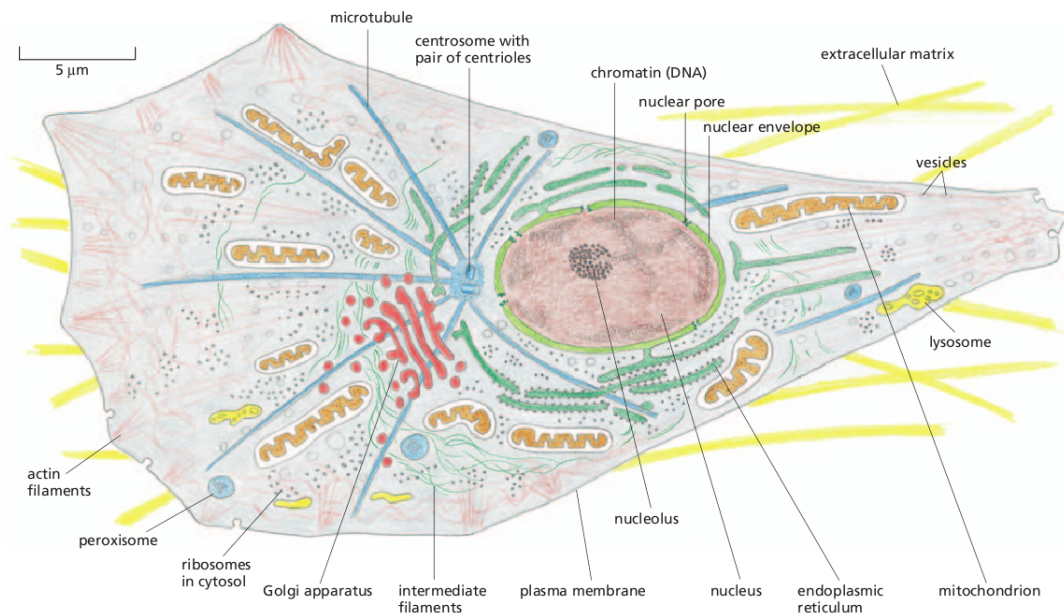


Figura 43: Principales características de una célula eucariota. Se pueden ver, entre otras cosas, la red de filamentos que constituye el citoesqueleto y diversos compartimientos internos, como el núcleo y las mitocondrias. Figura extraída de la Referencia [2].

c.1.2 La vida requiere energía libre

Una célula viva es un sistema fuera del equilibrio químico: tiene una gran cantidad de energía libre interna, lo que quiere decir que si la célula muere y decae al equilibrio químico, se libera una gran cantidad de energía, en forma de calor, al ambiente. Para que la célula haga una nueva célula a su propia imagen, debe tomar del ambiente

no sólo materia prima sino también energía libre, de modo de realizar las reacciones necesarias.

Se necesita energía libre para la propagación de información y hay una relación cuantitativa precisa entre ambas cantidades. Especificar un bit de información cuesta una cantidad de energía libre definida dependiendo de su temperatura.

C.1.2.1 Orden en la célula

Una propiedad por sobre todas las demás hace que las cosas vivas sean milagrosamente diferentes de la materia no animada: crean y mantienen orden, en un universo que está siempre tendiendo al desorden. Para crear este orden, las células de un organismo vivo deben realizar un sinnúmero de reacciones químicas (millones de reacciones cada segundo en cada célula).

Las células vivas, para sobrevivir, crecer y formar organismos complejos, generan orden y por lo tanto, parecen desafiar la segunda ley de la termodinámica, pero una célula no es un sistema aislado: obtiene energía en la forma de alimento o a través de fotones provenientes del sol (o incluso, como es el caso de algunas bacterias quimiosintéticas, de compuestos inorgánicos solamente) y usa esta energía para generar orden dentro de sí misma. En el transcurso de las reacciones químicas que generan orden, parte de la energía que la célula usa se convierte en calor que se libera al ambiente celular, desordenándolo de modo que la entropía total aumenta.

C.2 NUCLEÓTIDOS

Un *nucleótido* es una molécula constituida por un anillo que contiene nitrógeno unido a un azúcar con cinco carbonos. Este azúcar puede ser ribosa o desoxirribosa y puede tener uno o más grupos fosfato. Los anillos con nitrógeno se denominan *bases*. Las diferentes bases son A, T, C, G y U. El nucleótido más conocido es la molécula de [ADN](#).

C.2.1 Hidrólisis de ATP

Algunos nucleótidos pueden actuar como portadores de energía química a corto plazo. Por sobre todos se destaca el [ATP](#), que es utilizado para transferir energía en cientos de reacciones celulares diferentes.

Las moléculas de [ATP](#) se forman en reacciones que son llevadas adelante a partir de la energía liberada por el rompimiento oxidativo de alimentos, como se puede ver en la [Figura 44](#).

La molécula de [ATP](#) posee tres grupos fosfatos unidos en serie. La ruptura de estos enlaces libera grandes cantidades de energía utilizable. La situación más común es que el grupo fosfato terminal sea

separado por hidrólisis, transfiriendo el fosfato a otra molécula y liberando la energía para conducir alguna reacción biosintética que requiera energía. Este proceso se denomina *hidrólisis* porque una molécula de agua se agrega en la reacción, como se puede ver en la Figura 44.

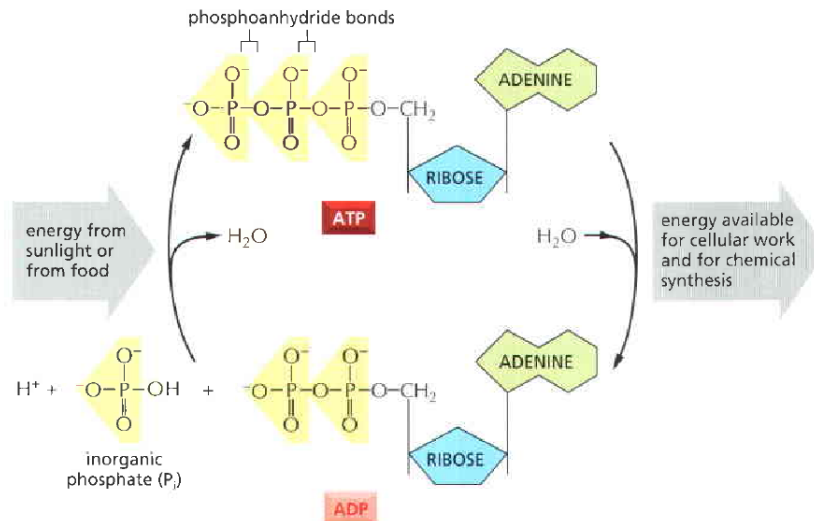


Figura 44: La molécula de **ATP** como portadora de energía en la célula. El **ATP** se sintetiza en una reacción de fosforilación energéticamente desfavorable en el cual un grupo fosfato se agrega a una molécula de **ADP**. Para que esto suceda la célula acopla esta reacción a una reacción energéticamente favorable como es la oxidación de alimentos. Cuando se lo requiere, la molécula de **ATP** entrega un paquete de energía a través de la reacción energéticamente favorable de la hidrólisis, en la cual resultan como productos una molécula de **ADP** y un fosfato inorgánico. Figura extraída de la Referencia [2].

C.3 PROTEÍNAS

Las proteínas son polímeros de aminoácidos unidos de forma ordenada (cabeza-con-cola) en una larga cadena que después se pliega en una estructura tridimensional. Esta estructura es única para cada tipo de proteína.

Cinco de los veinte aminoácidos tienen cadenas secundarias que pueden formar iones en solución y por lo tanto, pueden tener carga eléctrica. Los restantes no tienen carga eléctrica, algunos son polares e hidrofílicos, otros apolares e hidrofóbicos.

Las proteínas son especialmente abundantes y versátiles. Realizan miles de funciones diferentes en las células. Muchas sirven como enzimas que catalizan un gran número de reacciones que la célula necesita. Otras proteínas son utilizadas para construir componentes estruc-

turales, por ejemplo la *tubulina*, una proteína que se autoensambla para formar los *microtúbulos* de las células. Otras proteínas actúan como motores moleculares que producen fuerza y movimiento, como es el caso de la miosina y la kinesina.

C.3.1 Plegamiento de proteínas

Cada tipo de proteína tiene una única secuencia de aminoácidos (exactamente la misma para todas las moléculas). La forma tridimensional de la proteína queda especificada por su secuencia de aminoácidos. El plegamiento se da a partir de muchos conjuntos diferentes de ligaduras no covalentes actuando juntas. Cada proteína normalmente se pliega en una única conformación estable. Sin embargo, la conformación cambia ligeramente cuando la proteína interactúa con otras moléculas de la célula.

Las proteínas consisten en general de entre 50 y 2000 aminoácidos. Las proteínas más grandes consisten de varios *dominios*, cada dominio es una unidad estructural que se pliega de forma más o menos independiente del resto.

La *hélice- α* y la *hoja- β* son formas recurrentes de plegamiento. Esto se debe a que resultan de ligaduras que no involucran las cadenas secundarias sino sólo el núcleo de los aminoácidos, que, como dijimos es el mismo para todos.

Una *hoja- β* se puede formar por cadenas vecinas paralelas o por una única cadena que se pliega de ida y vuelta sobre sí misma (anti-paralela). Ambos tipos producen una estructura muy rígida.

Una *hélice- α* se genera cuando una cadena se tuerce sobre sí misma para formar un cilindro rígido. En algunas proteínas dos (o en algunos casos tres) *hélices- α* se enroscan para formar una estructura particularmente estable conocida como *hélice superenrollada*. Muchas proteínas elongadas tienen su base estructural en largas regiones tipo varilla de *hélices superenrolladas*. Éste es el caso de las kinesinas en general y de KIF1A en particular.

La cadena polimérica generalmente va y vuelve a lo largo de los dominios completos, haciendo giros agudos sólo en la superficie de la proteínas. Son estas protusiones en forma de lazo que se forman en las regiones donde la cadena gira las que conforman las regiones en las cuales la proteína tiene sitios de ligadura para otras moléculas.

Una vez que una proteína ha evolucionado para plegarse en una conformación estable con propiedades útiles, su estructura puede ser modificada durante la evolución para permitirle realizar nuevas funciones. Este tipo de eventos ha ocurrido de forma frecuente en el pasado, como resultado de esto, muchas proteínas actuales pueden agruparse en *familias*. Cada miembro de una familia tiene una secuencia de aminoácidos y una conformación tridimensional que se parece a los de los otros miembros de la familia.

c.3.2 *Interacciones de las proteínas*

Las proteínas no son trozos rígidos de material, pueden tener partes móviles cuyas acciones mecánicas se acoplan a eventos químicos. Estos acoplamientos subyacen a todos los procesos dinámicos que ocurren en las células vivas.

Las proteínas se ligan a otras moléculas. En algunos casos esta ligadura es fuerte, en otra es débil y de corta duración. Pero las ligaduras siempre muestran una alta especificidad: cada proteína puede ligar una o unas pocas moléculas de los miles de tipos con los que se puede encontrar. La substancia que se liga a la superficie de la proteína se conoce como *ligando*.

La habilidad de la proteína de ligarse selectivamente y con gran afinidad a un determinado ligando depende de la formación de un conjunto de enlaces débiles no covalentes (ligaduras puente hidrógeno, iónicas y de van der Waals) en conjunto con interacciones hidrofóbicas.

Cada ligadura es tan débil que se requieren muchas actuando de forma simultánea, por lo tanto, el contorno de la superficie del ligando necesita encajar muy bien con la proteína.

La región de la proteína que se asocia con un ligando se conoce como *sitio de ligadura* del ligando y normalmente consiste de una cavidad en la superficie de la proteína formada por un arreglo particular de los aminoácidos.

Regiones separadas de la superficie de la proteína proveen generalmente sitios de ligadura para ligandos diferentes.

c.3.2.1 *Enlaces covalentes y no covalentes*

Un enlace covalente típico es 100 veces más fuerte que la energía térmica, por lo tanto son resistentes a ser separados por movimiento térmico y normalmente se rompen sólo durante reacciones químicas específicas en las que participan otros átomos y moléculas.

La generación o rotura de enlaces covalentes son eventos violentos y en las células vivas son cuidadosamente controlados por catalistas altamente específicos llamadas *enzimas*.

Los enlaces que no son covalentes son, en general, mucho más débiles y son importantes en la célula en muchas situaciones en las cuales las moléculas se asocian y disocian rápidamente para llevar a cabo su función.

c.3.3 *Proteínas motoras*

Hay proteínas cuya función principal es la de mover otras moléculas. Estas *proteínas motoras* generan la fuerza responsable de la contracción muscular y del movimiento de las células.

Las proteínas motoras también son las responsables de los movimientos a menor escala que se dan en el interior celular: ayudan a mover cromosomas hacia lados opuestos de la célula durante la mitosis, mueven organelas a lo largo de las sendas moleculares en el interior celular y mueven enzimas a lo largo de las hebras de ADN durante la síntesis de una molécula de ADN.

Todos estos procesos fundamentales para la célula dependen de proteínas con partes móviles que operan como máquinas generadoras de fuerza.

Como estos procesos requieren movimiento dirigido y las proteínas involucradas hacen trabajo las leyes de la termodinámica exigen que tales movimientos utilicen alguna fuente de energía libre (de otro modo podrían constituir un móvil perpetuo).

Estas máquinas moleculares operan a través de una serie de cambios conformacionales. Para que este ciclo proceda en una única dirección alcanza con que uno de los pasos ocurra de forma irreversible.

La mayoría de las proteínas que caminan en una dirección por grandes distancias tienen acoplado uno de los cambios conformacionales a la hidrólisis de una molécula de ATP ligada a su superficie. Como una gran cantidad de energía se libera al hidrolizar una molécula de ATP es muy improbable que la proteína realice el proceso inverso necesario para moverse hacia atrás, puesto que esto requiere que también se invierta la hidrólisis de ATP (agregando un fosfato a una molécula de ADP para formar un ATP).

Muchas proteínas motoras generan movimiento direccionado de esta forma, incluyendo la proteína motora de los músculos, la *miosina*, que camina a lo largo de los *microfilamentos* para generar la contracción muscular, y la proteína *kinesina* que camina a lo largo de los *microtúbulos* transportando cargas.

En este apéndice se encuentran cuestiones accesorias al modelo de KIF_{1A} discutido en el Capítulo 4 y a la simulación que lo implementa. En la Sección D.1 se profundiza sobre los resultados experimentales que establecen los mecanismos de motilidad de KIF_{1A} y se discute la relación de los valores de los parámetros del modelo con las cantidades controlables experimentalmente y con el mecanismo *ratchet*. En la Sección D.2 se reinterpreta el modelo en término del estado de ocupación de cada sitio del filamento, presentando de esta forma el enfoque utilizado para desarrollar la simulación. En la Sección D.3 se explicitan los códigos fuentes de la simulación. Por último en la Sección D.4 se discute acerca de formas alternativas de construir las secuencias binarias con las que se trabaja.

D.1 PARÁMETROS DEL MODELO PARA LA DINÁMICA DE KIF_{1A}

En la Sección 4.2 se presentó el modelo utilizado para la dinámica de KIF_{1A}. Dicho modelo fue introducido por Nishinari *et al.* [47] y Greulich *et al.* [22] y está basado en la evidencia experimental acerca de los mecanismos de motilidad de KIF_{1A} acumulada a lo largo de varios años en el laboratorio de Hirokawa, que se puede encontrar en las Referencias [49, 50, 51, 48].

En la Referencia [49], Okada *et al.*, muestran que KIF_{1A} es un motor procesivo que se mueve distancias del orden de los micrómetros. También encuentran que el movimiento es más “ruidoso” que el de la kinesina convencional, esto es, que no sólo avanza hacia el extremo (+) sino que a menudo da pasos hacia sitios anteriores y que los datos se pueden ajustar por un movimiento browniano con deriva. Sugieren que la interacción con el MT es mediada por una atracción electrostática, a través del “lazo-K” (*K-loop*, en inglés) que es una parte de las proteínas motoras que está conservada en los monómeros de la subfamilia KIF₁ y no está presente en la kinesina convencional (que es un dímero).

En la Referencia [50], a través de un uso extensivo de mutantes¹, Okada *et al.* establecen que la interacción de KIF_{1A} con el filamento en el estado débil se produce a través del “lazo-K” que tiene residuos con carga positiva y que se relaciona con el “gancho-E” (*E-hook*, en inglés) una región flexible del MT con carga negativa. Además en-

¹ Esto es, proteínas similares a KIF_{1A} cuya secuencia genética ha sido modificada para cambiar alguna parte del polímero que constituye la proteína.

cuentran que esta región participa de la interacción sólo durante el estado débil.

