



Facultad de Ciencias

Modelo teórico de la liberación de neurotransmisor en las sinapsis colaterales de Schaffer

(Theoretical model of neurotransmitter release from Shaffer collateral synapses)

Trabajo de Fin de Máster
para acceder al

MÁSTER EN MATEMÁTICAS Y COMPUTACIÓN

Autor: Cristian Leonardo Ríos López

Director/es: Amparo Gil Gómez

Octubre - 2017

Título: Modelo teórico de la liberación de neurotransmisores en las sinapsis colaterales de Schaffer.

Resumen

Una comprensión cuantitativa de todos los procesos implicados en la liberación de neurotransmisores (aspecto clave para la comunicación entre neuronas) necesita de modelos teóricos del sistema que permitan confrontar datos experimentales de diversas fuentes y verificar hipótesis que se encuentran más allá de las posibilidades experimentales presentes.

El objetivo del presente trabajo es analizar modelos matemáticos y su implementación en algoritmos computacionales para el estudio de la neurotransmisión en un tipo particular de sinapsis: las colaterales de Schaffer. Estos tipos de sinapsis del hipocampo tienen una función muy importante en diversos aspectos relacionados con el aprendizaje y la memoria, y son muy utilizadas por grupos experimentales como modelo celular para el estudio de enfermedades neurodegenerativas como el Alzheimer.

Nuestro trabajo incluye, entre otros aspectos, el estudio de la dinámica de las ecuaciones de estado que describen los principales canales de calcio presentes en las sinapsis colaterales de Schaffer y la modelización de los mecanismos de reacción-difusión que participan en el proceso de neurosecreción.

Palabras clave: modelos matemáticos; neurotransmisión; mecanismos de reacción-difusión; colaterales de Schaffer.

Title: Theoretical model of neurotransmitter release from Schaffer collateral synapses.

Abstract

A quantitative understanding of all processes involved in neurotransmitter release needs a modelling study of the system in order to confront different experimental data and to test hypothesis that lie far beyond present experimental possibilities.

Our goal is to discuss mathematical models and their implementation in computational schemes for the study of neurotransmission in a particular type of synapses: the Schaffer collateral synapses. These synapses, which can be found at the hippocampus, play a very important role in different aspects related to learning and memory. They are very popular as a cellular model for experimental groups working in neurodegenerative diseases such as Alzheimer's disease.

Our work includes, among other aspects, the study of state models for describing the dynamics of the main calcium channels found in Schaffer collateral synapses and the modeling of the reaction-diffusion mechanisms involved in neurosecretion.

Key words: mathematical models; neurotransmission; reaction-diffusion mechanisms; Schaffer collaterals.

Índice general

1. Introducción	1
2. Conceptos Biológicos	3
2.1. Transporte a través de la membrana celular	3
2.1.1. Canales iónicos	6
2.1.2. Canales iónicos de Ca^{2+}	7
2.2. Potenciales de acción	8
2.3. Transporte dentro de la célula	11
2.3.1. Vesículas sinápticas	12
2.4. Liberación del transmisor	13
2.5. Hipocampo	16
3. Modelos matemáticos de los procesos implicados	19
3.1. Modelo de canales iónicos	19
3.1.1. Modelos de Estados Discretos de Markov(MEDMs)	20
3.2. Modelo de secreción	21
3.3. Modelo celular	22
3.3.1. Resolución numérica	23
4. Resultados numéricos	31
4.1. Canal iónico	31
4.2. Difusión	35
4.2.1. Validación	35
4.2.2. Colaterales de Schaffer	36
4.3. Secreción	41
4.4. Conclusiones	46
Referencias	49
A. Despeje de ecuaciones	51
A.1. ecuación (3.5) a (3.6)	51
A.2. ecuación (3.11) a (3.12)	52
A.3. ecuación (3.14) a (3.15)	52
A.4. ecuación (3.17) a (3.18)	53

A.5. ecuación (3.19)	54
A.6. ecuación (3.22) a (3.25)	54
A.7. ecuación (3.23) a (3.24)	56
B. Métodos numéricos para resolución de ecuaciones diferenciales	59
B.1. Método Crank-Nicolson	59
C. Código	61
C.1. Requerimientos	61
C.2. Canal	61
C.3. Secreción	70
C.4. Célula	74

Índice de tablas

2.1. Concentraciones iónicas típicas	3
2.2. Canales de Ca^{2+} dependientes de voltaje en neuronas	8
4.1. Parámetros del modelo de Markov para diferentes tipos de canales de Ca^{2+}	32
4.2. Parámetros usados en los resultados numéricos de validación.	36
4.3. Parámetros usados en los resultados numéricos de las colaterales de Schaffer	40

Índice de figuras

2.1. Neurona	9
2.2. Sinapsis	10
2.3. Hipocampo	17
2.4. Circuito del Hipocampo o circuito trisináptico	17
3.1. Modelo de estados discretos de Markov con dos estados y sus ecuaciones de estado asociadas	20
3.2. Algoritmo para la resolución del sistema $\mathcal{AW}(t + 1) = \mathcal{BW}(t)$	28
4.1. Modelo de canales de calcio y sus ecuaciones de estado asociadas	32
4.2. Intensidad de Corriente I_{Ca} a diferentes voltajes para canales P/Q, N y R.	33
4.3. Probabilidad de apertura a diferentes voltajes para canales P/Q, N y R.	33
4.4. Probabilidad de apertura para canales P/Q, N y R al aplicar un pulso de voltaje.	34
4.5. Concentración de calcio sin <i>buffer</i> a distintas distancias de la membrana celular al aplicar un pulso de corriente.	37
4.6. Concentración de calcio con <i>buffer</i> a distintas distancias de la membrana celular al aplicar un pulso de corriente.	38
4.7. Concentración de <i>buffer</i> a distintas distancias de la membrana celular al aplicar un pulso de corriente.	39
4.8. Concentración de calcio sin <i>buffer</i> a distintas distancias de la membrana celular al aplicar un pulso de voltaje. Colaterales de Schaffer	42
4.9. Concentración de calcio con <i>buffer</i> a distintas distancias de la membrana celular al aplicar un pulso de voltaje. Colaterales de Schaffer.	43
4.10. Concentración de <i>buffer</i> a distintas distancias de la membrana celular al aplicar un pulso de voltaje. Colaterales de Schaffer	44
4.11. Esquema de sensores de Ca^{2+} y sus ecuaciones de estado asociadas	45
4.12. Esquema de fusión vesicular y sus ecuaciones de estado asociadas	45
4.13. Cantidad de vesículas que liberan su contenido.	47

Capítulo 1

Introducción

La utilización de modelos matemáticos para explicar el comportamiento de sistemas biológicos se ha popularizado en los últimos años. Esto es debido, principalmente, al perfeccionamiento de las técnicas experimentales que han permitido una mayor comprensión de los elementos básicos que intervienen en el sistema.

Un ejemplo de ello es el proceso de liberación de neurotransmisor en terminales sinápticas, siendo este el objeto de estudio de esta tesis de máster. En particular, los objetivos de este trabajo son:

1. Identificar los canales iónicos principales involucrados en la dinámica del calcio en las neuronas del hipocampo. Analizar los modelos cinéticos propuestos, resolver las ecuaciones diferenciales que los describen y ser capaz de reproducir las curvas Intensidad-Voltaje que caracterizan a los canales.
2. Plantear un modelo de neurona esférica para describir los procesos de reacción-difusión que describen la dinámica del calcio en el citoplasma de la neurona. Para las reacciones en el medio se considerará la presencia de un “buffer” (quelante) endógeno inmóvil que interacciona con los iones de calcio de acuerdo a la reacción cinética reversible:



3. Discutir y plantear un método de resolución numérica para las ecuaciones de reacción-difusión en el modelo de neurona esférica.
4. Buscar en la literatura datos sobre densidad de canales típicas en las colaterales de Schaffer. Este dato, junto con el modelo estudiado en (1) determinará cómo es la corriente de entrada que se necesitará en nuestro modelo de neurona esférica para simular las condiciones de un experimento de fijación de voltaje.
5. Acoplar (2)+(4) para obtener numéricamente concentraciones de calcio a diferentes distancias de la membrana celular (las “capas” de nuestra neurona esférica).

6. Con las concentraciones de calcio obtenidas en la primera capa de la neurona esférica, resolver numéricamente las ecuaciones que describen la cinética de secreción de vesículas.

Capítulo 2

Conceptos Biológicos

2.1. Transporte a través de la membrana celular

Las membranas celulares son cruciales para la vida, ya que estas encierran la célula, definen sus límites y mantiene las diferencias esenciales entre lo que es el interior celular, conocido como citosol, y el ambiente extracelular. Las membranas no solo definen las células, también definen una variedad de organelas celulares dentro de la célula, diferenciado éstas del citosol circundante.

A través de la membrana plasmática se pueden ir acumulando cargas iónicas, esto debido a que la membrana es una bicapa lipídica con un interior hidrófobo altamente impermeable a moléculas polares. Estas cargas iónicas pueden ser usadas por la célula para efectuar diversos procesos como pueden ser la síntesis de ATP, el movimiento de solutos a través de la membrana, o incluso en células nerviosas y musculares, producir y transmitir señales eléctricas.

Eléctricamente, una célula debe de ser neutra, esto es, debe de contener igual cantidad de cargas negativas y positivas. En la tabla 2.1 se encuentran las concentraciones de algunos cationes y aniones en estado libre dentro de la célula (también pueden estar ligados a proteínas o a otras sustancias en la célula).

Ion	Concentración In-tracelular (mM)	Concentración Ex-tracelular (mM)
Na ⁺	5 - 15	145
K ⁺	140	5
Mg ²⁺	0.5	1 - 2
Ca ²⁺	10 ⁻⁴	1 - 2
H ⁺	7x10 ⁻⁵ o 10 ^{-7,2} M o pH 7.2	4x10 ⁻⁵ o 10 ^{-7,4} M o pH 7.4
Cl ⁻	5 - 15	110

Tabla 2.1: Concentraciones iónicas típicas. Adaptado de (Alberts y cols., 2008, pg. 652).

Gran cantidad de elementos deben de entrar y salir de la célula para su correcto funcionamiento, atravesando la barrera que forma la membrana celular. Estos elementos pueden ser azúcares, iones, aminoácidos, enzimas, etc, por lo que se hace necesario una manera de realizar este transporte de sustancias, algunas en una dirección y otras en la dirección opuesta a través de la membrana celular.

La importancia del transporte a través de la membrana se refleja en la gran cantidad de genes en todos los organismos que codifican proteínas de transporte, entre el 15 % y el 30 %, y en la cantidad de energía invertida en dicho proceso.

Si se diera el tiempo suficiente, virtualmente cualquier molécula pasaría la membrana celular siguiendo un gradiente de concentración, pero la tasa de difusión varía enormemente y depende parcialmente del tamaño de la molécula y de su solubilidad relativa en aceite (hidrofóbica o no polar). Entre más pequeña y más hidrófoba más rápido puede transportarse, esto debido en parte a que la membrana también es hidrofóbica. Por ejemplo, pequeñas moléculas no polares como el O_2 y el CO_2 se disuelven fácilmente en la bicapa lipídica, como también difunden fácilmente pequeñas moléculas polares no cargadas como el agua y la urea, aunque mucho más lentamente. La bicapa lipídica es altamente impermeable a moléculas cargadas (iones), no importa qué tan pequeñas sean.

Existen dos maneras principales en que puede ocurrir el transporte a través de la membrana, por transportadores o por canales. Estos están conformados por proteínas, todas las cuales son proteínas transmembrana (atravesan la membrana, en ocasiones múltiples veces) y lo que hacen es habilitar a solutos hidrofílicos específicos para que atraviesen la membrana sin entrar en contacto directo con el interior hidrofóbico de la bicapa lipídica.

Los transportadores, también conocidos como acarreadores, son específicos por cada tipo de soluto. Estos se ligan al soluto y después de una serie de cambios conformacionales, transfieren el soluto ligado a través de la membrana.

En contraste, los canales interactúan de una manera más débil con el soluto. Estos forman poros que se extienden a través de la bicapa lipídica y cuando se abren permiten que un soluto específico (son altamente selectivos) pase a través de ellos y por tanto atraviese la membrana. El transporte efectuado por estos es mucho más rápido que el llevado a cabo por los acarreadores. En las neuronas, los canales protéicos son altamente sofisticados.

El transporte puede ser realizado consumiendo o no energía, si se consume energía se le llama transporte activo, de lo contrario se le conoce como transporte pasivo.

El transporte pasivo, también llamado difusión facilitada, es dirigido por un gradiente de concentración o un gradiente de potencial eléctrico, lo que junto se conoce como gradiente electroquímico. En contraste, el transporte activo se hace en contra de un gradiente de concentración y la energía requerida para lograrlo se puede obtener por procesos metabólicos o por medio de un gradiente iónico.

Más específicamente, el transporte activo se puede llevar a cabo de alguna de las siguientes maneras:

- Por transporte acoplado, también llamado transporte activo secundario. Realiza

un transporte de una pareja de solutos, uno en contra y el otro a favor de un gradiente de concentración. Por ejemplo, el caso del Na^+ y la glucosa: se ingresa glucosa a la célula en contra de un gradiente de glucosa pero a favor de un gradiente de Na^+ y el gradiente de Na^+ es creado por una bomba Na^+/K^+ que sí requiere hidrólisis de ATP.

- Por bombas de ATP, llamado transporte activo primario. Realiza un transporte a través de la membrana en contra de un gradiente de concentración usando la energía proporcionada por la hidrólisis de ATP.

El movimiento transmembrana de moléculas pequeñas mediante transportadores puede ser tanto de forma activa como pasiva, mientras que el mediado por canales siempre se da de manera pasiva.

Los transportadores conocidos como *uniportes* mueven un sólo soluto a la vez en alguna dirección, mientras que los *simportes* y *antiportes* mueven dos solutos al tiempo, los primeros mueven ambos en la misma dirección, mientras que los segundos mueven un soluto en una dirección y el otro en la dirección opuesta.

La fuerte unión entre los dos solutos a transportar permite aprovechar la energía almacenada en el gradiente electroquímico de uno de los solutos, típicamente un ion, para transportar el otro soluto. Esto obliga a la célula a mantener los gradientes iónicos estables, trabajo que es realizado por las bombas iónicas. Este principio puede funcionar en cualquier dirección y entre mayor sea el gradiente electroquímico, mayor será la tasa de soluto que es transportado.

De manera general, las clases de bombas impulsadas por ATP, conocidas como ATPasas son:

- Bombas tipo P: Están funcional y estructuralmente relacionadas con las proteínas transmembrana. Son llamadas de tipo P porque se fosforilan a sí mismas durante el ciclo de bombeo. Esta clase incluye muchas de las bombas iónicas que son responsable de producir y mantener los gradientes de Na^+ , K^+ , Ca^{2+} e H^+ a través de las membranas celulares.
- Bombas tipo F: Son proteínas tipo turbina construidas a partir de múltiples subunidades protéicas. A menudo son llamadas ATP sintasas porque normalmente trabajan al reverso, o sea, en lugar de utilizar la hidrólisis de ATP para dirigir el bombeo de H^+ , usan el gradiente de H^+ a través de la membrana para dirigir la síntesis de ATP a partir de ADP y fosfato.
- Bombas tipo V: Son una familia de bombas relacionadas estructuralmente con las bombas tipo F, estas normalmente bombean H^+ en lugar de sintetizar ATP. Están presentes principalmente en membranas de organelas que deben ser ácidas, como por ejemplo las vesículas sinápticas.
- Transportadores ABC: Principalmente bombean pequeñas moléculas a través de la membrana, en contraste con las bombas tipo P y tipo F o V, las cuales exclusivamente transportan iones.

Las células eucariotas mantienen una muy baja concentración de Ca^{2+} libre en su citosol (alrededor de 10^{-7} M) en contraste con una concentración mucho mayor de Ca^{2+} extracelular (alrededor de 10^{-3} M). Incluso una pequeña entrada de Ca^{2+} incrementa significativamente la concentración de Ca^{2+} libre en el citosol, es por ello que el flujo de Ca^{2+} , siguiendo el gradiente de concentración en respuesta a señales extracelulares, es uno de los medios de transmisión de señales más rápido a través de la membrana plasmática. Es importante, por consiguiente, que la célula mantenga este pronunciado gradiente de concentración a través de la membrana plasmática, trabajo realizado por los transportadores de Ca^{2+} que activamente bombean Ca^{2+} fuera de la célula. Uno de estos es la ATPasa tipo P de Ca^{2+} , el otro es un *antiporter* que es impulsado por el gradiente electroquímico de Na^+ a través de la membrana, también llamado, intercambiador $\text{Ca}^{2+}/\text{Na}^+$.

2.1.1. Canales iónicos

La mayoría de canales protéicos son poros altamente selectivos que pueden abrirse y cerrarse rápidamente. Son llamados canales iónicos debido a que estas proteínas están involucradas específicamente con el transporte de iones inorgánicos.

Como se indica en (Kandel, Schwartz, Jessell, Siegelbaum, y Hudspeth, 2013, Capítulo 5), el mal funcionamiento de los canales iónicos puede desencadenar una amplia variedad de enfermedades, tales como la fibrosis quística, enfermedades del músculo esquelético y ciertos tipos de arritmias cardíacas. Además, los canales iónicos son a menudo el sitio de acción de drogas, toxinas y venenos, por lo que es más que claro la importancia que estos tienen tanto en la fisiología normal como en la patofisiología.

En cuestiones de eficiencia de transporte, los canales presentan una ventaja sobre los transportadores ya que hasta 100 millones de iones pueden pasar por un canal abierto cada segundo, lo que es una tasa 10^5 veces más grande que el más rápido transporte mediado por cualquier proteína de transporte conocida. El transporte que median los canales es siempre pasivo a favor de un gradiente de concentración.

Los canales iónicos presentan dos propiedades importantes. La primera es que muestran una alta selectividad, de esta manera, sólo los iones con el tamaño y la carga adecuada logran pasar a través de él. La zona más estrecha del canal es llamada filtro de selectividad y limita la tasa de paso de iones, lo que genera un nivel de saturación y por tanto una tasa máxima de flujo. La segunda es que los canales iónicos no están constantemente abiertos, por el contrario, estos están cerrados, se abren brevemente y se cierran de nuevo. Mas aún, con una estimulación prolongada, la mayoría de los canales entran en un estado cerrado desensibilizado o inactivado en el que son refractarios a una apertura adicional hasta que el estímulo se haya removido. Por tanto, de manera general, un canal iónico puede estar en alguno de tres estados: abierto, cerrado o inactivado.

En la mayoría de los casos, los canales se abren en respuesta a un estímulo específico. Los principales tipos de estímulo que son conocidos de causar que los canales iónicos se abran son el voltaje a través de la membrana (voltaje-gated channels), la tensión mecánica (mechanically-gated channels) o de uniones a ligando (ligand-gated

channels). El ligando puede ser un mediador extracelular, por ejemplo un neurotransmisor (transmitter-gated channels), o un mediador intracelular como un ion (ion-gated channels) o un nucleótido (nucleotide-gated channels).

Se han descrito más de 100 tipos de canales iónicos y cada día se descubren más, cada uno caracterizado por el tipo de ion que conduce, su mecanismo de apertura y su abundancia y localización en la célula.

Los canales iónicos son los responsables de la excitabilidad eléctrica de las células musculares y median la mayoría de formas de señales eléctricas en el sistema nervioso. Cada neurona puede tener típicamente 10 o más tipos de canales iónicos, localizados en diferentes dominios de su membrana plasmática. A pesar de todo, el canal iónico más común son los permeables a K^+ .

Un potencial de membrana surge cuando existe una diferencia en la carga eléctrica en los dos lados de la membrana, debido a un leve exceso de iones positivos sobre iones negativos a un lado de la membrana. Tal diferencia iónica resulta tanto del bombeo electrogénico activo como de la difusión pasiva de iones, aunque esta última es la que mayor contribución hace al potencial eléctrico a través de la membrana.

El número de iones que van a formar la capa de carga adyacente a la membrana es diminuto en comparación con la cantidad total de iones dentro de la célula. Por ejemplo, el movimiento de 6000 iones de Na^+ a través de un μm^2 de membrana pueden acarrear suficiente carga para cambiar el potencial de membrana en unos 100 mV (hay al rededor de 3×10^7 iones de Na^+ en una célula típica). Estos movimientos de carga son generalmente rápidos, tomando algunos pocos milisegundos o menos.

Aunque los gradientes de K^+ siempre tienen una mayor influencia sobre el potencial de reposo, los gradientes de otros iones también pueden afectar significativamente, entre más permeable la membrana para ese ion más fuertemente el potencial de membrana tiende a ser conducido a un valor de equilibrio para ese ion. De esta manera, los cambios en la permeabilidad de la membrana para algún ion pueden producir cambios significativos sobre el potencial de membrana.

Para más información acerca de la membrana celular y el transporte a través de ella se puede consultar (Alberts y cols., 2008, Capítulo 11).

2.1.2. Canales iónicos de Ca^{2+}

Los canales de Ca^{2+} son encontrados en todas las células nerviosas y en muchas otras células no nerviosas. Las neuronas en particular pueden tener 5 clases de canales de Ca^{2+} mediados por voltaje: los canales tipo L, los canales tipo P/Q, los canales tipo N, los canales tipo R y los canales tipo T; cada uno codificado por distintos genes o familias de genes.

Cada tipo de canal presenta distintas especificaciones biofísicas, propiedades farmacológicas y funciones fisiológicas. Estas se pueden observar en la tabla 2.2.

Los canales L, P/Q, N y R requieren una despolarización bastante fuerte para activarse y son comúnmente referidos como canales de Ca^{2+} activados por alto voltaje (high-voltaje-activated Ca^{2+} channels o HVA) mientras que los canales T se abren en respuesta a pequeñas despolarizaciones, al rededor del umbral que genera el potencial

de acción (-60mV y -40mV) y son llamados canales de Ca^{2+} activados por bajo voltaje (low-voltaje-activated Ca^{2+} channels o LVA).

Los canales T ayudan a controlar la excitabilidad en el potencial de reposo y son una importante fuente de corriente excitatoria que impulsa la actividad de marcapasos rítmicos en algunas células tanto del cerebro como del corazón.

En neuronas, la rápida liberación de transmisores convencionales como la ACh (Acetilcolina) y el glutamato, asociados con sinapsis de rápida transmisión es mediada principalmente por canales de Ca^{2+} P/Q y N porque estos son los tipos de canales concentrados en las zonas activas. Los canales L no son encontrados en las zonas activas por lo que normalmente no contribuyen a la rápida liberación de transmisores convencionales, sin embargo, el flujo de Ca^{2+} a través de los canales L es importante para formas lentas de liberación que no ocurren en zonas activas, como la liberación de neuropéptidos en neuronas y de hormonas en células endocrinas.

Para una descripción más detallada sobre los canales iónicos de Ca^{2+} puede consultarse (Kandel y cols., 2013, Capítulo 12).

Tipo de canal Ca^{2+}	Tejido	Bloqueador	Dependencia de voltaje	Función
L	Músculo, Neuronas	Dihidropiridonas	HVA	Contracción, liberación lenta
P/Q	Neuronas	ω -agatoxina (Veneno de araña)	HVA	Liberación rápida +++
N	Neuronas	ω -conotoxina (Veneno de serpiente)	HVA	Liberación rápida ++
R	Neuronas	SNR-482 (Veneno de tarántula)	HVA	Liberación rápida +
T	Músculo, Neuronas	Mibefradil	LVA	Disparador de marcapasos

Tabla 2.2: Canales de Ca^{2+} dependientes de voltaje en neuronas. Adaptado de (Kandel y cols., 2013, pg. 267).

2.2. Potenciales de acción

La tarea fundamental de una neurona o célula nerviosa es recibir, conducir y transmitir señales. Para realizar esta función, las neuronas son a menudo extremadamente alargadas. Una neurona consiste en un cuerpo celular (que contiene el núcleo) con un número de delgados procesos que irradian fuera de este. Usualmente uno largo, el axón,

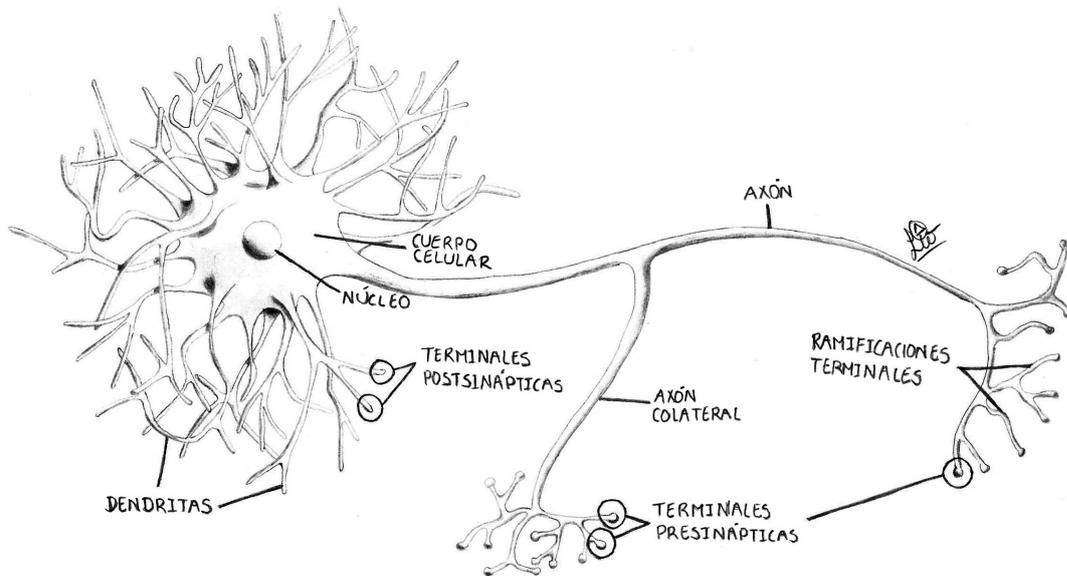


Figura 2.1: Neurona. Elaboración propia. Adaptado de (Kandel y cols., 2013, pg. 25).

conduce las señales lejos del cuerpo celular hacia el objetivo distante, y varios otros, las dendritas, se extienden desde el cuerpo celular como antenas, proveyendo una gran área para recibir señales de otras neuronas. Un axón típico se ramifica en su final pasando su mensaje a muchas células objetivos simultáneamente. Aunque, como se menciona en (Cucic, 2017), en algunas neuronas el axón puede presentar ramificaciones en su trayecto, dichas ramificaciones reciben el nombre de axones colaterales y usualmente hacen contacto con las dendritas o el cuerpo celular de sí misma o en ocasiones de otras neuronas, por lo que su función es proveer modulación y regulación en los patrones de disparo de la neurona. En la figura 2.1 se puede observar la representación de una neurona típica.

