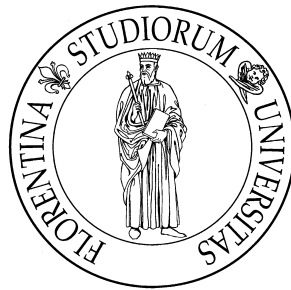


UNIVERSITÀ DEGLI STUDI DI FIRENZE
Facoltà di Scienze Matematiche, Fisiche e Naturali
Corso di Laurea Specialistica in Scienze e Tecnologie
dell'Informazione



Tesi di Laurea

DESCRIZIONE ED ANALISI DI SISTEMI
BIOLOGICI IN KLAIM

DARIO ANDREI

Relatore: *Michele Loreti*

Anno Accademico 2010-2011

Dario Andrei: *Descrizione ed analisi di sistemi biologici in Klaim*, Corso di
Laurea Specialistica in Scienze e Tecnologie dell'Informazione, © Anno
Accademico 2010-2011

INDICE

1	Introduzione	1
1.1	Un modello per il recettore CB2 sito nelle microglia	4
1.2	Un modello per le cellule elettricamente eccitabili	7
1.3	Panoramica della Tesi	9
1.4	Lavori connessi	10
2	Concetti biologici	11
2.1	Membrana cellulare	11
2.2	Recettori	13
2.2.1	Legame e attivazione	14
2.3	Il sistema endocannabinoide	16
2.3.1	I recettori cannabinoidi	18
2.3.2	Il recettore CB2	19
2.3.3	Ruolo di CB2 nell'attivazione della microglia	21
2.4	Canali ionici voltaggio dipendenti	22
2.5	Pompe ioniche	24
2.5.1	La pompa Sodio/Potassio	24
2.6	Le cellule eccitabili	25
2.6.1	Potenziale di membrana	26
2.6.2	Potenziale di equilibrio	29
2.6.3	Potenziale di riposo	30
2.7	Potenziale d'azione	31
2.7.1	Circuito equivalente	34
3	BioKLAIM, un linguaggio per modelli biologici	35
3.1	KLAIM	36

3.2	Concetti di teoria della probabilità	38
3.2.1	Distribuzione esponenziale	39
3.2.2	Catene di Markov a tempo continuo	40
3.2.3	Rate Transition System	41
3.3	BIOKLAIM	43
3.3.1	Categorie sintattiche	44
3.3.2	Reti e processi	44
3.3.3	Tuple e templates	45
3.3.4	Azioni	46
3.3.5	Specifiche	47
3.4	Semantica operativa di BIOKLAIM	48
3.5	Specificare proprietà per il global computing	49
3.5.1	Sintassi	51
3.5.2	Semantica	55
3.6	L'ambiente bksim	57
3.6.1	Dichiarazione di una specifica e sua esecuzione	58
3.6.2	Simulazione	61
3.6.3	Verifica proprietà	62
3.7	JSAM	63
4	Due casi di studio: microglia e cellule eccitabili	71
4.1	Microglia	72
4.1.1	Il modello	73
4.1.2	Risultati	81
4.2	Cellule eccitabili	87
4.2.1	Il modello	87
4.2.2	Risultati	95
5	Brevi approfondimenti conclusivi	103
5.1	Cosa è la Biologia Computazionale	103

5.1.1	Tecniche	105
5.2	Metodologia utilizzata	106
5.3	Descrivere sistemi con BioKLAIM	108
	Bibliografia	108

INTRODUZIONE

Negli ultimi trent'anni i computer hanno subito un significativo cambiamento: da sistemi indipendenti che eseguivano programmi autonomamente, si sono evoluti in una grande rete di calcolo dove ogni singolo dispositivo esegue un compito in modo cooperativo e coordinato. Caratteristiche che in precedenza erano assenti, oppure non venivano prese in considerazione, adesso giocano un ruolo prominente nel calcolo distribuito.

Esistono attività specializzate che mirano a essere da supporto alla ricerca e possono trarre beneficio da questa evoluzione. Un esempio di queste attività è la Biologia Computazionale, campo interdisciplinare che applica le tecniche dell'Informatica, Matematica e Statistica per destinarle a problemi di natura biologica.

Il principale scopo della Biologia Computazionale è la descrizione, in termini di algebre di calcolo e sistemi di simulazione, delle trasformazioni delle componenti delle cellule e di come queste interagiscano tra loro regolando la chimica della vita: un ponte fra la Biologia e la Chimica. Tale disciplina permette ai ricercatori di avere strumenti che, ad esempio, garantiscano l'efficacia o meno di un farmaco prima di passare alla sperimentazione sull'uomo, oppure rintraccino disfunzioni che diano la possibilità di individuare, preventivamente, future complicazioni.

Prima dell'introduzione di questo approccio, esisteva, e vengono tutt'ora impiegati, solo due metodi per effettuare esperimenti biologici:

all'interno di un organismo vivente (detti esperimenti *in vivo*) o in un ambiente artificiale (chiamati *in vitro*). È possibile affermare che la Biologia Computazionale è in realtà una Biologia *in silicio*, poiché per compiere l'esperimento vengono impiegati microprocessori (i quali sono costituiti da silicio).

Lo studio tradizionale dei sistemi biologici richiede metodi di riduzione¹, nei quali è prassi raccogliere dati per categorie. Un esempio di dati raccolti può riguardare le concentrazioni di sostanze chimiche nel tempo, in risposta ad un determinato stimolo. I computer sono di primaria importanza per realizzare e analizzare modelli di investigazione legati a ciò che è scaturito dall'esame dei dati in possesso. L'obiettivo è quello di creare modelli accurati di risposta agli stimoli interni e ambientali, al fine di comprendere come questi incidano sul sistema in analisi: si parla così di sistemi reattivi.

Un sistema reattivo sviluppa reazioni agli stimoli e produce stimoli per attivare altre reazioni: normalmente è costituito dalla composizione di sottosistemi, ognuno di essi reattivo. Tali sottosistemi formano l'ambiente del sistema.

Tra i sistemi reattivi ci sono anche quei sistemi naturali che si autoregolano: tra questi è possibile includere il sistema di regolamentazione climatica globale, tutti gli organismi viventi, molti processi cognitivi e anche molte istituzioni sociali umane. Riuscire a costruire un sistema artificiale che sia in grado di autoregolarsi è un traguardo pregevole. Non esiste ancora una ben identificabile metodologia scientifica che cerchi come sia possibile costruire classi di sistemi che siano in grado di autoregolarsi o che cerchi di scoprire condizioni che ne facilitino l'identificazione.