En la Referencia [51], Okada *et al.* muestran, utilizando técnicas de molécula individual (una pinza óptica) que el motor KIF1A hidroliza una molécula de ATP por cada paso y analizan la distribución de tamaño de los pasos (da sobre múltiplos de 8nm que es la periodicidad del MT) y establecen detalles cuantitativos del movimiento del motor.

En la Referencia [48], utilizando cristalografía de rayos-X, Nitta *et al.* resuelven la estructura atómica de KIF1A en los diferentes estados y muestran que utiliza dos lazos diferentes, de forma alternada, para relacionarse con el MT, uno en el estado fuerte y otro en el estado débil. La alternancia entre estados está acoplada al ciclo de hidrólisis, y esto da lugar a los cambios conformacionales en la proteína que permiten esta alternancia.

Resumiendo, los puntos más importantes de dichos resultados experimentales son que:

- KIF1A es una enzima y en cada ciclo hidroliza una molécula de ATP, siendo los productos de la hidrólisis ADP y fosfato inorgánico. Cada ciclo bioquímico del motor consiste en 4 estados:
 1. kinesina “desnuda” (K)
 2. kinesina ligada con ATP (KT)
 3. kinesina ligada con ADP y fosfato (KDP)
 4. kinesina ligada con ADP (KD), luego de liberar el fosfato
- La magnitud de la ligadura de la kinesina con el filamento depende de la etapa del ciclo de hidrólisis en que se encuentra. En los estados K y KT la ligadura es muy fuerte. Luego de la hidrólisis se pasa al estado KDP que tiene un tiempo de vida muy corto (enseguida se libera el fosfato) lo que lleva al estado KD, en este estado la ligadura al filamento es débil. Luego de liberar el ADP (volviendo al estado K) la ligadura con el filamento vuelve a ser fuerte, ligándose al sitio más cercano del MT.
- En el estado KD, el motor permanece “anclado” al MT por una atracción electrostática, esto impide que el motor se aleje del filamento, limitando su movimiento a una difusión unidimensional en la dirección paralela al MT. En cambio, en los estados de ligadura fuerte el motor no se mueve del sitio de ligadura (salvo cuando avanza de forma activa).

Estos resultados experimentales para el ciclo bioquímico de los motores KIF1A sugieren que es posible una descripción simplificada en término de un modelo de dos estados para describir las propiedades de transporte de estos motores. Se distingue un estado de ligadura fuerte con el filamento (el que llamamos estado 1), que incluye los

estados K y KT, y un estado de ligadura débil (estado 2), que corresponde al estado KT. KDP tiene un tiempo de vida muy corto por lo que su clasificación no es importante.

D.1.1 Valores de los parámetros del modelo

La elección de los valores de los parámetros para el modelo de KIF1A la hacemos siguiendo las Referencias [47, 22]:

- Cantidad de sitios en la red: $L = 600$.

La periodicidad del MT, que es la distancia entre sitios es 8nm , así $600 \cdot 8\text{nm} = 4800\text{nm} = 4,8\mu\text{m}$ que es del orden de los tamaños típicos de las células eucariotas.

- Tasa de ligadura: $\omega_a = 10^7 C_{\text{KIF1A}} \left[\frac{1}{\text{M}\cdot\text{s}} \right]$, donde C_{KIF1A} es la concentración de motores KIF1A en el citosol.

En células eucariotas *in vivo*, típicamente se tiene:

$$10\text{nM} \leq C_{\text{KIF1A}} \leq 1000\text{nM},$$

por lo tanto:

$$0,0001 \frac{1}{\text{ms}} \leq \omega_a \leq 0,01 \frac{1}{\text{ms}}.$$

NOTA: la unidad M se denomina *Molaridad* y es de amplio uso en biología; es el número de moles de soluto por litro de solvente (en general se toma como agua).

- Tasa de desligadura: $\omega_d = 0,0001 \frac{1}{\text{ms}}$.

Es independiente del número de motores, tiene que ver con la procesividad típica de cada superfamilia $\omega_d \sim \frac{1}{\tau}$ donde τ es el tiempo medio que pasa un motor ligado al filamento.

- Tasa de saltos por difusión: $\omega_b = 0,6 \left[\frac{1}{\text{ms}} \right]$.

Tiene que ver con el tiempo t que necesita el motor para difundir una distancia igual al parámetro de red, si D es el coeficiente de difusión:

$$\omega_b \sim \frac{1}{t} \sim \frac{D}{(8\text{nm})^2}.$$

- Tasa de permanencia: $\omega_s = 0,145 \left[\frac{1}{\text{ms}} \right]$.

- Tasa de avance activo: $\omega_f = 0,055 \left[\frac{1}{\text{ms}} \right]$.

Los valores de estas últimas dos tasas salen de las observaciones experimentales de que:

$$\frac{\omega_f}{\omega_s} \simeq \frac{3}{8}, \quad \omega_s + \omega_f \simeq 0,2 \frac{1}{\text{ms}}.$$

- Tasa de hidrólisis: los datos experimentales sugieren una cinética de tipo Michaelis-Menten², es decir:

$$\frac{1}{\omega_h} = \left[4 + 9 \left(\frac{0,1\text{mM}}{C_{\text{ATP}}} \right) \right] \text{ms},$$

por lo tanto, el rango biológicamente relevante para este parámetro es:

$$0 \leq \omega_h \leq 0,25 \frac{1}{\text{ms}}.$$

D.1.2 Relación con los procesos físicos del mecanismo ratchet

Las tasas pueden interpretarse en término de los procesos físicos correspondientes a un mecanismo *ratchet*, como en los modelos de *flashing-ratchets* [32, 58].

En estos modelos el motor se encuentra en un potencial efectivo dependiente del tiempo que, en un ciclo de hidrólisis cambia entre:

1. un potencial periódico asimétrico con forma de diente de sierra,
2. una constante o potencial plano.

La tasa de hidrólisis ω_h , corresponde a la tasa de transición del potencial en su forma 1 a su forma 2. Esta transición ocurre apenas se produce la hidrólisis de ATP, mientras que la transición de 2 a 1 ocurre cuando una molécula de ATP se liga a una kinesina desnuda.

La tasa ω_b , a la cual el motor salta a cualquiera de los vecinos cuando se encuentra en el estado 2, captura la dinámica del movimiento browniano del motor sujeto al potencial constante en su forma 2.

Las tasas ω_s y ω_f son proporcionales al solapamiento de la distribución de probabilidad gaussiana de la partícula libre, en el instante en que el potencial cambia de 2 a 1, con el pozo original y el pozo siguiente, respectivamente.

Así la imagen que podemos hacernos a la luz de los modelos tipo *flashing-ratchets* es la siguiente: supongamos que el motor se encuentra ligado a uno de los sitios del filamento en el estado 1, es decir en el fondo de uno de los pozos del potencial diente de sierra. Por lo tanto, está descrito por una distribución deltiforme centrada en dicha posición. Eventualmente una molécula de ATP se liga al motor y cuando se produce la hidrólisis y se libera el fosfato, el potencial pasa a ser plano. El motor se encuentra ahora en el estado 2, desligado del filamento pero anclado al mismo a través de una atracción electrostática y comienza a difundir en una dimensión, paralela al MT. Apenas se produce la transición del potencial, la distribución deltiforme comienza a ensancharse como en cualquier proceso difusivo [29]. En este ensanchamiento se convierte en una distribución gaussiana

² Por más información, ver por ejemplo la Referencia [29].

cuyo ancho es mayor mientras mayor es el tiempo transcurrido. Después de un tiempo, cuando se libera el [ADP](#), el potencial vuelve a ser diente de sierra, el motor pasa nuevamente al estado fuerte avanzando hacia el sitio siguiente o permaneciendo en el mismo de acuerdo al solapamiento de la distribución con el potencial diente de sierra.

D.2 NÚCLEO DE LA SIMULACIÓN

La simulación está pensada en torno a la actualización del estado de ocupación de los sitios del filamento, por lo tanto puede ser esclarecedor reescribir la dinámica del sistema en un lenguaje más cercano a la simulación misma.

Llamaremos $E(i, t)$ al estado del sitio i a tiempo t . Así, $E(i, t)$ podrá tomar los valores 0, 1 ó 2. Si el sitio i -ésimo está desocupado a tiempo t , toma el valor 0; si hay un motor ligado en estado 1, toma el valor 1; y si hay un motor ligado en estado 2, toma el valor 2.

Entonces, actualizar el sitio i -ésimo significa saber el estado de ocupación de dicho sitio y de sus primeros vecinos, a partir de esto, definir cuáles son sus posibilidades de actualización y sortear un número aleatorio para decidir cuál de todas las posibilidades efectivamente ocurre. A continuación se describirán de manera exhaustiva todas las distintas combinaciones que pueden ocurrir y cuáles son las posibilidades de actualización.

D.2.1 Sitio del bulk

Las reglas de evolución para la dinámica de un sitio que pertenece al *bulk* se pueden establecer de la siguiente manera:

1. Si $E(i, t) = 0$, el sitio i -ésimo se encuentra desocupado a tiempo t y entonces puede:
 - Ligarse un motor con probabilidad $\omega_a dt$
 $\implies E(i, t + dt) = 1.$
 - Permanecer igual con probabilidad $(1 - \omega_a dt)$
 $\implies E(i, t + dt) = 0.$
2. Si $E(i, t) = 1$ el sitio i -ésimo tiene un motor ligado en estado 1 y entonces puede:
 - Desligarse el motor con probabilidad $\omega_d dt$
 $\implies E(i, t + dt) = 0.$
 - Producirse la hidrólisis de [ATP](#) con probabilidad $\omega_h dt$
 $\implies E(i, t + dt) = 2.$
 - Permanecer igual con probabilidad $[1 - (\omega_h + \omega_d) dt]$
 $\implies E(i, t + dt) = 1.$

3. Si $E(i, t) = 2$, el sitio i -ésimo tiene un motor ligado en estado 2 entonces debe considerarse también el estado de ocupación de los vecinos, a saber

i) Caso 020: significa que $E(i-1, t) = 0$ y $E(i+1, t) = 0$ y puede:

- Liberar el ADP y permanecer en el mismo sitio con probabilidad $\omega_s dt$

$$\implies E(i-1, t+dt) = 0, E(i, t+dt) = 1, E(i+1, t+dt) = 0.$$

- Avanzar activamente con probabilidad $\omega_f dt$

$$\implies E(i-1, t+dt) = 0, E(i, t+dt) = 0, E(i+1, t+dt) = 1.$$

- Difundir a la izquierda con probabilidad $\omega_b dt$

$$\implies E(i-1, t+dt) = 2, E(i, t+dt) = 0, E(i+1, t+dt) = 0.$$

- Difundir a la derecha con probabilidad $\omega_b dt$

$$\implies E(i-1, t+dt) = 0, E(i, t+dt) = 0, E(i+1, t+dt) = 2.$$

- Permanecer igual con probabilidad $[1 - (\omega_s + \omega_f + 2\omega_b)dt]$

$$\implies E(i-1, t+dt) = 0, E(i, t+dt) = 2, E(i+1, t+dt) = 0.$$

ii) Caso X20: significa que $E(i-1, t) = 1$ ó $E(i-1, t) = 2$, y $E(i+1, t) = 0$ y puede:

- Liberar el ADP y permanecer en el mismo sitio con probabilidad $\omega_s dt$

$$\implies E(i-1, t+dt) = X, E(i, t+dt) = 1, E(i+1, t+dt) = 0.$$

- Avanzar activamente con probabilidad $\omega_f dt$

$$\implies E(i-1, t+dt) = X, E(i, t+dt) = 0, E(i+1, t+dt) = 1.$$

- Difundir a la derecha con probabilidad $\omega_b dt$

$$\implies E(i-1, t+dt) = X, E(i, t+dt) = 0, E(i+1, t+dt) = 2.$$

- Permanecer igual con probabilidad $[1 - (\omega_s + \omega_f + \omega_b)dt]$

$$\implies E(i-1, t+dt) = X, E(i, t+dt) = 2, E(i+1, t+dt) = 0.$$

iii) Caso 02X: significa que $E(i-1, t) = 0$ y $E(i+1, t) = 1$ ó $E(i+1, t) = 2$ y puede:

- Liberar el ADP y permanecer en el mismo sitio con probabilidad $\omega_s dt$

$$\implies E(i-1, t+dt) = 0, E(i, t+dt) = 1, E(i+1, t+dt) = X.$$

- Difundir a la izquierda con probabilidad $\omega_b dt$

$$\implies E(i-1, t+dt) = 2, E(i, t+dt) = 0, E(i+1, t+dt) = X.$$

- Permanecer igual con probabilidad $[1 - (\omega_s + \omega_b) dt]$

$$\implies E(i-1, t+dt) = 0, E(i, t+dt) = 2, E(i+1, t+dt) = X.$$

iv) Caso X2X: significa que $E(i-1, t) = 1$ ó $E(i-1, t) = 2$ y $E(i+1, t) = 1$ ó $E(i+1, t) = 2$ y puede:

- Liberar el ADP y permanecer en el mismo sitio con probabilidad $\omega_s dt$

$$\implies E(i-1, t+dt) = X, E(i, t+dt) = 1, E(i+1, t+dt) = X.$$

- Permanecer igual con probabilidad $(1 - \omega_s dt)$

$$\implies E(i-1, t+dt) = X, E(i, t+dt) = 2, E(i+1, t+dt) = X.$$

D.2.2 Sitios en los extremos

Los extremos deben considerarse separadamente pues no son equivalentes a un sitio cualquiera del *bulk*.

D.2.2.1 Extremo (-)

Reglas de evolución para la dinámica del extremo (-):

1. Si $E(1, t) = 0$, el sitio 1 está desocupado y puede:
 - Ligarse un motor con probabilidad αdt

$$\implies E(1, t+dt) = 1.$$
 - Permanecer igual con probabilidad $(1 - \alpha dt)$

$$\implies E(1, t+dt) = 0.$$
2. Si $E(1, t) = 1$, el sitio 1 tiene ligado un motor en estado 1 y puede:
 - Desligarse el motor con probabilidad $\gamma_1 dt$

$$\implies E(1, t+dt) = 0.$$
 - Producirse la hidrólisis de ATP con probabilidad $\omega_h dt$

$$\implies E(1, t+dt) = 2.$$

- Permanecer igual con probabilidad $[1 - (\gamma_1 + \omega_h)dt]$
 $\implies E(1, t + dt) = 1$
3. Si $E(1, t) = 2$, el sitio 1 tiene ligado un motor en estado 2 y resulta necesario considerar también el estado de ocupación del sitio 2:
- i) Caso 20: significa $E(2, t) = 0$ y puede:
- Difundir a la izquierda (soltándose del filamento) con probabilidad $\gamma_2 dt$
 $\implies E(1, t + dt) = 0, E(2, t + dt) = 0.$
 - Difundir a la derecha con probabilidad $\omega_b dt$
 $\implies E(1, t + dt) = 0, E(2, t + dt) = 2.$
 - Liberar el ADP y permanecer en el sitio con probabilidad $\omega_s dt$
 $\implies E(1, t + dt) = 1, E(2, t + dt) = 0.$
 - Avanzar activamente con probabilidad $\omega_f dt$
 $\implies E(1, t + dt) = 0, E(2, t + dt) = 1.$
 - Permanecer igual con probabilidad $[1 - (\gamma_2 + \omega_b + \omega_s + \omega_f)dt]$
 $\implies E(1, t + dt) = 2, E(2, t + dt) = 0.$
- ii) Caso 2X: significa $E(2, t) = 1$ ó $E(2, t) = 2$ y puede:
- Difundir a la izquierda (soltándose del filamento) con probabilidad $\gamma_2 dt$
 $\implies E(1, t + dt) = 0, E(2, t + dt) = X.$
 - Liberar el ADP y permanecer en el sitio con probabilidad $\omega_s dt$
 $\implies E(1, t + dt) = 1, E(2, t + dt) = X.$
 - Permanecer igual con probabilidad $[1 - (\gamma_2 + \omega_s)dt]$
 $\implies E(1, t + dt) = 2, E(2, t + dt) = X.$

D.2.2.2 Extremo (+)

Reglas de evolución para la dinámica del extremo (+):

1. Si $E(L, t) = 0$, el sitio L está desocupado y puede:
- Ligarse un motor con probabilidad δdt
 $\implies E(L, t + dt) = 1.$
 - Permanecer igual con probabilidad $(1 - \delta dt)$
 $\implies E(L, t + dt) = 0.$
2. Si $E(L, t) = 1$, el sitio L tiene ligado un motor en estado 1 y puede:

- Desligarse el motor con probabilidad $\beta_1 dt$
 $\implies E(L, t + dt) = 0.$
 - Producirse la hidrólisis de ATP con probabilidad $\omega_h dt$
 $\implies E(L, t + dt) = 2.$
 - Permanecer igual con probabilidad $[1 - (\beta_1 + \omega_h) dt]$
 $\implies E(L, t + dt) = 1.$
3. Si $E(L, t) = 2$, el sitio L tiene ligado un motor en estado 2 y resulta necesario considerar también el estado de ocupación del sitio L - 1:
- i) Caso 02: significa $E(L - 1, t) = 0$ y puede:
- Difundir a la derecha (soltándose del filamento) con probabilidad $\beta_2 dt$
 $\implies E(L - 1, t + dt) = 0, E(L, t + dt) = 0.$
 - Difundir a la izquierda con probabilidad $\omega_b dt$
 $\implies E(L - 1, t + dt) = 2, E(L, t + dt) = 0.$
 - Liberar el ADP y permanecer en el sitio con probabilidad $\omega_s dt$
 $\implies E(L - 1, t + dt) = 0, E(L, t + dt) = 1.$
 - Permanecer igual con probabilidad $[1 - (\beta_2 + \omega_b + \omega_s) dt]$
 $\implies E(L - 1, t + dt) = 0, E(L, t + dt) = 2.$
- ii) Caso X2: significa $E(L - 1, t) = 1$ ó $E(L - 1, t) = 2$ y puede:
- Difundir a la derecha (soltándose del filamento) con probabilidad $\beta_2 dt$
 $\implies E(L - 1, t + dt) = X, E(L, t + dt) = 0.$
 - Liberar el ADP y permanecer en el sitio con probabilidad $\omega_s dt$
 $\implies E(L - 1, t + dt) = X, E(L, t + dt) = 1.$
 - Permanecer igual con probabilidad $[1 - (\beta_2 + \omega_s) dt]$
 $\implies E(L, t + dt) = X, E(L, t + dt) = 2.$

D.3 CÓDIGO FUENTE PARA EL MODELO DE KIF1A

El programa para simular la dinámica de KIF1A fue desarrollado desde cero utilizando Fortran 90. El programa fue desarrollado de forma modular y está contenido en los siguientes archivos de código:

1. constantes.f90 : Es un módulo en el que se definen las variables más importantes del programa (las que conviene tener presentes y las que uno quiere modificar).