La forma de la señal transmitida por las neuronas es siempre la misma. Ésta consiste en cambios de potencial eléctrico a través de la membrana plasmática de la neurona generados por un estímulo eléctrico que excede cierto umbral. La señal se transmite porque la perturbación eléctrica producida en una parte de la membrana se propaga rápidamente a otra parte como una onda, aunque la señal se debilita conforme viaja o se aleja de la fuente, a menos que la neurona gaste energía amplificandola mientras viaja. Para distancias cortas, esta atenuación en la señal no tiene importancia, de hecho, muchas neuronas pequeñas conducen su señal pasivamente, sin amplificación. Las neuronas largas emplean un mecanismo de señalización activo, lo que es una de sus más sorprendentes características.

Esta onda de excitación eléctrica que viaja es conocida como potencial de acción o impulso nervioso y puede llevar un mensaje sin atenuación de una neurona a otra a una velocidad de 100 metros por segundo o más.

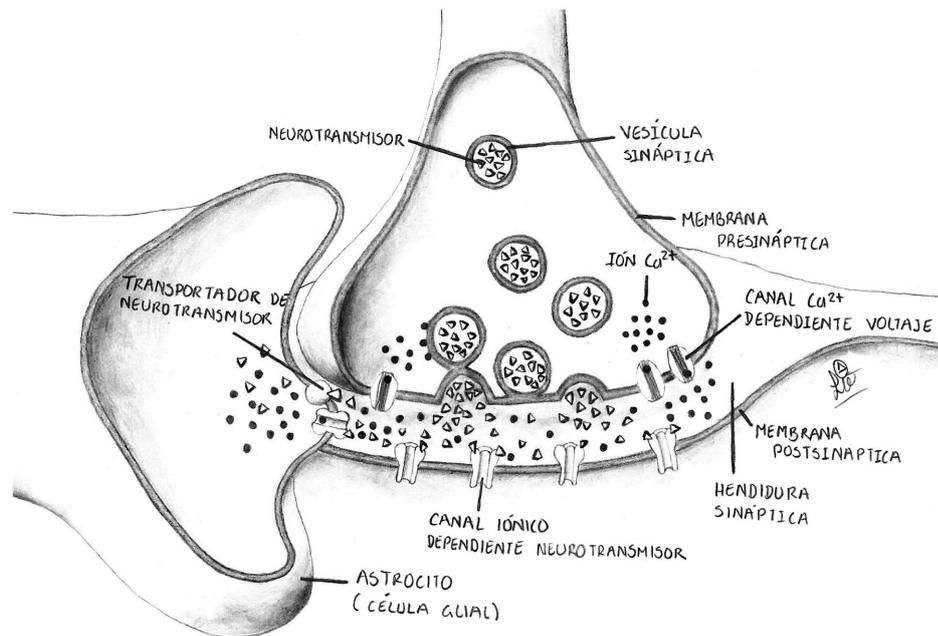


Figura 2.2: Sinapsis. Elaboración propia. Adaptado de (Alberts y cols., 2008, pgs. 94, 277 y 1149).

Los potenciales de acción son una consecuencia directa de los canales iónicos de Na^+ y K^+ dependientes de voltaje que se encuentran presentes en la membrana plasmática de todas las células excitables.

Las señales neuronales son transmitidas de una célula a otra en sitios especializados de contacto llamados sinapsis. El mecanismo usual de transmisión es indirecto. Las células son eléctricamente aisladas unas de otras, ya que la célula presináptica está separada de la postsináptica por la hendidura sináptica. En la figura 2.2 se puede observar la representación de una sinapsis. Un cambio en el potencial eléctrico de la célula presináptica desencadena la liberación de una pequeña señal molecular llamada neurotransmisor. El neurotransmisor se difunde rápidamente a través de la hendidura sináptica y provoca un cambio eléctrico en la célula postsináptica al ligarse a canales iónicos dependientes de neurotransmisor haciendo que se abran.

Una vez el neurotransmisor es liberado, éste rápidamente es eliminado al ser destruido por enzimas en la hendidura sináptica, reciclado por la terminal nerviosa que lo liberó o tomado por las células gliales que rodean la hendidura. El reciclaje es mediado por una variedad de transportadores de Na^+ dependientes de neurotransmisor. La rápida remoción asegura una precisión, tanto espacial como temporal, de la señal en una sinapsis.

Existen dos familias de neurotransmisores, los excitatorios y los inhibitorios. Los neurotransmisores excitatorios abren canales catiónicos, causando una entrada de Na^+ que despolariza la membrana postsináptica. Los neurotransmisores inhibitorios abren

canales de Cl^- o K^+ que suprimen el disparo y hace muy difícil la influencia excitatoria para despolarizar la membrana postsináptica. Muchos neurotransmisores pueden ser excitadores o inhibidores dependiendo de donde son liberados, a qué receptores se ligan y las condiciones iónicas en las que se encuentran.

La acetilcolina, el glutamato y la serotonina son usualmente neurotransmisores excitatorios. Mientras que el ácido gama-aminobutírico o GABA y la glicina son usualmente neurotransmisores inhibitorios.

Para más información sobre potenciales de acción y neurotransmisores, puede consultarse (Alberts y cols., 2008, Capítulo 11).

2.3. Transporte dentro de la célula

Al interior celular se deben de mover muchos componentes, caso especial son las proteínas sintetizadas que deben de ser llevadas bien sea a la membrana celular o fuera de la célula, si esta es un tipo de célula secretora.

Las vesículas destinadas a la membrana plasmática normalmente salen del TGN (Trans-Golgi Network) en un flujo constante como túbulos de forma irregular. Las proteínas de membrana y los lípidos de estas vesículas proveen nuevos componentes a la membrana plasmática mientras que las proteínas solubles dentro de las vesículas son liberadas al espacio extracelular. La fusión de la vesícula con la membrana plasmática es llamada exocitosis.

Lo anterior es conocido como la vía secretora constitutiva o vía por defecto, y todas las células llevan a cabo dicho proceso para su correcto funcionamiento, pero algunas células secretoras especializadas tienen un segundo camino de secreción en el que proteínas solubles y otras sustancias son almacenadas inicialmente en vesículas secretoras para una posterior liberación. Este mecanismo es conocido como la vía secretora regulada y se encuentra principalmente en células especializadas que secretan productos bajo rápida demanda, como pueden ser hormonas, neurotransmisores o enzimas digestivas.

Las vesículas secretoras también se forman en el TGN y liberan su contenido al exterior celular por exocitosis en respuesta de señales específicas.

La formación vesicular detallada es un intrincado proceso que se encuentra muy bien descrito en (Alberts y cols., 2008, Capítulo 13), el que puede ser consultado para una mayor información.

Finalizado el proceso de formación vesicular, las vesículas secretoras maduras están densamente llenas con su contenido y la célula puede secretar una gran cantidad de material por exocitosis inmediatamente ocurre alguna acción desencadenante.

Una vez la vesícula secretora se ha formado en el TGN y ha sido cargada, esta tiene que alcanzar el sitio de secreción, que en algunas células se encuentra muy lejos desde su lugar de formación. El ejemplo más extremo son las células nerviosas, ya que la terminal nerviosa al final del axón puede estar a un metro o más de distancia del cuerpo celular.

Mientras que las vesículas que contienen material constitutivo para la membrana celular se fusionan con esta una vez llegan allí, las vesículas secretoras en la vía regulada

esperan en la membrana hasta que la célula recibe la señal de secreción y entonces se fusionan.

En las terminales nerviosas, la señal inicial para la exocitosis es usualmente una excitación eléctrica (un potencial de acción) desencadenado por el ligado de un transmisor químico a los receptores en algún lugar de la superficie de la misma célula. Cuando el potencial de acción alcanza la terminal nerviosa, causa un flujo de entrada de Ca^{2+} a través de canales de Ca^{2+} dependientes de voltaje. Los iones de Ca^{2+} se unen a sensores específicos que desencadenan la fusión de las vesículas secretoras con la membrana plasmática produciendo la liberación de su contenido al medio extracelular.

La velocidad de la liberación del transmisor, que toma sólo milisegundos, indica que las proteínas que median la reacción de fusión no experimentan reorganizaciones complejas o de múltiples pasos, dicho de otra manera, las vesículas se encuentran acopladas a la membrana plasmática presináptica y preparadas para facilitar una rápida fusión.

En una sinapsis típica, sólo algunas pocas vesículas de las que se han acoplado parecen ser preparadas y quedar listas para la exocitosis, lo que permite que cada sinapsis dispare una y otra vez en una rápida sucesión. Con cada disparo, nuevas vesículas sinápticas son preparadas para reemplazar a las que se han fusionado y han liberado su contenido.

Cuando una vesícula secretora se fusiona con la membrana plasmática, su contenido es descargado de la célula por exocitosis, y su membrana pasa a ser parte de la membrana plasmática. Aunque esto puede incrementar enormemente la superficie de la membrana plasmática, es algo transitorio, porque los componentes de la membrana son removidos por endocitosis casi tan rápido como son agregados por exocitosis.

2.3.1. Vesículas sinápticas

Las células nerviosas y algunas células endocrinas contienen dos tipos de vesículas secretoras. Como todas las células secretoras, estas células empaquetan proteínas y péptidos en un núcleo denso en vesículas secretoras de la manera habitual, explicada anteriormente, para su posterior liberación por la vía secretora regulada. Sin embargo, también usan otra clase de minúsculas vesículas secretoras especializadas llamadas vesículas sinápticas, que son generadas de una manera diferente.

En las células nerviosas, estas vesículas almacenan pequeñas moléculas de neurotransmisor, como la acetilcolina, el glutamato, la glicina o el GABA, que median la rápida señalización de una célula a otra en sinapsis químicas. Algunas neuronas disparan más de mil veces por segundo, liberando neurotransmisor cada vez.

Debido a que las neuronas deben de responder rápida y repetidamente al estímulo que reciben, las vesículas se deben de reponer con gran velocidad después de cada descarga, es por esto que la mayoría de las vesículas sinápticas no son generadas a partir de la membrana del TGN en el cuerpo celular neuronal, sino que se generan por reciclaje de la membrana plasmática en la terminal nerviosa.

Se cree que los componentes de la membrana de las vesículas sinápticas son inicialmente entregados a la membrana plasmática por la vía secretora constitutiva y luego

recicladas por endocitosis, llenadas nuevamente con neurotransmisor por la acción de transportadores especializados que usan un gradiente de H^+ para captar neurotransmisores desde el citosol, y reenviadas a la membrana plasmática convirtiéndose en nuevas vesículas sinápticas.

Debido a que las vesículas sinápticas son abundantes y relativamente uniformes en tamaño, han podido ser purificadas en gran cantidad y por tanto son el organelo celular mejor caracterizado. Un análisis proteómico cuidadoso ha logrado identificar todos sus componentes. Cada vesícula contiene al rededor de 50 diferentes proteínas integrales de membrana con una amplia variedad y relativa abundancia. La sinaptoproteína v-SNARE es la más abundante, alrededor de 70 copias por vesícula. En contraste, la V-ATPasa que usa la hidrólisis de ATP para bombear H^+ al lumen vesicular, está presente en 1 o 2 copias por vesícula. El gradiente de H^+ provee energía para para importar el neurotransmisor por medio de un antiporter H^+ /neurotransmisor quien carga la vesícula con alrededor de 1800 moléculas de neurotransmisor.

Como se mencionó anteriormente, previo al paso de fusión dependiente de calcio de las vesículas sinápticas, estas experimentan un proceso de preparación para su liberación. Este proceso, como se detalla en (*Ciclo de las vesículas sinápticas*, 2017), puede ser explicado en dos pasos. Primero se produce un acercamiento espacial a las zonas donde se generan los microdominios de calcio en la zona activa, lo que se conoce con el nombre de amarre a la zona activa (del inglés docking) y un segundo paso posterior que permite el ensamblaje de complejos SNARE y recibe el nombre de preparación de las vesículas (del inglés priming). El papel del calcio sería inducir la reacción del complejo SNARE que lleva a la fusión de las membranas plasmática y vesicular.

El complejo SNARE que se forma entre proteínas de membrana plasmática y proteínas vesiculares supone un nexo físico entre la membrana presináptica y la vesicular, confiriendo proximidad y posibilitando el proceso de fusión. Por lo anterior, se podría definir el complejo SNARE como el verdadero motor molecular de la exocitosis al producir fuerzas de torsión que aproximan las membranas e inducen la curvatura necesaria para que suceda el evento de fusión.

2.4. Liberación del transmisor

Cuando una neurona presináptica es estimulada dispara un potencial de acción, y después de un breve retraso un Potencial Excitatorio Postsináptico o EPSP (Excitatory Postsynaptic Potencial) lo suficientemente grande para desencadenar un potencial de acción ocurre en la célula postsináptica.

La relación entre el pico de voltaje presináptico y el EPSP es logarítmica. Es necesario al menos 40 mV en el pico presináptico para empezar a producir un EPSP. Así, la cantidad de transmisor liberado está en función de la cantidad de despolarización presináptica, pero una despolarización más grande que cierto nivel no causa ninguna liberación de transmisor adicional.

Para la liberación del transmisor no es necesario ni el flujo de Na^+ ni el de K^+ que

generan el potencial de acción. Esto se logró determinar al bloquear dichos canales con toxinas e inyectando corriente en la terminal presináptica para producir un potencial de acción, observándose que se seguía produciendo un EPSP. También se observó que mantener una larga despolarización presináptica produce un largo y mantenido EPSP. Se encontró además que una gran cantidad de Ca^{2+} extracelular mejoraba la liberación de transmisor, mientras que un bajo nivel lo reducía llegando incluso a bloquear la liberación del transmisor. Se concluye entonces que el Ca^{2+} debe de estar implicado, pero dado que la liberación de transmisor es un proceso intracelular, implica que el Ca^{2+} debe de entrar a la célula.

Los canales de Ca^{2+} de apertura por voltaje fueron encontrados por primera vez en el axón del calamar gigante. Su apertura resultaba en un gran flujo de Ca^{2+} debido a la gran fuerza motriz electroquímica. La concentración extracelular de Ca^{2+} (aproximadamente 2 mM en vertebrados) es normalmente 4 veces mayor que la concentración intracelular (aproximadamente 0.2 μM) en reposo. Sin embargo, debido a que estos canales de Ca^{2+} están escasamente distribuidos a lo largo del axón, ellos no pueden, por si mismos, proveer una corriente suficiente para producir un potencial de acción regenerativo.

Los canales iónicos de Ca^{2+} están concentrados en la terminal presináptica en zonas activas, en estos sitios es donde se libera el neurotransmisor. Los iones de Ca^{2+} no difunden grandes distancias desde su sitio de entrada porque los iones libres de Ca^{2+} son rápidamente amortiguados (*buffered*) por proteínas que ligan al calcio. Como resultado, el flujo de entrada de Ca^{2+} crea un aumento local agudo en la concentración de Ca^{2+} en las zonas activas

Una característica en la liberación de neurotransmisor en todas las sinapsis es la pronunciada y no lineal dependencia del flujo de entrada de Ca^{2+} , por ejemplo, duplicar el flujo de entrada de Ca^{2+} puede incrementar la cantidad de neurotransmisor liberado 16 veces. Esta relación indica que en algún sitio, la unión cooperativa de varios iones de Ca^{2+} es requerida para desencadenar la liberación.

Las terminales sinápticas de una neurona típica en el cerebro contienen una única zona activa, pero algunas como el Cáliz de Held, pueden llegar a tener cerca de mil zonas activas y cada una funciona como una sinapsis independiente.

El retardo que existe entre el potencial de acción presináptico y el Potencial Excitatorio Postsináptico es llamado retardo sináptico y equivale a entre 1 y 2 ms. Esto se debe a que los canales de Ca^{2+} se abren más lentamente que los canales de Na^+ y los iones de Ca^{2+} no empieza a entrar a la terminal presináptica hasta que la membrana se ha empezado a repolarizar. Sorprendentemente, una vez el Ca^{2+} entra a la terminal, el transmisor es liberado rápidamente con un retardo de unos pocos cientos microsegundos. La sorprendente velocidad de acción del Ca^{2+} indica que antes de la entrada de Ca^{2+} , la maquinaria bioquímica que subyace al proceso de liberación ya debe de encontrarse ensamblada y lista, como se ha mencionado anteriormente.

Un potencial de acción presináptico normalmente produce solo un leve aumento de la concentración de Ca^{2+} presináptico porque los canales de Ca^{2+} se abren sólo por un corto periodo de tiempo además de que el flujo de entrada de Ca^{2+} está localizado

sólo en las zonas activas. Un aumento en la concentración de Ca^{2+} menor a $1 \mu\text{M}$ es suficiente para inducir la liberación de algún transmisor, pero aproximadamente de 10 a $30 \mu\text{M}$ de Ca^{2+} es necesario para liberar la cantidad de transmisor que normalmente es observada durante un potencial de acción.

La regulación del flujo de entrada de Ca^{2+} en la terminal presináptica controla la cantidad de transmisor liberado y por tanto la fuerza de la transmisión sináptica. El transmisor es liberado en cantidades discretas llamadas quantum. Cada quantum de transmisor produce un potencial postsináptico de tamaño fijo llamado potencial sináptico cuántico y el potencial postsináptico total es compuesto por un gran número de potenciales cuánticos. Los EPSP parecen ser suavemente graduados en amplitud solo porque cada quantum (unidad) de potencial es pequeño en comparación al potencial total.

Cada vesícula sináptica almacena un quantum de transmisor y libera su contenido a la hendidura sináptica en un sitio especializado para tal liberación, la zona activa, de una manera todo o nada. En las neuronas del sistema central las zonas activas tienen forma de disco con un área aproximada de $0.1 \mu\text{m}^2$ con un punto de proyección densa apuntando al citoplasma.

La eficacia de la liberación de neurotransmisor desde una única célula presináptica a una única célula postsináptica varía ampliamente en el sistema nervioso y depende de varios factores:

- El número de sinapsis entre un par de células presináptica y postsináptica.
- El número de zonas activas en una terminal sináptica individual.
- La probabilidad de que un potencial de acción presináptico desencadene la liberación de uno o más quantum de transmisor en la zona activa.

La liberación de un quantum de transmisor es un evento aleatorio. El destino de cada quantum de transmisor en respuesta a un potencial de acción sólo tiene dos posibles resultados, el quantum es o no es liberado. Este evento se parece a una binomial o a una prueba de Bernoulli. Además, la probabilidad de que un quantum sea liberado por un potencial de acción es independiente de la probabilidad de que otros quantums sean liberados por ese mismo potencial de acción. Por tanto, para una población de quantums liberables, cada potencial de acción representa una serie de ensayos binomiales independientes.

En una distribución binomial p significa la probabilidad promedio de éxito (la probabilidad de que cualquier quantum dado pueda ser liberado) y q representa la probabilidad media de fallo ($1 - p$). Tanto la probabilidad p como la cantidad n de quantums fácilmente liberables se asume constante. El producto de n y p produce una estimación m del número medio de quantums que se liberan para formar el potencial final. Los valores de m varían aproximadamente entre 100 y 300 en sinapsis neuromusculares de vertebrados y la sinapsis del calamar gigante, hasta unas pocas, entre 1 y 4, en las sinapsis de los ganglios simpáticos y la médula espinal en vertebrados.

El incremento de Ca^{2+} en la terminal presináptica en respuesta a un potencial de acción es rápidamente amortiguado por proteínas de unión a Ca^{2+} citoplasmáticas y mitocondrias, aunque el Ca^{2+} es también transportado activamente fuera de la neurona por bombas y transportadores.

Para más información acerca del proceso de liberación de transmisores puede consultarse (Kandel y cols., 2013, Capítulo 12).

2.5. Hipocampo

El hipocampo es una región del cerebro de la mayoría de los vertebrados localizado en la profundidad de la parte anterior del lóbulo temporal. Es una estructura pareada, lo que significa que existe un hipocampo en cada hemisferio cerebral, aunque esto sólo es cierto para humanos y otros mamíferos.

El hipocampo forma parte del sistema límbico y tiene importantes roles en la consolidación de la memoria tanto a corto como a largo plazo, también en la memoria espacial y la navegación (el cómo nos ubicamos cuando nos desplazamos). Además, el hipocampo está implicado en la regulación de las emociones, aunque esta no es su función principal.

El hipocampo está formado por el giro dentado, las regiones CA y el subículo, como se puede observar en la figura 2.3. A diferencia de la mayoría de la corteza, el hipocampo posee 3 capas neuronales bien definidas en el giro dentado y dependiendo del autor, se podrían indicar entre 3 y 5 capas neuronales en las regiones CA.

La principal fuente de aferencias y eferencias del hipocampo se dan con la corteza entorrinal formando lo que se conoce como el circuito del hipocampo o circuito trisináptico, de la siguiente manera: La vía perforante es el principal camino de salida de la corteza entorrinal, los axones de las células piramidales de la corteza entorrinal perforan el subículo y se proyectan principalmente a la capa granular del giro dentado (primera conexión sináptica), algunos otros axones se proyectan a la región CA3 y en menor medida a la región CA1. Los axones de las células granulosas del giro dentado, llamados fibras musgosas, pasan información a las dendritas de las células piramidales de la región CA3 (segunda conexión sináptica). Desde allí, los axones de las células de la región CA3 proyectan ramas colaterales, llamadas colaterales de Schaffer, que se enrollan hasta las dendritas apicales y posteriormente se extienden hasta la región CA1. Los axones de la región CA1 se proyectan de regreso a la corteza entorrinal (tercera conexión sináptica) completando de esta manera el circuito del hipocampo. Es de aclarar que existen otras conexiones que también tienen un rol importante en la función del hipocampo pero sus descripciones están fuera del alcance del presente trabajo y pueden ser consultadas en (Wright y cols., 2017). En la figura 2.4 se resume el circuito anteriormente descrito, aunque para tener clara su ubicación es mejor referirse a la figura 2.3.

En enfermedades degenerativas como el Alzheimer, el hipocampo es una de las primeras regiones en sufrir daño, de igual manera, un daño en la región del hipocampo produce amnesia anterógrada, esto es, la incapacidad de formar nuevos recuerdos sobre

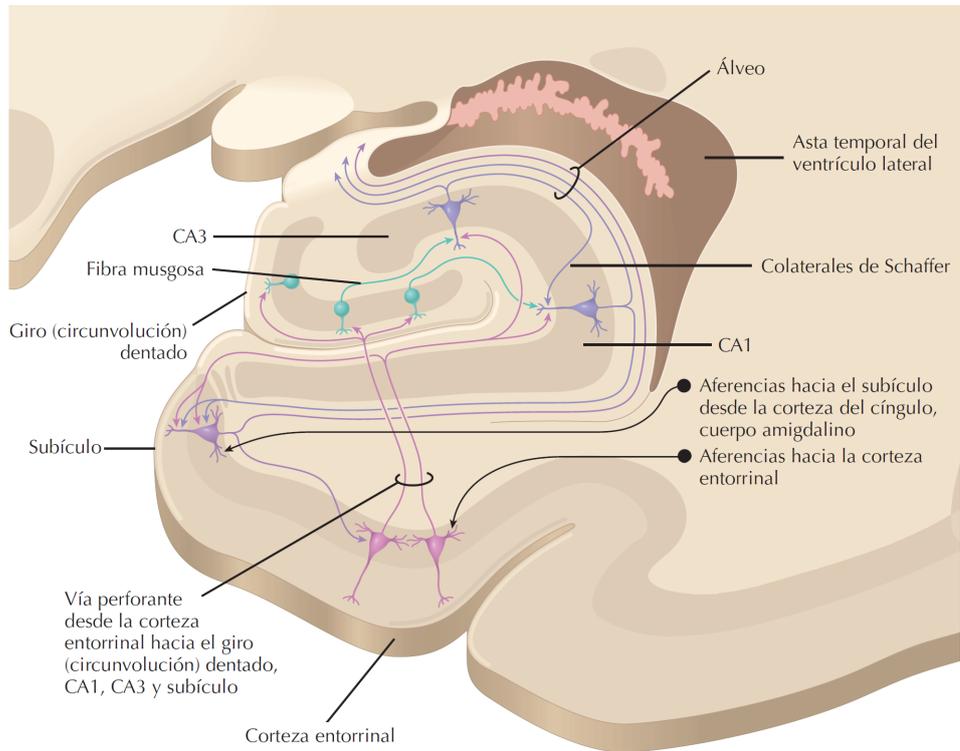


Figura 2.3: Hipocampo en corte coronal. Adaptado de (Rubin y Safdieh, 2009, pg. 205).