¹ Un approccio a comprendere un sistema complesso riducendolo a interazioni di sue parti, dove i risultati delle parti contribuiscono al risultato del sistema che le comprende

Data la forte presenza di sistemi con autoregolazione, e ponendo come elemento di spicco la loro possibile auto-manutenzione², qualsiasi approccio credibile, atto a stabilire basi per lo studio di tali sistemi, merita di essere preso in considerazione.

Sono stati sviluppati e implementati diversi framework per trattare il problema. Questi sono tutti basati sulla premessa che le entità costituenti un sistema godano dell'auto-mantenimento e abbiano come caratteristica quella di dedicarsi a interazioni il cui risultato è la costruzione di altri soggetti della stessa classe. L'auto-mantenimento è quindi la conseguenza di un *feedback* ciclico costruttivo[1, 2, 3]. Le relazioni funzionali specifiche tra entità, che collettivamente garantiscono la loro continua rigenerazione, vengono chiamate organizzazioni. L'obiettivo a lungo termine di questi studi è di permettere lo sviluppo formale e chiaro di un'organizzazione con auto-mantenimento.

Due delle domande a cui la Biologia Computazionale tenta di rispondere sono: la diversità biologica presente potrebbe essere sempre la stessa se "il nastro fosse letto una seconda volta"?[2] Se fosse possibile osservare l'evoluzione sulla terra, cosa potrebbe essere notato? Ad esempio, alcune delle risposte più plausibili potrebbero essere l'evoluzione dell'*Homo sapiens*, o l'evoluzione di qualcosa che sia identificabile con un animale oppure qualcosa di simile ad un eucariota[4]. La domanda si rivela importante se viene posta attenzione all'evoluzione della vita, vista come il prodotto di contingenza e necessità, che viene riassunta nella proverbiale *legge del più forte*.

La difficoltà fondamentale nell'analisi dei fenomeni è data dal fatto che questi possano essere accaduti nel passato (nel caso di studi evolutivisti), oppure si manifestino nell'infinitamente piccolo, oppure siano

² Proprietà di regolare la produzione degli elementi necessari al sistema per conservarsi.

espressi da eventi dalla durata minima. Gli esperimenti possono solo offrire una visione di cosa accade o è accaduto. L'unica alternativa è la costruzione di un modello con il quale sia possibile tale indagine, e che permetta di dimostrare se l'emergere di un risultato sia dovuto a particolari accadimenti. Sarebbe inoltre possibile interrogarsi sulla validità di tale investigazione.

Trovare una teoria per le organizzazioni è trovare una teoria per i sistemi che godono di auto-mantenimento. Un esempio è l'atmosfera oppure il sole a livello nucleare. Un altro esempio può essere rappresentato dalle molecole che si "muovono" nel nostro corpo. Gli organismi sono una particolare classe di sistemi con auto-manutenzione, proprietà che deriva dalle loro iterazioni costruttive.

1.1 UN MODELLO PER IL RECETTORE CB2 SITO NELLE MICROGLIA

Le cellule della microglia sono un tipo di cellule della glia³ che si occupano della prima e principale difesa immunitaria attiva nel sistema nervoso centrale.

La microglia, mostrata in figura 1, costituisce il 20% della popolazione totale di cellule gliali all'interno del cervello[6]. Le cellule microglia (e gli astrociti⁴) sono distribuite in grandi regioni non sovrapposte in tutta la

3 Sono cellule che, assieme ai neuroni, costituiscono il sistema nervoso. Hanno funzione nutritiva e di sostegno per i neuroni, assicurano l'isolamento dei tessuti nervosi e la protezione da corpi estranei in caso di lesioni. La riproduzione delle cellule della glia avviene molto frequentemente per mitosi, contrariamente ai neuroni per i quali il fenomeno si presenta raramente.

4 Sono cellule della glia con una forma simile ad una stella. Hanno molte funzioni tra le quali il supporto biochimico alle cellule endoteliali (cellule che formano la barriera tra sangue e cervello), provvedendo al nutrimento del tessuto nervoso, ed inoltre ricoprono

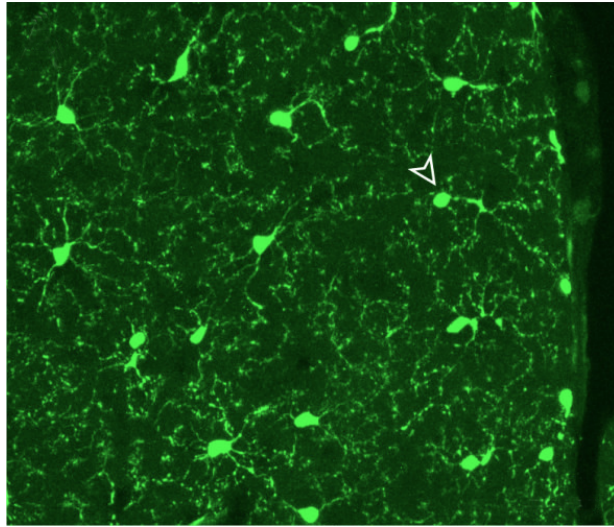


Figura 1: Immagine al microscopio del tessuto della glia. Le regioni in verde più chiaro sono le cellule della microglia. Immagine proveniente da [5].

colonna vertebrale e il cervello. Questo tipo di cellula scava costantemente il sistema centrale nervoso alla ricerca di neuroni danneggiati, placche e agenti infettivi.

Il cervello e la spina dorsale, data la loro palese importanza per il funzionamento di tutte le attività umane, sono considerate, in un certo senso, immuni alle infezioni perché separati dal resto del corpo dalle cellule endoteliali. Nel caso in cui un agente infettivo superasse la barriera endoteliale, le cellule microglia reagirebbero repentinamente per diminuire l'infiammazione e distruggere l'infezione prima che venga danneggiato sensibilmente il tessuto neurale. A causa dell'impossibilità dell'azione degli anticorpi provenienti dal resto del corpo (sono pochi a riuscire ad attraversare il rivestimento a causa della loro grande dimensione), le cellule microglia devono essere in grado di riconoscere i corpi estranei e inghiottirli.