2. opciones.f90 : Consiste en cuatro subrutinas que cubren las diferentes alternativas para un sitio del *bulk* que se encuentra en estado 2. Las diferentes posibilidades surgen de los distintos estados de ocupación posibles de los primeros vecinos. Se diferenciaron los casos en que: i) los dos sitios vecinos están desocupados: opción "o2o"; ii) el sitio anterior está ocupado (en cualquiera de los estados 1 ó 2) y, el sitio posterior, desocupado: opción "X2o", iii) el sitio anterior está desocupado y, el posterior, ocupado: opción "o2X", iv) ambos sitios vecinos están ocupados: opción "X2X".
3. extremos.f90 : Contiene dos subrutinas con las posibilidades de actualización para los sitios de los extremos.
4. main.f90 : Es el archivo principal, contiene la estructura del programa. Se pueden elegir varios detalles, entre ellos el número total de MCSs, el número de pasos que va a contener la imagen, etcétera.

El programa hace uso de un generador de números pseudo-aleatorios. Se utilizó el generador Mersenne-Twister³ que fue extraído de:

<http://www.math.sci.hiroshima-u.ac.jp/~m-mat/MT/VERSIONS/FORTRAN/mtfort90.f>

D.3.1 Código fuente constantes.f90

```

MODULE constantes !fig3d
INTEGER, PARAMETER :: L = 600
REAL(8), PARAMETER :: wd = 0.000071d0
REAL(8), PARAMETER :: wb = 0.4260d0
REAL(8), PARAMETER :: ws = 0.102950d0
REAL(8), PARAMETER :: wf = 0.03905d0
REAL(8), PARAMETER :: wa = 0.0000071d0
REAL(8), PARAMETER :: wh = 0.071d0
REAL(8), PARAMETER :: c = 1.0d0
REAL(8), PARAMETER :: alpha = wa
REAL(8), PARAMETER :: betha1 = wd
REAL(8), PARAMETER :: betha2 = wd
REAL(8), PARAMETER :: gamma1 = 0.d0
REAL(8), PARAMETER :: gamma2 = 0.d0
REAL(8), PARAMETER :: delta = 0.d0
END MODULE

```

NOTA: los valores aquí detallados corresponden a las probabilidades, no a las tasas de transición, es decir, son adimensionales y ya tienen una elección concreta del intervalo de tiempo. Corresponden

³ También se utilizaron otros generadores para chequear los resultados.

a un caso particular pero carecen de importancia aquí los valores numéricos que se muestran.

D.3.2 Código fuente opciones. f90

```

MODULE opciones_mod
USE constantes
USE mtmod, ONLY: PRNG => grnd
IMPLICIT NONE

CONTAINS

SUBROUTINE opcion1(estado, buffer)                                !020
IMPLICIT NONE
INTEGER :: estado(1:L)
INTEGER :: buffer
REAL(8) :: aleatorio

aleatorio = PRNG()
IF (aleatorio < wb) THEN
    ! se mueve a la izquierda por difusion
    estado(buffer-1) = 2
    estado(buffer) = 0
ELSEIF ( aleatorio < wb+wb ) THEN
    ! se mueve a la derecha por difusion
    estado(buffer) = 0
    estado(buffer+1) = 2
ELSEIF ( aleatorio < wb+wb+ws) THEN
    ! hidroliza pero permanece en el sitio, entonces pasa al
    estado 1
    estado(buffer) = 1
ELSEIF ( aleatorio < wb+wb+ws+wf ) THEN
    ! hidroliza y utiliza la energia para avanzar
    estado(buffer) = 0
    estado(buffer+1) = 1
! si no se cumple nada de eso entonces se queda todo igual
END IF

END SUBROUTINE

!*****

SUBROUTINE opcion2(estado, buffer)                                !X20
IMPLICIT NONE
INTEGER :: estado(1:L)
INTEGER :: buffer
REAL(8) :: aleatorio

aleatorio = PRNG()
IF (aleatorio < wb) THEN
    ! se mueve a la derecha por difusion

```

```

        estado(buffer) = 0
        estado(buffer+1) = 2
ELSEIF ( aleatorio < wb+ws) THEN
    ! hidroliza pero permanece en el sitio, entonces pasa al
    estado 1
    estado(buffer) = 1
ELSEIF ( aleatorio < wb+ws+wf ) THEN
    ! hidroliza y utiliza la energia para avanzar
    estado(buffer) = 0
    estado(buffer+1) = 1
! si no se cumple nada de eso entonces se queda todo igual
END IF

END SUBROUTINE

!*****

SUBROUTINE opcion3(estado, buffer)          !02X
IMPLICIT NONE
INTEGER :: estado(1:L)
INTEGER :: buffer
REAL(8) :: aleatorio

aleatorio = PRNG()
IF (aleatorio < wb) THEN
    ! se mueve a la izquierda por difusion
    estado(buffer-1) = 2
    estado(buffer) = 0
ELSEIF ( aleatorio < wb+ws+(1-c)*wf) THEN
    ! hidroliza pero permanece en el sitio, ya sea porque
    quiere (ws) como porque el sitio siguiente esta
    ocupado (wf)
    estado(buffer) = 1
! si no se cumple nada de eso entonces se queda todo igual
END IF

END SUBROUTINE

!*****

SUBROUTINE opcion4(estado, buffer)          ! X2X
IMPLICIT NONE
INTEGER :: estado(1:L)
INTEGER :: buffer
REAL(8) :: aleatorio

aleatorio = PRNG()
IF ( aleatorio < ws+(1-c)*wf) THEN
    ! hidroliza pero permanece en el sitio, ya sea porque
    quiere (ws) como porque el sitio siguiente esta
    ocupado (wf)
    estado(buffer) = 1

```



```

! si no se cumple eso entonces se queda todo igual
END IF

END SUBROUTINE

END MODULE opciones_mod

```

D.3.3 Código fuente extremos.f90

```

MODULE extremos_mod
USE mtmod, ONLY: PRNG => grnd
USE constantes
IMPLICIT NONE

CONTAINS
SUBROUTINE extremo_izq(estado, buffer)
IMPLICIT NONE

INTEGER :: estado(1:L)
INTEGER :: buffer
REAL(8) :: aleatorio

aleatorio = PRNG()

SELECT CASE (estado(buffer))
CASE(0) ! cuando el sitio esta en el estado 0
  IF (aleatorio<alpha) estado(buffer)=1
  ! sino queda como esta (en el estado 0)
CASE(1) ! sitio en estado 1
  IF (aleatorio < gamma1) THEN
    estado(buffer) = 0
  ELSEIF (aleatorio < (gamma1+wh) ) THEN
    estado(buffer) = 2
  ! sino queda como esta (en el estado 1)
  ENDIF
CASE(2)
! solo hay vecino a la derecha
  IF (estado(buffer+1) == 0 ) THEN
    ! 20 *****
    IF (aleatorio < gamma2) THEN
      ! se puede 'ir' del MT por difusion (a la
      izquierda: a la nada!)
      estado(buffer) = 0
    ELSEIF (aleatorio < gamma2+wb) THEN
      ! se mueve a la derecha por difusion
      estado(buffer) = 0
      estado(buffer+1) = 2
    ELSEIF (aleatorio < gamma2+wb+ws) THEN
      ! hidroliza pero permanece en el sitio,
      pasa al estado 1
      estado(buffer) = 1
    
```

```

        ELSEIF (aleatorio < gamma2+wb+ws+wf) THEN
            ! hidroliza y utiliza la energia para
              avanzar
            estado(buffer) = 0
            estado(buffer+1) = 1
            ! si no se cumple nada de eso entonces se queda
              todo igual
        END IF
        ! *****
ELSE
        ! 2X *****
        IF (aleatorio < gamma2) THEN
            ! se mueve a la izquierda por difusion
              saliendo del filamento
            estado(buffer) = 0
        ELSEIF ( aleatorio < gamma2+ws+(1-c)*wf) THEN
            ! hidroliza pero permanece en el sitio,
              ya sea porque 'quiere' (ws) como
              porque el sitio siguiente esta
              ocupado (wf)
            estado(buffer) = 1
            ! si no se cumple nada de eso entonces qda todo
              igual
        END IF
        ! *****
        END IF
CASE DEFAULT
        WRITE(*,*) 'El estado del sitio debe ser 0, 1 o 2. Algo
          anda mal, revisar!'
        STOP
END SELECT

END SUBROUTINE extremo_izq

!*****

SUBROUTINE extremo_der(estado, buffer)
IMPLICIT NONE

INTEGER :: estado(1:L)
INTEGER :: buffer
REAL(8) :: aleatorio

aleatorio = PRNG()
SELECT CASE (estado(buffer))
CASE(0) ! cuando el sitio esta en el estado 0
        IF (aleatorio<delta) estado(buffer)=1
          ! sino queda como esta (en el estado 0)
CASE(1) ! sitio en estado 1
        IF (aleatorio < betha1) THEN
            estado(buffer) = 0
        ELSEIF (aleatorio < (betha1+wh) ) THEN

```

```

        estado(buffer) = 2
    ! sino queda como esta (en el estado 1)
    ENDIF
CASE(2)
    ! recordar q solo hay vecino a la izquierda
    IF (estado(buffer-1) == 0 ) THEN
        ! 02 *****
        IF (aleatorio < betha2) THEN
            ! se puede 'ir' del MT por difusion (a la
            ! derecha: a la nada!)
            estado(buffer) = 0
        ELSEIF (aleatorio < betha2+wb) THEN
            ! se mueve a la izquierda por difusion
            estado(buffer) = 0
            estado(buffer-1) = 2
        ELSEIF (aleatorio < betha2+wb+ws) THEN
            ! hidroliza pero permanece en el sitio (
            ! porque 'quiere', no porque este
            ! ocupado), entonces pasa al estado 1
            estado(buffer) = 1
        ! si no se cumple nada de eso entonces se queda
        ! todo igual
        END IF
        ! *****
    ELSE
        ! X2 *****
        IF (aleatorio < betha2) THEN
            ! se puede 'ir' del MT por difusion (a la
            ! derecha: a la nada!)
            estado(buffer) = 0
        ELSEIF (aleatorio < betha2+ws) THEN
            ! hidroliza pero permanece en el sitio (
            ! porque 'quiere', no porque este
            ! ocupado), entonces pasa al estado 1
            estado(buffer) = 1
        ! si no se cumple nada de eso entonces se queda
        ! todo igual
        END IF
        ! *****
    END IF
CASE DEFAULT
    WRITE(*,*) 'El estado del sitio debe ser 0, 1 o 2. Algo
    anda mal, revisar!'
    STOP
END SELECT

END SUBROUTINE extremo_der

END MODULE extremos_mod

```

D.3.4 Código fuente main.f90

```

PROGRAM motores
USE constantes
!USE aleatorio_mod , ONLY: Rand0, Rand
USE mtmod, ONLY: sgrnd, PRNG => grnd
USE extremos_mod ,ONLY: extremo_izq, extremo_der
USE opciones_mod, ONLY: opcion1, opcion2, opcion3, opcion4

IMPLICIT NONE
INTEGER                :: sitio(1:L) , estado(1:L)
! 'sitio' p/saber cuales actualice
! 'estados': 0 (sin motor), 1 (mot. en est 1) y 2 (mot. en est 2)
INTEGER                :: ii , aleat_entero, buffer, MCS
, MCStotal, kk, mm, nn, semilla !, cero, uno, dos
REAL(8)                :: aleatorio
CHARACTER(30)          :: filename = "3d/imagen.ppm" , filename2
INTEGER, PARAMETER     :: n_importan = 100

MCStotal = 200000

!CALL Rand0      ! inicializa el generador de numeros aleatorios

!*****
! Uso el generador de PseudoAleatorios Mersenne-Twister
! aca lo inicializo
! pasandole la semilla por teclado (archivo <semilla.in)
!*****
READ(*,*) semilla
!WRITE(*,*) semilla
CALL sgrnd(semilla)
!*****

!*****
! valores iniciales. Ningun motor ligado inicialmente
!*****
DO ii=1,L
    sitio(ii) = ii
    estado(ii) = 0
END DO

!*****
! para sacar el dibujito
!*****
OPEN(UNIT=9 , FILE = filename , ACTION = "write", STATUS = "new")
WRITE(9,'(A2)') "P3"
! le dice que es un ppm
! sin formato explicito queda espacio adelante y Xnview se queja
WRITE(9,'(I3,X,I6)') L, n_importan
! bits horizontales y verticales ! OJO EL FORMATO
WRITE(9,'(I3)') 256
! RGB de 256 colores, van de 0 a 255

```

```

!*****
!
kk=-1
mm=1
!nn=1

MCStep: DO MCS= 1, MCStotal
  filamento:DO ii = 1 , L      ! un bucle es un 'dt' (todos
    los sitios tuvieron oportunidad de actualizarse)

    aleat_entero = INT(PRNG)*(L-ii+1) + 1
    ! sitio a actualizar
    buffer = sitio(aleat_entero)
    !reordeno (al final pongo los ya visitados, para
      no actualizarlos otra vez en el mismo MCS)
    sitio(aleat_entero) = sitio(L-ii+1)
    sitio(L-ii+1) = buffer

    IF (buffer>1 .AND. buffer<L) THEN
      ! vale para el bulk nomas
      SELECT CASE (estado(buffer))
      CASE(0) ! sitio en estado 0
        aleatorio = PRNG()
        !WRITE(*,*) wa, aleatorio
        IF(aleatorio<wa) estado(buffer)=1
        ! sino queda como esta
      CASE(1) ! sitio en estado 1
        aleatorio = PRNG()
        IF(aleatorio<wd) THEN
          estado(buffer) = 0
        ELSEIF(aleatorio<(wd+wh)) THEN
          estado(buffer) = 2
        ! sino queda como esta
        ENDIF
      CASE(2)
        IF(estado(buffer-1)==0) THEN
          IF(estado(buffer+1)==0) THEN
            CALL opcion1(estado, buffer)
            !020
          ELSE
            CALL opcion3(estado, buffer)
            !02X
          END IF
        ELSE
          IF(estado(buffer+1)==0) THEN
            CALL opcion2(estado, buffer)
            !X20
          ELSE
            CALL opcion4(estado,
              buffer) !X2X
          END IF
        END IF
      END IF
    END IF
  END DO
END