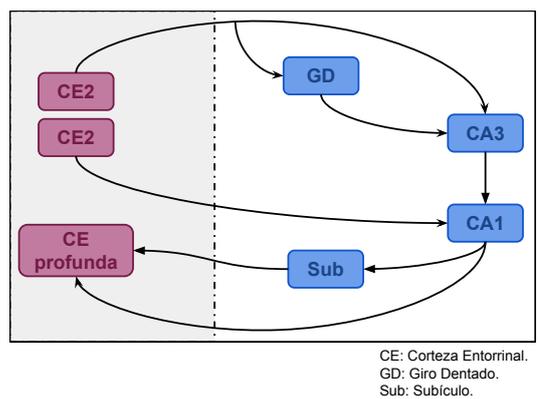


Figura 2.4: Circuito del Hipocampo o circuito trisináptico. Elaboración propia. Adaptado de “neural circuitry of the rodent hippocampus” (y Cajal, 1909-1911).

hechos específicos.

La mayoría de las sinapsis del hipocampo son glutamatérgicas y su liberación está mediada por diferentes tipos de canales de Ca^{2+} . El tipo de canal y su proporción en cada especie celular varía según la fisiología de la sinapsis. Por ejemplo, las fibras musgosas, según (Li, Bischofberger, y Jonas, 2007), presentan canales P/Q, N y R en una proporción de 61 %, 24 % y 15 % respectivamente, lo que en una zona activa se correspondería con 44 canales P/Q, 17 canales N y 5 canales R. En contraste con las colaterales de Schaffer, según (Scimemi y Diamond, 2012), la mayor contribución se debe a canales P/Q y N, y aunque no indica su proporción exacta, menciona que la liberación de neurotransmisor es iniciada por muy pocos canales de Ca^{2+} en una zona activa, posiblemente 3 canales y no más de 10 canales.

Capítulo 3

Modelos matemáticos de los procesos implicados

3.1. Modelo de canales iónicos

Tal como se describe en (Arning, 2009), posiblemente el primer modelo de canal iónico fue el propuesto por Hodgking y Huxley, donde se trata de describir el flujo de corriente a través de la membrana del axón del calamar gigante como la suma de las corrientes de sodio I_{Na} , potasio I_K y una corriente de fuga I_L . Las corrientes individuales estarían dadas por:

$$I_K = g_K n^4 (V - E_K) \quad , \quad (3.1)$$

$$I_{Na} = g_{Na} m^3 h (V - E_{Na}) \quad , \quad (3.2)$$

$$I_L = g_L (V - E_L) \quad , \quad (3.3)$$

donde g_α denota la conductancia máxima con respecto de la especie iónica α , V es el potencial electrostático aplicado y E_α es el potencial de reposo para el ion α . Las variables n , m , y h son las variables de apertura o *gating* e indican el comportamiento dinámico del canal y pueden ser interpretadas como partículas de apertura ficticias, donde por ejemplo, n^4 significa que es necesario la activación de cuatro partículas para permitir que el canal de potasio conduzca iones (esto se puede pensar como las cuatro subunidades protéicas del canal). Del mismo modo, m es una partícula de activación, pero h es una partícula de inactivación. De esta manera, las funciones $n(t)$, $m(t)$ y $h(t)$ se deben de determinar como soluciones de ecuaciones diferenciales ordinarias de primer orden.

Desde entonces, muchos otros modelos de canal se han propuesto, como los modelos de estado discreto de Markov, los modelos Fokker-Planck o los modelos estadísticos de un sólo canal. Puede consultarse (Arning, 2009) para una descripción detallada de cada uno. En este trabajo se considerarán los modelos de estados discretos de Markov para describir los canales y poder obtener resultados cuantitativos de conductancias.

3.1.1. Modelos de Estados Discretos de Markov(MEDMs)

Son los modelos más ampliamente usados en el contexto de los canales iónicos. Este modelo se basa en la hipótesis de que el sistema que representa el canal puede estar en algún estado bien definido (estos son los estados discretos correspondientes a la energía mínima local del sistema) y que para pasar a un estado diferente se debe de superar cierta barrera energética donde cada paso entre diferentes estados está caracterizado por alguna tasa de transición.

De esta manera, un MEDMs está básicamente compuesto por dos elementos, la cantidad de estados discretos y la correspondiente tasa de transición entre ellos. El modelo más simple puede ser un modelo con sólo dos estados y sus correspondientes tasas de avance y retroceso entre ellos, como se indica en la figura 3.1a. De manera general, los MEDMs pueden tener cualquier cantidad de estados con conexiones arbitrarias entre dichos estados y la notación k_{ij} se refiere a la tasa de transición entre el estado i y el estado j .

Un punto crucial en el uso de los MEDMs es la apropiada definición de las tasas de transición. Una aproximación comúnmente usada está basada o en la teoría de transición de Eyring o en la teoría de transición de la reacción de Kramers (ver (Arning, 2009) para más detalles). Éstas, aplicadas en el contexto de los MEDMs para la apertura de canales inducidos por voltaje, generan tasas de transición que tiene la forma:

$$k_{ij} = k_{ij}^0 \exp\left(\pm \frac{z_{ij}e_0V}{K_B T}\right) \quad , \quad (3.4)$$

donde K_B es la constante de Boltzmann, T es la temperatura absoluta, e_0 es la carga unitaria, z_{ij} es la cantidad de carga que se mueve en el correspondiente tiempo de transición como fracción del campo que atraviesa y V es el potencial electrostático a través de la membrana. Los prefactores k_{ij}^0 son constantes por determinar.

Una vez se han definido el número de estados y la estructura del modelo de Markov en consideración, se puede obtener un sistema de ecuaciones diferenciales ordinarias (EDOs) que representan el modelo. Éstas describen la evolución en el tiempo de las probabilidades de estado, es decir, la probabilidad de estar en un estado particular. Un ejemplo de lo anterior se encuentra en la figura 3.1b.

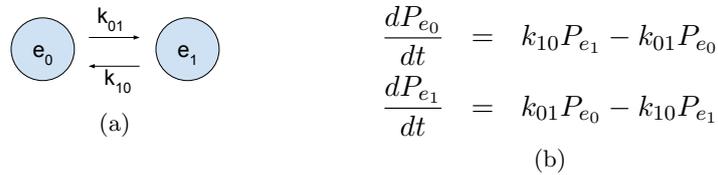


Figura 3.1: Modelo de estados discretos de Markov con dos estados (3.1a) y sus ecuaciones de estado asociadas (3.1b).

Cuando se usan los modelos de Markov para describir el comportamiento de los canales iónicos, usualmente las tasas de transición no pueden ser medidas directamente

de los experimentos, por lo que deben ser determinadas al ajustar la salida del modelo a los datos medidos. Los datos que se derivan de los experimentos son voluminosos y se corresponden con corrientes iónicas macroscópicas, y en algunos casos, de corrientes iónicas en un sólo canal. De esta manera, las tasas de transición representan los parámetros libres del sistema.

En algunas ocasiones, ciertas propiedades de los datos obtenidos necesitan modelos con un gran número de estados para lograr un resultado satisfactorio, pero la introducción de más estados naturalmente incrementará la cantidad de tasas de transición necesarias para poder describir el sistema. Con más parámetros libres para afinar el sistema, se hace mucho más fácil ajustar un gran rango de datos, aunque el inconveniente que esto presenta es que en un espacio de parámetros grande pueden existir varios mínimos locales o globales por lo que diferentes combinaciones pueden llevar al resultado deseado. De esta manera, la identificación única de parámetros con significancia física se hace menos probable cuando se incrementa el número de estados.

3.2. Modelo de secreción

La secreción celular o exocitosis, proceso mediante el cual la membrana de las vesículas que contienen el componente a secretar se fusiona con la membrana celular y el contenido de la vesícula es secretado al espacio extracelular, puede ser modelado como una reacción química entre los iones de calcio y las proteínas presentes en las membranas vesiculares, que son las que realmente inducen la fusión de las membranas celular y plasmática (ver secciones 2.3.1 y 2.4 para más detalles biológicos).

Podemos encontrar dos aproximaciones de modelos de estados discretos de Markov para las interacciones entre los iones de Ca^{2+} y las vesículas sinápticas. En un primer modelo, cada estado puede representar la cantidad de sensores de calcio que han ligado a iones Ca^{2+} y las tasas de transición indican las velocidades cinéticas de la reacción entre los sensores de calcio y los iones de Ca^{2+} . Un ejemplo de este tipo de modelo es el propuesto en (Bollmann, Sakmann, y Borst, 2000). En un segundo modelo, cada estado del modelo representa el grupo de vesículas que se tienen disponibles según la cantidad de iones Ca^{2+} que se les ha ligado pero en su forma de capacitancia, esto es, la capacitancia que pueden aportar las membranas vesiculares a la membrana celular una vez ocurre la exocitosis, y las tasas de transición también indican velocidades cinéticas. Un ejemplo de este tipo de modelos se encuentra en (Klingauf y Neher, 1997a) y (Heinemann, Crow, Neher, y Zucker, 1994).

Estos últimos modelos son los más usados puesto que medir el aumento de la capacitancia de la membrana celular es una práctica común en el estudio de la exocitosis en células *in vitro*. Una vez tenida la capacitancia total es fácil saber cuántas vesículas secretaron su contenido al exterior celular.

3.3. Modelo celular

El modelo neuronal propuesto es un modelo de neurona esférica similar a los propuestos por (Sala y Hernández-Cruz, 1990) o (Nowycky y Pinter, 1993). En este modelo, se considera la neurona como una esfera de radio r que puede ser discretizada en capas concéntricas de grosor Δr como si fuese una cebolla. Es claro que una neurona no tiene una forma esférica, pero debemos de recordar que nuestro interés son las concentraciones de Ca^{2+} muy cerca de la membrana celular, ya que como bien se mencionó anteriormente, la cantidad de Ca^{2+} que ingresa a la célula en la zona activa es poca y por la presencia de *buffers* este calcio no logra difundirse muy lejos de la membrana celular. De esta manera, seleccionando un valor adecuado para r y enfocándonos sólo en las capas más externas, podemos obtener los perfiles de Ca^{2+} deseados. Además y principalmente, la elección de un modelo de neurona esférica facilita enormemente la resolución numérica de las ecuaciones diferenciales en derivadas parciales que se obtienen de los procesos celulares modelados.

El modelo toma en cuenta el Ca^{2+} y un *buffer* inmóvil en sus dos formas, tanto libre como ligado a calcio. Dado que el *buffer* es inmóvil, el proceso de difusión en nuestra neurona esférica sólo ocurre para el calcio libre.

Si suponemos que la difusión solo ocurre en dirección radial, podemos ignorar los componentes tangenciales de la difusión (sobre las coordenadas angulares), con lo que la ecuación de difusión en tres dimensiones en una esfera se podría reducir a una ecuación en derivadas parciales unidimensional llegando entonces a la ecuación (Crank, 1975):

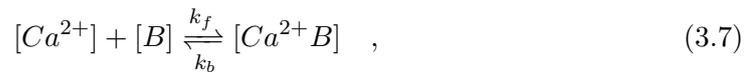
$$\frac{\partial[\text{Ca}^{2+}]}{\partial t} = D_{\text{Ca}} \left(\frac{\partial^2[\text{Ca}^{2+}]}{\partial r^2} + \frac{2}{r} \frac{\partial[\text{Ca}^{2+}]}{\partial r} \right) , \quad (3.5)$$

donde $[\text{Ca}^{2+}]$ indica la concentración de Ca^{2+} , r es el la dirección radial, t es el tiempo y D_{Ca} es el coeficiente de difusión del calcio. Y definiendo $u = [\text{Ca}^{2+}]r$, la ecuación (3.5) puede reescribirse como (el procedimiento completo se puede ver en el Apéndice A):

$$\frac{\partial u}{\partial t} = D_{\text{Ca}} \frac{\partial^2 u}{\partial r^2} . \quad (3.6)$$

La discretización, tanto en espacio como en tiempo se hacen necesarios para encontrar una solución a la ecuación (3.6). La simplificación realizada al considerar la neurona como una esfera de capas concéntricas de grosor uniforme, asume que cada una de estas capas es un sistema bien distribuido (las moléculas que se encuentran en dicha capa están uniformemente distribuidas en todo el volumen de la capa) donde la difusión puede ser ignorada.

El *buffer* es una molécula que reacciona con el Ca^{2+} libre y se une o se separa de este, de tal manera que el amortiguamiento o *buffering* se puede modelar como una reacción cinética de primer orden, como indica el siguiente esquema cinético:



donde $[B]$ es la concentración de *buffer*, $[Ca^{2+}B]$ es la concentración de calcio ligado a *buffer*, y k_f y k_b son la tasa de avance y retroceso de la reacción respectivamente. La constante de disociación K_D queda entonces definida como k_b/k_f . El efecto del *buffering* sobre la concentración del calcio y la concentración de *buffer* está dado por las siguientes ecuaciones diferenciales que se derivan del esquema (3.7):

$$\frac{d[Ca^{2+}]}{dt} = \frac{d[B]}{dt} = k_b \cdot [Ca^{2+}B] - k_f \cdot [Ca^{2+}][B] \quad , \quad (3.8)$$

$$[B]_T = [Ca^{2+}B] + [B] \quad , \quad (3.9)$$

donde $[B]_T$ es la concentración total de *buffer*. La ecuación (3.8) es obtenida aplicando la ley de acción de masas, que viene a decir que el ritmo de la reacción es proporcional al producto de los reactantes (se puede consultar (*Law of mass action*, 2017) para más información).

La concentración de Ca^{2+} en el interior celular cambia debido a la entrada de Ca^{2+} por los canales iónicos de Ca^{2+} dependientes de voltaje en la membrana celular, o en el caso del modelo propuesto, en la capa más externa. La concentración de Ca^{2+} debido a este flujo en la capa más externa queda entonces determinado por la siguiente ecuación:

$$[Ca^{2+}] = -\frac{I_{Ca}}{2FV} \quad , \quad (3.10)$$

donde I_{Ca} es la corriente de calcio transmembrana, F es la constante de Faraday y V es el volumen del compartimiento. Esto nos proporciona la condición de frontera que usaremos en el modelo.

El modelo propuesto no considera ningún mecanismo de extrusión de Ca^{2+} como podrían ser las bombas de Ca^{2+} , ni tampoco ningún mecanismo de fuga como lo son los canales de fuga de Ca^{2+} . Esto debido a que los tiempos que nos interesan son muy cortos y los mecanismos mencionados no entrarían a funcionar en dicho tiempo por lo que es irrelevante si están o no presentes.

Finalmente, al tener las concentraciones de calcio en la capa subyacente a la membrana celular, donde se encuentran las vesículas sinápticas, se calcula la cantidad de vesículas sinápticas que son excitadas.

3.3.1. Resolución numérica

Difusión

Como se mencionó anteriormente, la difusión depende del tiempo y el espacio, generando de esta manera una ecuación en derivadas parciales. Para su resolución numérica se hace necesario entonces una discretización tanto en el espacio como en el tiempo. Para lograrlo se utiliza el esquema de resolución numérica propuesto por Crank y Nicolson (Crank y Nicolson, 1947) (Ver Apéndice B.1). Este esquema es implícito en el espacio pero explícito en el tiempo, y dado que es incondicionalmente estable, da la

liberad de elegir los valores para Δt y Δr arbitrariamente si comprometer en ningún momento la estabilidad.

Discretizando la ecuación (3.6) con el esquema indicado, se llega a la siguiente ecuación:

$$\begin{aligned} \frac{r_i}{\Delta t} ([Ca^{2+}]_{i,t+1} - [Ca^{2+}]_{i,t}) &= \frac{DCa}{2\Delta^2 r} [(r_i + \Delta r)([Ca^{2+}]_{i+1,t+1} + [Ca^{2+}]_{i+1,t}) \\ &\quad - 2r_i([Ca^{2+}]_{i,t+1} + [Ca^{2+}]_{i,t}) \\ &\quad + (r_i - \Delta r)([Ca^{2+}]_{i-1,t+1} + [Ca^{2+}]_{i-1,t})] \quad , \quad (3.11) \end{aligned}$$

donde r_i es la distancia radial desde el centro de la esfera hasta el punto medio de la capa i . Si la capa central tiene un radio r_0 , entonces $r_i = r_0 + i\Delta r$ para $i = 0, 1, \dots, n-1$. Además, si el tiempo inicial es t_0 , podemos definir $t_i = t_0 + j\Delta t$ y $t + 1 = t_i + \Delta t$ para $j = 0, 1, \dots, m-1$.

Multiplicando la ecuación (3.11) por $\Delta t/r_i$, dejando al lado izquierdo de la ecuación los términos correspondiente al tiempo $t + 1$ y en el derecho los correspondientes a t , y considerando $r_{i+1} = r_i + \Delta r$, se llega a la siguiente ecuación (el despeje completo se puede ver en el Apéndice A):

$$\begin{aligned} -\frac{DCa\Delta t}{2r_i\Delta^2 r}(r_{i-1})[Ca^{2+}]_{i-1,t+1} + \left(1 + \frac{DCa\Delta t}{\Delta^2 r}\right) [Ca^{2+}]_{i,t+1} - \frac{DCa\Delta t}{2r_i\Delta^2 r}(r_{i+1})[Ca^{2+}]_{i+1,t+1} \\ = \frac{DCa\Delta t}{2r_i\Delta^2 r}(r_{i-1})[Ca^{2+}]_{i-1,t} + \left(1 - \frac{DCa\Delta t}{\Delta^2 r}\right) [Ca^{2+}]_{i,t} + \frac{DCa\Delta t}{2r_i\Delta^2 r}(r_{i+1})[Ca^{2+}]_{i+1,t} \quad . \quad (3.12) \end{aligned}$$

Se puede observar que para cada t se debe de resolver un sistema de $n+1$ ecuaciones con $n+1$ incógnitas. Dicho sistema se puede organizar en una matriz tridiagonal.

Para poder resolver el sistema se deben de definir las condiciones iniciales y de frontera. Se asume que la concentración inicial de Ca^{2+} es conocida y está uniformemente distribuida a través de toda la célula y que también se conoce la concentración total de *buffer*.

Las condiciones de frontera de la capa más externa pueden ser obtenidas al considerar una capa imaginaria externa $n+1$ por encima de la capa n que está siempre en equilibrio con la capa subyacente, esto es, que $[Ca^{2+}]_{n+1} = [Ca^{2+}]_n$ en cualquier tiempo t . Desarrollando la ecuación (3.11) para la capa n al introducir esta condición y sustituyendo $[Ca^{2+}]_{n+1}$ por $[Ca^{2+}]_n$ obtenemos:

$$\begin{aligned} \frac{r_n}{\Delta t} ([Ca^{2+}]_{n,t+1} - [Ca^{2+}]_{n,t}) &= \frac{DCa}{2\Delta^2 r} [(r_n + \Delta r)([Ca^{2+}]_{n,t+1} + [Ca^{2+}]_{n,t}) \\ &\quad - 2r_n([Ca^{2+}]_{n,t+1} + [Ca^{2+}]_{n,t}) \\ &\quad + (r_n - \Delta r)([Ca^{2+}]_{n-1,t+1} + [Ca^{2+}]_{n-1,t})] \quad . \quad (3.13) \end{aligned}$$

Reduciendo términos semejantes, la ecuación (3.13) se puede escribir como:

$$\begin{aligned} \frac{r_n}{\Delta t} ([Ca^{2+}]_{n,t+1} - [Ca^{2+}]_{n,t}) &= \frac{DCa}{2\Delta^2 r} [(r_n - \Delta r)([Ca^{2+}]_{n-1,t+1} + [Ca^{2+}]_{n-1,t}) \\ &\quad - (r_n - \Delta r)([Ca^{2+}]_{n,t+1} + [Ca^{2+}]_{n,t})] \quad . \quad (3.14) \end{aligned}$$

Multiplicando la ecuación (3.14) por $\Delta t/r_n$, dejando del lado izquierdo de la ecuación los términos correspondientes al tiempo $t + 1$ y en el lado derecho los correspondientes a t y considerando $r_{n-1} = r_n - \Delta r$, se llega finalmente a la siguiente ecuación (el despeje completo se puede ver en el Apéndice A):

$$\begin{aligned} -\frac{DCa\Delta t}{2r_n\Delta^2 r}(r_{n-1})[Ca^{2+}]_{n-1,t+1} + \left(1 + \frac{DCa\Delta t}{2r_n\Delta^2 r}(r_{n-1})\right) [Ca^{2+}]_{n,t+1} \\ = \frac{DCa\Delta t}{2r_n\Delta^2 r}(r_{n-1})[Ca^{2+}]_{n-1,t} + \left(1 - \frac{DCa\Delta t}{2r_n\Delta^2 r}(r_{n-1})\right) [Ca^{2+}]_{n,t} \quad . \quad (3.15) \end{aligned}$$

Las condiciones de frontera de la capa más interna son obtenidas al considerar una capa imaginaria en el centro de la esfera por debajo de la capa 0 que siempre está en equilibrio con la capa más interna, esto es, que $[Ca^{2+}]_0 = [Ca^{2+}]_{-1}$ en cualquier tiempo t y que el flujo a través del centro de la esfera es tres veces mayor que el que atraviesa a cualquier otra capa (Sala y Hernández-Cruz, 1990). Desarrollando la ecuación (3.11) para la capa 0 al introducir esta condición y sustituyendo $[Ca^{2+}]_{-1}$ por $[Ca^{2+}]_0$ llegamos a:

$$\begin{aligned} \frac{r_0}{\Delta t} ([Ca^{2+}]_{0,t+1} - [Ca^{2+}]_{0,t}) &= \frac{3DCa}{2\Delta^2 r} [(r_0 + \Delta r)([Ca^{2+}]_{1,t+1} + [Ca^{2+}]_{1,t}) \\ &\quad - 2r_0([Ca^{2+}]_{0,t+1} + [Ca^{2+}]_{0,t}) \\ &\quad + (r_0 - \Delta r)([Ca^{2+}]_{0,t+1} + [Ca^{2+}]_{0,t})] \quad . \quad (3.16) \end{aligned}$$

Reduciendo términos semejantes y teniendo en cuenta que $-2r_0 + (r_0 - \Delta r) = -r_0 - \Delta r = -(r_0 + \Delta r)$ la ecuación (3.16) se puede escribir como:

$$\begin{aligned} \frac{r_0}{\Delta t} ([Ca^{2+}]_{0,t+1} - [Ca^{2+}]_{0,t}) &= \frac{3DCa}{2\Delta^2 r} [(r_0 + \Delta r)([Ca^{2+}]_{1,t+1} + [Ca^{2+}]_{1,t}) \\ &\quad - (r_0 + \Delta r)([Ca^{2+}]_{0,t+1} + [Ca^{2+}]_{0,t})] \quad . \quad (3.17) \end{aligned}$$

Multiplicando la ecuación (3.17) por $\Delta t/r_0$, dejando del lado izquierdo de la ecuación los términos correspondientes al tiempo $t + 1$ y en el lado derecho los correspondientes a t y considerando $r_1 = r_0 + \Delta r$, finalmente se llega a la siguiente ecuación (el despeje completo se puede ver en el Apéndice A):

$$\begin{aligned} -\frac{3DCa\Delta t}{2r_0\Delta^2 r}r_1[Ca^{2+}]_{1,t+1} + \left(1 + \frac{3DCa\Delta t}{2r_0\Delta^2 r}r_1\right) [Ca^{2+}]_{0,t+1} \\ = \frac{3DCa\Delta t}{2r_0\Delta^2 r}r_1[Ca^{2+}]_{1,t} + \left(1 - \frac{3DCa\Delta t}{2r_0\Delta^2 r}r_1\right) [Ca^{2+}]_{0,t} \quad . \quad (3.18) \end{aligned}$$

Quedando de esta manera definas las condiciones de frontera tanto en la capa más interna como en la capa más externa del modelo.