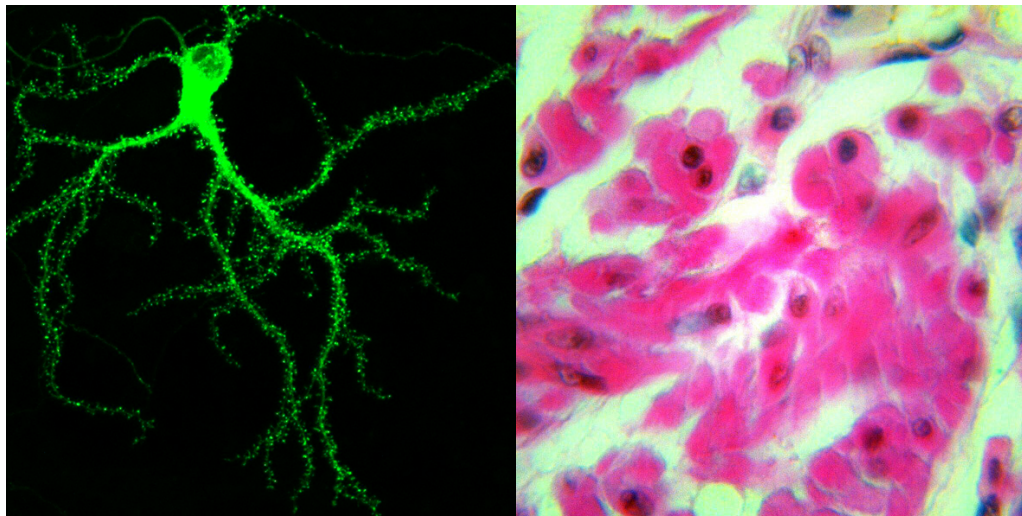
un ruolo nel processo di riparazione del cervello e della spina dorsale in seguito a traumi.

In questa tesi verrà presentato, in termini di un modello KLAIM[7], uno studio riguardante il recettore CB₂⁵ presente nelle cellule microglia, permettendo una ricerca e un'analisi di particolari aspetti:

- mostrare il comportamento di alcune funzionalità della microglia, analizzando alcuni parametri del sistema;
- studiare come un farmaco possa incidere nel caso in cui il sistema in analisi presenti delle mutazioni, in modo che, conosciute le caratteristiche del farmaco, si possa sapere quanto sia la concentrazione migliore a sopperire alla disfunzione;
- permettere alla ricerca, che in questo settore sta ancora muovendo i primi passi, di essere più efficiente con il supporto di simulazioni che siano da premessa e verifica a successivi (e alcune volte costosi) studi *in vivo* e *in vitro*.

L'attivazione del CB₂ non ha apparenti effetti comportamentali negli animali. Tuttavia il ruolo di questo recettore può avere possibili scopi terapeutici nel trattamento di malattie neurodegenerative (ad esempio il morbo di Alzheimer), nel trattamento di depressione e dipendenza da droghe e inoltre per il trattamento dei dolori che rispondono difficilmente ai trattamenti convenzionali, come il dolore neuropatico, causato dalla disfunzione o dal danneggiamento del sistema nervoso.

⁵ Un recettore è una proteina che si lega con un fattore specifico, definito *ligando*, causando in esso una variazione conformazionale in seguito alla quale si ha l'insorgenza di una risposta cellulare o un effetto biologico. Il recettore CB₂, conosciuto anche come CNR₂, fa parte, insieme al recettore CB₁, della famiglia dei recettori dei cannabinoidi.



(a) Un neurone

(b) Tessuto muscolare

Figura 2: Diversi tipi di cellule eccitabili.

Le immagini provengono dal Dipartimento di Anatomia Istologia e Medicina Legale, Università degli Studi di Firenze^a.

^a <http://www.dipaiml.unifi.it/istologia/nerv.htm>

1.2 UN MODELLO PER LE CELLULE ELETTRICAMENTE ECCITABILI

Una cellula eccitabile, la figura 2 ne riporta due tipi, ha la capacità di propagare un segnale elettrico alle cellule che la circondano. Esempi di cellule eccitabili sono i neuroni, i miociti⁶ cardiaci. La circolazione degli ioni tra l'interno e l'esterno della cellula provoca un cambiamento del potenziale elettrostatico lungo la membrana della cellula, che viene chiamato *potenziale d'azione*[6]. L'esterno è caricato positivamente, l'interno negativamente.

Le principali classi di ioni che interessano il fenomeno legato alle cellule eccitabili sono il sodio (Na^+) e il potassio (K^+); questi si spostano

⁶ Sono cellule proprie del tessuto muscolare e ne regolano il funzionamento.

utilizzando sia dei canali ionici voltaggio dipendenti che delle pompe presenti sulla membrana delle cellule eccitabili. Disturbi nell'eccitazione possono verificarsi nel comportamento di questi canali ionici a livello cellulare o nella propagazione di onde elettriche a livello di rete cellulare. In generale, un potenziale d'azione è un evento d'innescò dall'esterno: una cellula scatena un potenziale d'azione come risposta ad uno stimolo.

È importante osservare il comportamento delle cellule eccitabili per comprendere come alcune malattie, che interessano queste cellule, possano essere controllate e studiate. Un'anormale attività elettrica dei neuroni può causare epilessia⁷[8]: le crisi epilettiche sono le improvvise, rapide ed eccessive scariche elettriche di una popolazione più o meno estesa di neuroni che compongono la materia grigia del nostro cervello.

Per quanto riguarda lo studio delle cellule eccitabili, questa tesi si propone di indagare su come viene causato un potenziale d'azione e come questo incida nel momento in cui venga considerato un tessuto.

Lo scopo viene raggiunto attraverso la creazione di un modello KLAIM che risponda agli stimoli interni ed esterni allo stesso modo di una cellula eccitabile, permettendo così di ricreare il fenomeno sia singolarmente che nel momento in cui venga considerato in un sistema più complesso, come un tessuto.

Lo studio di tessuti composti da cellule eccitabili ha una forte rilevanza in numerose applicazioni, dal momento che regolano alcune delle più importanti attività fisiologiche di un organismo: basti pensare che molti animali, per difendersi o attaccare, hanno sviluppato mezzi che vanno a interagire sul loro funzionamento. Inoltre le cellule eccitabili sono oggetto di interesse di ricerche sulla preparazione di anestetici.

⁷ Disordine neurologico cronico, caratterizzato da frequenti e improvvise convulsioni.

1.3 PANORAMICA DELLA TESI

L'intento di questa tesi è quello di presentare come sia possibile descrivere e analizzare i sistemi biologici attraverso un'estensione di KLAIM, appositamente creata per compiere tali studi. Per riuscire in questo intento, dopo aver assimilato concetti che permettano la comprensione dei fenomeni e degli strumenti trattati, vengono presentate le due applicazioni precedentemente introdotte.

Il capitolo 2 ha il compito di introdurre i concetti biologici necessari per poter successivamente comprendere i modelli che verranno mostrati. Tra gli argomenti trattati verrà data una descrizione di alcune importanti "infrastrutture" e come queste operino tra di loro, garantendo alcuni fenomeni.

Nel capitolo 3 viene presentato in modo approfondito il linguaggio BioKLAIM, estensione di KLAIM accennata in precedenza, spiegando come funziona e che cosa fornisce all'utilizzatore. Inoltre viene presentato l'ambiente `bksim`, implementazione del linguaggio e della serie di strumenti messi a disposizione per lavorare con i modelli.