```

```

                                END IF

                                CASE DEFAULT
                                    WRITE(*,*) 'El estado del sitio
                                                debe ser 0, 1 o 2. Algo anda
                                                mal, revisar!'
                                    STOP
                                END SELECT
ELSEIF (buffer==1) THEN
    ! extremo izquierdo
    CALL extremo_izq(estado, buffer)
ELSEIF (buffer==L) THEN
    ! extremo derecho
    CALL extremo_der(estado, buffer)
END IF
END DO filamento

IF (MCS>(MCStotal-n_importan)) THEN
    !cero = 0
    !uno = 0
    !dos = 0

    kk=kk+1

    filename2 = "3d/secuencia"//CHAR(mm+47)//CHAR(kk+48)//".
                dat"
    ! sirve cuando saco hasta 100
    !filename2 = "SB7/secuencia"//CHAR(mm+47)//CHAR(kk+48)
                //" .dat" !//CHAR(n+47)

    IF (kk==9) THEN
        kk=-1
        mm=mm+1
        !IF (mm==11) THEN
            !     mm=1
            !     nn=nn+1
        !ENDIF
    ENDIF

    !*****
    !para la secuencia
    !
    OPEN(UNIT=11,FILE=filename2,ACTION="write",STATUS="new")
    !*****

    DO ii=1,L
        IF (estado(ii)==0) THEN
            !cero=cero+1
            WRITE(11,'(I1)') 0
            WRITE(9,'(I3,X,I3,X,I3)') 255, 255, 255
            !estado 0: blanco
        ELSEIF (estado(ii)==1) THEN

```

```

                !uno=uno+1
                WRITE(11, '(I1)') 1
                WRITE(9, '(I3,X,I3,X,I3)') 0, 0 , 255
                !estado 1 : azul
            ELSEIF (estado(ii)==2) THEN
                ! elseif por control
                !dos=dos+1
                WRITE(11, '(I1)') 1
                !1 si el sitio esta ocupado
                WRITE(9, '(I3,X,I3,X,I3)') 255, 0 , 0
                !estado 2 : rojo
            ELSE
                WRITE(*,*) "algo va mal"
                STOP
            ENDIF
        END DO
    CLOSE(11)

    !WRITE(*,*) "MCStep: " , MCS
    !WRITE(*,*) " 0, 1, 2 " , cero , uno , dos
    !WRITE(*,*) "600?: " , (cero+uno+dos)
END IF

END DO MCStep

CLOSE(9)

END PROGRAM

```

D.4 FORMAS ALTERNATIVAS PARA CONSTRUIR LAS SECUENCIAS

Como discutimos en la Sección 4.2.2.3, el propósito detrás de la construcción de las secuencias simbólicas es que permitan identificar si se produjo un *shock* en el filamento y en caso afirmativo localizar su posición. En la mencionada sección se explicó la forma en que esto se llevó a cabo en todos los casos mostrados en la tesis. Aquí presentaremos, a la luz de un ejemplo concreto, dos alternativas para la construcción de secuencias que también fueron exploradas.

MAPEO UTILIZADO: el carácter *i*-ésimo de la secuencia será un 0 si el sitio de ligadura *i*-ésimo del filamento estaba vacío en ese tiempo, y será un 1 si el sitio estaba ocupado (en cualquiera de los dos estados).

ALTERNATIVA 1: el carácter *i*-ésimo de la secuencia será un 0 si, en ese tiempo, el sitio de ligadura *i*-ésimo del filamento estaba vacío u ocupado con un motor en estado 2, y será un 1 si el sitio estaba ocupado con un motor en estado 1.

ALTERNATIVA 2: el caracter i -ésimo de la secuencia será un 0 si, en ese tiempo, el sitio de ligadura i -ésimo del filamento estaba vacío u ocupado con un motor en estado 1, y será un 1 si el sitio estaba ocupado con un motor en estado 2.

Los resultados para un caso particular, en el cual se utilizaron los parámetros que se detallan en la Tabla 12, se muestran en las Figuras 45, 46 y 47. Para los resultados que se muestran en las tres figuras se utilizaron no sólo los mismos parámetros sino también la misma realización de la simulación.

$\omega_a \left[\frac{1}{\text{ms}} \right]$	$\omega_d \left[\frac{1}{\text{ms}} \right]$	$\omega_h \left[\frac{1}{\text{ms}} \right]$	$\omega_s \left[\frac{1}{\text{ms}} \right]$	$\omega_f \left[\frac{1}{\text{ms}} \right]$	$\omega_b \left[\frac{1}{\text{ms}} \right]$
0.00001	0.0001	0.2	0.145	0.055	0.6

Tabla 12: Parámetros del modelo para el caso de la Figura 45. El número de sitios de la red es $L = 600$ y los parámetros correspondientes a los extremos se utilizan de la siguiente forma: $\beta_1 = \beta_2 = \omega_d$, $\gamma_1 = \gamma_2 = \delta = 0$ y $\alpha = \omega_a$.

En la Figura 45 se muestra el método funcionando a la manera estándar (como se utilizó en el resto de la tesis), en la Figura 46 se muestra los resultados obtenidos mapeando las secuencias siguiendo la alternativa 1, y en la Figura 47 los que se obtuvieron siguiendo la alternativa 2.

En los dos mapeos alternativos los resultados obtenidos son equivalentes a los del caso estándar. De hecho, para este caso se obtuvo $\langle D_{JS_{\max}} \rangle = 0,2781$ en la posición $x_{\max} = 531$, calculando la significancia máxima allí obtuvimos $s_{\max}(N = 600, \langle D_{JS_{\max}} \rangle = 0,2781) = 1 > 0,999$.

Para la alternativa 1, se obtuvo un valor máximo $\langle D_{JS_{\max}} \rangle = 0,08653$ en la posición $x_{\max} = 531$, que da lugar a una significancia máxima $s_{\max}(N = 600, \langle D_{JS_{\max}} \rangle = 0,08653) = 1 > 0,999$.

Por último, para la alternativa 2, se obtuvo $\langle D_{JS_{\max}} \rangle = 0,1366$ en la posición $x_{\max} = 530$, que da lugar a una significancia máxima $s_{\max}(N = 600, \langle D_{JS_{\max}} \rangle = 0,1366) = 1 > 0,999$.

Como el caso que aquí se discute, se corresponde con el de la Figura 22(a), de acuerdo a lo que se puede ver en la Tabla 5, el error asociado es de 12 sitios. Por lo tanto, en las tres formas de construir las secuencias, el *shock* se localizo en posiciones indistinguibles.

Los mapeos alternativos aquí presentados son equivalentes conceptualmente a considerar las densidades de ocupación de los motores en los dos estados internos de forma separada. La alternativa 1 sería como considerar sólo la densidad de ocupación en el estado 1 y la alternativa 2 como considerar sólo la densidad de ocupación en el estado 2, mientras que el caso estándar es equivalente a considerar la densidad total.

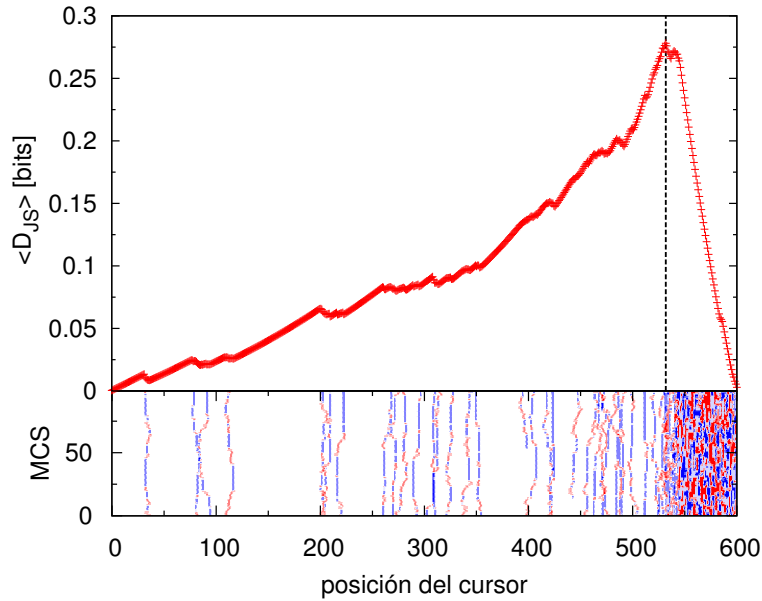


Figura 45: Valores de la *DJS* promedio para cada posición del cursor correspondientes a un caso en el cual el estado estacionario presenta coexistencia de fases. Los parámetros del modelo para este caso se explicitan en la Tabla 12. Las secuencias fueron construidas en la forma en que se utilizó a lo largo de la tesis, ver texto principal para más detalles. La línea de puntos vertical corresponde a la posición 531 en la cual se detectó un *shock*. En la parte inferior de la figura se puede ver un gráfico espacial en el que cada fila representa el estado del filamento en un dado instante temporal. Se muestran cien MCSs una vez que el sistema alcanzó el estado estacionario. Un punto blanco representa un sitio vacío para ese tiempo, un punto azul, un sitio ocupado con un motor en estado 1 y un punto rojo, un sitio ocupado con un motor en estado 2.

Los resultados son consistentes con el hecho de que en un modelo con dos estados internos, si se produce un *shock* en ambas componentes, la posición debe coincidir porque de otro modo el perfil global tendría dos *shocks*.

Para la combinación de parámetros elegida los resultados son equivalentes para las tres formas de construir las secuencias, pero la detección cuando se utiliza sólo uno de los estados internos es más ruidosa y no es robusta. Los valores de la *DJS* son menores cuando se considera sólo un estado interno, en comparación a cuando se consideran ambos (Figuras 46 y 47, en comparación a la Figura 45). Si bien en el ejemplo que presentamos, esto no impide la detección, es relevante en situaciones en las que la coexistencia de fases no es tan clara. En particular, en un caso donde la mayoría de los sitios estuvieren ocupados con motores en un mismo estado una de las alternativas no detectaría el *shock*.

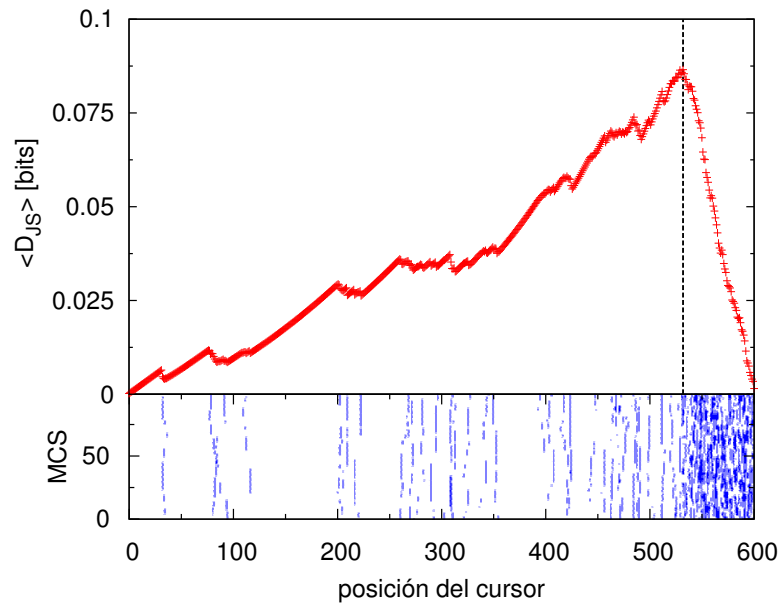


Figura 46: Valores de la DJS promedio para cada posición del cursor correspondientes a un caso en el cual el estado estacionario presenta coexistencia de fases. Los parámetros del modelo para este caso se explicitan en la Tabla 12. Las secuencias fueron construidas en la forma de la alternativa 1, ver texto principal para más detalles. La línea de puntos vertical corresponde a la posición 531 en la cual se detectó un *shock*. En la parte inferior de la figura se puede ver un gráfico espacial en el que cada fila representa el estado del filamento en un dado instante temporal. Se muestran cien MCS una vez que el sistema alcanzó el estado estacionario. Un punto blanco representa o bien un sitio vacío o un sitio ocupado con un motor en estado 2, y un punto azul, un sitio ocupado con un motor en estado 1.

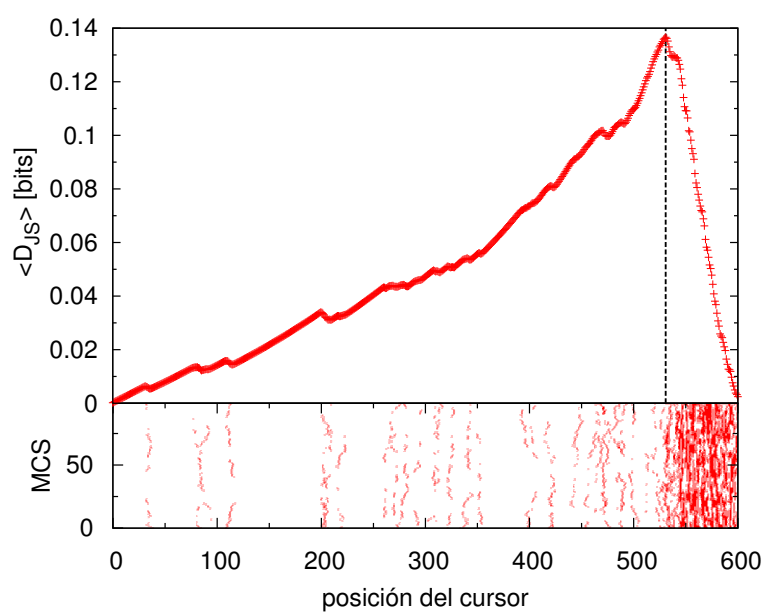


Figura 47: Valores de la DJS promedio para cada posición del cursor correspondientes a un caso en el cual el estado estacionario presenta coexistencia de fases. Los parámetros del modelo para este caso se explicitan en la Tabla 12. Las secuencias fueron construidas en la forma de la alternativa 2, ver texto principal para más detalles. La línea de puntos vertical corresponde a la posición 581 en la cual se detectó un *shock*. En la parte inferior de la figura se puede ver un gráfico espacial en el que cada fila representa el estado del filamento en un dado instante temporal. Se muestran cien MCS una vez que el sistema alcanzó el estado estacionario. Un punto blanco representa un sitio vacío o un sitio ocupado con un motor en estado 1, y un punto rojo, un sitio ocupado con un motor en estado 2.

SOBRE LA INTERACCIÓN KIF₁A-TAU

E.1 CÓDIGO FUENTE PARA EL MODELO DE KIF₁A EN PRESENCIA DE TAU

El programa para simular la dinámica de KIF₁A en presencia de tau fue desarrollado utilizando Fortran 90. Se modificaron los códigos desarrollados para implementar el modelo de KIF₁A sin tau, que se pueden ver en la Sección D.3. Toda la estructura de actualización de los sitios fue modificada pero el programa conserva la forma modular original en que fue desarrollado, agregando un módulo para generar la distribución de tau en el filamento. El programa está contenido en los siguientes archivos de código:

1. `constantes.f90` : es un módulo en el que se definen las variables más importantes del programa (las que conviene tener presentes y las que uno quiere modificar).
2. `concentracion.f90` : es un módulo que genera la distribución de tau en el filamento.
3. `opciones.f90` : consiste en cuatro subrutinas que cubren las diferentes alternativas para un sitio del *bulk* que se encuentra en estado 2. Las diferentes posibilidades surgen de los distintos estados de ocupación posibles de los primeros vecinos. Se diferenciaron los casos en que: i) los dos sitios vecinos están desocupados: opción "o2o"; ii) el sitio anterior está ocupado (en cualquiera de los estados 1 ó 2) y, el sitio posterior, desocupado: opción "X2o", iii) el sitio anterior está desocupado y, el posterior, ocupado: opción "o2X", iv) ambos sitios vecinos están ocupados: opción "X2X".
4. `extremos.f90` : contiene dos subrutinas con las posibilidades de actualización para los sitios de los extremos.
5. `main.f90` : es el archivo principal, contiene la estructura del programa. Se pueden elegir varios detalles, entre ellos el número total de MCSs, el número de pasos que va a contener la imagen, etcétera.

El programa hace uso de un generador de números pseudo-aleatorios. Se utilizó el generador Mersenne-Twister¹ que fue extraído de:

<http://www.math.sci.hiroshima-u.ac.jp/~m-mat/MT/VERSIONS/FORTRAN/mtfort90.f>

¹ También se utilizaron otros generadores para chequear los resultados.