El sistema de $n + 1$ ecuaciones lineales y $n + 1$ incógnitas puede ser representado en forma matricial como $\mathcal{A}\mathcal{W}(t + \Delta t) = \mathcal{B}\mathcal{W}(t)$ para $t = 0, \Delta t, 2\Delta t, \dots, n$ donde $\mathcal{W}(t) = (w_{0,t}, w_{1,t}, \dots, w_{n,t})^t$ y $w_{i,t} = [Ca^{2+}]_{i,t}$. Las matrices \mathcal{A} y \mathcal{B} se definen a continuación a partir de las ecuaciones (3.12), (3.15) y (3.18) considerando $R = \frac{DCa\Delta t}{2\Delta^2 r}$ e $Y = \frac{3D_{Ca}\Delta t r_1}{2r_0\Delta^2 r}$.

$$\mathcal{A} = \begin{bmatrix} d_0 & u_0 & 0 & \cdots & \cdots & 0 \\ l_1 & d_1 & u_1 & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & u_{n-1} \\ 0 & \cdots & \cdots & 0 & l_n & d_n \end{bmatrix} = \begin{bmatrix} 1 + Y & -Y & 0 & \cdots & \cdots & 0 \\ -\frac{r_0}{r_1}R & 1 + 2R & -\frac{r_2}{r_1}R & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & -\frac{r_n}{r_{n-1}}R \\ 0 & \cdots & \cdots & 0 & -\frac{r_{n-1}}{r_n}R & 1 + \frac{r_{n-1}}{r_n}R \end{bmatrix}$$

$$\mathcal{B} = \begin{bmatrix} D_0 & b_0 & 0 & \cdots & \cdots & 0 \\ h_1 & D_1 & b_1 & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & b_{n-1} \\ 0 & \cdots & \cdots & 0 & h_n & D_n \end{bmatrix} = \begin{bmatrix} 1 - Y & Y & 0 & \cdots & \cdots & 0 \\ \frac{r_0}{r_1}R & 1 - 2R & \frac{r_2}{r_1}R & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \frac{r_n}{r_{n-1}}R \\ 0 & \cdots & \cdots & 0 & \frac{r_{n-1}}{r_n}R & 1 - \frac{r_{n-1}}{r_n}R \end{bmatrix}$$

Este sistema puede ser resuelto utilizando el bien conocido método de reducción de Crout o factorización LU de la siguiente manera:

1. Se factoriza la matriz \mathcal{A} como el producto de dos matrices \mathcal{L} y \mathcal{U} , una diagonal inferior y la otra diagonal superior.
2. Se resuelve el sistema $\mathcal{L}z = \mathcal{B}\mathcal{W}(t)$ para z aplicando sustitución hacia adelante.
3. Se resuelve el sistema $\mathcal{U}\mathcal{W}(t + \Delta t) = z$ para $\mathcal{W}(t + \Delta t)$ aplicando sustitución hacia atrás.

Las matrices triangulares \mathcal{L} y \mathcal{U} tienen la siguiente forma:

$$\mathcal{L} = \begin{bmatrix} d_0 & 0 & \cdots & \cdots & \cdots & 0 \\ l_1 & d_1 = d_1 - l_1 v_0 & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & l_n & d_n = d_n - l_n v_{n-1} \end{bmatrix}$$

$$\mathcal{U} = \begin{bmatrix} 1 & v_0 = \frac{u_0}{d_0} & 0 & \cdots & \cdots & 0 \\ 0 & 1 & v_1 = \frac{u_1}{d_1} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & v_{n-1} = \frac{u_{n-1}}{d_{n-1}} \\ 0 & \cdots & \cdots & \cdots & 0 & 1 \end{bmatrix}$$

El esquema general del algoritmo usado para la resolución del sistema se puede observar en la figura 3.2

Las matrices \mathcal{A} y \mathcal{B} originales comparten los elementos l_i y u_i con h_i y b_i respectivamente, donde lo único que cambia es el signo.

Amortiguamiento

Las ecuaciones de amortiguamiento o *buffering*, a pesar de ser EDOs también se discretizan utilizando el esquema de Crank-Nicolson, de esta manera se asegura que no se pierde la ganancia de orden de error previamente ganada al aplicar este esquema a la ecuación de difusión.

Conociendo la concentración inicial de Ca^{2+} , las constantes k_b y k_f , y la concentración de *buffer* total se puede calcular la concentración inicial de *buffer*. Esto se logra al tener en cuenta que la reacción cinética que se indica en el esquema (3.7) entre el Ca^{2+} y el *buffer* en un inicio está en equilibrio, por lo que al aplicar la ley de acción de masas se puede obtener la concentración inicial de *buffer* libre (el procedimiento se puede ver en el Apéndice A)

$$[B]_{i,0} = \frac{[B]_T k_b}{[Ca^{2+}]_{i,0} k_f + k_b} \quad . \quad (3.19)$$

La ecuación (3.9) se despeja para $[Ca^{2+}B]$ y se sustituye en la ecuación (3.8) obteniendo de esta manera la ecuación:

$$\frac{d[Ca^{2+}]}{dt} = k_b[B]_T - k_b[B] - k_f[Ca^{2+}][B] \quad . \quad (3.20)$$

Como se mencionó anteriormente, la ecuación (3.20) se discretiza siguiendo el esquema Crank-Nicolson obteniéndose la ecuación:

$$\begin{aligned} \frac{1}{\Delta t} ([Ca^{2+}]_{i,t+1} - [Ca^{2+}]_{i,t}) &= k_b[B]_T - \frac{k_b}{2} ([B]_{i,t+1} + [B]_{i,t}) \\ &\quad - \frac{k_f}{2} ([B]_{i,t+1}[Ca^{2+}]_{i,t+1} + [B]_{i,t}[Ca^{2+}]_{i,t}) \quad . \quad (3.21) \end{aligned}$$

Definición de variables

$$Y = 3D_{Ca}\Delta tr_1/2r_0\Delta^2r$$

$$R = D_{Ca}\Delta t/2\Delta^2r$$

Definición de elementos de la matriz \mathcal{B}

$$D_0 = 1 - Y$$

$$D_i = 1 - 2R \text{ para } i = 1, \dots, n - 1$$

$$D_n = 1 - r_{n-1}R/r_n$$

Definición de elementos de la matriz \mathcal{A}

$$d_0 = 1 + Y$$

$$d_i = 1 + 2R \text{ para } i = 1, \dots, n - 1$$

$$d_n = 1 + r_{n-1}R/r_n$$

$$l_i = -r_{i-1}R/r_i \text{ para } i = 1, \dots, n$$

$$u_0 = -Y$$

$$u_i = -r_{i+1}R/r_i \text{ para } i = 1, \dots, n - 1$$

Definición de los elementos de las matrices \mathcal{L} y \mathcal{U}

$$v_0 = u_0/d_0$$

Para $i = 1, \dots, n - 1$

$$d_i = d_i - l_i v_{i-1}$$

$$v_i = u_i/d_i$$

$$d_n = d_n - l_n v_{n-1}$$

Para $t \leq T_{max}$

Sustitución hacia adelante para resolver $\mathcal{L}z = \mathcal{B}\mathcal{W}(t)$ para z

$$z_0 = (D_0 w_{0,t} - u_0 w_{1,t})/d_0$$

$$z_i = (-l_i w_{i-1,t} + D_i w_{i,t} - u_i w_{i+1,t} - l_i z_{i-1})/d_i \text{ para } i = 1, \dots, n - 1$$

$$z_n = (-l_n w_{n-1,t} + D_n w_{n,t} - l_n z_{n-1})/d_n$$

Sustitución hacia atrás para resolver $\mathcal{U}\mathcal{W}(t + \Delta t) = z$ para $\mathcal{W}(t + \Delta t)$

$$w_{n,t+1} = z_n$$

$$w_{i,t+1} = z_i - v_i w_{i+1,t+1} \text{ para } i = n - 1, \dots, 0$$

Salida: $w_{i,t+1}$ para $i = 0, \dots, n$

Figura 3.2: Algoritmo para la resolución del sistema $\mathcal{A}\mathcal{W}(t + 1) = \mathcal{B}\mathcal{W}(t)$

Hay un problema en la ecuación (3.21) con el término $[B][Ca^{2+}]$ evaluado en $t + 1$, porque para calcular uno es necesario el otro. Para resolver este tipo de ecuación se usa un esquema de integración explícito-implícito en el cual los productos $[B][Ca^{2+}]$ ya no representan ningún problema dado que $[B]$ y $[Ca^{2+}]$ se evalúan para tiempos distintos. De esta manera se llega a la siguiente ecuación:

$$\begin{aligned} \frac{1}{\Delta t}([Ca^{2+}]_{i,t+1} - [Ca^{2+}]_{i,t}) &= k_b[B]_T - \frac{k_b}{2}([B]_{i,t+1} + [B]_{i,t}) \\ &\quad - \frac{k_f}{2}([B]_{i,t+1}[Ca^{2+}]_{i,t} + [B]_{i,t}[Ca^{2+}]_{i,t+1}) \quad . \quad (3.22) \end{aligned}$$

Se aclara que aunque existen otras posibilidades para ecuaciones con términos no lineales que también son modificaciones del método de Crank-Nicolson (Ames, 1977) se adopta esta aproximación por su simplicidad. Otra posibilidad, como lo hacen en (Klingauf y Neher, 1997b), es dado $[Ca^{2+}]_{i,n}$, la concentración de calcio para la capa i y paso de tiempo n , se avanza $[B]_{i,n-1/2}$ de $n - 1/2$ a $n + 1/2$ con un primer paso de la regla trapezoidal y luego se utiliza $[B]_{i,n+1/2}$ para avanzar $[Ca]_{i,n}$ de n a $n + 1$ con otro paso de la regla trapezoidal.

La ecuación (3.8) para actualizar $[B]$ tiene una forma análoga a la ecuación (3.22), definiendo así la ecuación:

$$\begin{aligned} \frac{1}{\Delta t}([B]_{i,t+1} - [B]_{i,t}) &= k_b[B]_T - \frac{k_b}{2}([B]_{i,t+1} + [B]_{i,t}) \\ &\quad - \frac{k_f}{2}([B]_{i,t+1}[Ca^{2+}]_{i,t} + [B]_{i,t}[Ca^{2+}]_{i,t+1}) \quad . \quad (3.23) \end{aligned}$$

De esta manera, se ha generado un sistema de dos ecuaciones, (3.22) y (3.23), con dos incógnitas, $[Ca^{2+}]_i$ y $[B]_i$ para cada capa del modelo. Podemos utilizar el método de eliminación por sustitución para resolver dicho sistema, primero se despeja $[B]_{i,t}$ a partir de la ecuación (3.23) obteniendo la siguiente ecuación (el despeje completo se puede ver en el Apéndice A):

$$[B]_{i,t+1} = \frac{2k_b[B]_T + (\frac{2}{\Delta t} - k_b - k_f[Ca^{2+}]_{i,t+1})[B]_{i,t}}{\frac{2}{\Delta t} + k_b + k_f[Ca^{2+}]_{i,t}} \quad . \quad (3.24)$$

Posteriormente se sustituye la ecuación (3.24) en la ecuación (3.22) y despejando para $[Ca^{2+}]_{i,t+1}$ se llega a la siguiente ecuación (el despeje completo se puede ver en el Apéndice A):

$$[Ca^{2+}]_{i,t+1} = \frac{2k_b[B]_T - 2k_b[B]_{i,t} + (\frac{2}{\Delta t} + k_b + k_f[Ca^{2+}]_{i,t} - k_f[B]_{i,t})[Ca^{2+}]_{i,t}}{\frac{2}{\Delta t} + k_b + k_f[Ca^{2+}]_{i,t} + k_f[B]_{i,t}} \quad . \quad (3.25)$$

Por tanto, podemos actualizar $[Ca^{2+}]_{i,t+1}$ a partir de $[Ca^{2+}]_{i,t}$ y $[B]_{i,t}$ con la ecuación (3.25) para después actualizar $[B]_{i,t+1}$ a partir de $[Ca^{2+}]_{i,t+1}$, $[Ca^{2+}]_{i,t}$ y $[B]_{i,t}$ con la ecuación (3.24).

Corriente de Ca^{2+}

La ecuación (3.10) es una EDO pero como en el caso anterior es discretizada como se indica a continuación:

$$\frac{1}{\Delta t}([Ca^{2+}]_{n,t+1} - [Ca^{2+}]_{n,t}) = -\frac{ICa}{2FV} \quad . \quad (3.26)$$

Recordando que la ecuación (3.26) sólo aplica para la capa n y despejando llegamos a la ecuación:

$$[Ca^{2+}]_{n,t+1} = -\frac{ICa}{2FV}\Delta t + [Ca^{2+}]_{n,t} \quad . \quad (3.27)$$

Completando de esta manera una esquema de resolución numérica para todas las ecuaciones implicadas en el modelo celular.

Capítulo 4

Resultados numéricos

4.1. Canal iónico

Los canales iónicos que son de nuestro interés son los canales de Ca^{2+} ya que son estos los implicados en la liberación de neurotransmisor. El modelo propuesto está basado en el trabajo realizado en (Li y cols., 2007) sobre canales de Ca^{2+} en el hipocampo, más precisamente en los botones de las fibras musgosas, aunque este modelo también es usado en (Scimemi y Diamond, 2012) para las colaterales de Schaffer, obteniendo buenos resultados.

El modelo propuesto es un modelo de estados discretos de Markov de seis estados. Cinco estados para el estado cerrado del canal y un único estado para el estado abierto del canal. El esquema del modelo se puede observar en la figura 4.1a y sus ecuaciones de estado asociadas en la figura 4.1b. El modelo no considera estados inactivados, debido a que los canales de Ca^{2+} en este tipo de sinapsis sólo muestran una inactivación mínima durante el tiempo que duran los pulsos de test. Todas las transiciones, excepto al última, la que precede al estado abierto, se consideran dependientes de voltaje. Las tasas de avance α y retroceso β dependientes del voltaje se calculan según las siguientes ecuaciones:

$$\alpha_i(V) = \alpha_{i,0} \exp(V/k_i) \quad , \quad (4.1)$$

$$\beta_i(V) = \beta_{i,0} \exp(-V/k_i) \quad . \quad (4.2)$$

donde $\alpha_{i,0}$ y $\beta_{i,0}$ son las tasas de avance y retroceso a 0 mV, k_i es un factor de pendiente e $i = 1..4$.

La corriente a través de un canal iónico a un potencial de membrana específico V_m es proporcional tanto a la fuerza motriz como a la probabilidad de apertura del canal P_O . Lo anterior queda definido en la siguiente ecuación:

$$I_{Ca} = P_O \cdot (V_m - E_{Ca}) \quad , \quad (4.3)$$

donde E_{Ca} es el potencial de reposo de la membrana y la fuerza motriz se define como la diferencia entre V_m y E_{Ca} .

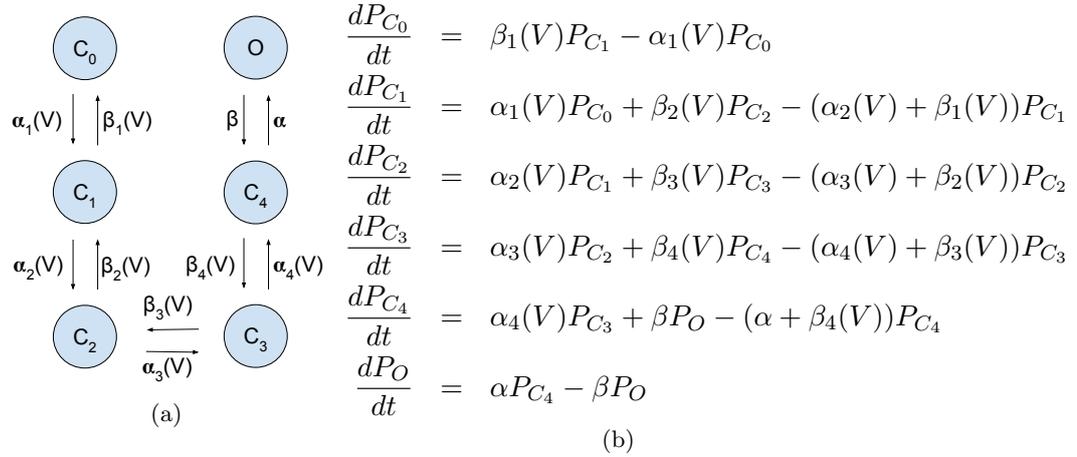


Figura 4.1: Modelo de canales de calcio (4.1a) y sus ecuaciones de estado asociadas (4.1b).

$\alpha_{i,0}(\text{ms}^{-1})$	$\beta_{i,0}(\text{ms}^{-1})$	$k_i(\text{mV})$
Canal P/Q-type		
$\alpha_{1,0} = 5.89$	$\beta_{1,0} = 14.99$	$k_1 = 62.61$
$\alpha_{2,0} = 9.21$	$\beta_{2,0} = 6.63$	$k_2 = 33.92$
$\alpha_{3,0} = 5.20$	$\beta_{3,0} = 132.80$	$k_3 = 135.08$
$\alpha_{4,0} = 1823.18$	$\beta_{4,0} = 248.58$	$k_4 = 20.86$
$\alpha = 247.71$	$\beta = 8.28$	
Canal N-type		
$\alpha_{1,0} = 4.29$	$\beta_{1,0} = 5.23$	$k_1 = 68.75$
$\alpha_{2,0} = 5.24$	$\beta_{2,0} = 6.63$	$k_2 = 39.53$
$\alpha_{3,0} = 4.98$	$\beta_{3,0} = 73.89$	$k_3 = 281.62$
$\alpha_{4,0} = 772.63$	$\beta_{4,0} = 692.18$	$k_4 = 18.46$
$\alpha = 615.01$	$\beta = 7.68$	
Canal R-type		
$\alpha_{1,0} = 9911.36$	$\beta_{1,0} = 0.62$	$k_1 = 67.75$
$\alpha_{2,0} = 4.88$	$\beta_{2,0} = 21.91$	$k_2 = 50.94$
$\alpha_{3,0} = 4.00$	$\beta_{3,0} = 51.30$	$k_3 = 173.29$
$\alpha_{4,0} = 256.41$	$\beta_{4,0} = 116.97$	$k_4 = 16.92$
$\alpha = 228.83$	$\beta = 1.78$	

Tabla 4.1: Parámetros del modelo de Markov para diferentes tipos de canales de Ca^{2+} . Tomado de (Li y cols., 2007).

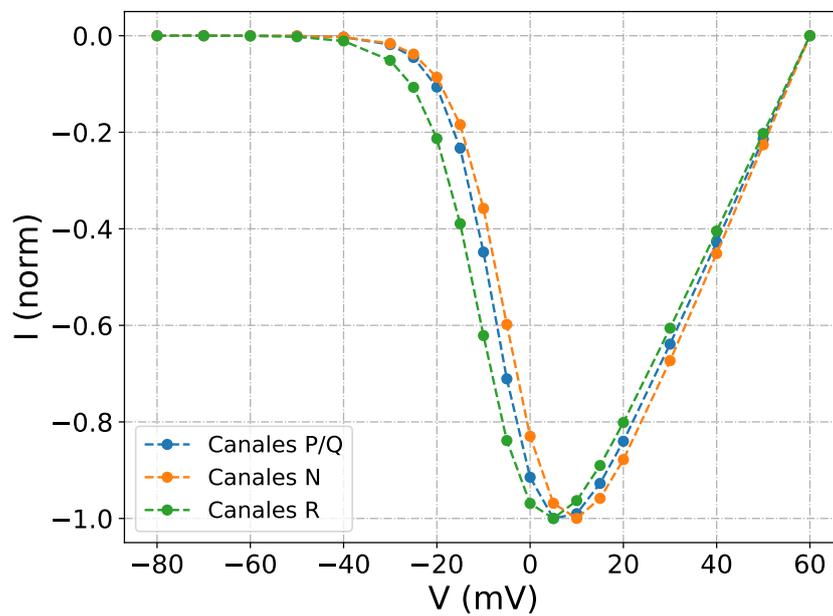
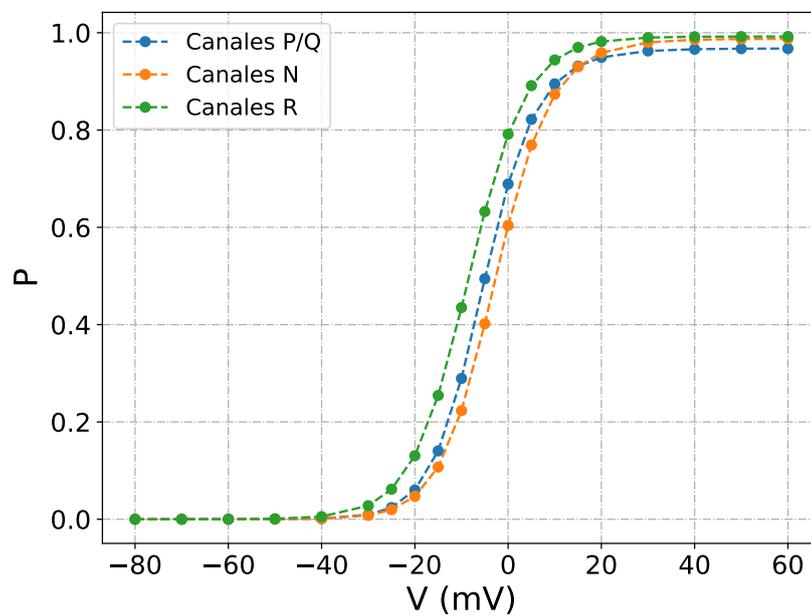
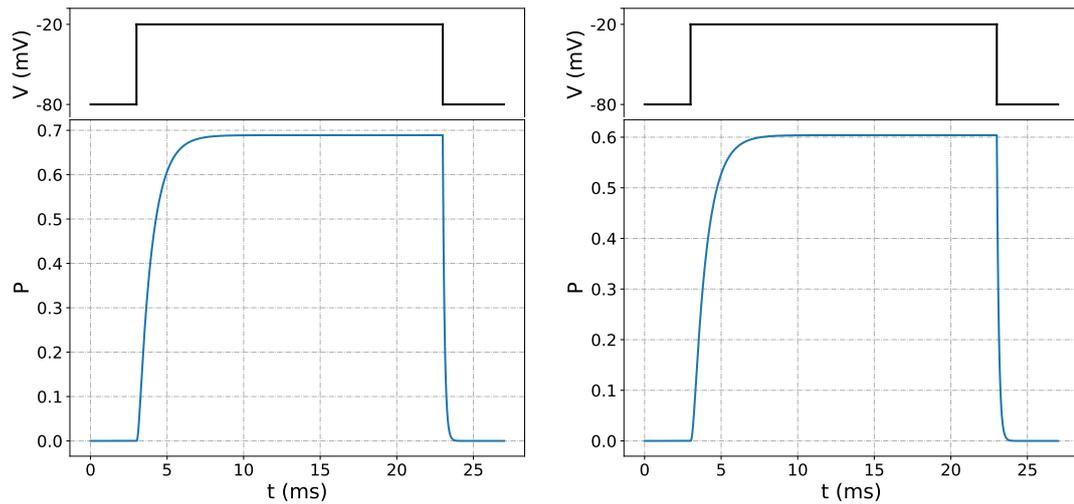
Figura 4.2: Intensidad de Corriente I_{Ca} a diferentes voltajes para canales P/Q, N y R.

Figura 4.3: Probabilidad de apertura a diferentes voltajes para canales P/Q, N y R.

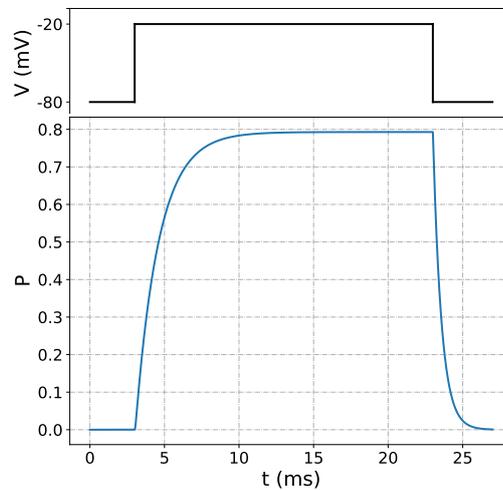
En la figura 4.2 se muestran las corrientes de Ca^{2+} para cada tipo de canal a diferentes potenciales de membrana. Las corrientes han sido normalizadas al valor de la corriente a 10 mV para los canales P/Q y N, y a 0 mV para los canales R.

Las correspondientes probabilidades de apertura para diferentes potenciales de membrana y para diferentes tipos de canales se muestran en la figura 4.3.



(a) Canales P/Q.

(b) Canales N.



(c) Canales R.

Figura 4.4: Probabilidad de apertura para canales P/Q, N y R al aplicar un pulso de voltaje.

En los resultados numéricos se usan los parámetros indicados en la Tabla 4.1 y un potencial de reposo de 60 mV. Este tipo de gráficas representan un comportamiento estático del canal, en contraste con los comportamientos que se dan cuando existen pulsos de voltaje, como es el caso de la figura 4.4 donde se pueden apreciar las probabilidades de apertura para los distintos tipos de canales al aplicar un pulso de voltaje de 0 mV con una duración de 20 ms.

Se puede observar que las cinéticas de los canales P/Q y N son muy similares, con una rápida activación cuando inicia el pulso de voltaje y una rápida desactivación cuando este finaliza, a diferencia de los canales R que presentan una cinética más lenta, tardando más tiempo tanto en activarse como en desactivarse cuando se aplica el pulso de voltaje.

Los resultados obtenidos, tanto para los comportamientos estáticos como dinámicos del canal, son los mismos obtenidos en (Li y cols., 2007), con lo que aseguramos que el modelo propuesto de canal funciona correctamente.

4.2. Difusión

4.2.1. Validación

Para comprobar que nuestro modelo es correcto, se calculan resultados numéricos con los mismos parámetros que los usados por (Nowycky y Pinter, 1993), los cuales se encuentran en la tabla 4.2.