Nel capitolo 4 sono riportati i modelli BioKLAIM delle interazioni presentate. Successivamente alla descrizione dei modelli, vengono riportati gli studi relativi a questi, mostrando poi i risultati delle ricerche.

Infine nel capitolo 5 vengono tratte alcune considerazioni sulla natura della Biologia Computazionale, su come sia stato svolto il lavoro, su come la natura del linguaggio abbia facilitato la rappresentazione dei fenomeni e su come questa disciplina, seppur dimostrandosi in alcuni frangenti ostica, possa essere fonte di enormi passi avanti per la ricerca.

1.4 LAVORI CONNESSI

Esistono molteplici approcci per tentare di definire i comportamenti che caratterizzano il funzionamento della meccanica regolatrice delle interazioni nei processi biologici. Alcuni di questi sono gli Hybrid Automata [9, 10], i P-Systems [11], SBML [12] e Beta Binders [13]. Tali approcci si basano sulla rappresentazione di un sistema come un'interazione, più o meno articolata, di componenti più piccoli che insieme vanno a operare per ottenere un risultato.

CONCETTI BIOLOGICI

Questo capitolo introduce i concetti biologici necessari a comprendere i fenomeni descritti mediante BioKLAIM. In particolare verranno illustrati alcuni apparati fondamentali per le attività che risiedono nelle cellule: verranno messe in risalto alcune delle caratteristiche della membrana cellulare, strumento di appoggio e regolatore della vita e delle attività cellulari. In seguito saranno presi in esame alcuni degli elementi che risiedono in essa come i recettori, i canali ionici voltaggio dipendenti e le pompe ioniche.

Tali strutture saranno necessarie per poter comprendere il ruolo che il recettore CB₂ della microglia ha nel sistema endocannabinoide e il funzionamento delle cellule eccitabili. Per quanto riguarda queste ultime verranno inoltre mostrati i concetti fisici che stanno alla base del potenziale d'azione, con accenni al modello di Hodgkin-Huxley [14], che descrive l'andamento della differenza di potenziale negli assoni dei calamari giganti, dando una formalizzazione della fisiologia che regola le cellule eccitabili.

2.1 MEMBRANA CELLULARE

La membrana cellulare, della quale viene riportato uno schema in figura 3, è una struttura che separa l'interno delle cellule dall'ambiente esterno. Essa circonda le cellule ed ha la caratteristica di controllare le

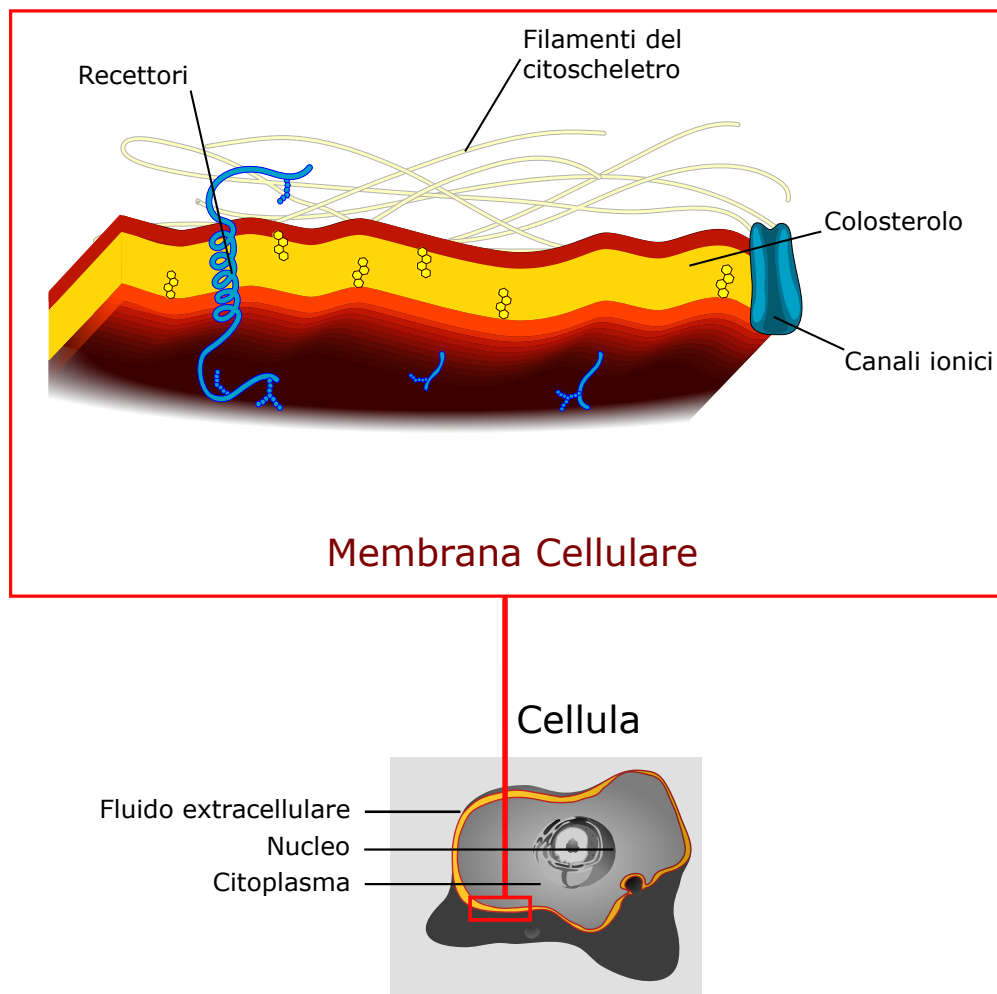


Figura 3: Cellula eucariota con un particolare della membrana. Immagine proveniente da Wikipedia.

sostanze in ingresso e in uscita. Questa barriera contiene un'estesa varietà di elementi, principalmente proteine e lipidi, interessati dalle attività cellulari come la gestione dei canali ionici e l'invio di segnali alle altre cellule. La membrana fa anche da appoggio al citoscheletro, una struttura proteica dinamica che mantiene la forma della cellula, la protegge e le permette il movimento (usando strutture come flagelli o ciglia).

Il movimento delle sostanze attraverso la barriera cellulare può essere attivo, quando sfrutta fonti di energia presenti all'interno della cellula, o passivo, quando avviene senza esplicito comando da parte della cellula o senza usare in qualche modo energia.

La membrana mantiene anche un potenziale elettrostatico cellulare che è chiamato *potenziale di membrana*.