E.1.1 *Código fuente constantes.f90*

```

MODULE constantes
  IMPLICIT NONE
!
  INTEGER, PARAMETER          :: L = 600
  REAL(KIND(0.d0)), PARAMETER :: wd = 0.000071d0
  REAL(KIND(0.d0)), PARAMETER :: wb = 0.4260d0
  REAL(KIND(0.d0)), PARAMETER :: ws = 0.102950d0
  REAL(KIND(0.d0)), PARAMETER :: wf = 0.03905d0
  REAL(KIND(0.d0)), PARAMETER :: wa = 0.0000071d0
  REAL(KIND(0.d0)), PARAMETER :: wh = 0.071d0
  REAL(KIND(0.d0)), PARAMETER :: c = 1.d0
  REAL(KIND(0.d0)), PARAMETER :: alpha = wa
  REAL(KIND(0.d0)), PARAMETER :: betha1 = wd
  REAL(KIND(0.d0)), PARAMETER :: betha2 = wd
  REAL(KIND(0.d0)), PARAMETER :: gamma1 = 0.d0
  REAL(KIND(0.d0)), PARAMETER :: gamma2 = 0.d0
  REAL(KIND(0.d0)), PARAMETER :: delta = 0.d0
  REAL(KIND(0.d0)), PARAMETER :: Cmin=0.1d0, Cmax=0.1d0
    ! concentraciones minima y maxima de tau
  INTEGER, PARAMETER          :: n_segmentos = 6
    ! debe ser divisor de L
  REAL(KIND(0.d0)), PARAMETER :: prob_tau_wa = 0.67d0
    ! prob. de que tau impida la ligadura
  REAL(KIND(0.d0)), PARAMETER :: prob_tau_mov = 0.5d0
    ! prob. de que tau impida el movimiento
END MODULE

```

NOTA: los valores aquí detallados corresponden a las probabilidades, no a las tasas de transición, es decir, son adimensionales y ya tienen una elección concreta del intervalo de tiempo. Corresponden a un caso particular pero carecen de importancia aquí los valores numéricos que se muestran.

E.1.2 *Código fuente concentracion.f90*

```

MODULE concentracion_mod
  USE constantes
  USE mtmod, ONLY: sgrnd, PRNG => grnd
  IMPLICIT NONE
!
CONTAINS
SUBROUTINE DistrTau(tau)
  IMPLICIT NONE
!
  INTEGER, INTENT(OUT) :: tau(1:L)
  INTEGER :: semilla, segmento, ubicacion, i, j
! 'L' y 'n_segmentos' se definen en constates.f90

```

```

INTEGER :: long_segmento = L/n_segmentos ! ojo
REAL(KIND(0.d0)) :: auxiliar, aleatorio, p
!
tau(:)=0
!
auxiliar= DBLE(long_segmento)*0.5d0
DO i=1,n_segmentos
  p = Cmin ! estoy usando constante C=Cmin=Cmax
  DO j=1,long_segmento
    ubicacion=long_segmento*(i-1)+j
    aleatorio=PRNG()
    IF(aleatorio<p) THEN
      tau(ubicacion)=1
    ENDIF
  ENDDO
  auxiliar = auxiliar + long_segmento
ENDDO
!
END SUBROUTINE DistrTau
END MODULE concentracion_mod

```

E.1.3 Código fuente opciones.f90

```

MODULE opciones_tau_mod
  USE constantes
  USE mtmod, ONLY: PRNG => grnd
  IMPLICIT NONE
!
CONTAINS
!
SUBROUTINE opcion1(estado, buffer, tau, tiempo, tiempo_todos,
  contador, MCS, tiempo_control, tiempo_todos_cuad, long, long_
  todos, veloc_todos) !020
  IMPLICIT NONE
!
  INTEGER, INTENT(INOUT) :: estado(1:L)
  INTEGER, INTENT(IN) :: tau(1:L)
  INTEGER, INTENT(IN) :: buffer, MCS
  REAL(KIND(0.d0)) :: aleatorio, aleatorio_tau
  INTEGER, INTENT(INOUT) :: tiempo(1:L), tiempo_control(1:L), long
    (1:L)
  INTEGER, INTENT(INOUT) :: tiempo_todos, contador, long_todos
  INTEGER*8, INTENT(INOUT) :: tiempo_todos_cuad
  REAL(KIND(0.d0)),INTENT(INOUT) :: veloc_todos
  REAL(KIND(0.d0)) :: auxiliar
!
  aleatorio = PRNG()
!
  IF (aleatorio < wb) THEN
    ! se quiere mover a la izquierda por difusion
    IF(tau(buffer-1)==0) THEN

```

```

estado(buffer-1) = 2
estado(buffer) = 0
! retrocede, calculo para el tiempo de ligadura <-
IF(tiempo_control(buffer)==MCS-1) THEN
    tiempo(buffer-1) = tiempo(buffer) + 1
    tiempo_control(buffer-1) = MCS
ELSE
    tiempo(buffer-1) = tiempo(buffer)
    tiempo_control(buffer-1) = MCS
ENDIF
long(buffer-1) = long(buffer)-1
! <-
ELSE
aleatorio_tau=PRNG()
IF(aleatorio_tau<prob_tau_mov) THEN
    !tau impide el movimiento, el motor se desliga
    estado(buffer)=0
    ! se desliga, acumulo tiempo de ligadura <-
    tiempo_todos = tiempo_todos + tiempo(buffer) + 1
    contador = contador + 1
    tiempo_todos_cuad = tiempo_todos_cuad + (tiempo(buffer)
        + 1)**2
    long_todos = long_todos + long(buffer)
    auxiliar = DBLE(long(buffer))/DBLE(tiempo(buffer)+1)
    veloc_todos = veloc_todos + auxiliar
ELSE
    ! tau no impide el movimiento
    estado(buffer-1) = 2
    estado(buffer) = 0
    ! retrocede, calculo para el tiempo de ligadura <-
    IF(tiempo_control(buffer)==MCS-1) THEN
        tiempo(buffer-1) = tiempo(buffer) + 1
        tiempo_control(buffer-1) = MCS
    ELSE
        tiempo(buffer-1) = tiempo(buffer)
        tiempo_control(buffer-1) = MCS
    ENDIF
    long(buffer-1) = long(buffer)-1
    ! <-
ENDIF
ENDIF
ELSEIF ( aleatorio < wb+wb ) THEN
    ! se quiere mover a la derecha por difusion
    IF(tau(buffer+1)==0) THEN
        estado(buffer) = 0
        estado(buffer+1) = 2
        ! avanza, calculo para el tiempo de ligadura <-
        IF(tiempo_control(buffer)==MCS-1) THEN
            tiempo(buffer+1) = tiempo(buffer) + 1
            tiempo_control(buffer+1) = MCS
        ELSE
            tiempo(buffer+1) = tiempo(buffer)

```



```

        tiempo_control(buffer+1) = MCS
    ENDIF
    long(buffer+1) = long(buffer) + 1
    ! <-
ELSE
    aleatorio_tau=PRNG()
    IF(aleatorio_tau<prob_tau_mov) THEN
        !tau impide el movimiento, el motor se desliga
        estado(buffer)=0
        ! se desliga, acumulo tiempo de ligadura <-
        tiempo_todos = tiempo_todos + tiempo(buffer) + 1
        contador = contador +1
        tiempo_todos_cuad = tiempo_todos_cuad + (tiempo(buffer)
            + 1)**2
        long_todos = long_todos + long(buffer)
        auxiliar = DBLE(long(buffer))/DBLE(tiempo(buffer)+1)
        veloc_todos = veloc_todos + auxiliar
    ELSE
        ! tau no impide el movimiento
        estado(buffer) = 0
        estado(buffer+1) = 2
        ! avanza, calculo para el tiempo de ligadura <-
        IF(tiempo_control(buffer)==MCS-1) THEN
            tiempo(buffer+1) = tiempo(buffer) + 1
            tiempo_control(buffer+1) = MCS
        ELSE
            tiempo(buffer+1) = tiempo(buffer)
            tiempo_control(buffer+1) = MCS
        ENDIF
        long(buffer+1) = long(buffer) + 1
        ! <-
    ENDIF
ENDIF
ELSEIF ( aleatorio < wb+wb+ws) THEN
    ! hidroliza pero permanece en el sitio, pasa al estado 1
    estado(buffer) = 1
    ! permanece, acumulo en tiempo de ligadura <-
    IF(tiempo_control(buffer)==MCS-1) THEN
        tiempo(buffer) = tiempo(buffer) + 1
        ! no puede volver a cambiar pero puede haber cambiado
        antes
        ! a la long le tendria que sumar cero
    ENDIF
    !<-
ELSEIF ( aleatorio < wb+wb+ws+wf ) THEN
    ! hidroliza e intenta utilizar la energia para avanzar
    IF(tau(buffer+1)==0) THEN
        estado(buffer) = 0
        estado(buffer+1) = 1
        ! avanza, calculo para el tiempo de ligadura <-
        IF(tiempo_control(buffer)==MCS-1) THEN
            tiempo(buffer+1) = tiempo(buffer) + 1

```

```

        tiempo_control(buffer+1) = MCS
    ELSE
        tiempo(buffer+1) = tiempo(buffer)
        tiempo_control(buffer+1) = MCS
    ENDIF
    long(buffer+1) = long(buffer) + 1
    ! <-
ELSE
    ! hay tau, tiene prob de impedir el movimiento
    aleatorio_tau=PRNG()
    IF(aleatorio_tau<prob_tau_mov) THEN
        !tau impide el movimiento, el motor se desliga
        estado(buffer)=0
        ! se desliga, acumulo tiempo de ligadura <-
        tiempo_todos = tiempo_todos + tiempo(buffer) + 1
        contador = contador +1
        tiempo_todos_cuad = tiempo_todos_cuad + (tiempo(buffer)
            + 1)**2
        long_todos = long_todos + long(buffer)
        auxiliar = DBLE(long(buffer))/DBLE(tiempo(buffer)+1)
        veloc_todos = veloc_todos + auxiliar
    ELSE
        ! tau no impide el movimiento
        estado(buffer) = 0
        estado(buffer+1) = 1
        ! avanza, calculo para el tiempo de ligadura <-
        IF(tiempo_control(buffer)==MCS-1) THEN
            tiempo(buffer+1) = tiempo(buffer) + 1
            tiempo_control(buffer+1) = MCS
        ELSE
            tiempo(buffer+1) = tiempo(buffer)
            tiempo_control(buffer+1) = MCS
        ENDIF
        long(buffer+1) = long(buffer) + 1
        ! <-
    ENDIF
ENDIF
ELSE
    ! si no se cumple nada de eso entonces se queda todo igual
    ! permanece, acumulo en tiempo de ligadura <-
    tiempo(buffer) = tiempo(buffer) + 1
    ! a la long le tendria que sumar cero
    !<-
END IF
!
END SUBROUTINE opcion1
!
!*****
!
SUBROUTINE opcion2(estado, buffer, tau, tiempo, tiempo_todos,
    contador, MCS, tiempo_control, tiempo_todos_cuad, long, long_
    todos, veloc_todos) !X20

```

```

IMPLICIT NONE
!
INTEGER, INTENT(INOUT) :: estado(1:L)
INTEGER, INTENT(IN) :: tau(1:L)
INTEGER, INTENT(IN) :: buffer, MCS
REAL(KIND(0.d0)) :: aleatorio, aleatorio_tau
INTEGER, INTENT(INOUT) :: tiempo(1:L), tiempo_control(1:L), long
    (1:L)
INTEGER, INTENT(INOUT) :: tiempo_todos, contador, long_todos
INTEGER*8, INTENT(INOUT) :: tiempo_todos_cuad
REAL(KIND(0.d0)), INTENT(INOUT) :: veloc_todos
REAL(KIND(0.d0)) :: auxiliar
!
aleatorio = PRNG()
!
IF (aleatorio < wb) THEN
! se quiere mover a la derecha por difusion
IF(tau(buffer+1)==0) THEN
    estado(buffer) = 0
    estado(buffer+1) = 2
! avanza, calculo para el tiempo de ligadura <-
IF(tiempo_control(buffer)==MCS-1) THEN
    tiempo(buffer+1) = tiempo(buffer) + 1
    tiempo_control(buffer+1) = MCS
ELSE
    tiempo(buffer+1) = tiempo(buffer)
    tiempo_control(buffer+1) = MCS
ENDIF
long(buffer+1) = long(buffer) + 1
! <-
ELSE
! hay tau, tiene prob de impedir el movimiento
aleatorio_tau=PRNG()
IF(aleatorio_tau<prob_tau_mov) THEN
!tau impide el movimiento, el motor se desliga
estado(buffer)=0
! se desliga, acumulo tiempo de ligadura <-
tiempo_todos = tiempo_todos + tiempo(buffer) + 1
contador = contador +1
tiempo_todos_cuad = tiempo_todos_cuad + (tiempo(buffer) +
    1)**2
long_todos = long_todos + long(buffer)
auxiliar = DBLE(long(buffer))/DBLE(tiempo(buffer)+1)
veloc_todos = veloc_todos + auxiliar
ELSE
! tau no impide el movimiento
estado(buffer) = 0
estado(buffer+1) = 2
! avanza, calculo para el tiempo de ligadura <-
IF(tiempo_control(buffer)==MCS-1) THEN
    tiempo(buffer+1) = tiempo(buffer) + 1
    tiempo_control(buffer+1) = MCS

```

```

        ELSE
            tiempo(buffer+1) = tiempo(buffer)
            tiempo_control(buffer+1) = MCS
        ENDIF
        long(buffer+1) = long(buffer) + 1
        ! <-
    ENDIF
ENDIF
ELSEIF ( aleatorio < wb+ws) THEN
    ! hidroliza pero permanece en el sitio, pasa al estado 1
    estado(buffer) = 1
    !permanece, acumulo en tiempo de ligadura <-
    IF(tiempo_control(buffer)==MCS-1) THEN
        tiempo(buffer) = tiempo(buffer) + 1
        ! no puede volver a cambiar pero puede haber cambiado
        antes
        ! a la long le tendria que sumar cero
    ENDIF
    !<-
ELSEIF ( aleatorio < wb+ws+wf ) THEN
    ! hidroliza e intenta utilizar la energia para avanzar
    IF(tau(buffer+1)==0) THEN
        estado(buffer) = 0
        estado(buffer+1) = 1
        ! avanza, calculo para el tiempo de ligadura <-
        IF(tiempo_control(buffer)==MCS-1) THEN
            tiempo(buffer+1) = tiempo(buffer) + 1
            tiempo_control(buffer+1) = MCS
        ELSE
            tiempo(buffer+1) = tiempo(buffer)
            tiempo_control(buffer+1) = MCS
        ENDIF
        long(buffer+1) = long(buffer) + 1
        ! <-
    ELSE
        ! hay tau, tiene prob de impedir el movimiento
        aleatorio_tau=PRNG()
        IF(aleatorio_tau<prob_tau_mov) THEN
            !tau impide el movimiento, el motor se desliga
            estado(buffer)=0
            ! se desliga, acumulo tiempo de ligadura <-
            tiempo_todos = tiempo_todos + tiempo(buffer) + 1
            contador = contador +1
            tiempo_todos_cuad = tiempo_todos_cuad + (tiempo(buffer) +
                1)**2
            long_todos = long_todos + long(buffer)
            auxiliar = DBLE(long(buffer))/DBLE(tiempo(buffer)+1)
            veloc_todos = veloc_todos + auxiliar
        ELSE
            ! tau no impide el movimiento
            estado(buffer) = 0
            estado(buffer+1) = 1

```

```

        ! avanza, calculo para el tiempo de ligadura <-
        IF(tiempo_control(buffer)==MCS-1) THEN
            tiempo(buffer+1) = tiempo(buffer) + 1
            tiempo_control(buffer+1) = MCS
        ELSE
            tiempo(buffer+1) = tiempo(buffer)
            tiempo_control(buffer+1) = MCS
        ENDIF
        long(buffer+1) = long(buffer) + 1
        ! <-
    ENDIF
ENDIF
ELSE
    ! si no se cumple nada de eso entonces se queda todo igual
    ! permanece, acumulo en tiempo de ligadura <-
    tiempo(buffer) = tiempo(buffer) + 1
    ! a la long le tendria que sumar cero
    !<-
END IF
!
END SUBROUTINE opcion2
!
!*****
!
SUBROUTINE opcion3(estado, buffer, tau, tiempo, tiempo_todos,
    contador, MCS, tiempo_control, tiempo_todos_cuad, long, long_
    todos, veloc_todos) !02X
IMPLICIT NONE
!
INTEGER, INTENT(INOUT) :: estado(1:L)
INTEGER, INTENT(IN) :: tau(1:L)
INTEGER, INTENT(IN) :: buffer, MCS
REAL(KIND(0.d0)) :: aleatorio, aleatorio_tau
INTEGER, INTENT(INOUT) :: tiempo(1:L), tiempo_control(1:L), long
    (1:L)
INTEGER, INTENT(INOUT) :: tiempo_todos, contador, long_todos
INTEGER*8, INTENT(INOUT) :: tiempo_todos_cuad
REAL(KIND(0.d0)), INTENT(INOUT) :: veloc_todos
REAL(KIND(0.d0)) :: auxiliar
!
REAL(KIND(0.d0)) :: prob_tau_wa
REAL(KIND(0.d0)) :: prob_tau_mov
!
aleatorio = PRNG()
!
IF (aleatorio < wb) THEN
    ! se quiere mover a la izquierda por difusion
    ! me fijo si hay tau en el sitio a la izq
    IF(tau(buffer-1)==0) THEN
        estado(buffer-1) = 2
        estado(buffer) = 0
        ! retrocede, calculo para el tiempo de ligadura <-