Primero se calcula un resultado numérico sin *buffer*, con lo que sólo intervienen el pulso de corriente y la difusión del calcio. Los resultados obtenidos se pueden observar en la figura 4.5. Estos resultados son coherentes con los obtenidos en (Nowycky y Pinter, 1993). La única diferencia es que en dicho trabajo la concentración de Ca^{2+} se estabiliza a $72 \mu\text{M}$ mientras que en el presente trabajo la concentración de Ca^{2+} se estabiliza a $74 \mu\text{M}$. Esto se debe a que a diferencia del presente trabajo, el modelo propuesto en (Nowycky y Pinter, 1993) sí considera la extrusión de Ca^{2+} por medio de canales de fuga y bombas de Ca^{2+} lo que hace que la concentración final de calcio sea un poco menor.

Posterior a ello se calculan resultados numéricos donde si está presente el *buffer*. Los resultados obtenidos para la concentración de calcio se pueden observar en la figura 4.6 mientras que los obtenidos para el buffer se pueden observar en la figura 4.7.

Nuevamente, los resultados que proporciona el modelo son coherentes con los obtenidos en (Nowycky y Pinter, 1993).

Parámetro	Valor
Radio celular	7.5 μm
Ca^{2+}	
Concentración inicial	0.1 μM
D_{Ca}	200 $\mu\text{m}^2/\text{s}$
Buffer	
$[\text{B}]_T$	0.5 μM
k_f	$1 \times 10^8 \text{ M}^{-1} \text{ s}^{-1}$
k_D	5 μM
Corriente de Ca	
Duración	50 ms
Intensidad	0.5 nA
Pasos	
Δr	0.1 μm
Δt	0.1 μs

Tabla 4.2: Parámetros usados en los resultados numéricos de validación. Tomados de (Nowycky y Pinter, 1993)

4.2.2. Colaterales de Schaffer

Una vez el modelo se ha validado se procede a calcular un resultado numérico para las colaterales de Schaffer y obtener de este modo los perfiles de calcio bajo la membrana celular, necesarios para calcular los eventos de secreción de neurotransmisor.

Los parámetros para el resultado numérico de las colaterales de Schaffer fueron tomados de (Scimemi y Diamond, 2012) y se encuentran en la tabla 4.3.

Como cuestiones relevantes se puede mencionar lo siguiente:

- En promedio, el radio de los botones presinápticos hipocampales tienen un radio 345 nm.
- Presentan en promedio 260 vesículas listas para ser liberadas fácilmente, del inglés RRP(Readily Releasable Pool).
- Los canales que más contribuyen a la liberación de las sinapsis colaterales de Schaffer son los canales P/Q y N.
- La presencia de muy baja cooperatividad en los canales de calcio que desencadenan la liberación de neurotransmisor indica que en el hipocampo, a temperatura fisiológica, la transmisión excitatoria es iniciada por un solo canal de calcio y en general la terminal presináptica tiene entre 3 y 10 canales de calcio.
- Las mediciones fueron hechas en ratón.

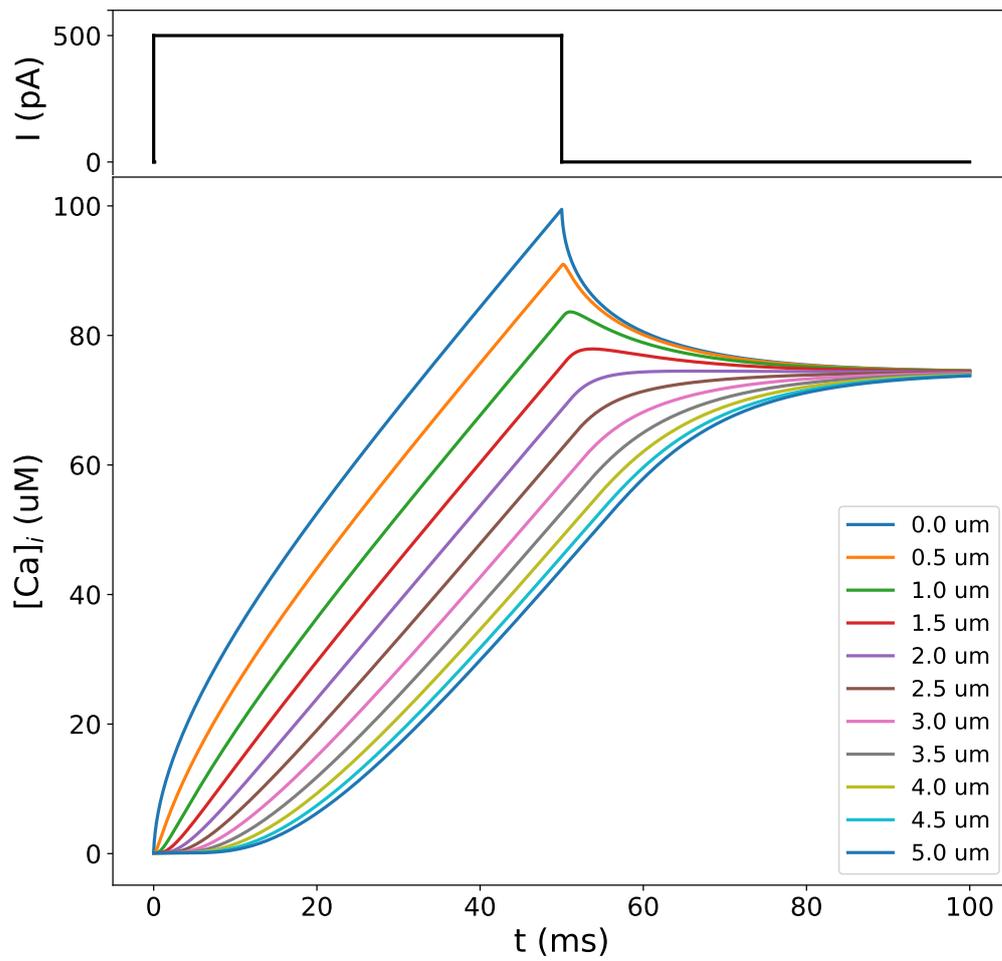


Figura 4.5: Concentración de calcio sin *buffer* a distintas distancias de la membrana celular al aplicar un pulso de corriente.

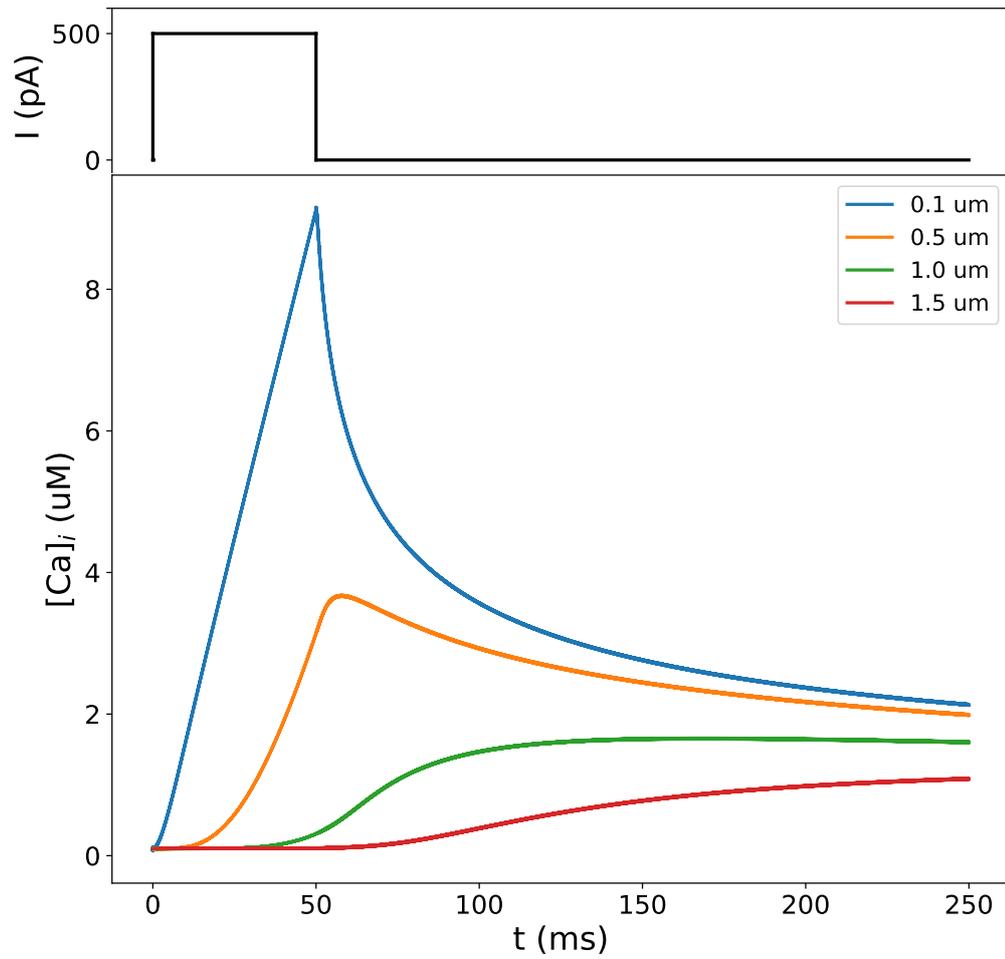


Figura 4.6: Concentración de calcio con *buffer* a distintas distancias de la membrana celular al aplicar un pulso de corriente.

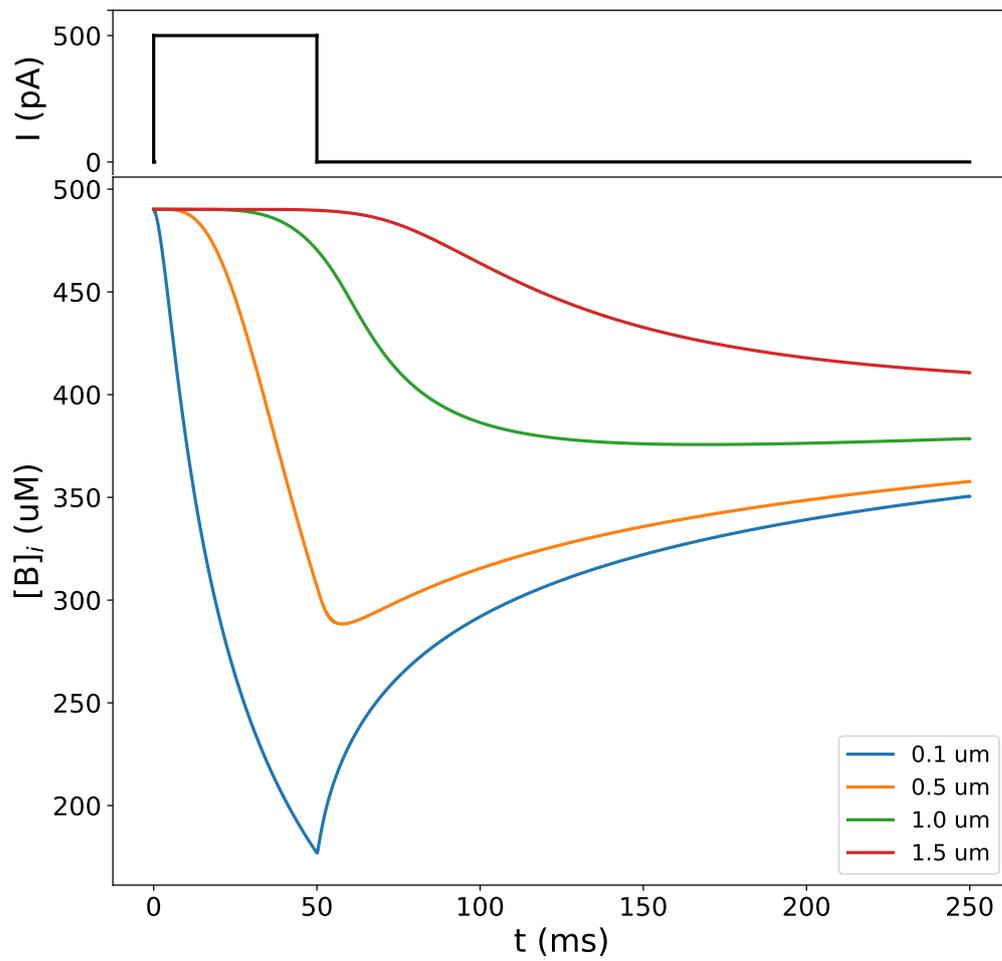


Figura 4.7: Concentración de *buffer* a distintas distancias de la membrana celular al aplicar un pulso de corriente.

Parámetro	Valor
Radio celular	345 nm
Ca ²⁺	
Concentración inicial	0.1 μM
D_{Ca}	220 $\mu\text{m}^2/\text{s}$
Buffer	
$[B]_T$	310 μM
k_f^B	$5 \times 10^8 \text{ M}^{-1}\text{s}^{-1}$
k_b^B	$5 \times 10^3 \text{ s}^{-1}$
k_D^B	10 μM
Corriente de Ca	
Duración	1 ms
Tipo de canal de Ca ²⁺	P/Q
Cantidad de canales de Ca ²⁺	3
Vesículas	
Cantidad de Vesículas	260
α	$0.3 \mu\text{M}^{-1}\text{ms}^{-1}$
β	3 ms^{-1}
k_D^V	$\beta/\alpha = 10 \mu\text{M}$
γ	30 ms^{-1}
δ	8 ms^{-1}
Pasos	
Δr	5 nm
Δt	0.1 ns

Tabla 4.3: Parámetros usados en los resultados numéricos de las colaterales de Schaffer. Tomado de (Scimemi y Diamond, 2012) y (Bollmann y cols., 2000)

Para imitar la despolarización de un potencial presináptico, a $t = 0$ ms se mantuvo un potencial de membrana $V_m = -70$ mV y a $t = 1$ ms, V_m fue elevado a 0 mV por una duración de 1 ms. Las tasas de apertura de los canales iónicos de Ca^{2+} fueron ajustadas para esta escala temporal.

Los resultados obtenidos para la concentración de Ca^{2+} sin *buffer* a diferentes distancias de la membrana celular se pueden observar en la figura 4.8.

En las figuras 4.9 y 4.10 se pueden observar los resultados obtenidos para la concentración de Ca^{2+} en presencia de *buffer* y la concentración de *buffer* respectivamente, a diferentes distancias de la membrana celular.

4.3. Secreción

El modelo de secreción propuesto está basado en el trabajo realizado en (Bollmann y cols., 2000). El modelo presentado en dicho trabajo se realiza para sinapsis glutamatérgicas llamadas Cáliz de Held. Estas son sinapsis gigantes en el sistema nervioso central auditivo de los mamíferos.

De esta manera, al ser un modelo de secreción neuronal, se adapta de mejor manera al modelo propuesto en el presente trabajo para colaterales de Schaffer, además, en ambos casos las sinapsis son glutamatérgicas.

El modelo presenta dos partes. La primer parte consiste en un esquema secuencial que describe la unión de los iones de Ca^{2+} a los sensores de Ca^{2+} (\mathbf{X}) presentes en las vesículas, con unas tasas de avance y retroceso indicadas por α y β respectivamente. El esquema de sensores de Ca^{2+} y sus ecuaciones de estado asociadas se pueden observar en la figura 4.11.

Como se indica en el esquema, son necesarios cinco iones de Ca^{2+} para propiciar la exocitosis de la vesícula.

Adicional a los sensores de Ca^{2+} , en el esquema se considera una última etapa de isomerización reversible e independiente del Ca^{2+} , común en las proteínas de unión a calcio existentes (Bollmann y cols., 2000). Las tasas de avance y retroceso para la isomerización están indicadas por γ y δ respectivamente. Una vez se concluye esta etapa se obtiene la fracción de sensores de Ca^{2+} que están en un estado que promueve la fusión vesicular, representado en el esquema por \mathbf{C} .

La isomerización es un proceso químico mediante el cual una molécula se transforma en otra que posee los mismos átomos que la inicial pero dispuestos de una manera distinta (*Isomerization*, 2017).

La segunda parte del modelo consiste en un esquema que representa la fusión de las vesículas a la membrana celular. El proceso de fusión no es reversible y la tasa de fusión (ρ) está escalada por \mathbf{C} . El esquema de fusión y sus ecuaciones de estado asociadas se pueden observar en la figura 4.12. \mathbf{V} son las vesículas liberables y \mathbf{F} son las vesículas fusionadas.

Finalmente, teniendo los perfiles de Ca^{2+} bajo la membrana celular obtenidos de los resultados numéricos de las colaterales de Schaffer, que se encuentran en la figura 4.9, se aplica el modelo de secreción (unión de iones de Ca^{2+} a sensores de Ca^{2+} vesicular y

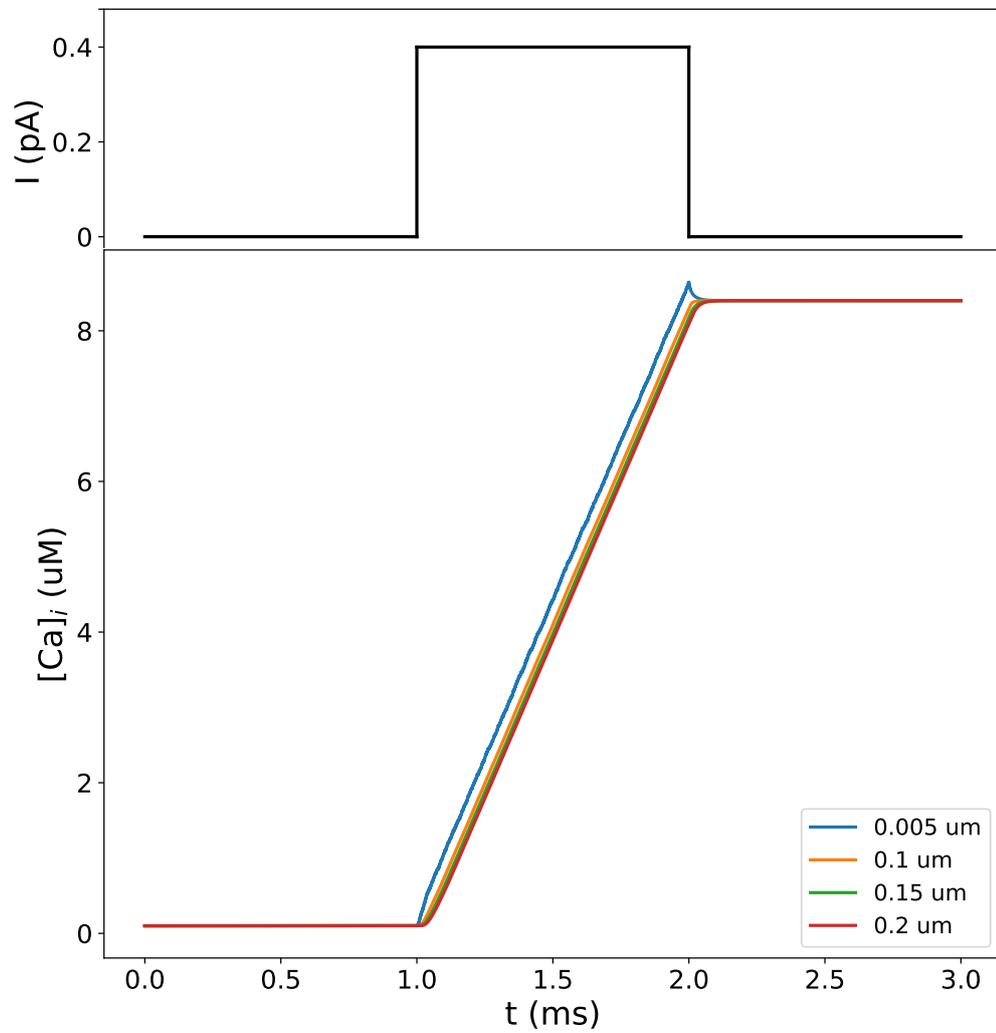


Figura 4.8: Concentración de calcio sin *buffer* a distintas distancias de la membrana celular al aplicar un pulso de voltaje (se muestra la corriente generada por el pulso). Colaterales de Schaffer.

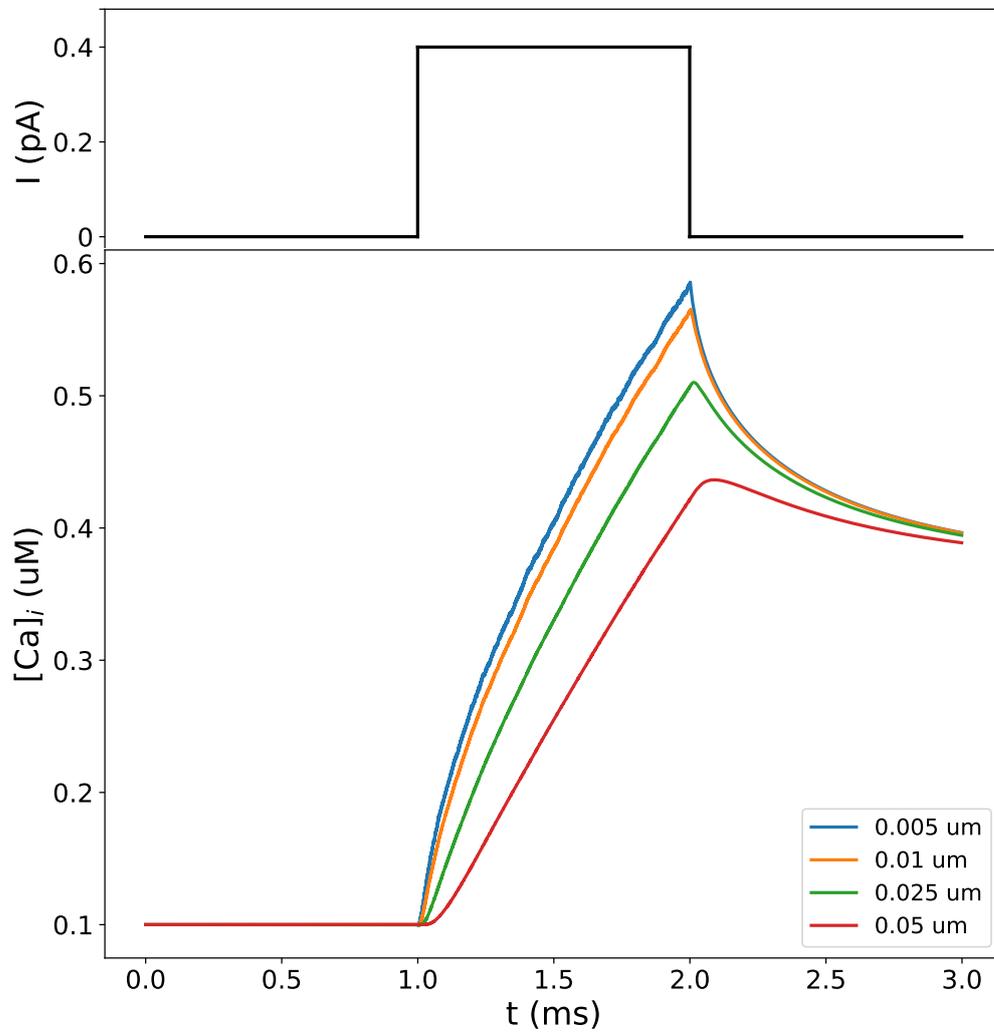


Figura 4.9: Concentración de calcio con *buffer* a distintas distancias de la membrana celular al aplicar un pulso de voltaje (se muestra la corriente generada por el pulso). Colaterales de Schaffer.