2.2 RECETTORI

Un recettore è una proteina, immersa nella membrana o nel citoplasma¹ di una cellula, a cui possono legarsi uno o più tipi di molecole mediatrici di informazioni. Una molecola che si lega al recettore prende il nome di *ligando* e può essere di vari tipi: una molecola, un ormone, un farmaco oppure una tossina. Ogni specie di recettore può legarsi soltanto con alcuni ligandi; una cellula, di norma, possiede molti recettori, anche di tipi diversi.

Quando avviene il legame, il recettore subisce un cambiamento conformazionale (un cambiamento nella forma tridimensionale della proteina, senza però intaccare la sequenza di amminoacidi² che la compone e la caratterizza) che di solito avvia una risposta intracellulare. Alcuni ligandi,

¹ Liquido cellulare di supporto alla cellula.

² Sono gli elementi costitutivi delle proteine.

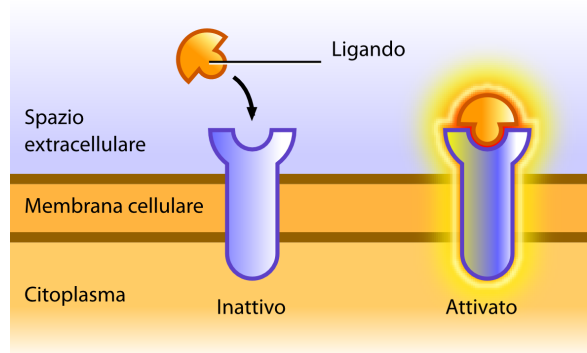


Figura 4: Schema di interazione di un recettore e il suo ligando.

invece, si legano al recettore senza indurre alcuna reazione, questi sono detti antagonisti.

2.2.1 Legame e attivazione

Il legame di un ligando ad un recettore è un processo dinamico di equilibrio chimico³ (in figura 4 viene mostrata l'attivazione di un recettore in seguito al legame).



I ligandi si legano e si dissociano dai recettori secondo la legge di azione di massa, riportata in (2.1), nella quale le parentesi quadre intendono le concentrazioni degli elementi[15] e il valore K_d , chiamato costante di dissociazione, dà una misura di quanto bene una molecola si adatti al recettore (più è alta l'affinità, più basso sarà il valore di dissociazione).

Non tutti i ligandi che si possono legare a un recettore provocano in esso la reazione di risposta. Esiste una classificazione per i ligandi[16]:

³ Stato in cui le attività chimiche o le concentrazioni dei reagenti e dei prodotti non hanno, nel tempo, alcun cambiamento netto.

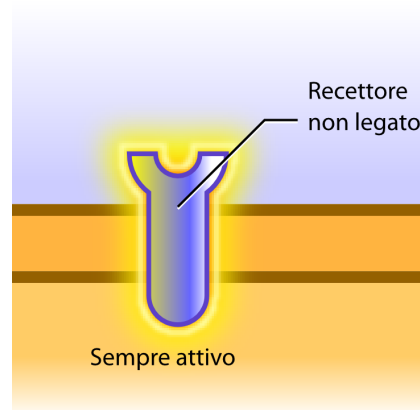


Figura 5: Schema di recettore che presenta attività costitutiva.

- Agonista totale: ha la capacità di attivare il recettore e il risultato del legame è la massima risposta biologica. In natura, la maggior parte dei ligandi è agonista totale.
- Agonista parziale: non attiva completamente il recettore, provocando una risposta parziale se confrontata con quella data dagli agonisti totali.
- Antagonista: si lega al recettore ma non lo attiva. Inibisce il legame dei recettori con gli agonisti.
- Agonista inverso: riduce l'attività del recettore, inibendo la sua "attività costitutiva"⁴. In figura 5 viene mostrato uno schema che rappresenta questa classe di recettori.

⁴ Capacità manifestata da una specie di recettori che hanno la caratteristica di produrre la risposta biologica in assenza di un ligando legato.

2.3 IL SISTEMA ENDOCANNABINOIDE

Gli endocannabinoidi rappresentano una classe di molecole che hanno la comune capacità di legarsi a particolari recettori. Il primo endocannabinoide ad essere stato identificato, nel 1992, è l'*anandamide* (AEA), che risulta essere un agonista parziale. In seguito è stata scoperta la presenza del *2-arachidonoilglicerolo* (2-AG), agonista totale, e soltanto recentemente (negli ultimi 10 anni) sono stati identificati altri cannabinoidi, ma il loro studio è tutt'ora argomento acceso per la ricerca biologica.[17]

Il 2-AG e l'AEA principalmente sono generati all'interno delle cellule neuronali: non vengono immagazzinati ma prodotti su richiesta e, interagendo tra loro con processi di sintesi, trasporto e degradazione, costituiscono il sistema endocannabinoide. Ad oggi è noto che questo sistema ha il compito di proteggere l'organismo dai danni causati da varie situazioni patologiche e può esercitare un'azione antiossidante, ipotensiva, immunosoppressiva, antinfiammatoria e antidolorifica.

La scoperta che il sistema endocannabinoide si estende anche a cellule del cervello ha suggerito che tale sistema ricopra un ruolo nel controllo del movimento e della percezione, nell'alterazione dei processi di apprendimento e memoria e nella regolazione degli stati emotivi.

Una volta che gli endocannabinoidi sono stati sintetizzati, vengono immediatamente rilasciati dalla cellula all'esterno di essa. Così i recettori cannabinoidi⁵, presenti nella membrana delle cellule limitrofe o sulla membrana della stessa cellula che li ha generati, hanno la possibilità di interagire con i ligandi prodotti.

Gli endocannabinoidi vengono rapidamente smaltiti attraverso un meccanismo comprendente vari passaggi:

⁵ Recettori presenti nelle cellule che interagiscono nel sistema endocannabinoide.

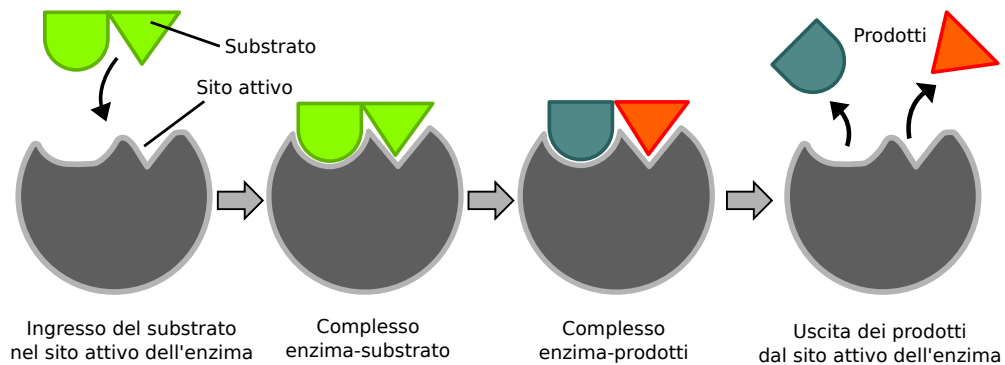


Figura 6: Fasi di azione di un enzima su di un substrato. Immagine proveniente da Wikipedia.