```

```

IF(tiempo_control(buffer)==MCS-1) THEN
  tiempo(buffer-1) = tiempo(buffer) + 1
  tiempo_control(buffer-1) = MCS
ELSE
  tiempo(buffer-1) = tiempo(buffer)
  tiempo_control(buffer-1) = MCS
ENDIF
long(buffer-1) = long(buffer) - 1
! <-
ELSE
! hay tau, tiene prob de impedir el movimiento
aleatorio_tau=PRNG()
IF(aleatorio_tau<prob_tau_mov) THEN
!tau impide el movimiento, el motor se desliga
estado(buffer)=0
! se desliga, acumulo tiempo de ligadura <-
tiempo_todos = tiempo_todos + tiempo(buffer) + 1
contador = contador +1
tiempo_todos_cuad = tiempo_todos_cuad + (tiempo(buffer) + 1)
**2
long_todos = long_todos + long(buffer)
auxiliar = DBLE(long(buffer))/DBLE(tiempo(buffer)+1)
veloc_todos = veloc_todos + auxiliar
ELSE
! tau no impide el movimiento
estado(buffer-1) = 2
estado(buffer) = 0
! retrocede, calculo para el tiempo de ligadura <-
IF(tiempo_control(buffer)==MCS-1) THEN
  tiempo(buffer-1) = tiempo(buffer) + 1
  tiempo_control(buffer-1) = MCS
ELSE
  tiempo(buffer-1) = tiempo(buffer)
  tiempo_control(buffer-1) = MCS
ENDIF
long(buffer-1) = long(buffer) - 1
! <-
ENDIF
ENDIF
ELSEIF ( aleatorio < wb+ws+(1-c)*wf) THEN
! hidroliza pero permanece en el sitio
estado(buffer) = 1
! permanece, acumulo en tiempo de ligadura <-
IF(tiempo_control(buffer)==MCS-1) THEN
  tiempo(buffer) = tiempo(buffer) + 1
! no puede volver a cambiar pero puede haber cambiado antes
! a la long le tendria que sumar cero
ENDIF
!<-
ELSE
! si no se cumple nada de eso entonces se queda todo igual
! permanece, acumulo en tiempo de ligadura <-

```

```

    tiempo(buffer) = tiempo(buffer) + 1
    ! a la long le tendria que sumar cero
    !<-
END IF
!
END SUBROUTINE opcion3
!
!*****
!
SUBROUTINE opcion4(estado, buffer, tiempo, MCS, tiempo_control)
    ! X2X
    IMPLICIT NONE
    !
    INTEGER, INTENT(INOUT) :: estado(1:L)
    INTEGER, INTENT(IN) :: buffer, MCS
    REAL(KIND(0.d0)) :: aleatorio
    INTEGER, INTENT(INOUT) :: tiempo(1:L)
    INTEGER, INTENT(INOUT) :: tiempo_control(1:L)
    !
    aleatorio = PRNG()
    !
    IF ( aleatorio < ws+(1-c)*wf) THEN
        ! hidroliza pero permanece en el sitio
        estado(buffer) = 1
        ! si no se cumple eso entonces se queda todo igual
    END IF
    !
    ! permanece, acumulo en tiempo de ligadura <-
    IF(tiempo_control(buffer)==MCS-1) THEN
        tiempo(buffer) = tiempo(buffer) + 1
        ! no puede volver a cambiar pero puede haber cambiado antes
    ENDIF
    !
END SUBROUTINE opcion4
!
END MODULE opciones_tau_mod

```

E.1.4 Código fuente extremos.f90

```

MODULE extremos_tau_mod
    USE mtmod, ONLY: PRNG => grnd
    USE constantes
    IMPLICIT NONE
    !
    CONTAINS
    !
    SUBROUTINE extremo_izq(estado, buffer, tau, tiempo, tiempo_todos,
        contador, MCS, tiempo_control, tiempo_todos_cuad, long, long
        _todos, veloc_todos)
        IMPLICIT NONE
        !

```

```

INTEGER, INTENT(INOUT) :: estado(1:L)
INTEGER, INTENT(IN) :: tau(1:L)
INTEGER, INTENT(IN) :: buffer, MCS
REAL(KIND(0.d0)) :: aleatorio, aleatorio_tau
INTEGER, INTENT(INOUT) :: tiempo(1:L), tiempo_control(1:L), long
      (1:L)
INTEGER, INTENT(INOUT) :: tiempo_todos, contador, long_todos
INTEGER*8, INTENT(INOUT) :: tiempo_todos_cuad
REAL(KIND(0.d0)), INTENT(INOUT) :: veloc_todos
REAL(KIND(0.d0)) :: auxiliar
!
aleatorio = PRNG()
!
SELECT CASE (estado(buffer))
CASE(0) ! cuando el sitio esta en el estado 0
IF (aleatorio<alpha) THEN
! me fijo si hay tau en ese sitio
IF(tau(buffer)==0) THEN
estado(buffer)=1
!se liga, empieza cuenta t medio de lig <-
tiempo(buffer) = 0
long(buffer) = 0
!<-
ELSE
! hay tau, tiene prob de impedir la ligadura
aleatorio_tau=PRNG()
IF(aleatorio_tau>prob_tau_wa) THEN
estado(buffer)=1
!se liga, empieza cuenta t medio de lig <-
tiempo(buffer) = 0
long(buffer) = 0
!<-
ENDIF
ENDIF
! sino queda como esta (en el estado 0)
ENDIF
CASE(1) ! sitio en estado 1
IF (aleatorio < gamma1) THEN
estado(buffer) = 0
! se desliga, acumulo tiempo de ligadura <-
tiempo_todos = tiempo_todos + tiempo(buffer) + 1
contador = contador + 1
tiempo_todos_cuad = tiempo_todos_cuad + (tiempo(buffer) + 1)**2
long_todos = long_todos + long(buffer)
auxiliar = DBLE(long(buffer))/DBLE(tiempo(buffer)+1)
veloc_todos = veloc_todos + auxiliar
ELSEIF (aleatorio < (gamma1+wh) ) THEN
estado(buffer) = 2
! permanece, acumulo en tiempo de ligadura <-
tiempo(buffer) = tiempo(buffer) + 1
! a la long le tendria que sumar cero
!<-

```



```

ELSE
  ! sino queda como esta (en el estado 1)
  ! permanece, acumulo en tiempo de ligadura <-
  tiempo(buffer) = tiempo(buffer) + 1
  ! a la long le tendria que sumar cero
  !<-
ENDIF
CASE(2)
! solo hay vecino a la derecha!!!!!!!!!!!!!!
IF (estado(buffer+1) == 0 ) THEN
  ! 20 *****
  IF (aleatorio < gamma2) THEN
    ! se puede ir por difusion del filamento
    estado(buffer) = 0
    ! se desliga, acumulo tiempo de ligadura <-
    tiempo_todos = tiempo_todos + tiempo(buffer) + 1
    contador = contador +1
    tiempo_todos_cuad = tiempo_todos_cuad + (tiempo(buffer) + 1)
      **2
    long_todos = long_todos + long(buffer)
    auxiliar = DBLE(long(buffer))/DBLE(tiempo(buffer)+1)
    veloc_todos = veloc_todos + auxiliar
  ELSEIF (aleatorio < gamma2+wb) THEN
    ! se quiere mover a la derecha por difusion
    ! me fijo si hay tau en ese sitio
    IF(tau(buffer)==0) THEN
      estado(buffer) = 0
      estado(buffer+1) = 2
      ! avanza, calculo para el tiempo de ligadura <-
      IF(tiempo_control(buffer)==MCS-1) THEN
        tiempo(buffer+1) = tiempo(buffer) + 1
        tiempo_control(buffer+1) = MCS
      ELSE
        tiempo(buffer+1) = tiempo(buffer)
        tiempo_control(buffer+1) = MCS
      ENDIF
      long(buffer+1) = long(buffer) + 1
      ! <-
    ELSE
      ! hay tau, tiene prob de impedir el movimiento
      aleatorio_tau=PRNG()
      IF(aleatorio_tau<prob_tau_mov) THEN
        !tau impide el movimiento, el motor se desliga
        estado(buffer)=0
        ! se desliga, acumulo tiempo de ligadura <-
        tiempo_todos = tiempo_todos + tiempo(buffer) + 1
        contador = contador +1
        tiempo_todos_cuad = tiempo_todos_cuad + (tiempo(buffer)
          + 1)**2
        long_todos = long_todos + long(buffer)
        auxiliar = DBLE(long(buffer))/DBLE(tiempo(buffer)+1)
        veloc_todos = veloc_todos + auxiliar
      ELSE

```

```

ELSE
  ! tau no impide el movimiento
  estado(buffer) = 0
  estado(buffer+1) = 2
  ! avanza, calculo para el tiempo de ligadura <-
  IF(tiempo_control(buffer)==MCS-1) THEN
    tiempo(buffer+1) = tiempo(buffer) + 1
  tiempo_control(buffer+1) = MCS
  ELSE
    tiempo(buffer+1) = tiempo(buffer)
    tiempo_control(buffer+1) = MCS
  ENDIF
  long(buffer+1) = long(buffer) + 1
  ! <-
ENDIF
ELSEIF (aleatorio < gamma2+wb+ws) THEN
  ! hidroliza pero permanece en el sitio, entonces pasa al
  estado 1
  estado(buffer) = 1
  ! permanece, acumulo en tiempo de ligadura <-
  IF(tiempo_control(buffer)==MCS-1) THEN
    tiempo(buffer) = tiempo(buffer) + 1
    ! no puede volver a cambiar pero puede haber cambiado
    antes
    ! a la long le tendria que sumar cero
  ENDIF
  !<-
ELSEIF (aleatorio < gamma2+wb+ws+wf) THEN
  ! hidroliza e intenta utilizar la energia para avanzar
  ! me fijo si hay tau en ese sitio
  IF(tau(buffer)==0) THEN
    estado(buffer) = 0
    estado(buffer+1) = 1
    ! avanza, calculo para el tiempo de ligadura <-
    IF(tiempo_control(buffer)==MCS-1) THEN
      tiempo(buffer+1) = tiempo(buffer) + 1
      tiempo_control(buffer+1) = MCS
    ELSE
      tiempo(buffer+1) = tiempo(buffer)
      tiempo_control(buffer+1) = MCS
    ENDIF
    long(buffer+1) = long(buffer) + 1
    ! <-
  ELSE
    ! hay tau, tiene prob de impedir el movimiento
    aleatorio_tau=PRNG()
    IF(aleatorio_tau<prob_tau_mov) THEN
      !tau impide el movimiento, el motor se desliga
      estado(buffer)=0
      ! se desliga, acumulo tiempo de ligadura <-
      tiempo_todos = tiempo_todos + tiempo(buffer) + 1
    
```

```

        contador = contador +1
        tiempo_todos_cuad = tiempo_todos_cuad + (tiempo(buffer)
            + 1)**2
        long_todos = long_todos + long(buffer)
        auxiliar = DBLE(long(buffer))/DBLE(tiempo(buffer)+1)
        veloc_todos = veloc_todos + auxiliar
    ELSE
        ! tau no impide el movimiento
        estado(buffer) = 0
        estado(buffer+1) = 1
        ! avanza, calculo para el tiempo de ligadura <-
        IF(tiempo_control(buffer)==MCS-1) THEN
            tiempo(buffer+1) = tiempo(buffer) + 1
            tiempo_control(buffer+1) = MCS
        ELSE
            tiempo(buffer+1) = tiempo(buffer)
            tiempo_control(buffer+1) = MCS
        ENDIF
        long(buffer+1) = long(buffer) + 1
        ! <-
    ENDIF
    ENDIF
    ! si no se cumple nada de eso entonces se queda todo igual
    ! permanece, acumulo en tiempo de ligadura <-
    tiempo(buffer) = tiempo(buffer) + 1
    ! a la long le tendria que sumar cero
    !<-
    END IF
    !*****
ELSE
    ! 2X *****
    IF (aleatorio < gamma2) THEN
        ! se va por difusion del filamento
        estado(buffer) = 0
        ! se desliga, acumulo tiempo de ligadura <-
        tiempo_todos = tiempo_todos + tiempo(buffer) + 1
        contador = contador +1
        tiempo_todos_cuad = tiempo_todos_cuad + (tiempo(buffer) + 1)**2
        long_todos = long_todos + long(buffer)
        auxiliar = DBLE(long(buffer))/DBLE(tiempo(buffer)+1)
        veloc_todos = veloc_todos + auxiliar
    ELSEIF ( aleatorio < gamma2+ws+(1-c)*wf) THEN
        ! hidroliza pero permanece en el sitio
        estado(buffer) = 1
        ! permanece, acumulo en tiempo de ligadura <-
        IF(tiempo_control(buffer)==MCS-1) THEN
            tiempo(buffer) = tiempo(buffer) + 1
            ! no puede volver a cambiar pero puede haber cambiado antes
            ! a la long le tendria que sumar cero
        ENDIF
        !<-
    ELSE

```

```

! si no se cumple nada de eso entonces se queda todo igual
! permanece, acumulo en tiempo de ligadura <-
tiempo(buffer) = tiempo(buffer) + 1
! a la long le tendria que sumar cero
!<-
END IF
!*****
END IF
CASE DEFAULT
WRITE(*,*) 'El estado del sitio debe ser 0, 1 o 2. Algo anda mal
, revisar!'
STOP
END SELECT
!
END SUBROUTINE extremo_izq
!
!*****
!
SUBROUTINE extremo_der(estado, buffer, tau, tiempo, tiempo_todos,
contador, MCS, tiempo_control, tiempo_todos_cuad, long, long
_todos, veloc_todos)
IMPLICIT NONE
!
INTEGER, INTENT(INOUT) :: estado(1:L)
INTEGER, INTENT(IN) :: tau(1:L)
INTEGER, INTENT(IN) :: buffer, MCS
REAL(KIND(0.d0)) :: aleatorio, aleatorio_tau
INTEGER, INTENT(INOUT) :: tiempo(1:L), tiempo_control(1:L), long
(1:L)
INTEGER, INTENT(INOUT) :: tiempo_todos, contador, long_todos
INTEGER*8, INTENT(INOUT):: tiempo_todos_cuad
REAL(KIND(0.d0)), INTENT(INOUT) :: veloc_todos
REAL(KIND(0.d0)) :: auxiliar
!
aleatorio = PRNG()
SELECT CASE (estado(buffer))
CASE(0) ! cuando el sitio esta en el estado 0
IF (aleatorio<delta) THEN
! me fijo si hay tau en ese sitio
IF(tau(buffer)==0) THEN
estado(buffer)=1
!se liga, empieza cuenta t medio de lig <-
tiempo(buffer) = 0
long(buffer) = 0
!<-
ELSE
! hay tau, tiene prob de impedir la ligadura
aleatorio_tau=PRNG()
IF(aleatorio_tau>prob_tau_wa) THEN
estado(buffer) = 1
!se liga, empieza cuenta t medio de lig <-
tiempo(buffer) = 0

```

```

        long(buffer) = 0
        !<-
    ENDIF
ENDIF
! sino queda como esta (en el estado 0)
ENDIF
CASE(1) ! sitio en estado 1
IF (aleatorio < betha1) THEN
! se desliga
estado(buffer) = 0
! se desliga, acumulo tiempo de ligadura <-
tiempo_todos = tiempo_todos + tiempo(buffer) + 1
contador = contador +1
tiempo_todos_cuad = tiempo_todos_cuad + (tiempo(buffer) + 1)**2
long_todos = long_todos + long(buffer)
auxiliar = DBLE(long(buffer))/DBLE(tiempo(buffer)+1)
veloc_todos = veloc_todos + auxiliar
ELSEIF (aleatorio < (betha1+wh) ) THEN
! hidroliza
estado(buffer) = 2
! permanece, acumulo en tiempo de ligadura <-
tiempo(buffer) = tiempo(buffer) + 1
! a la long le tendria que sumar cero
!<-
ELSE
! sino queda como esta (en el estado 1)
! permanece, acumulo en tiempo de ligadura <-
tiempo(buffer) = tiempo(buffer) + 1
! a la long le tendria que sumar cero
!<-
ENDIF
CASE(2)
! recordar q solo hay vecino a la izquierda
IF (estado(buffer-1) == 0 ) THEN
! 02 *****
IF (aleatorio < betha2) THEN
! se puede ir por difusion del filamento
! no hace falta ver si tau lo impide, porque si lo hiciera el
! resultado es el mismo
! i.e. se desliga, lo mismo que ocurre si 'salta a la nada' en
! el extremo
estado(buffer) = 0
! se desliga, acumulo tiempo de ligadura <-
tiempo_todos = tiempo_todos + tiempo(buffer) + 1
contador = contador +1
tiempo_todos_cuad = tiempo_todos_cuad + (tiempo(buffer) + 1)
**2
long_todos = long_todos + long(buffer)
auxiliar = DBLE(long(buffer))/DBLE(tiempo(buffer)+1)
veloc_todos = veloc_todos + auxiliar
ELSEIF (aleatorio < betha2+wb) THEN
! se quiere mover a la izquierda por difusion