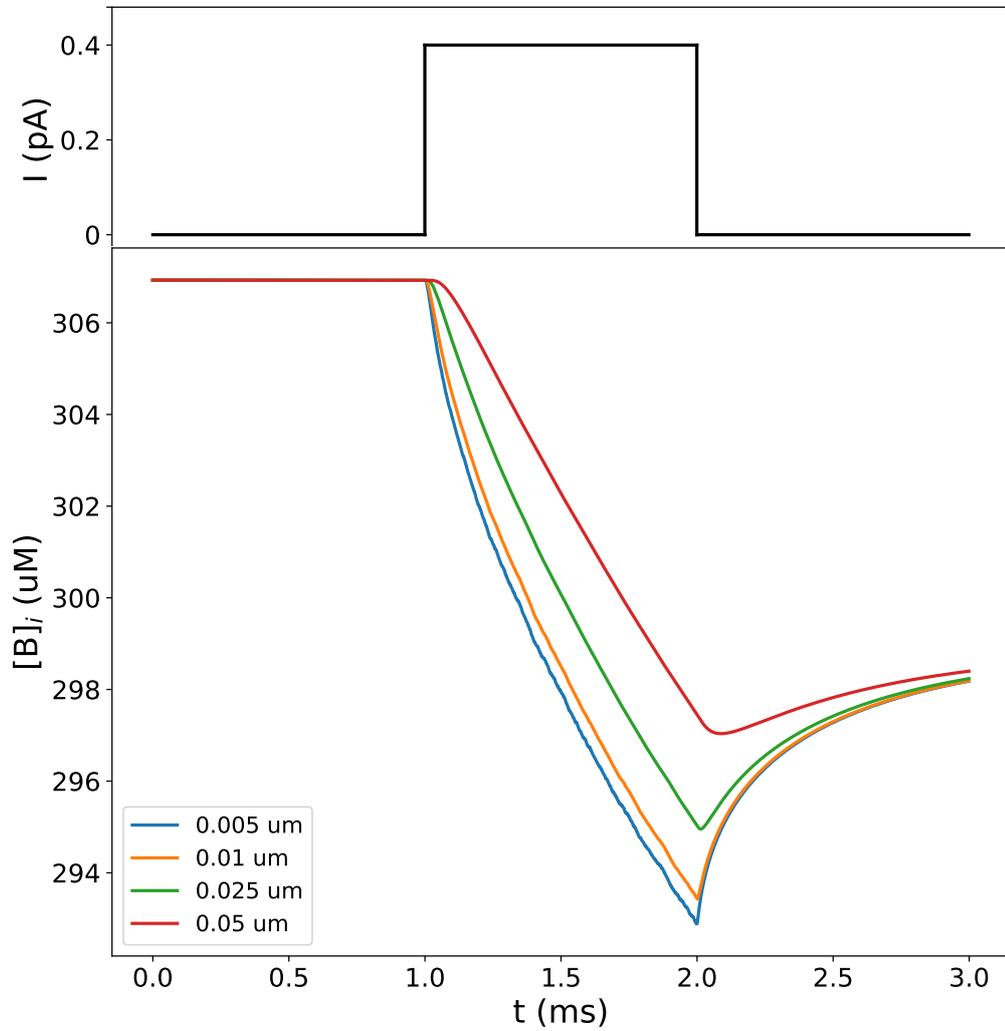
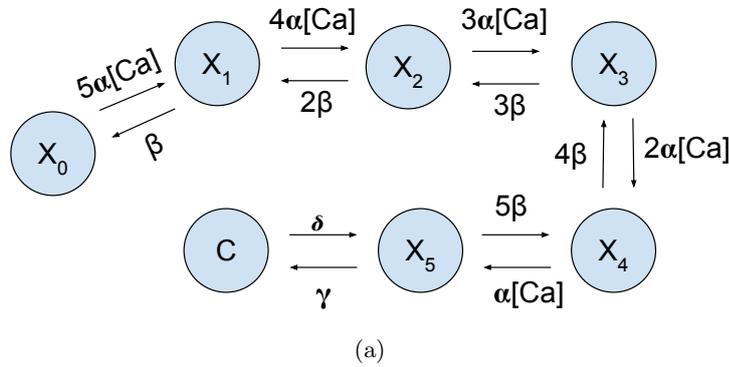


Figura 4.10: Concentración de *buffer* a distintas distancias de la membrana celular al aplicar un pulso de voltaje (se muestra la corriente generada por el pulso). Colaterales de Schaffer.



$$\begin{aligned}
 \frac{dX_0}{dt} &= \beta X_1 - 5\alpha[\text{Ca}]X_0 \\
 \frac{dX_1}{dt} &= 5\alpha[\text{Ca}]X_0 + 2\beta X_2 - (\beta + 4\alpha[\text{Ca}])X_1 \\
 \frac{dX_2}{dt} &= 4\alpha[\text{Ca}]X_1 + 3\beta X_3 - (2\beta + 3\alpha[\text{Ca}])X_2 \\
 \frac{dX_3}{dt} &= 3\alpha[\text{Ca}]X_2 + 4\beta X_4 - (3\beta + 2\alpha[\text{Ca}])X_3 \\
 \frac{dX_4}{dt} &= 2\alpha[\text{Ca}]X_3 + 5\beta X_5 - (4\beta + \alpha[\text{Ca}])X_4 \\
 \frac{dX_5}{dt} &= \alpha[\text{Ca}]X_4 + \delta C - (5\beta + \gamma)X_5 \\
 \frac{dC}{dt} &= \gamma X_5 - \delta C
 \end{aligned}$$

(b)

Figura 4.11: Esquema de sensores de Ca^{2+} (4.11a) y sus ecuaciones de estado asociadas (4.11b).

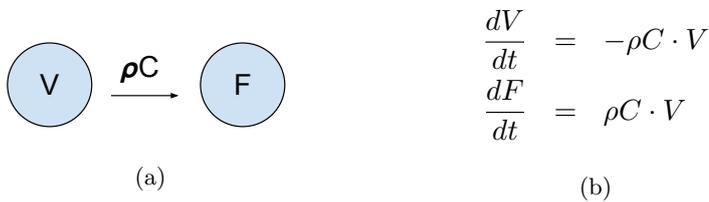


Figura 4.12: Esquema de fusión vesicular (4.12a) y sus ecuaciones de estado asociadas (4.12b).

fusión vesicular) para obtener la cantidad de vesículas que liberan su contenido debido al potencial de acción simulado. Los parámetros usados para el resultado numérico de secreción se encuentran en la tabla 4.3. Las tasas de avance, de retroceso y de fusión fueron ajustadas para esta escala temporal.

Los resultados de la secreción se pueden observar en la figura 4.13.

Como el proceso continua 1 ms después de que finaliza el potencial de acción y el modelo no tiene ningún modo de retornar la concentración de Ca^{2+} intracelular a su estado basal, se observa que la secreción continua hasta agotar todas las vesículas disponibles. Para saber aproximadamente cuántas vesículas se liberan con el potencial de acción aplicado, se puede observar qué valor se corresponde al tiempo de 0.1 ms después de haber finalizado el potencial de acción, ya que después de ese tiempo en un contexto fisiológico, la concentración de Ca^{2+} comenzaría a normalizarse (Scimemi y Diamond, 2012). De esta manera se puede decir que el potencial de acción aplicado genera una secreción de aproximadamente 245 vesículas, que se corresponde con el 94% de la población inicial de vesículas.

Es interesante ver que la secreción vesicular no se dispara hasta aproximadamente los 1.4 ms, lo que evidencia la acción cooperativa del calcio necesaria para que se de la fusión de las membranas.

4.4. Conclusiones

De los resultados obtenidos se puede apreciar que las concentraciones de calcio obtenidos cerca de la membrana son pequeños en comparación con otras terminales presinápticas.

El *buffer* reduce de manera significativa las concentraciones de calcio, de modo que sus características van a tener gran importancia en el comportamiento del sistema.

Respecto a la secreción, esta parece ser más lenta que en otras sinapsis lo que puede tener un impacto en la funcionalidad del sistema.

En cualquier caso, un modelo microscópico permitiría profundizar en el estudio del sistema y obtener resultados mucho más precisos. Este consistiría en modelar cada una de las partículas involucradas en el proceso, en contraste con el modelo macroscópico propuesto en el presente trabajo que en lugar de partículas maneja concentraciones.

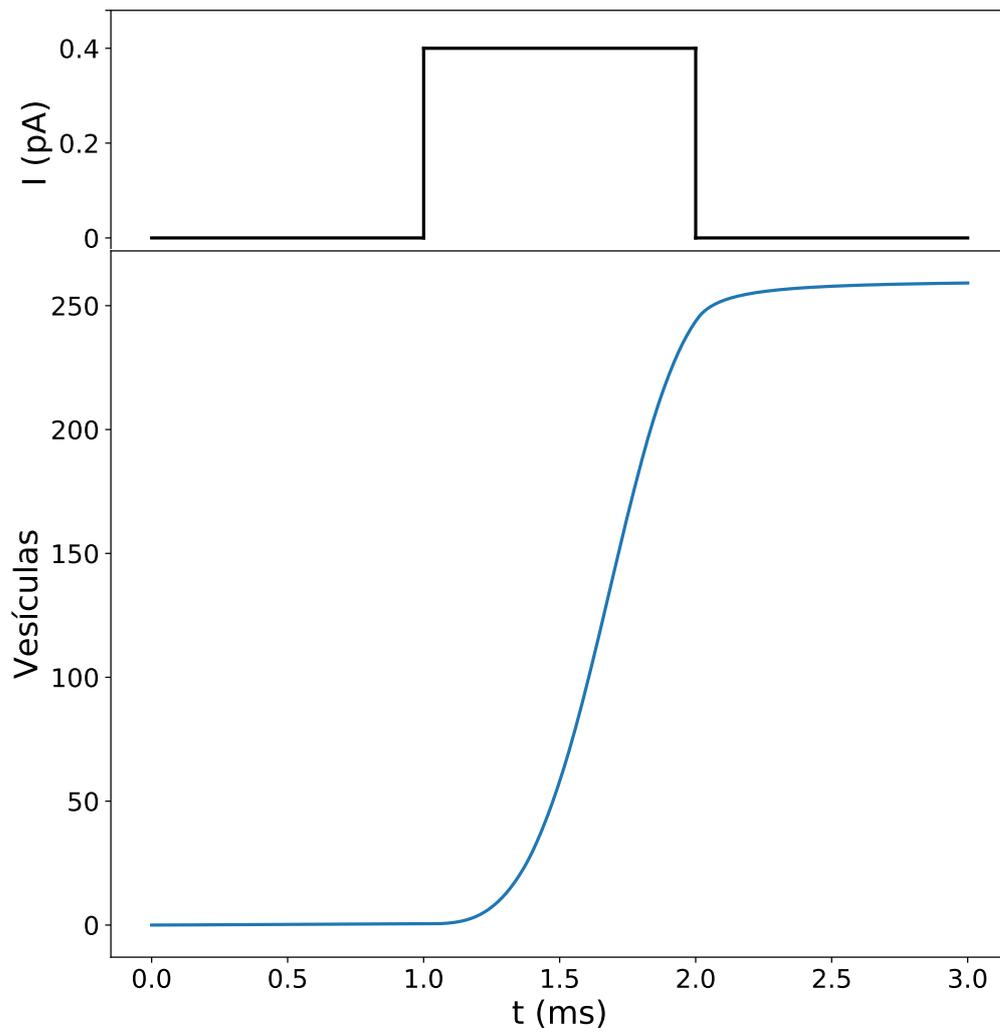


Figura 4.13: Cantidad de vesículas que liberan su contenido.

Referencias

- Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K., y Walter, P. (2008). *Molecular biology of the cell* (5.^a ed.). Garland Science.
- Ames, W. F. (1977). *Numerical methods for partial differential equations*. Academic press, inc.
- Arning, K. (2009). *Mathematical modelling and simulation of ion channels* (PhD Tesis). Radon Institute for Computational and Applied Mathematics.
- Bollmann, J. H., Sakmann, B., y Borst, J. G. G. (2000, Agosto). Calcium sensitivity of glutamate release in calyx-type terminal. *Science*, 289. Descargado de <http://www.sciencemag.org/>
- Ciclo de las vesículas sinápticas [Página web]. (2017, 6). Wikipedia. Descargado de https://es.wikipedia.org/wiki/Ciclo_de_las_vesículas_sinápticas
- Crank, J. (1975). *The mathematics of diffusion* (2.^a ed.). Clarendon press Oxford.
- Crank, J., y Nicolson, P. (1947). A practical method for numerical evaluation of solutions of partial differential equations of heat-conduction type. *Mathematical Proceedings of the Cambridge Philosophical Society*, 43, 50-67.
- Cucic, A. (2017, 8). *Axon collateral* [Página web]. Med Health Daily. Descargado de <https://www.medhealthdaily.com/axon-collateral/>
- Heinemann, C., Crow, R. H., Neher, E., y Zucker, R. S. (1994, Diciembre). Kinetics of the secretory response in bovine chromaffin cells following flash photolysis of caged ca^{2+} . *Biophysical Journal*, 67, 2546-2557.
- Isomerization [Página web]. (2017, 10). Wikipedia. Descargado de <https://en.wikipedia.org/wiki/Isomerization>
- Kandel, E. R., Schwartz, J. H., Jessell, T. M., Siegelbaum, S. A., y Hudspeth, A. J. (2013). *Principles of neural science* (5.^a ed.). Mc Graw Hill Medical.
- Klingauf, J., y Neher, E. (1997a, Febrero). Modeling buffered ca^{2+} diffusion near the membrane: Implications for secretion in neuroendocrine cells. *Biophysical Journal*, 72, 674-690.
- Klingauf, J., y Neher, E. (1997b, Febrero). Time courses of calcium and calcium-bound buffers following calcium influx in a model cell. *Biophysical journal*, 72, 674-690.
- Law of mass action [Página web]. (2017, 9). Wikipedia. Descargado de https://en.wikipedia.org/wiki/Law_of_mass_action
- Li, L., Bischofberger, J., y Jonas, P. (2007, Diciembre). Differential gating and recruitment of p/q-,n, and r-type ca^{2+} channels in hippocampal mossy fiber boutons.

- The journal of neuroscience*, 27(49), 13420-13429.
- Nowycky, M. C., y Pinter, M. J. (1993, Enero). Time courses of calcium and calcium-bound buffers following calcium influx in a model cell. *Biophysical journal*, 64, 77-91.
- Rubin, M., y Safdieh, J. (2009). *Netter. neuroanatomía esencial*. Elsevier Masson.
- Sala, F., y Hernández-Cruz, A. (1990, Febrero). Calcium diffusion modeling in a spherical neuron. *Biophysical journal*, 57, 313-324.
- Scimemi, A., y Diamond, J. S. (2012, Diciembre). The number and organization of Ca^{2+} channels in the active zone shapes neurotransmitter release from schaffer collateral synapses. *The journal of neuroscience*, 32(50).
- Wright, M., y cols. (2017). The hippocampus. *WikiJournal of Medicine*. Descargado de https://en.wikiversity.org/wiki/WikiJournal_of_Medicine/The_Hippocampus
- y Cajal, S. R. (1909-1911). *Histologie du système nerveux de l'homme et vertébrés* (Vol. 1 y 2). A. Maloine. Descargado de <https://www.biodiversitylibrary.org/bibliography/48637>

Apéndice A

Despeje de ecuaciones

A.1. ecuación (3.5) a (3.6)

Considerando $u = [Ca^{2+}]r = Cr$ tenemos que

$$\frac{\partial u}{\partial r} = \frac{\partial C}{\partial r}r + C$$

Y

$$\begin{aligned}\frac{\partial^2 u}{\partial r^2} &= \frac{\partial}{\partial r} \left(\frac{\partial u}{\partial r} \right) \\ &= \frac{\partial}{\partial r} \left(\frac{\partial C}{\partial r}r + C \right) \\ &= \frac{\partial^2 C}{\partial r^2}r + \frac{\partial C}{\partial r} + \frac{\partial C}{\partial r} \\ &= \frac{\partial^2 C}{\partial r^2}r + 2\frac{\partial C}{\partial r} \quad .\end{aligned}$$

Multiplicando la expresión anterior por $\frac{1}{r}$ llegamos a:

$$\frac{1}{r} \frac{\partial^2 u}{\partial r^2} = \frac{\partial^2 C}{\partial r^2} + \frac{2}{r} \frac{\partial C}{\partial r} \quad .$$

Por otra parte, de la ecuación 3.5 tenemos que:

$$\frac{\partial^2 C}{\partial r^2} + \frac{2}{r} \frac{\partial C}{\partial r} = \frac{1}{D_{Ca}} \frac{\partial C}{\partial t}$$

y usando el resultado encontrado previamente, finalmente llegamos a que:

$$\begin{aligned}\frac{1}{D_{Ca}} \frac{\partial C}{\partial t} &= \frac{1}{r} \frac{\partial^2 u}{\partial r^2} \\ \frac{1}{D_{Ca}} \frac{\partial u}{\partial t} \frac{1}{r} &= \frac{1}{r} \frac{\partial^2 u}{\partial r^2} \\ \frac{\partial u}{\partial t} &= D_{Ca} \frac{\partial^2 u}{\partial r^2}\end{aligned}$$

A.2. ecuación (3.11) a (3.12)

$$\begin{aligned} \frac{r_i}{\Delta t} ([Ca^{2+}]_{i,t+1} - [Ca^{2+}]_{i,t}) &= \frac{DCa}{2\Delta^2 r} [(r_i + \Delta r)([Ca^{2+}]_{i+1,t+1} + [Ca^{2+}]_{i+1,t}) \\ &\quad - 2r_i([Ca^{2+}]_{i,t+1} + [Ca^{2+}]_{i,t}) \\ &\quad + (r_i - \Delta r)([Ca^{2+}]_{i-1,t+1} + [Ca^{2+}]_{i-1,t})] \end{aligned}$$

Se multiplica por $\frac{\Delta t}{r_i}$

$$\begin{aligned} [Ca^{2+}]_{i,t+1} - [Ca^{2+}]_{i,t} &= \frac{DCa\Delta t}{2r_i\Delta^2 r} [(r_{i+1})([Ca^{2+}]_{i+1,t+1} + [Ca^{2+}]_{i+1,t}) \\ &\quad - 2r_i([Ca^{2+}]_{i,t+1} + [Ca^{2+}]_{i,t}) + (r_{i-1})([Ca^{2+}]_{i-1,t+1} + [Ca^{2+}]_{i-1,t})] \end{aligned}$$

Se eliminan los paréntesis al realizar la multiplicación correspondiente al lado derecho de la igualdad

$$\begin{aligned} [Ca^{2+}]_{i,t+1} - [Ca^{2+}]_{i,t} &= \frac{DCa\Delta tr_{i+1}}{2r_i\Delta^2 r} [Ca^{2+}]_{i+1,t+1} + \frac{DCa\Delta tr_{i+1}}{2r_i\Delta^2 r} [Ca^{2+}]_{i+1,t} - \frac{DCa\Delta t}{\Delta^2 r} [Ca^{2+}]_{i,t+1} \\ &\quad - \frac{DCa\Delta t}{\Delta^2 r} [Ca^{2+}]_{i,t} + \frac{DCa\Delta tr_{i-1}}{2r_i\Delta^2 r} [Ca^{2+}]_{i-1,t+1} + \frac{DCa\Delta tr_{i-1}}{2r_i\Delta^2 r} [Ca^{2+}]_{i-1,t} \end{aligned}$$

Se agrupan los términos correspondientes a $t+1$ al lado izquierdo y los correspondientes a t al lado derecho de la igualdad

$$\begin{aligned} [Ca^{2+}]_{i,t+1} - \frac{DCa\Delta tr_{i+1}}{2r_i\Delta^2 r} [Ca^{2+}]_{i+1,t+1} + \frac{DCa\Delta t}{\Delta^2 r} [Ca^{2+}]_{i,t+1} - \frac{DCa\Delta tr_{i-1}}{2r_i\Delta^2 r} [Ca^{2+}]_{i-1,t+1} \\ = \frac{DCa\Delta tr_{i+1}}{2r_i\Delta^2 r} [Ca^{2+}]_{i+1,t} - \frac{DCa\Delta t}{\Delta^2 r} [Ca^{2+}]_{i,t} + \frac{DCa\Delta tr_{i-1}}{2r_i\Delta^2 r} [Ca^{2+}]_{i-1,t} + [Ca^{2+}]_{i,t} \end{aligned}$$

Se reorganizan los términos

$$\begin{aligned} - \frac{DCa\Delta t}{2r_i\Delta^2 r} (r_{i-1}) [Ca^{2+}]_{i-1,t+1} + \left(1 + \frac{DCa\Delta t}{\Delta^2 r}\right) [Ca^{2+}]_{i,t+1} - \frac{DCa\Delta t}{2r_i\Delta^2 r} (r_{i+1}) [Ca^{2+}]_{i+1,t+1} \\ = \frac{DCa\Delta t}{2r_i\Delta^2 r} (r_{i-1}) [Ca^{2+}]_{i-1,t} + \left(1 - \frac{DCa\Delta t}{\Delta^2 r}\right) [Ca^{2+}]_{i,t} + \frac{DCa\Delta t}{2r_i\Delta^2 r} (r_{i+1}) [Ca^{2+}]_{i+1,t} \end{aligned}$$

A.3. ecuación (3.14) a (3.15)

$$\begin{aligned} \frac{r_n}{\Delta t} ([Ca^{2+}]_{n,t+1} - [Ca^{2+}]_{n,t}) &= \frac{DCa}{2\Delta^2 r} [(r_n - \Delta r)([Ca^{2+}]_{n-1,t+1} + [Ca^{2+}]_{n-1,t}) \\ &\quad - (r_n - \Delta r)([Ca^{2+}]_{n,t+1} + [Ca^{2+}]_{n,t})] \end{aligned}$$

Multiplicamos la ecuación por $\Delta t/r_n$

$$\begin{aligned} [Ca^{2+}]_{n,t+1} - [Ca^{2+}]_{n,t} &= \frac{DCa\Delta t}{2r_n\Delta^2 r}(r_{n-1})([Ca^{2+}]_{n-1,t+1} + [Ca^{2+}]_{n-1,t}) \\ &\quad - \frac{DCa\Delta t}{2r_n\Delta^2 r}(r_{n-1})([Ca^{2+}]_{n,t+1} + [Ca^{2+}]_{n,t}) \end{aligned}$$

Se eliminan los paréntesis realizando la multiplicación correspondiente y se agrupan los términos correspondientes a $t + 1$ del lado izquierdo y los correspondientes a t del lado derecho de la ecuación

$$\begin{aligned} [Ca^{2+}]_{n,t+1} - \frac{DCa\Delta t}{2r_n\Delta^2 r}(r_{n-1})[Ca^{2+}]_{n-1,t+1} + \frac{DCa\Delta t}{2r_n\Delta^2 r}(r_{n-1})[Ca^{2+}]_{n,t+1} \\ = \frac{DCa\Delta t}{2r_n\Delta^2 r}(r_{n-1})[Ca^{2+}]_{n-1,t} - \frac{DCa\Delta t}{2r_n\Delta^2 r}(r_{n-1})[Ca^{2+}]_{n,t} + [Ca^{2+}]_{n,t} \end{aligned}$$

Finalmente se reorganizan los términos

$$\begin{aligned} - \frac{DCa\Delta t}{2r_n\Delta^2 r}(r_{n-1})[Ca^{2+}]_{n-1,t+1} + \left(1 + \frac{DCa\Delta t}{2r_n\Delta^2 r}(r_{n-1})\right) [Ca^{2+}]_{n,t+1} \\ = \frac{DCa\Delta t}{2r_n\Delta^2 r}(r_{n-1})[Ca^{2+}]_{n-1,t} + \left(1 - \frac{DCa\Delta t}{2r_n\Delta^2 r}(r_{n-1})\right) [Ca^{2+}]_{n,t} \end{aligned}$$

A.4. ecuación (3.17) a (3.18)

$$\begin{aligned} \frac{r_0}{\Delta t}([Ca^{2+}]_{0,t+1} - [Ca^{2+}]_{0,t}) &= \frac{3DCa}{2\Delta^2 r}[(r_0 + \Delta r)([Ca^{2+}]_{1,t+1} + [Ca^{2+}]_{1,t}) \\ &\quad - (r_0 + \Delta r)([Ca^{2+}]_{0,t+1} + [Ca^{2+}]_{0,t})] \end{aligned}$$

Multiplicamos la ecuación por $\Delta t/r_0$

$$\begin{aligned} [Ca^{2+}]_{0,t+1} - [Ca^{2+}]_{0,t} &= \frac{3DCa\Delta t}{2r_0\Delta^2 r}r_1([Ca^{2+}]_{1,t+1} + [Ca^{2+}]_{1,t}) \\ &\quad - \frac{3DCa\Delta t}{2r_0\Delta^2 r}r_1([Ca^{2+}]_{0,t+1} + [Ca^{2+}]_{0,t}) \end{aligned}$$

Se eliminan los paréntesis realizando la multiplicación correspondiente y se agrupan los términos correspondientes a $t + 1$ del lado izquierdo y los correspondientes a t del lado derecho de la ecuación

$$\begin{aligned} [Ca^{2+}]_{0,t+1} - \frac{3DCa\Delta t}{2r_0\Delta^2 r}r_1[Ca^{2+}]_{1,t+1} + \frac{3DCa\Delta t}{2r_0\Delta^2 r}r_1[Ca^{2+}]_{0,t+1} \\ = \frac{3DCa\Delta t}{2r_0\Delta^2 r}r_1[Ca^{2+}]_{1,t} - \frac{3DCa\Delta t}{2r_0\Delta^2 r}r_1[Ca^{2+}]_{0,t} + [Ca^{2+}]_{0,t} \end{aligned}$$

Finalmente se reorganizan los términos

$$\begin{aligned} & -\frac{3D_{Ca}\Delta t}{2r_0\Delta^2 r}r_1[Ca^{2+}]_{1,t+1} + \left(1 + \frac{3D_{Ca}\Delta t}{2r_0\Delta^2 r}r_1\right)[Ca^{2+}]_{0,t+1} \\ & = \frac{3D_{Ca}\Delta t}{2r_0\Delta^2 r}r_1[Ca^{2+}]_{1,t} + \left(1 - \frac{3D_{Ca}\Delta t}{2r_0\Delta^2 r}r_1\right)[Ca^{2+}]_{0,t} \end{aligned}$$

A.5. ecuación (3.19)

Partiendo del esquema (3.7) y considerando $[B] = [B]_{i,0}$, $[Ca^{2+}B] = [Ca^{2+}B]_{i,0}$ y $[Ca^{2+}] = [Ca^{2+}]_{i,0}$ se tiene que

$$\begin{aligned} k_d &= \frac{k_b}{k_f} = \frac{[B][Ca^{2+}]}{[Ca^{2+}B]} \\ [B] &= \frac{[Ca^{2+}B]k_b}{[Ca]k_f} \end{aligned}$$

Despejando $[Ca^{2+}B]$ de la ecuación (3.9) y reemplazándolo en la ecuación anterior se tiene

$$[B] = \frac{[B]_T k_b - [B]k_b}{[Ca^{2+}]k_f}$$

y despejando para $[B]$

$$\begin{aligned} [B][Ca^{2+}]k_f &= [B]_T k_b - [B]k_b \\ [B][Ca^{2+}]k_f + [B]k_b &= [B]_T k_b \\ [B]([Ca^{2+}]k_f + k_b) &= [B]_T k_b \\ [B] &= \frac{[B]_T k_b}{[Ca^{2+}]k_f + k_b} \end{aligned}$$