RE-UP TAKE - gli endocannabinoidi vengono trasportati all'interno della cellula;

IDROLISI ENZIMATICA - gli enzimi⁶ distruggono l'elemento, agendo sul substrato come mostrato in figura 6;

ESTERIFICAZIONE - i prodotti dell'operato degli enzimi vengono immagazzinati nella membrana.

L'enzima che catalizza l'idrolisi⁷ dell'AEA è il *fatty acid amide hydrolase* (FAAH), mentre l'idrolisi del 2-AG è catalizzata dall'enzima *monoacylglycerololipasi* (MAGL). Successivamente all'idrolisi del 2-AG e dell'AEA,

⁶ Sono proteine in grado di accelerare una reazione chimica. Un enzima accelera unicamente le velocità delle reazioni chimiche senza intervenire sui processi che ne regolano la spontaneità. Il suo ruolo consiste nel facilitare le reazioni attraverso l'interazione tra il substrato (la molecola o le molecole che partecipano alla reazione) e il proprio sito attivo (la parte di enzima in cui avvengono le reazioni). Avvenuta la reazione, il prodotto viene allontanato dall'enzima che rimane disponibile per iniziarne una nuova. L'enzima infatti non viene consumato durante la reazione.

⁷ Reazione chimica in cui la molecola viene scissa in due o più parti per effetto di una molecola d'acqua. Non è da confondersi con l'idratazione, in cui ad una molecola viene addizionata una molecola d'acqua.

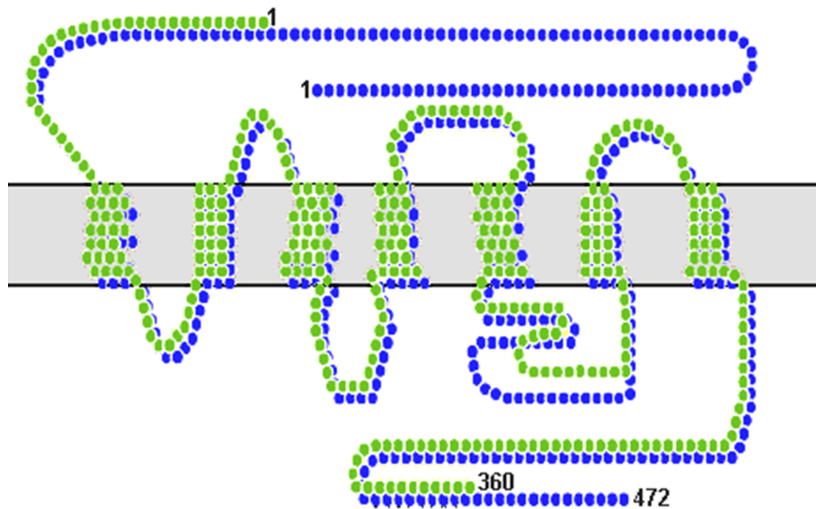


Figura 7: Posizione nella membrana cellulare dei recettori CB₁ (blu) e CB₂ (verde). I cerchi sono gli amminoacidi che compongono le due strutture. Immagine proveniente da Wikipedia.

viene prodotta una nuova sostanza che prende il nome di *acido arachidonico* (AA). Quest'ultimo viene coinvolto dall'enzima *prostaglandina-endoperossido sintasi* (COX-2), che catalizza la conversione di AA in *prostaglandine* (PGE)[18].

2.3.1 I recettori cannabinoidi

I recettori cannabinoidi sono una classe di recettori attivati da tre gruppi di ligandi: gli endocannabinoidi (prodotti dal corpo dei mammiferi), i cannabinoidi provenienti da piante (come il THC prodotto dalla pianta *cannabis*) e da cannabinoidi sintetici. Ad oggi sono stati scoperti due sottotipi di recettori cannabinoidi: il CB₁ ed il CB₂. La figura 7 mostra la struttura dei recettori nella membrana cellulare.

Il recettore CB₁ è principalmente espresso⁸ nel cervello mentre il re-

⁸ Termine proveniente dalla Biologia Molecolare che indica la presenza di un elemento

cettore CB2 è espresso nel sistema immunitario, ma ad oggi la ricerca biologica sta identificando altre regioni dell'organismo dove sono presenti recettori cannabinoidi (ad esempio: CB1 è espresso anche nei polmoni, nei reni, nel fegato e nelle cellule dell'apparato riproduttivo sia femminile che maschile, CB2 nelle cellule emopoietiche⁹).[19, 20]

2.3.2 *Il recettore CB2*

Questo recettore è stato scoperto recentemente grazie a esperimenti su ratti mutati geneticamente in modo da avere basse concentrazioni di CB1. Non sono state notate grandi differenze nella metabolizzazione degli endocannabinoidi e quindi è chiaro che CB1 non è il solo recettore del sistema.

La stimolazione del CB2 è la principale responsabile dell'azione antinfiammatoria e immunomodulatrice dei cannabinoidi e, tramite recenti studi su quali cellule lo esprimano, è risultata l'importanza che questo recettore ricopre nel funzionamento del sistema immunitario, nello sviluppo embrionale, nell'osteoporosi, nell'apoptosi¹⁰ cellulare e nella chemiotassi¹¹. Inoltre è stato osservato che ligandi non psicoattivi del

all'interno di una singola cellula.

⁹ Sono cellule che hanno il compito di originare tutti i tipi di cellule del sangue.

¹⁰ Un processo di morte cellulare programmata. In condizioni normali contribuisce al mantenimento del numero di cellule di un sistema. Al contrario della necrosi, che è una forma di morte cellulare risultante da un acuto stress o trauma cellulare, l'apoptosi è portata avanti in modo ordinato e regolato, richiede consumo di energia e generalmente porta ad un vantaggio durante il ciclo vitale dell'organismo (è infatti chiamata da alcuni *morte altruista* o *morte pulita*).

¹¹ Fenomeno con cui i corpi cellulari, batteri ed altri organismi uni o pluricellulari direzionano i loro movimenti a seconda della presenza di alcune sostanze chimiche nel loro ambiente. È molto importante per i batteri al fine di trovare cibo, nuotando verso

CB2 hanno effetto sulla degenerazione ossea, nella gotta, nei processi neuroprotettivi e nell'aterosclerosi.