```

```

! me fijo si hay tau en ese sitio
IF(tau(buffer)==0) THEN
  estado(buffer-1) = 2
  estado(buffer) = 0
  ! retrocede, calculo para el tiempo de ligadura <-
  IF(tiempo_control(buffer)==MCS-1) THEN
    tiempo(buffer-1) = tiempo(buffer) + 1
    tiempo_control(buffer-1) = MCS
  ELSE
    tiempo(buffer-1) = tiempo(buffer)
    tiempo_control(buffer-1) = MCS
  ENDIF
  long(buffer-1) = long(buffer)-1
  ! <-
ELSE
  ! hay tau, tiene prob de impedir el movimiento
  aleatorio_tau=PRNG()
  IF(aleatorio_tau<prob_tau_mov) THEN
    !tau impide el movimiento, el motor se desliga
    estado(buffer)=0
    ! se desliga, acumulo tiempo de ligadura <-
    tiempo_todos = tiempo_todos + tiempo(buffer) + 1
    contador = contador +1
    tiempo_todos_cuad = tiempo_todos_cuad + (tiempo(buffer) + 1)
      **2
    long_todos = long_todos + long(buffer)
    auxiliar = DBLE(long(buffer))/DBLE(tiempo(buffer)+1)
    veloc_todos = veloc_todos + auxiliar
  ELSE
    ! tau no impide el movimiento
    estado(buffer-1) = 2
    estado(buffer) = 0
    ! retrocede, calculo para el tiempo de ligadura <-
    IF(tiempo_control(buffer)==MCS-1) THEN
      tiempo(buffer-1) = tiempo(buffer) + 1
      tiempo_control(buffer-1) = MCS
    ELSE
      tiempo(buffer-1) = tiempo(buffer)
      tiempo_control(buffer-1) = MCS
    ENDIF
    long(buffer-1) = long(buffer)-1
    ! <-
  ENDIF
ENDIF
ELSEIF (aleatorio < betha2+wb+ws) THEN
  ! hidroliza pero permanece en el sitio
  estado(buffer) = 1
  ! permanece, acumulo en tiempo de ligadura <-
  IF(tiempo_control(buffer)==MCS-1) THEN
    tiempo(buffer) = tiempo(buffer) + 1
    ! no puede volver a cambiar pero puede haber cambiado antes
    ! a la long le tendria que sumar cero

```

```

    ENDIF
    !<-
ELSE
    ! si no se cumple nada de eso entonces se queda todo igual
    ! permanece, acumulo en tiempo de ligadura <-
    tiempo(buffer) = tiempo(buffer) + 1
    ! a la long le tendria que sumar cero
    !<-
END IF
    !*****
ELSE
    ! X2 *****
    IF (aleatorio < betha2) THEN
        ! se puede ir por difusion del filamento
        estado(buffer) = 0
        ! se desliga, acumulo tiempo de ligadura <-
        tiempo_todos = tiempo_todos + tiempo(buffer) + 1
        contador = contador +1
        tiempo_todos_cuad = tiempo_todos_cuad + (tiempo(buffer) + 1)
            **2
        long_todos = long_todos + long(buffer)
        auxiliar = DBLE(long(buffer))/DBLE(tiempo(buffer)+1)
        veloc_todos = veloc_todos + auxiliar
    ELSEIF (aleatorio < betha2+ws) THEN
        ! hidroliza pero permanece en el sitio
        estado(buffer) = 1
        ! permanece, acumulo en tiempo de ligadura <-
        IF(tiempo_control(buffer)==MCS-1) THEN
            tiempo(buffer) = tiempo(buffer) + 1
            ! no puede volver a cambiar pero puede haber cambiado antes
            ! a la long le tendria que sumar cero
        ENDIF
    !<-
ELSE
    ! si no se cumple nada de eso entonces se queda todo igual
    ! permanece, acumulo en tiempo de ligadura <-
    tiempo(buffer) = tiempo(buffer) + 1
    ! a la long le tendria que sumar cero
    !<-
END IF
    !*****
END IF
CASE DEFAULT
    WRITE(*,*) 'El estado del sitio debe ser 0, 1 o 2. Algo anda mal
        , revisar!'
    STOP
END SELECT
!
END SUBROUTINE extremo_der
!
END MODULE extremos_tau_mod

```

E.1.5 Código fuente *main.f90*

```

PROGRAM conta
  USE constantes
  USE mtmod, ONLY: sgrnd, PRNG => grnd
  USE extremos_tau_mod, ONLY: extremo_izq, extremo_der
  USE opciones_tau_mod, ONLY: opcion1, opcion2, opcion3, opcion4
  USE concentracion_mod, ONLY: Distrtau
  !
  IMPLICIT NONE
  INTEGER :: sitio(1:L) , estado(1:L), tau(1:L)
  ! 'sitio' p/saber actualizados
  ! 'estados': 0 (sin motor), 1 (mot. en est 1) y 2 (mot. en est 2)
  ! 'tau': 0 (no tau), 1 (si tau)
  INTEGER :: ii , aleat_entero, buffer, MCS, MCStotal,
            MCStermalizacion, kk, mm, nn, semilla, repeticiones ,
            barrido
  REAL(KIND(0.d0)) :: aleatorio, aleat_real
  CHARACTER(30) :: archivo_salida = 'DJSmedia_max.dat'
  CHARACTER(30) :: archivo_tligadura = 'tligadura.dat'
  CHARACTER(30) :: archivo_secuencia
  INTEGER, PARAMETER :: n_importan = 100
  INTEGER :: pos_max
  REAL(KIND(0.d0)) :: val_max
  ! las variables que siguen son para el calculo del tiempo medio
  ! de ligadura
  INTEGER :: tiempo(1:L), tiempo_control(1:L), long(1:L)
  INTEGER :: tiempo_todos, contador, contador_buffer, long_todos
  INTEGER*8 :: tiempo_todos_cuad
  REAL(KIND(0.d0)) :: veloc_todos
  REAL(KIND(0.d0)) :: auxiliar
  !
  !*****
  ! Uso el generador de PseudoAleatorios Mersenne-Twister
  ! aca lo inicializo
  ! pasandole la semilla por teclado (archivo <semilla.in)
  !*****
      READ(*,*) semilla
      CALL sgrnd(semilla)
  !*****
  !
  tiempo_todos = 0
  contador = 0
  tiempo(:) = 0
  tiempo_control(:) = 0
  contador_buffer = 0
  tiempo_todos_cuad = 0
  long(:) = 0
  long_todos = 0
  veloc_todos = 0.d0
  !
  MCStotal = 2000000

```



```

MCStermalizacion = 200000
!
WRITE(*,*) '*****'
WRITE(*,*) '*      Parametros del modelo      *'
WRITE(*,*) '*****'
WRITE(*,*) 'L: ', L
WRITE(*,*) 'n_segmentos: ', n_segmentos
WRITE(*,*) 'long_segmento:L/n_segmentos ', L/n_segmentos
WRITE(*,*) 'Cmin: ', Cmin
WRITE(*,*) 'Cmax: ', Cmax
WRITE(*,*) 'prob_tau_wa: ', prob_tau_wa , ' prob. de que tau
impida la ligadura'
WRITE(*,*) 'prob_tau_mov: ', prob_tau_mov, ' prob. de que tau
impida el movimiento'
WRITE(*,*) 'wd=betha1=betha2: ', wd
WRITE(*,*) 'wb: ', wb
WRITE(*,*) 'ws: ', ws
WRITE(*,*) 'wf: ', wf
WRITE(*,*) 'wa=alpha: ', wa
WRITE(*,*) 'wh: ', wh
WRITE(*,*) 'c: ', c
WRITE(*,*) 'gamma1: ', gamma1
WRITE(*,*) 'gamma2: ', gamma2
WRITE(*,*) 'delta: ', delta
WRITE(*,*) 'semilla del PRNG: ', semilla
WRITE(*,*) 'MCStep total: ', MCStotal
WRITE(*,*) 'MCStep termalizacion: ', MCStermalizacion
WRITE(*,*) '*****'
!
!*****
! para almacenar el valor de la DJSmaxima y en que posicion
ocurre
!*****
!
OPEN(UNIT=34 , FILE = archivo_salida , STATUS = 'new' , ACTION =
'write')
WRITE(34,*) '# semilla ', ' C_min ', ' C_max ', ' x_
max ', ' DJS_max'
!
!*****
! para almacenar el tiempo de ligadura, la longitud y la
velocidad
!*****
OPEN(UNIT=33 , FILE = archivo_tligadura , ACTION = 'write',
STATUS = 'new')
WRITE(33,*) 'Ctau ', 'nro. eventos, Running Average de tlig , R.
A. tlig**2, R.A long, R.A. veloc '
WRITE(33,*) '# Todos estan sin unidades dt=0.71ms, dl=8nm'
!
!*****
! Hago 1000 repeticiones para 'semillas' diferentes.
!En la practica no cambio la semilla porque

```

```

! el generador ya esta andando, tira nros independientes
! mientras no supere el periodo, que en el caso del MT es
  altisimo.
! Para cada caso genero una nueva distribucion de tau
!*****
!
realizaciones: DO repeticiones=1,1000 ! <---- <---- <----
!
!*****
! valores iniciales. Ningun motor ligado inicialmente
!*****
!
DO ii=1,L
  sitio(ii) = ii
  estado(ii) = 0
END DO
!
!*****
! Genero la distribucion de tau en el filamento
!*****
!
CALL DistrTau(tau)
!
!*****
! sirven para nombrar los archivos de las secuencias
!*****
kk=-1
mm=1
!nn=1
!
!*****
! Un bucle de 'filamento' es un 'dt'
! Cada 'dt', es decir cada paso temporal es un MCS
! Actualizamos los sitios en orden aleatorio
!*****
!
MCStep: DO MCS= 1, MCStotal
  filamento:DO ii = 1 , L
  !
  !-----
  aleat_real = PRNG()
! deberian ocurrir dos milagros para que esto falle
  IF(aleat_real==1.d0) aleat_real=PRNG()
  !-----
  aleat_entero = INT(aleat_real*(L-ii+1)) + 1 ! da el sitio
  que vamos a actualizar
  !-----
  buffer = sitio(aleat_entero)
! re ordeno el vector para ir actualizando todos (el prox
  aleatorio va a tener en cuenta solo los sitios que no visite)
  sitio(aleat_entero) = sitio(L-ii+1)
  sitio(L-ii+1) = buffer

```

```

!
IF (buffer>1 .AND. buffer<L) THEN
  ! vale para el bulk nomas
  SELECT CASE (estado(buffer))
  CASE(0) ! cuando el sitio esta en el estado 0
    aleatorio = PRNG()
    IF (aleatorio<wa) THEN
      ! me fijo si hay tau en ese sitio
    IF(tau(buffer)==0) THEN
      estado(buffer)=1
      !se liga,empieza cuenta t medio de lig <-
      tiempo(buffer) = 0
      long(buffer) = 0
      !<-
    ELSE
      ! hay tau, tiene prob de impedir la ligadura
      aleatorio=PRNG()
      IF(aleatorio>prob_tau_wa) THEN
        estado(buffer)=1
        !se liga,empieza cuenta t medio de lig <-
        tiempo(buffer) = 0
        long(buffer) = 0
        !<-
      ENDIF
    ENDIF
    ! sino queda como esta (en el estado 0)
  ENDIF
CASE(1) ! sitio en estado 1
  aleatorio = PRNG()
  IF (aleatorio < wd) THEN
    estado(buffer) = 0
    ! se desliga, acumulo tiempo de ligadura <-
    tiempo_todos = tiempo_todos + tiempo(buffer) + 1
    contador = contador +1
    tiempo_todos_cuad = tiempo_todos_cuad + (tiempo(buffer)+1)
      **2
    long_todos = long_todos + long(buffer)
    auxiliar = DBLE(long(buffer))/DBLE(tiempo(buffer)+1)
    veloc_todos = veloc_todos + auxiliar
    !<-
  ELSEIF (aleatorio < (wd+wh) ) THEN
    estado(buffer) = 2
    ! permanece, acumulo en tiempo de ligadura <-
    tiempo(buffer) = tiempo(buffer) + 1
    ! a la long le tendria que sumar cero
    !<-
  ELSE
    ! queda como esta (en el estado 1)
    ! permanece, acumulo en tiempo de ligadura <-
    tiempo(buffer) = tiempo(buffer) + 1
    ! a la long le tendria que sumar cero
    !<-

```

```

ENDIF
CASE(2)
IF (estado(buffer-1) == 0 ) THEN
  IF (estado(buffer+1) == 0 ) THEN
    ! 020
    CALL opcion1(estado,buffer,tau,tiempo,tiempo_todos,
      contador,MCS,tiempo_control,tiempo_todos_cuad,long,
      long_todos,veloc_todos)
  ELSE
    ! 02X
    CALL opcion3(estado,buffer,tau,tiempo,tiempo_todos,
      contador,MCS,tiempo_control,tiempo_todos_cuad,long,
      long_todos,veloc_todos)
  END IF
ELSE
  IF (estado(buffer+1) == 0 ) THEN
    ! X20
    CALL opcion2(estado,buffer,tau,tiempo,tiempo_todos,
      contador,MCS,tiempo_control,tiempo_todos_cuad,long,
      long_todos,veloc_todos)
  ELSE
    ! X2X
    CALL opcion4(estado,buffer,tiempo,MCS,tiempo_control)
  END IF
END IF
CASE DEFAULT
  WRITE(*,*) 'El estado del sitio debe ser 0, 1 o 2. Algo anda
    mal, revisar!'
  STOP
END SELECT
!
ELSEIF (buffer==1) THEN
! extremo izquierdo
  CALL extremo_izq(estado,buffer,tau,tiempo,tiempo_todos,
    contador,MCS,tiempo_control,tiempo_todos_cuad,long,
    long_todos,veloc_todos)
ELSEIF (buffer==L) THEN
! extremo derecho
  CALL extremo_der(estado,buffer,tau,tiempo,tiempo_todos,
    contador,MCS,tiempo_control,tiempo_todos_cuad,long,
    long_todos,veloc_todos)
END IF
END DO filamento
!
!*****
!
! tiempo control=MCS que actualice -> control: q se actualicen
  solo 1 vez
  tiempo_control(:)= MCS
!
! reinicio porque aca recién empiezo a medir, medio tosco pero no
  se me ocurrió nada mejor

```

```
!  
IF(MCS==MCStermalizacion) THEN  
  contador = 0  
  tiempo_todos = 0  
  tiempo_todos_cuad = 0  
  long_todos = 0  
  veloc_todos = 0.d0  
  ! ya hay tiempos de ligadura puestos, los dejo  
ENDIF  
!  
! para los ultimos 'n_importan' pasos dibujo el filamento y  
  calculo la DJS promedio  
IF (MCS>(MCStotal-n_importan)) THEN  
!  
kk=kk+1  
!  
archivo_secuencia = 'sec/secuencia'//CHAR(mm+47)//CHAR(kk+48)  
  //' .dat' ! hasta 100  
!  
IF (kk==9) THEN  
  kk=-1  
  mm=mm+1  
  !IF (mm==11) THEN  
    ! mm=1  
    ! nn=nn+1  
  !ENDIF  
ENDIF  
!  
!*****  
!para la secuencia  
!  
OPEN(UNIT=11 , FILE = archivo_secuencia, ACTION = 'write' ,  
  STATUS = 'replace')  
!*****  
!  
DO ii=1,L  
  IF (estado(ii)==0) THEN  
    ! estado 0: blanco  
    WRITE(11, '(I1)') 0  
  ELSEIF (estado(ii)==1) THEN  
    ! estado 1 : azul  
    WRITE(11, '(I1)') 1  
  ELSEIF (estado(ii)==2) THEN ! elseif por control  
    ! estado 2 : rojo  
    WRITE(11, '(I1)') 1  
    ! 1 = sitio ocupado, sin importar en que estado esta  
  ELSE  
    WRITE(*,*) 'algo va mal'  
    STOP  
  ENDIF  
END DO  
CLOSE(11)
```

```

!
  END IF
!
END DO MCStep
!
!*****
! promedios finales para el archivo de tligadura
!*****
WRITE(33, '(E14.7,2X,I5,4(2X,E14.7))') Cmin, contador, DBLE(tiempo
  _todos)/DBLE(contador), DBLE(tiempo_todos_cuad)/DBLE(contador
  ), DBLE(long_todos)/DBLE(contador), DBLE(veloc_todos)/DBLE(
  contador)
!
!
CALL segmentador(pos_max , val_max)
WRITE(34, '(I4,2(2X,E14.7),I3,2X,E14.7)')      semilla, Cmin,
  Cmax, pos_max , val_max
!
ENDDO realizaciones ! <---- <---- <----
!
CLOSE(34)
CLOSE(33)
!
CONTAINS
!
SUBROUTINE cursor_fijo(prob1,prob2, cursor , long_sec , filename)
  IMPLICIT NONE
  CHARACTER(35) :: filename
  INTEGER :: i , long_sec, lector , cursor
  INTEGER :: cuentas(0:1)
  REAL(KIND(0.d0)) :: prob1(0:1) , prob2(0:1)
  !      OPEN( UNIT = 9 , FILE = filename , STATUS = 'old' ,
    ACTION = 'read')
  !
  cuentas(0) = 0
  cuentas(1) = 0
  DO i = 1, cursor
    READ(9, '(I1)') lector
    !
    IF (lector==0) THEN
      cuentas(0) = cuentas(0) + 1
    ELSE
      cuentas(1) = cuentas(1) + 1 ! si no es cero es 1
    END IF
  END DO
  prob1(0) = DBLE(cuentas(0)) / DBLE(cursor)
  prob1(1) = DBLE(cuentas(1)) / DBLE(cursor)
  !
  cuentas(0) = 0
  cuentas(1) = 0
  DO i = (cursor+1), long_sec
    READ(9, '(I1)') lector