A.6. ecuación (3.22) a (3.25)

$$\begin{aligned} \frac{1}{\Delta t}([Ca^{2+}]_{i,t+1} - [Ca^{2+}]_{i,t}) &= k_b[B]_T - \frac{k_b}{2}([B]_{i,t+1} + [B]_{i,t}) \\ &\quad - \frac{k_f}{2}([B]_{i,t+1}[Ca^{2+}]_{i,t} + [B]_{i,t}[Ca^{2+}]_{i,t+1}) \end{aligned}$$

Se eliminan los paréntesis al realizar las multiplicaciones necesarias

$$\begin{aligned} \frac{1}{\Delta t}[Ca^{2+}]_{i,t+1} - \frac{1}{\Delta t}[Ca^{2+}]_{i,t} &= k_b[B]_T - \frac{k_b}{2}[B]_{i,t+1} - \frac{k_b}{2}[B]_{i,t} \\ &\quad - \frac{k_f}{2}[B]_{i,t+1}[Ca^{2+}]_{i,t} - \frac{k_f}{2}[B]_{i,t}[Ca^{2+}]_{i,t+1} \end{aligned}$$

Se agrupan los términos correspondientes a $t + 1$ del lado izquierdo y los correspondientes a t del lado derecho de la ecuación con respecto a $[Ca^{2+}]$

$$\begin{aligned} \frac{1}{\Delta t}[Ca^{2+}]_{i,t+1} + \frac{k_f}{2}[B]_{i,t}[Ca^{2+}]_{i,t+1} = k_b[B]_T - \left(\frac{k_b}{2} + \frac{k_f}{2}[Ca^{2+}]_{i,t} \right) [B]_{i,t+1} \\ - \frac{k_b}{2}[B]_{i,t} + \frac{1}{\Delta t}[Ca^{2+}]_{i,t} \end{aligned}$$

Se reemplaza $[B]_{i,t+1}$ por el que define la ecuación (3.24)

$$\begin{aligned} \frac{1}{\Delta t}[Ca^{2+}]_{i,t+1} + \frac{k_f}{2}[B]_{i,t}[Ca^{2+}]_{i,t+1} = k_b[B]_T - \frac{k_b}{2}[B]_{i,t} + \frac{1}{\Delta t}[Ca^{2+}]_{i,t} \\ - \left(\frac{k_b}{2} + \frac{k_f}{2}[Ca^{2+}]_{i,t} \right) \left(\frac{2k_b[B]_T + \left(\frac{2}{\Delta t} - k_b - k_f[Ca^{2+}]_{i,t+1} \right) [B]_{i,t}}{\frac{2}{\Delta t} + k_b + k_f[Ca^{2+}]_{i,t}} \right) \end{aligned}$$

Se multiplica por $\frac{2}{\Delta t} + k_b + k_f[Ca^{2+}]_{i,t}$

$$\begin{aligned} (1) &= \left(\frac{1}{\Delta t}[Ca^{2+}]_{i,t+1} + \frac{k_f}{2}[B]_{i,t}[Ca^{2+}]_{i,t+1} \right) \left(\frac{2}{\Delta t} + k_b + k_f[Ca^{2+}]_{i,t} \right) \\ &= \left(\frac{2}{\Delta t} + k_b + k_f[Ca^{2+}]_{i,t} \right) \left(k_b[B]_T - \frac{k_b}{2}[B]_{i,t} + \frac{1}{\Delta t}[Ca^{2+}]_{i,t} \right) \\ &- \left(\frac{k_b}{2} + \frac{k_f}{2}[Ca^{2+}]_{i,t} \right) \left(2k_b[B]_T + \left(\frac{2}{\Delta t} - k_b - k_f[Ca^{2+}]_{i,t+1} \right) [B]_{i,t} \right) = (2) \end{aligned}$$

Se eliminan los paréntesis al realizar las multiplicaciones necesarias para (2)

$$\begin{aligned} (2) &= \frac{2k_b}{\Delta t}[B]_T + \underbrace{k_b^2[B]_T}_a + \underbrace{k_f k_b [Ca^{2+}]_{i,t} [B]_T}_b - \frac{k_b}{\Delta t}[B]_{i,t} - \underbrace{\frac{k_b^2}{2}[B]_{i,t}}_d - \underbrace{\frac{k_b k_f}{2}[Ca^{2+}]_{i,t} [B]_{i,t}}_e \\ &\underbrace{-k_b^2[B]_T}_a - \underbrace{k_f k_b [Ca^{2+}]_{i,t} [B]_T}_b - \frac{k_b}{\Delta t}[B]_{i,t} + \underbrace{\frac{k_b^2}{2}[B]_{i,t}}_d + \frac{k_b k_f}{2}[Ca^{2+}]_{i,t+1} [B]_{i,t} \\ &- \frac{k_f}{\Delta t}[Ca^{2+}]_{i,t} [B]_{i,t} + \underbrace{\frac{k_f k_b}{2}[Ca^{2+}]_{i,t} [B]_{i,t}}_e + \frac{k_f^2}{2}[Ca^{2+}]_{i,t} [Ca^{2+}]_{i,t+1} [B]_{i,t} \\ &+ \frac{1}{\Delta t}[Ca^{2+}]_{i,t} \left(\frac{2}{\Delta t} + k_b + k_f[Ca^{2+}]_{i,t} \right) \end{aligned}$$

Se simplifica y se agrupan términos semejantes

$$\begin{aligned} (2) &= \frac{2k_b}{\Delta t}[B]_T - \frac{2k_b}{\Delta t}[B]_{i,t} + \frac{1}{\Delta t}[Ca^{2+}]_{i,t} \left(\frac{2}{\Delta t} + k_b + k_f[Ca^{2+}]_{i,t} - k_f[B]_{i,t} \right) \\ &+ \left(\frac{k_b k_f}{2}[B]_{i,t} + \frac{k_f^2}{2}[Ca^{2+}]_{i,t} [B]_{i,t} \right) [Ca^{2+}]_{i,t+1} \end{aligned}$$

Se eliminan los paréntesis al realizar las multiplicaciones necesarias para (1)

$$(1) = \frac{2}{\Delta t^2} [Ca^{2+}]_{i,t+1} + \frac{k_b}{\Delta t} [Ca^{2+}]_{i,t+1} + \frac{k_f}{\Delta t} [Ca^{2+}]_{i,t+1} [Ca^{2+}]_{i,t} + \frac{k_f}{\Delta t} [Ca^{2+}]_{i,t+1} [B]_{i,t} \\ + \frac{k_f k_b}{2} [Ca^{2+}]_{i,t+1} [B]_{i,t} + \frac{k_f^2}{2} [Ca^{2+}]_{i,t+1} [B]_{i,t} [Ca^{2+}]_{i,t}$$

Se agrupan términos semejantes

$$(1) = \left(\frac{2}{\Delta t^2} + \frac{k_b}{\Delta t} + \frac{k_f}{\Delta t} [Ca^{2+}]_{i,t} + \frac{k_f}{\Delta t} [B]_{i,t} + \frac{k_f k_b}{2} [B]_{i,t} + \frac{k_f^2}{2} [B]_{i,t} [Ca^{2+}]_{i,t} \right) [Ca^{2+}]_{i,t+1}$$

Se agrupan los términos correspondientes a $t + 1$ del lado izquierdo y los correspondientes a t del lado derecho de la ecuación con respecto a $[Ca^{2+}]$

$$\left(\frac{2}{\Delta t^2} + \frac{k_b}{\Delta t} + \frac{k_f}{\Delta t} [Ca^{2+}]_{i,t} + \frac{k_f}{\Delta t} [B]_{i,t} + \underbrace{\frac{k_f k_b}{2} [B]_{i,t}}_a + \underbrace{\frac{k_f^2}{2} [B]_{i,t} [Ca^{2+}]_{i,t}}_b \right) [Ca^{2+}]_{i,t+1} \\ - \left(\underbrace{\frac{k_b k_f}{2} [B]_{i,t}}_a + \underbrace{\frac{k_f^2}{2} [Ca^{2+}]_{i,t} [B]_{i,t}}_b \right) [Ca^{2+}]_{i,t+1} \\ = \frac{2k_b}{\Delta t} [B]_T - \frac{2k_b}{\Delta t} [B]_{i,t} + \frac{1}{\Delta t} [Ca^{2+}]_{i,t} \left(\frac{2}{\Delta t} + k_b + k_f [Ca^{2+}]_{i,t} - k_f [B]_{i,t} \right)$$

Se simplifica y se multiplica por Δt

$$\left(\frac{2}{\Delta t} + k_b + k_f [Ca^{2+}]_{i,t} + k_f [B]_{i,t} \right) [Ca^{2+}]_{i,t+1} \\ = 2k_b [B]_T - 2k_b [B]_{i,t} + \left(\frac{2}{\Delta t} + k_b + k_f [Ca^{2+}]_{i,t} - k_f [B]_{i,t} \right) [Ca^{2+}]_{i,t}$$

Finalmente se despeja para $[Ca^{2+}]_{i,t+1}$

$$[Ca^{2+}]_{i,t+1} = \frac{2k_b [B]_T - 2k_b [B]_{i,t} + \left(\frac{2}{\Delta t} + k_b + k_f [Ca^{2+}]_{i,t} - k_f [B]_{i,t} \right) [Ca^{2+}]_{i,t}}{\frac{2}{\Delta t} + k_b + k_f [Ca^{2+}]_{i,t} + k_f [B]_{i,t}}$$

A.7. ecuación (3.23) a (3.24)

$$\frac{1}{\Delta t} (B)_{i,t+1} - (B)_{i,t} = k_b [B]_T - \frac{k_b}{2} ((B)_{i,t+1} + (B)_{i,t}) \\ - \frac{k_f}{2} ((B)_{i,t+1} [Ca^{2+}]_{i,t} + (B)_{i,t} [Ca^{2+}]_{i,t+1})$$

Se eliminan los paréntesis al realizar las multiplicaciones necesarias

$$\begin{aligned} \frac{1}{\Delta t}[B]_{i,t+1} - \frac{1}{\Delta t}[B]_{i,t} &= k_b[B]_T - \frac{k_b}{2}[B]_{i,t+1} - \frac{k_b}{2}[B]_{i,t} \\ &\quad - \frac{k_f}{2}[B]_{i,t+1}[Ca^{2+}]_{i,t} - \frac{k_f}{2}[B]_{i,t}[Ca^{2+}]_{i,t+1} \end{aligned}$$

Se agrupan los términos correspondientes a $t + 1$ del lado izquierdo y los correspondientes a t del lado derecho de la ecuación con respecto a $[B]$

$$\begin{aligned} \frac{1}{\Delta t}[B]_{i,t+1} + \frac{k_b}{2}[B]_{i,t+1} + \frac{k_f}{2}[B]_{i,t+1}[Ca^{2+}]_{i,t} \\ = k_b[B]_T - \frac{k_b}{2}[B]_{i,t} - \frac{k_f}{2}[B]_{i,t}[Ca^{2+}]_{i,t+1} + \frac{1}{\Delta t}[B]_{i,t} \end{aligned}$$

Se agrupan términos semejantes

$$\left(\frac{1}{\Delta t} + \frac{k_b}{2} + \frac{k_f}{2}[Ca^{2+}]_{i,t}\right)[B]_{i,t+1} = k_b[B]_T - \left(\frac{k_b}{2} + \frac{k_f}{2}[Ca^{2+}]_{i,t+1} - \frac{1}{\Delta t}\right)[B]_{i,t}$$

Se despeja para $[B]_{i,t+1}$ y se multiplica el lado derecho de la ecuación por $\frac{2}{2}$

$$[B]_{i,t+1} = \frac{2k_b[B]_T + \left(\frac{2}{\Delta t} - k_b - k_f[Ca^{2+}]_{i,t+1}\right)[B]_{i,t}}{\frac{2}{\Delta t} + k_b + k_f[Ca^{2+}]_{i,t}}$$

Apéndice B

Métodos numéricos para resolución de ecuaciones diferenciales

B.1. Método Crank-Nicolson (Crank y Nicolson, 1947)

Es un método de diferencias finitas usado para la resolución de ecuaciones en derivadas parciales, como puede ser el caso de la ecuación del calor o en general de una ecuación de difusión.

El método se basa en diferencias centrales en el espacio y en la regla del trapecio en el tiempo, dando como resultado un método con convergencia de segundo orden en tiempo. El método es implícito en el tiempo, explícito en el espacio y numéricamente estable.

Por ejemplo, para una dimensión, la ecuación en derivadas parciales definida en (B.1) y tomando $u(i\Delta x, n\Delta t) = u_i^n$ puede ser aproximada al aplicar el método de Crank-Nicolson como indica la ecuación (B.2).

$$\frac{\partial u}{\partial t} = F\left(u, x, t, \frac{\partial u}{\partial x}, \frac{\partial^2 u}{\partial x^2}\right) \quad (\text{B.1})$$

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{1}{2} \left[F_i^{n+1}\left(u, x, t, \frac{\partial u}{\partial x}, \frac{\partial^2 u}{\partial x^2}\right) + F_i^n\left(u, x, t, \frac{\partial u}{\partial x}, \frac{\partial^2 u}{\partial x^2}\right) \right] \quad (\text{B.2})$$

La función F debe ser discretizada espacialmente aplicando diferencias centradas.

En el caso del presente trabajo, se usa para resolver una difusión lineal en una dimensión, que de manera general tiene la forma:

$$\frac{\partial u}{\partial t} = a \frac{\partial^2 u}{\partial x^2}$$

cuya discretización mediante Crank-Nicolson es:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{a}{2\Delta^2 x} [(u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}) + (u_{i+1}^n - 2u_i^n + u_{i-1}^n)]$$

ahora, tomando $r = \frac{a\Delta t}{2\Delta^2 x}$ podemos escribirla como:

$$-ru_{i+1}^{n+1} + (1 + 2r)u_i^{n+1} - ru_{i-1}^{n+1} = ru_{i+1}^n + (1 - 2r)u_i^n + ru_{i-1}^n$$

que es un sistema con matriz tridiagonal.

Apéndice C

Código

En el presente Apéndice se encuentra el código en lenguaje Python donde se implementan todos los modelos y con los cuales se efectúan las simulaciones.

C.1. Requerimientos

```
1 import numpy as np
2 from scipy.integrate import ode
3 import matplotlib.pyplot as plt
4 from numpy.random import random
```

C.2. Canal

```
1 class Canal:
2     '''
3     Modela un Canal iónico de Calcio según se define en (Li y cols.,
4     2007)
5     INPUTS
6     tipo: string
7         indica el tipo de canal, puede ser "PQ", "N" o "R"
8     cantidad: int
9         indica la cantidad de canales de ese tipo, afectará la
10    corriente
11    escala: float
12        indica la resolución temporal partiendo desde ms que es el
13    valor por defecto
14    ejemplo: si escala = 1e3 el tiempo estaría dado en us porque
15    ms*1e3 = us
16    '''
17    #tipos de canal soportados
18    PQ = "PQ"
19    N = "N"
20    R = "R"
21
22    def __init__(self, tipo, cantidad, escala=1):
23        self.tipo = tipo
```

```

20     self.cantidad = cantidad
21     self.escala = escala
22
23     if self.tipo == self.PQ:
24         self.a = [247.71, 5.89, 9.21, 5.20, 1823.18]
25         self.b = [8.28, 14.99, 6.63, 132.80, 248.58]
26         self.k = [62.61, 33.92, 135.08, 20.86]
27         self.gCa = 2.2 #capacitancia pS
28         self.uCorriente = 0.13 #corriente unitaria pA
29     elif self.tipo == self.N:
30         self.a = [615.01, 4.29, 5.24, 4.98, 772.63]
31         self.b = [7.68, 5.23, 6.63, 73.89, 692.18]
32         self.k = [68.75, 39.53, 281.62, 18.46]
33         self.gCa = 2.2
34         self.uCorriente = 0.13
35     elif self.tipo == self.R:
36         self.a = [228.83, 9911.36, 4.88, 4.0, 256.41]
37         self.b = [1.78, 0.62, 21.91, 51.30, 116.97]
38         self.k = [67.75, 50.94, 173.29, 16.92]
39         self.gCa = 3.5
40         self.uCorriente = 0.21
41     else:
42         raise Exception("Tipo de canal no soportado")
43
44     #se aplica el cambio de escala temporal
45     self.a = np.array(self.a)*escala
46     self.b = np.array(self.b)*escala
47
48     self.ECa = 60 #potencial de reposo del Calcio
49     self.nv = 6 #cantidad de variables
50     self.y0 = [1.0, 0, 0, 0, 0, 0] #condiciones iniciales
51     self.t0 = 0 #tiempo inicial
52     self.r = ode(self._modelo).set_integrator('dopri5') #definición
del sistema de ecuaciones. el método de solución es Runge-Kutta de
orden 5
53
54     def __alfai(self, ia, v):
55         '''
56         Función de transición de avance del modelo de estados discreto de
markov del canal iónico
57         INPUTS
58             ia: int
59                 índice del alfa deseado, entre 0 y 4
60             v: float
61                 voltaje en mV
62         OUTPUT
63             alfa_{ia}: float
64                 el valor correspondiente de alfa
65         '''
66         if ia == 0:
67             alfa = self.a[ia]
68         else:
69             alfa = self.a[ia]*np.exp(v/self.k[ia-1])

```

```

70     return alfa
71
72 def __betai(self, ib, v):
73     '''
74     Función de transición de retroceso del modelo de estados discreto
de markov del canal iónico
75     INPUTS
76         ib: int
77             índice del beta deseado, entre 0 y 4
78         v: float
79             voltaje en mV
80     OUTPUT
81         beta_{ia}: float
82             el valor correspondiente de beta
83     '''
84     if ib == 0:
85         beta = self.b[ib]
86     else:
87         beta = self.b[ib]*np.exp(-v/self.k[ib-1])
88     return beta
89
90 def __modelo(self, t, y, v):
91     '''
92     Modelo de estados discretos de markov para el canal iónico de
Calcio.
93     Es un modelo de seis estados, cinco estados para el estado
cerrado del canal y un estado para el estado abierto del canal
94     INPUTS
95         t: float
96             tiempo en ms
97         y: array float
98             de longitud seis para las variables del sistema
99         v: float
100             voltaje en mV
101     OUTPUT
102         f: array float
103             definición de las ecuaciones de estados asociadas al
modelo
104     '''
105     beta = self.__betai(0,v)
106     beta1 = self.__betai(1,v)
107     beta2 = self.__betai(2,v)
108     beta3 = self.__betai(3,v)
109     beta4 = self.__betai(4,v)
110     alfa = self.__alfai(0,v)
111     alfa1 = self.__alfai(1,v)
112     alfa2 = self.__alfai(2,v)
113     alfa3 = self.__alfai(3,v)
114     alfa4 = self.__alfai(4,v)
115
116     C0 = y[0]
117     C1 = y[1]
118     C2 = y[2]

```

```

119     C3 = y[3]
120     C4 = y[4]
121     O = y[5]
122
123     f = np.zeros(6)
124     f[0] = beta1*C1 - alfa1*C0
125     f[1] = alfa1*C0 + beta2*C2 - (alfa2 + beta1)*C1
126     f[2] = alfa2*C1 + beta3*C3 - (alfa3 + beta2)*C2
127     f[3] = alfa3*C2 + beta4*C4 - (alfa4 + beta3)*C3
128     f[4] = alfa4*C3 + beta*O - (alfa + beta4)*C4
129     f[5] = alfa*C4 - beta*O
130
131     return f
132
133     def integrar(self, t, dt, pulsoInit, pulsoLong, v, vBase):
134         '''
135         Calcula las soluciones numéricas para el modelo bajo las
136         condiciones indicadas
137         INPUTS
138             t: float
139                 tiempo final en ms
140             dt: float
141                 paso de tiempo en ms
142             pulsoInit: float
143                 tiempo de inicio del pulso de voltaje en mV
144             pulsoLong: float
145                 duración del pulso de voltaje en mV
146             v: float
147                 intensidad del pulso de voltaje en mV
148             vBase: float
149                 potencial base(el existente antes del pulso de voltaje)
150         en mV
151         OUTPUT
152             NONE:
153             inicializa los atributos tiempo(t), probabilidad de
154             apertura(o) y corriente(i)
155         '''
156         #se aplica el cambio de escala temporal
157         t *= self.escala
158         dt *= self.escala
159         pulsoInit *= self.escala
160         pulsoLong *= self.escala
161
162         pulsoFinal = pulsoInit + pulsoLong
163         if t < self.t0:
164             raise ValueError("El tiempo final debe de ser mayor que el
165             tiempo inicial")
166         if pulsoInit < self.t0:
167             raise ValueError("El tiempo de inicio del pulso debe ser
168             mayor que el tiempo inicial")
169         if pulsoInit > t:
170             raise ValueError("El tiempo de inicio del pulso debe de ser
171             menor que el tiempo final")

```

```

166     if pulsoFinal > t:
167         raise ValueError("La duración del pulso no debe exceder el
tiempo final")
168
169     n = int(t/dt)
170     self.o = np.zeros(n+1) #probabilidad de apertura
171     self.t = np.zeros(n+1) #tiempo
172     self.i = np.zeros(n+1) #corriente
173     j = 0
174     self.r.set_initial_value(self.y0, self.t0) #se inicializan las
condiciones iniciales
175     while self.r.successful() and self.r.t < t:
176         if self.r.t < pulsoInit or self.r.t > pulsoFinal:
177             v_ = vBase
178         else:
179             v_ = v
180         self.r.set_f_params(v_) #se inicializan los parámetros
iniciales del modelo
181         self.r.integrate(self.r.t+dt) #se integra
182         self.t[j] = self.r.t
183         self.o[j] = self.r.y[5]
184         self.i[j] = self._getCorriente(self.o[j])
185         j+=1
186
187     def getTiempos(self):
188         '''
189         Retorna los tiempos de la simulación
190         INPUTS
191             NONE:
192         OUTPUT
193             t: array float
194         '''
195         return self.t
196
197     def getAperturas(self):
198         '''
199         Retorna la probabilidad de apertura como función del tiempo
200         INPUTS
201             NONE:
202         OUTPUT
203             o: array float
204         '''
205         return self.o
206
207     def getCorrientes(self):
208         '''
209         Retorna la corriente como función del tiempo
210         INPUTS
211             NONE:
212         OUTPUT
213             i: array float
214         '''
215         return self.i

```

```

216
217 def __getCorriente(self,o):
218     '''
219     Dada una probabilidad de apertura calcula la corriente
dependiendo de la cantidad de canales y la corriente unitaria.
220     El método es privado
221     INPUTS
222         o: float
223             probabilidad de apertura
224     OUIPUT
225         corr: float
226             corriente
227     '''
228     cantidad = 0
229     for i in range(0,self.cantidad):
230         rr = random()
231         if rr < o:
232             cantidad += 1
233     return cantidad*-self.uCorriente
234
235 def plotEstatico(self,t,dt,v,norm):
236     '''
237     Realiza el gráfico del voltaje vs intesidad de corriente para
varios voltajes
238     Se muestra el valor máximo de corriente no la dinámica del canal
239     INPUTS
240         t: float
241             tiempo final en ms
242         dt: float
243             paso de tiempo en ms
244         v: array float
245             intensidad de los voltajes en mV
246         norm: int
247             indice del voltaje para el cual se normaliza la corriente
248     OUIPUT
249         NONE:
250     '''
251     pulsoInit = 0
252     pulsoLong = t
253     vBase = 0
254     o_ = np.zeros(len(v))
255     j = 0
256     for v_ in v:
257         self.integrar(t,dt,pulsoInit,pulsoLong,v_,vBase)
258         o_[j] = np.max(self.getAperturas())
259         j+=1
260     i_ = o_*(np.array(v)-self.ECa) #intensidad de corriente
261     #gráfica V vs I
262     plt.figure(figsize=(8,6))
263     plt.plot(v,i_/i_[norm]*-1,'o')
264     plt.grid(linestyle='-.')
265     plt.xlabel("V (mV)")
266     plt.ylabel("I (norm)")

```

```

267     plt.show()
268     #probabilidad de apertura
269     plt.figure(figsize=(8,6))
270     plt.plot(v,o_,'o')
271     plt.xlabel("V (mV)")
272     plt.ylabel("P")
273     plt.grid(linestyle='-.')
274     plt.show()
275
276     def plotDinamico(self,t,dt,pulsoInit,pulsoLong,v,vBase):
277         '''
278         Realiza el gráfico de la probabilidad de apertura del canal
279         Permite ver la dinámica del canal
280         INPUTS
281             t: float
282                 tiempo final en ms
283             dt: float
284                 paso de tiempo en ms
285             pulsoInit: float
286                 tiempo de inicio del pulso de voltaje en mV
287             pulsoLong: float
288                 duración del pulso de voltaje en mV
289             v: float
290                 intensidad del pulso de voltaje en mV
291             vBase: float
292                 potencial base(el existente antes del pulso de voltaje)
293         en mV
294         OUTPUT
295         NONE:
296         '''
297         self.integrar(t,dt,pulsoInit,pulsoLong,v,vBase)
298         tt = self.getTiempos()
299         o = self.getAperturas()
300         end = len(tt)-1
301         tt = tt[0:end]
302         o = o[0:end]
303         plt.figure(figsize=(8,6))
304         plt.plot(tt/self.escala,o)
305         plt.xlabel("t (ms)")
306         plt.ylabel("P")
307         plt.grid(linestyle='-.')
308         plt.show()