Ruolo di CB2

Lo studio del CB2 permette alla ricerca di combattere una serie di malattie.

CANCRO.[21] I cannabinoidi possono ridurre la crescita tumorale nel cancro al polmone. Una molecola molto importante nell'attività antitumorale del CB2 è la ceramide, che si trova in alte concentrazioni nelle membrane cellulari e che regola l'apoptosi delle cellule. In seguito alla somministrazione di cannabinoidi è da notare un aumento della ceramide intracellulare.

Le cellule maligne mostrano un sostanziale aumento nella sintesi *de-novo*¹² di ceramide, che provoca l'inibizione delle AKT¹³, portando, insieme ad altri processi, alla apoptosi cellulare.

L'attivazione del CB2 nelle cellule cancerose del cancro alla prostata e al seno mostra una sostanziale attivazione del sistema regolato da AKT, permettendo di controllare la crescita della popolazione cellulare; nel melanoma i livelli di AKT si mantengono pressoché costanti mentre nel tumore al polmone e nel cancro al colon diminuiscono. La citotossicità

le concentrazioni massime di molecole di nutrimento, o sfuggendo da veleni. Negli organismi pluricellulari la chemiotassi è critica nelle prime fasi dello sviluppo (ad esempio il movimento dello sperma verso l'ovulo durante la fecondazione). Inoltre è stato riconosciuto che i meccanismi che permettono la chemiotassi negli animali possono essere sovvertiti durante la metastasi di un cancro.

¹² Sintesi per duplicazione semiconservativa: nel caso del DNA, una doppia elica figlia presenta un filamento della molecola parentale e un filamento di nuova sintesi.

¹³ Una famiglia di proteine che hanno un ruolo importante nei mammiferi per la segnalazione cellulare

dei cannabinoidi è comunque dipendente dalla concentrazione con cui vengono somministrati.

OSTEOPOROSI. L'osteoporosi è una delle malattie degenerative più comuni. Gli affetti da questa malattia hanno come caratteristica comune quella di avere una bassa densità minerale nelle ossa, con un aumento dei rischi di fratture. I fattori genetici alla base dell'osteoporosi sono ancora in gran parte sconosciuti; recentemente è stato dimostrato in [22] una correlazione tra il sistema endocannabinoide e la regolazione della massa ossea.

DEPRESSIONE. A livello del sistema nervoso centrale, il CB₂ si trova espresso, come già accennato, nelle cellule della microglia dove varia la sua espressione in base allo stadio in cui la cellula si trova. Nello stadio non attivato, l'espressione del CB₂ è limitata; appena le cellule vengono stimulate per compiere il loro impiego di difesa dalle infezioni, l'espressione del CB₂ aumenta. È stato notato in [23] che uno stadio infiammatorio prolungato può causare alterazioni neurodegenerative e cambiamenti nell'azione di neurotrasmettitori che contribuiscono alla determinazione dell'umore, come la serotonina e il glutammato.

2.3.3 *Ruolo di CB₂ nell'attivazione della microglia*

Il recettore CB₂ è presente nelle cellule della microglia ed ha lo scopo di consentire ad elementi prodotti da tali cellule di migrare verso le zone dove c'è necessità di combattere neuroinfiammazioni¹⁴.

La microglia, dopo essersi attivata, prende una forma ameboide e

¹⁴ Situazioni croniche del sistema nervoso centrale.

incrementa la presenza dei recettori nella sua membrana. Inoltre produce una serie di substrati (come l'AEA e il 2-AG) che permetteranno alla cellula di compiere il suo lavoro di contrattacco eliminando, tramite alcuni linfociti citotossici¹⁵ chiamati *Natural Killers Cells* (NK), gli agenti patogeni e i tessuti infettati. Gli NK agiscono attivando l'apoptosi della cellula a cui si legano.

Nel modello riportato sappiamo che il 2-AG risulta essere un agonista totale e il suo legame con il CB2 sviluppa un rilascio di proteine all'interno della cellula che, raggiunto il nucleo, attivano la produzione delle interleuchine, proteine dalla funzione di mediatrici di informazione (in questo caso apoptosi), che successivamente verranno secrete all'esterno della microglia nei pressi dell'infiammazione. L'AEA invece inibisce la produzione delle interleuchine e quindi rallenta l'efficacia di attacchi ad agenti patogeni. Il recettore CB2 presente nella microglia ha una costante di dissociazione più alta con l'AEA che con il 2-AG, quindi la microglia risulta più occupata ad attivare NK che a essere inibita dall'AEA.

2.4 CANALI IONICI VOLTAGGIO DIPENDENTI

Qualsiasi cellula animale possiede una grande resistività elettrica o, detto in altro modo, una bassa permeabilità agli ioni. I canali ionici (figura 8) sono delle particolari proteine (o assemblati di proteine) fissate permanentemente alla membrana dove formano dei pori. Questi permettono di stabilire e controllare il gradiente della forza elettrica da una parte all'altra della membrana cellulare, consentendo il flusso di determinati ioni

¹⁵ Sono cellule del sistema immunitario dei vertebrati. La caratteristica dei citotossici è quella di essere mortali per le cellule: un tipo di questi linfociti costituisce ciò che è chiamato *sistema immunitario innato*.

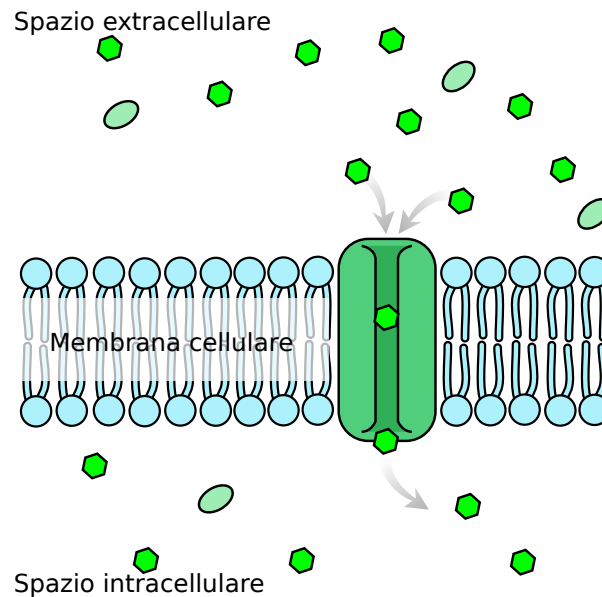


Figura 8: Schema di un canale ionico. Immagine proveniente da Wikipedia.

che danno il nome al canale. Un tipo di canale ionico è quello voltaggio dipendente: questo si attiva in base al cambiamento nella differenza di potenziale nelle vicinanze della cellula. Un altro tipo di canale ionico è quello detto di *perdita*, dove la permeabilità è più o meno costante[24].