```

```

!
IF (lector==0) THEN
  cuentas(0) = cuentas(0) + 1
ELSE
  cuentas(1) = cuentas(1) + 1      ! si no es cero es 1
END IF
END DO
prob2(0) = DBLE(cuentas(0)) / DBLE(long_sec-cursor)
prob2(1) = DBLE(cuentas(1)) / DBLE(long_sec-cursor)
!
CLOSE(9)
END SUBROUTINE
!
!*****
!
SUBROUTINE DJS(prob1 , prob2 , JS , pi1 , pi2)
  IMPLICIT NONE
  !
  REAL(KIND(0.d0)) :: prob1(0:1) , prob2(0:1)
  REAL(KIND(0.d0)) :: JS , pi1 , pi2
  !
  JS=0.0D0
  IF (prob1(0).NE.0)      JS = pi1 * prob1(0) * LOG2(prob1(0))
    !1er termino de - H(prob1)/2
  IF (prob1(1).NE.0)      JS = JS + pi1 * prob1(1) * LOG2(prob1(1))
    ) ! 2do termino de - H(prob1)/2
  IF (prob2(0).NE.0)      JS = JS + pi2 * prob2(0) * LOG2(prob2(0))
    ! 1er termino de - H(prob2)/2
  IF (prob2(1).NE.0)      JS = JS + pi2 * prob2(1) * LOG2(prob2(1))
    ! 2do termino de - H(prob2)/2
  !
  IF (prob1(0).NE.0 .OR. prob2(0).NE.0) JS=JS-(pi1*prob1(0)+pi2*
    prob2(0))*LOG2(pi1*prob1(0)+pi2*prob2(0))
    ! 1er termino de + H ((prob1+prob2)/2)
  IF (prob1(1).NE.0 .OR. prob2(1).NE.0) JS=JS-(pi1*prob1(1)+pi2*
    prob2(1))*LOG2(pi1*prob1(1)+pi2*prob2(1))
    ! 2do termino de + H((prob1+prob2)/2)
  !
END SUBROUTINE
!
!*****
!
FUNCTION LOG2(x)
  IMPLICIT NONE
  REAL(KIND(0.d0)) :: LOG2, x
  !
  LOG2 = LOG(x)/LOG(2.d0)
  !
END FUNCTION LOG2
!
!*****
!

```

```

SUBROUTINE segmentador(lugar_max,valor_max)
  IMPLICIT NONE
  !
  REAL(KIND(0.d0)) :: probl1(0:1) , probl2(0:1) , pi1 , pi2 , suma
  INTEGER :: pos_cursor , j , k , l
  INTEGER,PARAMETER :: long_sec = 600
  CHARACTER(45) :: filename
  REAL(KIND(0.d0)):: JS(1:100,1:long_sec-1)      ! estoy
      promediando 100 sec
  REAL(KIND(0.d0)) :: JSpromedio(1:long_sec-1)
  INTEGER, INTENT(OUT):: lugar_max
  REAL(KIND(0.d0)), INTENT(OUT):: valor_max
  !
DO k=1,10
  DO l=1,10
    filename = 'sec/secuencia'//CHAR(k+47)//CHAR(l+47)//'.dat'
    DO j = 1 , (long_sec-1)
      pos_cursor = j
      CALL cursor_fijo(probl1,probl2, pos_cursor, long_sec, filename
        )
      pi1 = DBLE(pos_cursor)/DBLE(long_sec)
      pi2 = 1.0D0 - DBLE(pos_cursor)/DBLE(long_sec)
      CALL DJS( probl1 , probl2 , JS(((k-1)*10+l),j) , pi1 , pi2)
    END DO
  END DO
END DO
!
!*****
! calculo la JS promedio y la almaceno en un vector
!*****
DO pos_cursor = 1, long_sec-1
  suma = 0.0D0
  DO l=1,100
    suma = suma + JS(l,pos_cursor)
  END DO
  suma = suma/100.0D0
  JSpromedio(pos_cursor)=suma
END DO
!
!*****
! busco el maximo y la posicion del maximo
!*****
lugar_max=MAXLOC(JSpromedio,1)
valor_max=MAXVAL(JSpromedio,1)
! MAXLOC es lugar_max , 1 es la DIM, si no lo pones 1 da un
  vector de la misma dim. que el que le pasas
! MAXVAL, al igual que en MAXLOC poner DIM=1, sino MAXVAL no sale
  como un escalar!!!
!
END SUBROUTINE segmentador
!
END PROGRAM contau

```


BIBLIOGRAFÍA

- [1] Estadística de la Real Federación Española de Fútbol. URL <http://www.rfef.es/>.
- [2] B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. *The Molecular Biology of the Cell, fourth edition*. Garland, New York, 2002.
- [3] G. B Arfken and H.J. Weber. *Mathematical Methods for Physicists*. Academic Press, 4 edition, 1995.
- [4] I. Bengtsson and K. Życzkowski. *Geometry of Quantum States: An Introduction to Quantum Entanglement*. Cambridge University Press, Cambridge, England, 2006.
- [5] W. Bialek. *Biophysics: Searching for Principles*. Princeton University Press, 2012.
- [6] M.A. Bielsa. Conferencia de prensa.
- [7] S. L. Braunstein and C. M. Caves. *Phys. Rev. Lett.*, 72:3439–3443, 1994.
- [8] J. Briët and P. Harremoës. *Phys. Rev. A*, 79:052311, 2009.
- [9] C. Bustamante, W. Cheng, and Y.X. Mejia. *Cell*, 144:480–497, 2011.
- [10] H.B. Callen. *Thermodynamics and an Introduction to Thermostatistics*. John Wiley Sons, Inc., 2 edition, 1985.
- [11] E. Chevalier-Larsen and E.L.F. Holzbaur. *Biochimica et Biophysica Acta (BBA) - Molecular Basis of Disease*, 1762(11-12):1094 – 1108, 2006.
- [12] J. I. Cirac, A. K. Ekert, and C. Macchiavello. *Phys. Rev. Lett.*, 82: 4344–4347, 1999.
- [13] C. Conde and A. Cáceres. *Nature Reviews Neuroscience*, 10(5):319–332, 2009.
- [14] T. Constantinescu and V. Ramakrishna. *Quantum Inf. Process.*, 1: 409–424, 2002.
- [15] Julio Cortázar. *Historia de Cronopios y de Famas*. Alfaguara, 1965.
- [16] T. M. Cover and J. A. Thomas. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, 2006.

- [17] B. Derrida. *Physics Reports*, 301(1–3):65 – 83, 1998.
- [18] R. Dixit, J.L. Ross, Y.E. Goldman, and E.L.F. Holzbaur. *Science*, 319(5866):1086–1089, 2008.
- [19] K. Doya, S. Ishii, A. Pouget, and R. P. N. Rao, editors. *Bayesian Brain: Probabilistic Approaches to Neural Coding*. The MIT Press, 1 edition, 2007.
- [20] A. Ebner, R. Godemann, K. Stamer, S. Illenberger, B. Trinczek, E.-M. Mandelkow, and E. Mandelkow. *The Journal of Cell Biology*, 143(3):777–794, 1998.
- [21] D.M. Endres and J.E. Schindelin. *Information Theory, IEEE Transactions on*, 49(7):1858 – 1860, 2003.
- [22] P. Greulich, A. Garai, K. Nishinari, A. Schadschneider, and D. Chowdhury. *Phys. Rev. E*, 75(4):041905, 2007.
- [23] I. Grosse, P. Bernaola-Galván, P. Carpena, R. Román-Roldán, J. Oliver, and H. E. Stanley. *Phys. Rev. E*, 65(4):041905, 2002.
- [24] E. M. Guizzo. The essential message : Claude shannon and the making of information theory. Thesis (S.M. in Science Writing)–Massachusetts Institute of Technology, Dept. of Humanities, Program in Writing and Humanistic Studies, 2003. URL <http://hdl.handle.net/1721.1/39429>.
- [25] N. Hirokawa. *Current Opinion in Cell Biology*, 6(1):74 – 81, 1994.
- [26] N. Hirokawa and Y. Noda. *Physiol. Rev.*, 88(3):1089–1118, 2008.
- [27] N. Hirokawa, Y. Noda, Y. Tanaka, and S. Niwa. *Nat. Rev. Mol. Cell Biol.*, 10(10).
- [28] N. Hirokawa, R. Nitta, and Y. Okada. *Nat. Rev. Mol. Cell Biol.*, 10(12):877–884, 2009.
- [29] J. Howard. *Mechanics of Motor Proteins and the Cytoskeleton*. Sinauer Associates, Sunderland, MA, 2001.
- [30] R. Jozsa. *J. Mod. Opt.*, 41(12):2315–2323, 1994.
- [31] R. Jozsa and J. Schlienz. *Phys. Rev. A*, 62:012301, 2000.
- [32] F. Jülicher, A. Ajdari, and J. Prost. *Rev. Mod. Phys.*, 69:1269–1282, 1997.
- [33] P. W. Lamberti, A. P. Majtey, A. Borrás, M. Casas, and A. Plastino. *Phys. Rev. A*, 77:052311, 2008.
- [34] P. W. Lamberti, M. Portesi, and J. Sparacino. *Int. J. Quantum Inform.*, 07(05):1009–1019, 2009.

- [35] C. J. Lawrence, R. K. Dawe, K. R. Christie, D. W. Cleveland, S. C. Dawson, S. A. Endow, L. S.B. Goldstein, H. V. Goodson, N. Hirokawa, J. Howard, R. L. Malmberg, J. R. McIntosh, H. Miki, T. J. Mitchison, Y. Okada, A.S.N. Reddy, W.M. Saxton, M. Schliwa, J.M. Scholey, R.D. Vale, C.E. Walczak, and L. Wordeman. *The Journal of Cell Biology*, 167(1):19–22, 2004.
- [36] G. Lee. *Current Opinion in Cell Biology*, 5(1):88 – 91, 1993.
- [37] J. Lee, M. S. Kim, and Č. Brukner. *Phys. Rev. Lett.*, 91:087902, 2003.
- [38] G.C. Lichtenberg. *Aforismos*. Longseller, 2001.
- [39] J. Lin. *Information Theory, IEEE Transactions on*, 37(1):145–151, jan 1991.
- [40] G. Lindblad. *Commun. Math. Phys.*, 33:305–322, 1973.
- [41] Z.-H. Ma, F.-L. Zhang, and J.-L. Chen. arXiv:0811.3453v1, 2008.
- [42] A. Majtey, P. W. Lamberti, M. T. Martin, and A. Plastino. *Eur. Phys. J. D*, 32:413–419, 2005.
- [43] A. P. Majtey, P. W. Lamberti, and D. P. Prato. *Phys. Rev. A*, 72:052310, 2005.
- [44] A. P. Majtey, A. Borrás, M. Casas, P. W. Lamberti, and A. Plastino. *Int. J. Quantum Inform.*, 6:715–720, 2008.
- [45] P. E. M. F. Mendonça, R. d. J. Napolitano, M. A. Marchioli, C. J. Foster, and Y.-C. Liang. *Phys. Rev. A*, 78:052330, 2008.
- [46] M.A. Nielsen and I.L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, Cambridge, England, 2000.
- [47] K. Nishinari, Y. Okada, A. Schadschneider, and D. Chowdhury. *Phys. Rev. Lett.*, 95(11):118101, 2005.
- [48] R. Nitta, M. Kikkawa, Y. Okada, and N. Hirokawa. *Science*, 305(5684):678–683, 2004.
- [49] Y. Okada and N. Hirokawa. *Science*, 283(5405):1152–1157, 1999.
- [50] Y. Okada and N. Hirokawa. *Proc. Natl. Acad. Sci. U.S.A.*, 97(2):640–645, 2000.
- [51] Y. Okada, H. Higuchi, and N. Hirokawa. *Nature (London)*, 424:574–577, 2003.
- [52] G. Paglini, L. Peris, F. Mascotti, S. Quiroga, and A. Caceres. *Neurochemical Research*, 25(1):37–42, 2000.

- [53] S. Ch. Papasozomenos and L. I. Binder. *Cell Motility and the Cytoskeleton*, 8(3):210–226, 1987.
- [54] A. Parmeggiani, F. Jülicher, L. Peliti, and J. Prost. *EPL (Europhysics Letters)*, 56(4):603, 2001.
- [55] A. Parmeggiani, T. Franosch, and E. Frey. *Phys. Rev. Lett.*, 90(8):086601, 2003.
- [56] W. H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in Fortran. The Art of Scientific Computing. Second Edition*. Cambridge University Press, 1992.
- [57] C.R. Rao. *Theoretical Population Biology*, 21(1):24 – 43, 1982.
- [58] P. Reimann. *Physics Reports*, 361(2–4):57 – 265, 2002.
- [59] M. Ricci, F. De Martini, N. J. Cerf, R. Filip, J. Fiurášek, and C. Macchiavello. *Phys. Rev. Lett.*, 93:170501, 2004.
- [60] W. Roga, M. Fannes, and K. Życzkowski. *Phys. Rev. Lett.*, 105:040505, 2010.
- [61] P. H. Schönemann. *Psychometrika*, 31:1–10, 1966.
- [62] E. Schrödinger. *¿Qué es la vida?* Tusquets Editores, 7 edition, 2008.
- [63] B. Schumacher. *Phys. Rev. A*, 51:2738–2747, 1995.
- [64] C.E. Shannon. *Bell System Technical Journal*, 27:379–423, 623–656, 1948.
- [65] J. Sparacino, P. W. Lamberti, and C. M. Arizmendi. *Phys. Rev. E*, 84:041907, 2011.
- [66] J. Sparacino, M.G. Farías, and P.W. Lamberti. arXiv:1302.2653, 2013.
- [67] I. Toyoshima, K. Kato, M. Sugawara, C. Wada, and O. Masamune. *Neuroscience Letters*, 249(2-3):103 – 106, 1998.
- [68] M. Tribus and E.C. McIrvine. *Scientific American*, 224(3):179–184, 1971.
- [69] B. Trinczek, A. Ebneith, E.M. Mandelkow, and E. Mandelkow. *Journal of Cell Science*, 112(14):2355–2367, 1999.
- [70] A. Uhlmann. *Rep. Math. Phys.*, 9(2):273 – 279, 1976.
- [71] V. Vedral. *Rev. Mod. Phys.*, 74:197–234, 2002.
- [72] V. Vedral, M. B. Plenio, K. Jacobs, and P. L. Knight. *Phys. Rev. A*, 56:4452–4455, 1997.

- [73] A. Wehrl. *Rev. Mod. Phys.*, 50:221–260, 1978.
- [74] W. K. Wootters. *Phys. Rev. D*, 23:357–362, 1981.
- [75] W. H. Zurek. *Rev. Mod. Phys.*, 75:715–775, 2003.

COLOFÓN

Este documento fue tipeado utilizando `classicthesis` desarrollado por André Miede. El estilo fue inspirado por el libro de tipografía "*The Elements of Typographic Style*" de Robert Bringhurst. `classicthesis` está disponible tanto para \LaTeX como para \LyX :

<http://code.google.com/p/classicthesis/>