```

```

1 class Canales:
2     '''
3     Modela un conjunto de canales iónicos de Calcio según se define en (
4     Li y cols., 2007)
5     INPUTS
6         canalCantidad: dict int
7             cantidad de canales por cada tipo de canal, de la forma {PQ:
8             int, N:int, R:int}
9             si no se especifica el canal el valor por defecto es 0
10        escala: float

```

```

9         indica la resolución temporal partiendo desde ms que es el
10        valor por defecto
11        ejemplo: si escala = 1e3 el tiempo estaría dado en us porque
12        ms*1e3 = us
13        '''
14    def __init__(self, canalCantidad, escala=1):
15        self.PQ = canalCantidad.get(Canal.PQ, 0)
16        self.N = canalCantidad.get(Canal.N, 0)
17        self.R = canalCantidad.get(Canal.R, 0)
18        self.escala = escala
19        #validar que los canales sean todos cero
20        self.canalPQ = Canal(Canal.PQ, self.PQ, escala)
21        self.canalN = Canal(Canal.N, self.N, escala)
22        self.canalR = Canal(Canal.R, self.R, escala)
23
24    def integrar(self, t, dt, pulsoInit, pulsoLong, v, vBase):
25        '''
26        Calcula las soluciones numéricas para cada tipo de canal bajo las
27        condiciones indicadas
28        INPUTS
29            t: float
30                tiempo final en ms
31            dt: float
32                paso de tiempo en ms
33            pulsoInit: float
34                tiempo de inicio del pulso de voltaje en mV
35            pulsoLong: float
36                duración del pulso de voltaje en mV
37            v: float
38                intensidad del pulso de voltaje en mV
39            vBase: float
40                potencial base(el existente antes del pulso de voltaje)
41        en mV
42        OUTPUT
43            NONE:
44                inicializa los atributo corrientes, que es la suma de las
45                corrientes proporcionada por cada tipo de canal
46        '''
47        n = int(t/dt)
48        self.corrientes = np.zeros(n+1)
49        if self.PQ != 0:
50            self.canalPQ.integrar(t, dt, pulsoInit, pulsoLong, v, vBase)
51            self.corrientes += self.canalPQ.getCorrientes()
52        if self.N != 0:
53            self.canalN.integrar(t, dt, pulsoInit, pulsoLong, v, vBase)
54            self.corrientes += self.canalN.getCorrientes()
55        if self.R != 0:
56            self.canalR.integrar(t, dt, pulsoInit, pulsoLong, v, vBase)
57            self.corrientes += self.canalR.getCorrientes()
58
59    def getCorriente(self, index):
60        '''
61        Retorna la corriente para un índice dado

```

```

57     INPUTS
58         index: int
59         indice del array de corrientes
60     OUPUT
61         i: float
62         corriente
63     '''
64     return self.corrientes[index]
65
66 def getCorrientes(self):
67     '''
68     Retorna la corriente como función del tiempo
69     INPUTS
70         NONE:
71     OUPUT
72         i: array float
73         corrientes
74     '''
75     return self.corrientes
76
77 def getTiempos(self):
78     '''
79     Retorna los tiempos de la simulación. Son tomadas de alguno de
80     los canales existentes, todos tienen el mismo tiempo.
81     INPUTS
82         NONE:
83     OUTPUT
84         t: array float
85         tiempos
86     '''
87     if self.PQ !=0:
88         return self.canalPQ.getTiempos()
89     elif self.N !=0:
90         return self.canalN.getTiempos()
91     else:
92         return self.canalR.getTiempos()
93
94 def plotCorriente(self, s=1):
95     '''
96     Realiza un gráfico de la corriente que generaron los canales
97     INPUTS
98         s: int
99         indica el sentido de la corriente, puede ser 1 o -1. Por
100        defecto el valor es 1 que indica una corriente negativa.
101    '''
102    tt = self.getTiempos()
103    plt.figure(figsize=(8,6))
104    plt.plot(tt/self.escala, self.corrientes*s, color='black')
105    plt.ylabel("I (pA)")
106    plt.xlabel("t (ms)")
107    plt.show()

```

C.3. Secreción

```

1 class Secrecion:
2     '''
3     Modela eventos de secreción según se define en (Bollmann y cols.,
4     2007)
5     INPUTS
6         vesiculas: int
7             cantidad de vesiculas que se tendrán en la RRP(Readily
8             Releasable Pool)
9     '''
10
11     OP_SENSOR = "sensor"
12     OP_EXOCITOSIS = "exocitosis"
13
14     def __init__(self, vesiculas, escala=1):
15         self.vesiculas = vesiculas
16         self.escala = escala
17         self.alpha = 0.3*self.escala #uM^-1 ms^-1
18         self.beta = 3*self.escala #ms^-1
19         self.gamma = 30*self.escala #ms^-1
20         self.delta = 8*self.escala #ms^-1
21         self.pho = 40*self.escala #ms^-1
22         self.t0 = 0 #tiempo inicial
23         self.y0Sensor = [self.vesiculas*5, 0, 0, 0, 0, 0, 0] #cantidad de
24         sensores de Ca(cada vesicula tiene 5 sensores)
25         self.y0Exocitosis = [self.vesiculas, 0]
26         self.rSensor = ode(self.__modeloSensor).set_integrator('dopri5')
27         #definición del sistema de ecuaciones. el método de solución es Runge-
28         Kutta de orden 5
29         self.rExocitosis = ode(self.__modeloExocitosis).set_integrator('
30         dopri5')
31
32     def __modeloSensor(self, t, y, ca):
33         '''
34         Modelo de estados discretos para los sensores de calcio de las
35         vesículas sinápticas.
36         Es un modelo de siete estados. Los primeros seis para los
37         sensores con 0,1,2,3,4 y 5 Ca ligado y un último estado para un
38         paso de isomerización independiente de calcio
39         El método es privado
40         INPUTS
41             t: float
42                 tiempo en ms
43             y: array float
44                 de longitud siete para las variables del sistema
45             ca: concentración de calcio en uM para el tiempo t
46         OUTPUT
47             f: array float
48                 definición de las ecuaciones de estados asociadas al
49         modelo
50         '''
51         X0 = y[0]
52         X1 = y[1]

```

```

43     X2 = y[2]
44     X3 = y[3]
45     X4 = y[4]
46     X5 = y[5]
47     C = y[6]
48
49     f = np.zeros(7)
50     f[0] = self.beta*X1 - 5*self.alpha*ca*X0
51     f[1] = 5*self.alpha*ca*X0 + 2*self.beta*X2 - (self.beta + 4*self.alpha*ca)*X1
52     f[2] = 4*self.alpha*ca*X1 + 3*self.beta*X3 - (2*self.beta + 3*self.alpha*ca)*X2
53     f[3] = 3*self.alpha*ca*X2 + 4*self.beta*X4 - (3*self.beta + 2*self.alpha*ca)*X3
54     f[4] = 2*self.alpha*ca*X3 + 5*self.beta*X5 - (4*self.beta + self.alpha*ca)*X4
55     f[5] = self.alpha*ca*X4 + self.delta*C - (5*self.beta + self.gamma)*X5
56     f[6] = self.gamma*X5 - self.delta*C
57
58     return f
59
60     def __modeloExocitosis(self, t, y, x):
61         '''
62         Modelo de estados discretos para la exocitosis o liberación de
63         las vesículas sinápticas.
64         Es un modelo de dos estados, uno para las vesículas listas para
65         ser liberadas o RRP y otro para las vesículas que se
66         han liberado, esto es, que se han fusionado con la membrana
67         El método es privado
68         INPUTS
69         t: float
70             tiempo en ms
71         y: array float
72             de longitud siete para las variables del sistema
73         x: fracción de los sensores de Ca que promueven el estado de
74         liberación(salida del modelo de sensor)
75         OUTPUT
76         f: array float
77             definición de las ecuaciones de estados asociadas al
78         modelo
79         '''
80         f = np.zeros(2)
81         f[0] = -self.pho*x*y[0]
82         f[1] = self.pho*x*y[0]
83         return f
84
85     def __integrar(self, t1, dt, z, tipo):
86         '''
87         Calcula las soluciones numéricas para el modelo bajo las
88         condiciones indicadas
89         El método es privado
90         INPUTS

```

```

86         t1: float
87             tiempo final en ms
88         dt: float
89             paso de tiempo en ms
90         z: float array
91             parámetros del modelo para cada tiempo t.
92             Si tipo es OP_SENSOR z será la concentración de calcio
93             Si tipo es OP_EXOCITOSIS z será la fracción de sensores
de Ca
94     OUTPUT
95         (t,e): (float array,float array)
96             tiempo y soluciones numéricas encontradas
97     '''
98     if tipo == self.OP_SENSOR:
99         r = self.rSensor
100        y0 = self.y0Sensor
101        salida = 6
102    elif tipo == self.OP_EXOCITOSIS:
103        r = self.rExocitosis
104        y0 = self.y0Exocitosis
105        salida = 1
106    n = int(t1/dt)
107    t = np.zeros(n+1) #tiempo
108    e = np.zeros(n+1) #sensor o exocitosis
109    r.set_initial_value(y0, self.t0) #se inicializan las condiciones
iniciales
110    j = 0
111    while r.successful() and r.t < t1:
112        try:
113            x = z[j]
114        except:
115            x = 0
116        r.set_f_params(x) #se inicializan los parámetros iniciales
del modelo
117        r.integrate(r.t+dt) #se integra
118        t[j] = r.t
119        e[j] = r.y[salida]
120        j+=1
121    return (t,e)
122
123    def integrar(self,t,dt,calcio,escala=1):
124        '''
125        Calcula las soluciones numéricas para el modelo bajo las
condiciones indicadas
126        INPUTS
127            t: float
128                tiempo final en ms
129            dt: float
130                paso de tiempo en ms
131            calcio: float array
132                concentración de Ca en uM para cada tiempo t
133            escala: float

```

```

134         indica la resolución temporal partiendo desde ms que es
el valor por defecto
135     OUTPUT
136         NONE:
137         inicializa los atributos tiempo, sensor y exocitosis
138     '''
139     t *= self.escala
140     dt *= self.escala
141     [W, self.sensor] = self._integrar(t, dt, calcio, self.OP_SENSOR)
142     [self.t, self.exocitosis] = self._integrar(t, dt, self.sensor, self.
OP_EXOCITOSIS)
143
144     def getTiempos(self):
145     '''
146         Retorna los tiempos de la simulación.
147     INPUTS
148         NONE:
149     OUTPUT
150         t: array float
151         tiempos
152     '''
153     return self.t
154
155     def getSensores(self):
156     '''
157         Retorna los resultados obtenidos para el modelo de sensores de
calcio
158     INPUTS
159         NONE:
160     OUTPUT
161         sensor: array float
162         resultados de la simulación
163     '''
164     return self.sensor
165
166     def getExocitosis(self):
167     '''
168         Retorna los resultados obtenidos para el modelo de fusión
vesicular
169     INPUTS
170         NONE:
171     OUTPUT
172         exocitosis: array float
173         resultados de la simulación
174     '''
175     return self.exocitosis
176
177     def plot(self, tipo):
178     '''
179         Realiza un gráfico de los resultados obtenidos para las
simulaciones
180     INPUTS
181         tipo: string

```

```

182         indica el tipo de grafica a realizar.
183         Secrecion.OP_SENSOR para graficar el resultado de la
simulación de sensores de calcio
184         Secrecion.OP_EXOCITOSIS para graficar el resultado de la
simulación de fusión vesicular
185     OUTPUT
186     NONE:
187     '''
188     if self.vesiculas != 0:
189         tt = self.getTiempos()
190         if tipo == self.OP_SENSOR:
191             y = self.getSensores()
192             ylabel = "Sensores"
193         elif tipo == self.OP_EXOCITOSIS:
194             y = self.getExocitosis()
195             ylabel = "Vesículas"
196         end = len(tt)-1
197         tt = tt[0:end]
198         y = y[0:end]
199         plt.figure(figsize=(8,6))
200         plt.plot(tt/self.escala,y)
201         plt.xlabel("t (ms)")
202         plt.ylabel(ylabel)
203         #plt.grid(linestyle='-.')
204         plt.show()
205     else:
206         print("No hay datos para graficar")

```

C.4. Célula

```

1 class Celula:
2     '''
3     Modela una célula esférica
4     INPUTS
5         canalCantidad: dict int
6             cantidad de canales que tendrá la célula por cada tipo de
canal, de la forma {PQ:int, N:int, R:int}
7             si no se especifica el canal el valor por defecto es 0
8         radioIn: float
9             radio interno de la célula en um(micrometros)
10        radioOut: float
11            radio externo de la célula en um(micrometros)
12        Kf: float
13            constante de asociación(binding o forward) del buffer en 1/(M
*seg) (1/molar*seg)
14        Kd: float
15            constante de disociación(equilibrio) del buffer uM(micromolar
)
16        Btotal: float
17            concentración inicial total del buffer en uM(micromolar)
18        CaBasal: float
19            concentración inicial de calcio en uM(micromolar)
20        DCa: float

```

```

21         constante de difusión del calcio
22     '''
23     def __init__(self, canalCantidad, radioIn, radioOut, Kf, Kd, Btotal, CaBasal
24     ,DCa, vesiculas, escala=1):
25         #Constantes
26         self.F = 0.096485 #constante de Faraday en C/uM
27         self.DCa = DCa #coeficiente de difusión del Ca en um^2/s
28
29         self.radioCelOut = radioOut #um
30         self.radioCelIn = radioIn #um
31         self.Kf = Kf*1e6 #1/(M*seg) a 1/(uM*seg)
32         self.Kd = Kd #uM
33         self.Kb = self.Kd*self.Kf #1/seg
34         self.Btotal = Btotal #uM
35         self.CaBasal = CaBasal #uM
36         self.vesiculas = vesiculas
37
38         #Se calcula la cantidad de buffer inicial que está en equilibrio
39         con el Ca aplicando la ley de acción de masas
40         self.Binicial = (self.Btotal * self.Kb)/((self.CaBasal * self.Kf)
41         + self.Kb)
42
43         self.canales = Canales(canalCantidad, escala)
44         self.secrecion = Secrecion(self.vesiculas, escala)
45
46     def __initVariables(self):
47         '''
48         Inicializa las variables necesarias para la simulación y
49         dependientes del valor de dt y dr
50         El método es privado
51         INPUTS
52         NONE:
53         OUTPUT
54         NONE:
55         variables inicializadas
56     '''
57     #cantidad de capas de la célula
58     self.nCapas = int(np.floor((self.radioCelOut-self.radioCelIn)/
59     self.dr))
60     self.nCapas1 = self.nCapas+1
61     #variables
62     self.Cai = np.ones(self.nCapas1) #concentración de Ca en cada
63     capa
64     self.l = np.zeros(self.nCapas1) #elementos l de la matriz A
65     self.d = np.zeros(self.nCapas1) #elementos d de la matriz A y
66     posteriormente de la matriz L
67     self.u = np.zeros(self.nCapas) #elementos u de la matriz A
68     self.v = np.zeros(self.nCapas) #elementos v de la matriz U
69     self.D = np.zeros(3) #elementos D de la matriz B
70     self.Bi = np.ones(self.nCapas1) #concentración de buffer en cada
71     capa
72     self.Cait = np.zeros((self.nCapas1, self.pasos))
73     self.Bit = np.zeros((self.nCapas1, self.pasos))

```

```

66     #se inicializan los radios de cada una de las capas
67     self.radioi = np.zeros(self.nCapas1)
68     n = np.arange(0, self.nCapas1)
69     self.radioi = self.radioi + self.radioCelIn
70     self.radioi = self.radioi + n*self.dr
71     #se calcula el volumen de la última capa
72     self.Vn = (4/3)*np.pi*((self.radioCelOut*1e-6)**3-((self.
radioCelOut-self.dr)*1e-6)**3)*1000 # ((um^3)ul) litros REVISAR
73     #se inicializa el calcio y el buffer
74     self.Cai = self.Cai*self.CaBasal
75     self.Bi = self.Bi*self.Binicial
76     #se inicializan los elementos de las matrices A y B
77     self.R = (self.DCa*self.dt)/(2*self.dr**2)
78     self.Y = 3*self.R * self.radioi[1] / self.radioi[0]
79     self.u[0] = -self.Y
80     self.u[1:self.nCapas] = -(self.radioi[2:self.nCapas1]*self.R)/
self.radioi[1:self.nCapas]
81     self.l[1:self.nCapas1] = -(self.radioi[0:self.nCapas]*self.R)/
self.radioi[1:self.nCapas]
82     self.D[0] = 1 - self.Y
83     self.D[1] = 1 - 2*self.R
84     self.D[2] = 1 + self.l[self.nCapas]
85     #se factoriza la matriz A = LU
86     self.d[0] = 1 + self.Y
87     self.v[0] = self.u[0]/self.d[0]
88     for i in range(1, self.nCapas):
89         self.d[i] = (1 + 2*self.R) - self.l[i]*self.v[i-1]
90         self.v[i] = self.u[i]/self.d[i]
91     self.d[self.nCapas] = (1-self.l[self.nCapas]) - (self.l[self.
nCapas]*self.v[self.nCapas-1])
92
93     def difusion(self):
94         '''
95         Calcula la difusión del calcio al resolver el sistema AW(t+dt) =
BW(t) usando
96         reducción de Crout, de tal manera que Lz = BW(t) y UW(t+dt) = z
97         Posteriormente calcula la concentración de calcio y buffer que se
encuentra en equilibrio
98         INPUTS
99             NONE:
100        OUTPUT
101            NONE:
102            valores de calcio y buffer actualizado
103        '''
104        #sustitución hacia adelante
105        z = np.zeros(self.nCapas1)
106        z[0] = (self.D[0]*self.Cai[0] - self.u[0]*self.Cai[1])/self.d[0]
107        for i in range(1, self.nCapas):
108            z[i] = (-self.l[i]*self.Cai[i-1] + self.D[1]*self.Cai[i] -
self.u[i]*self.Cai[i+1] -self.l[i]*z[i-1])/self.d[i]
109        z[self.nCapas] = (-self.l[self.nCapas]*self.Cai[self.nCapas-1] +
self.D[2]*self.Cai[self.nCapas] - self.l[self.nCapas]*z[self.nCapas
-1])/self.d[self.nCapas]

```

```

110     #sustitución hacia atrás
111     self.Cai[self.nCapas] = z[self.nCapas]
112     for i in range(self.nCapas-1,-1,-1):
113         self.Cai[i] = z[i] - self.v[i]*self.Cai[i+1]
114     #calculo de calcio y buffer en equilibrio
115     a = 2*self.Kb*self.Btotal
116     b = 2/self.dt + self.Kb + self.Kf*self.Cai
117     c = self.Kf*self.Bi
118     if self.buffer:
119         self.Cai = (a - 2*self.Kb*self.Bi + (b-c)*self.Cai)/(b+c)
120         self.Bi = (a + (2/self.dt - self.Kb - self.Kf*self.Cai)*self.
Bi)/b
121
122     def simular(self ,dr ,t ,dt ,pulsoInit ,pulsoLong ,pulsoVoltaje ,buffer ,v=0,
vBase=0,corriente=0):
123         '''
124         Realiza una simulación del proceso de difusión.
125         La simulación se puede realizar al aplicar un pulso de voltaje o
un pulso de corriente
126         INPUTS
127             dr: float
128                 paso espacial(radius) en um
129             t: float
130                 tiempo de la simulación en ms
131             dt: float
132                 paso de tiempo en ms
133             pulsoInit: float
134                 tiempo de inicio del pulso de voltaje o corriente
135             pulsoLong: float
136                 tiempo de duración del pulso de voltaje o corriente
137             pulsoVoltaje: bool
138                 True si el pulso es de voltaje , de los contrario(pulso de
corriente) False
139             buffer: bool
140                 True si se desea que se use el buffer en la simulación
141             v: float
142                 intensidad del pulso de voltaje en mV
143             vBase: float
144                 potencial base(el existente antes del pulso de voltaje)
en mV
145             corriente: function
146                 función que recibe como argumento el tiempo y retorna la
intensidad de la corriente en pA para ese tiempo
147         OUTPUT
148             NONE:
149                 calcula el valor para los atributos Cai y Bi
150         '''
151         self.dr = dr #um
152         self.t = t*1e-3 #ms a s
153         self.dt = dt*1e-3 #ms a s
154         self.pasos = int(self.t/self.dt)
155         self.__initVariables()
156         self.buffer = buffer

```

```

157
158     #si es un pulso de voltaje se calcula la corriente que entraría
por los canales
159     if pulsoVoltaje:
160         self.canales.integrar(t,dt,pulsoInit,pulsoLong,v,vBase)
161         #I = self.canal.getCorrientes()*1e-12 #pA a A
162
163     paso = 0
164     while paso < self.pasos:
165         if not pulsoVoltaje:
166             if paso*dt < pulsoInit or paso*dt > (pulsoInit+pulsoLong)
:
167                 I = 0
168             else:
169                 I = corriente(paso)*-1e-12
170             else: #pulso de voltaje
171                 I = self.canales.getCorriente(paso)*1e-12 #pA a A
172                 #aumenta el calcio en la última capa
173                 dca=-I*self.dt/(2*self.F*self.Vn) #uM
174                 self.Cai[self.nCapas] = self.Cai[self.nCapas]+dca
175                 #difunde el calcio y se actualiza la concentración de calcio
y buffer
176                 self.difusion()
177                 #se guarda la concentración de calcio y buffer para el paso
actual
178                 self.Cait[:,paso]= self.Cai
179                 self.Bit[:,paso] = self.Bi
180                 paso+=1
181
182     #se calcula la cantidad de vesículas que se fusionan con la
membrana
183     if self.vesiculas != 0:
184         caVesiculas = self.Cait[self.nCapas-1,:]
185         self.secrecion.integrar(t,dt,caVesiculas)
186
187     def plotConcentracion(self, especie, capas=[], suavizar=False, **kwargs):
188         '''
189         Realiza la gráfica de la concentración de calcio y la concentraci
ón de buffer
190         INPUTS
191             especie: string
192                 "Ca" para graficar la concentración de calcio
193                 "B" para graficar la concentración de buffer
194             capas: array float
195                 distancia de las capas que se quieren dibujar por debajo
de la membrana en um.
196                 Deben ser multiples del paso espacial dr
197                 Si capas no se especifica indica que se quiere dibujar
todas las capas
198                 ejemplo: si dr=0.1um capas=[0.1,0.5,1.0] indica las capas
a 0.1um, 0.5um y 1um debajo de la ultima capa(membrana)
199             suavizar: bool

```

```

200         Si True se realiza un suavizado de los datos al
re calcular cada punto como un promedio de los datos circundantes.
201         Por defecto False, no se realiza ningún suavizado
202         kwargs: dic
203             permite pasar argumentos adicionales arbitrarios. En
particular la ubicación del legend de la gráfica
204     OUTPUT
205     NONE:
206     '''
207     espCa = "Ca"
208     espB = "B"
209     ylabel = "$_i$ (uM)"
210     if especie == espCa:
211         esp = self.Cait
212         ylabel = "[Ca]" + ylabel
213     elif especie == espB:
214         esp = self.Bit
215         ylabel = "[B]" + ylabel
216     else:
217         raise Exception("La especie que indica para graficar no
existe")
218
219     if suavizar:
220         window = 10
221         espCapas = esp.shape[0]
222         espElem = esp.shape[1]
223         temp = np.zeros((espCapas, espElem-window))
224         for capa in range(0, espCapas):
225             for ele in range(0, espElem-window):
226                 temp[capa, ele] = np.mean(esp[capa, ele:ele+window])
227         esp = temp
228
229     tt = np.arange(0, esp.shape[1]*self.dt, self.dt)*1e3
230     legends = []
231     plt.figure(figsize=(10,8))
232     if not capas:
233         for i in range(ce.nCapas, -1, -1):
234             plt.plot(tt, esp[i, :])
235     else:
236         for d in capas:
237             #if d%ce.dr != 0:
238             #    raise Exception("Las distancias de las capas deben
de ser multiples del paso espacial dr")
239             #validar que no se supere la profundida de la célula o el
radio interno
240             i = int(ce.nCapas - d/ce.dr)
241             plt.plot(tt, esp[i, :])
242             legends.append(str(d)+" um")
243     if legends:
244         plt.legend(legends, loc=kwargs.get('loc', 'upper left'))
245     plt.xlabel("t (ms)")
246     plt.ylabel(ylabel)
247     plt.show()

```

```
248
249 def plotCorriente(self):
250     '''
251     Realiza un gráfico de la corriente que proporcionó los canales de
252     calcio. Usa el método de la clase Canales
253     INPUTS
254     NONE:
255     OUTPUT
256     NONE:
257     '''
258     self.canales.plotCorriente(s=-1)
259
260 def plotSecrecion(self):
261     '''
262     Realiza un gráfico de la cantidad de vesículas excitadas. Usa el
263     método de la clase Secrecion
264     INPUTS
265     NONE:
266     OUTPUT
267     NONE:
268     '''
269     self.secrecion.plot(Secrecion.OP_EXOCITOSIS)
```