Il ruolo biologico assunto da tali strutture è quello di essere alla base degli impulsi nervosi, mediare la conduzione dei segnali nelle sinapsi e dell'attività cardiaca, inoltre contribuiscono alla locomozione e al mantenimento, insieme al meccanismo dell'equilibrio, di una posizione stazionaria. La maggior parte delle tossine che gli organismi hanno sviluppato per difendersi da attacchi o per attaccare (il veleno prodotto dai ragni, scorpioni, etc...) ha effetto poiché opera sul funzionamento di questi canali. Nel campo della ricerca di nuovi farmaci, lo studio dei canali ionici è molto frequente ([25, 26, 27]).

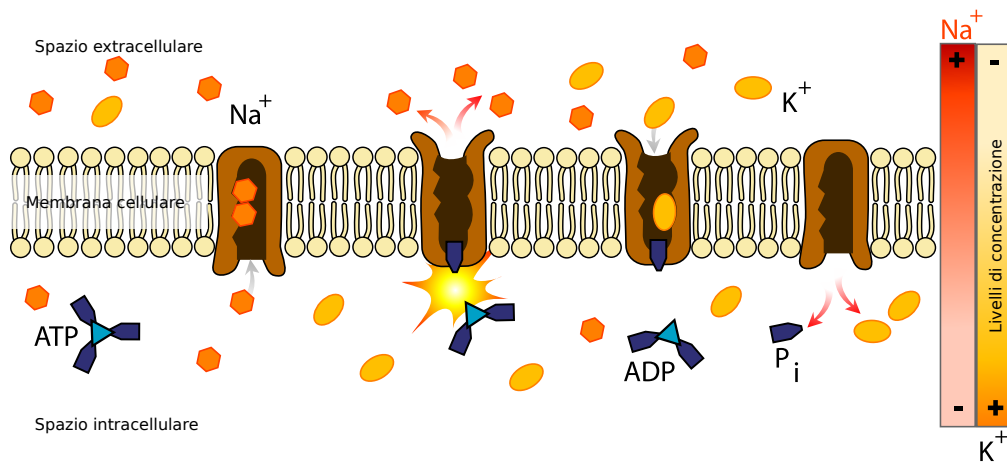


Figura 9: La pompa Na⁺/K⁺ usa l'energia derivata dall'ATP per scambiare ioni di Na⁺ per K⁺ lungo la membrana. Immagine proveniente da Wikipedia.

2.5 POMPE IONICHE

Le pompe ioniche sono delle proteine fissate permanentemente alla membrana che, quando attive, muovono gli ioni andando contro il gradiente, in contrasto con i canali ionici che invece sono caratterizzati da un trasporto passivo.

Queste pompe, per contrastare il gradiente, sfruttano l'*adenosina trifosfato*¹⁶ (ATP)

2.5.1 La pompa Sodio/Potassio

È stato rilevato da alcuni studi che una cellula ha al suo interno grandi concentrazioni di K⁺ e basse concentrazioni di Na⁺. Il meccanismo responsabile di questo è la pompa ionica Na⁺/K⁺ (figura 9) che muove i

¹⁶ L'ATP è il composto ad alta energia richiesto dalla maggioranza delle reazioni metaboliche.

due ioni in direzione opposta attraverso la membrana cellulare, trasportando 3 ioni di Na^+ al di fuori della cellula per ogni 2 ioni di K^+ pompato dentro. Tra le varie funzioni ricoperte dalla pompa Na^+/K^+ , una è quella di mantenere il volume della cellula, poiché molti elementi, se non ci fosse la pompa, migrerebbero fuori dalla cellula, dato che possiedono la stessa carica (la maggior parte degli elementi organici sono caricati negativamente).

2.6 LE CELLULE ECCITABILI

Nelle cellule non eccitabili e nelle eccitabili a riposo il potenziale di membrana è all'incirca stabile al valore di riposo. Per i neuroni tipicamente il valore di *potenziale di riposo*¹⁷ va da -70 a -80 mV (milliVolts). Come accennato, una cellula eccitabile ha la capacità intrinseca di generare un potenziale d'azione, ovvero un breve evento "tutto o nulla"¹⁸ (caratterizzato da un rapido aumento e da un'altrettanta rapida caduta di potenziale, dalla traiettoria stereotipata, in risposta ad uno stimolo) capace di trasmettere un impulso a cellule simili a sé.

Inizialmente, quando il potenziale di membrana è vicino al potenziale di riposo, i canali ionici sono chiusi, pronti ad attivarsi rapidamente nel momento in cui l'aumento del potenziale di membrana superi un valore soglia. Con i canali aperti viene permesso un flusso ionico verso l'interno, che produce un ulteriore aumento del potenziale di membrana, portando brevemente la membrana ad invertire la sua polarità, per riportarla poi

¹⁷ È il potenziale di membrana di una cellula in stato quiescente, dato dalla differenza del potenziale elettrico esterno e quello interno ($V_{\text{out}} - V_{\text{in}}$).

¹⁸ Reazione chimica caratterizzata dall'avere solo due possibili soluzioni: un totale successo oppure un completo fallimento. Questo tipo di legge regola la trasmissione nervosa.

al suo livello base.

L'apertura o la chiusura dei canali ionici inducono (oltre a un cambiamento locale nel potenziale di membrana, che scatena rapidi flussi di corrente negli altri punti della membrana) l'inizio di fenomeni di depolarizzazione, ovvero la fase in cui il potenziale di membrana aumenta, o di iperpolarizzazione, dove si ha una diminuzione del potenziale.

2.6.1 *Potenziale di membrana*

Il citoplasma e il fluido esterno alla cellula contengono alte concentrazioni di ioni mobili come il Na^+ e K^+ . Il potenziale di membrana è il risultato dell'interazione dei canali ionici e delle pompe ioniche immerse nella membrana cellulare; i canali e le pompe mantengono le diverse concentrazioni ioniche sui due lati della membrana.

Il potenziale di membrana ha due principali funzioni base:

- permette a una cellula di funzionare come una batteria, fornendo energia per operare ad una varietà di "dispositivi molecolari" immersi nella membrana;
- nelle cellule elettricamente eccitabili, come i neuroni, viene usato per trasmettere segnali a parti della cellula o a cellule limitrofe.

Il potenziale di membrana in una cellula deriva fondamentalmente da due fattori: l'attrazione elettrica e la diffusione. La forza elettrica deriva dalla mutua attrazione tra le particelle di carica opposta e dalla repulsione tra particelle di carica uguale. La diffusione deriva invece dalla statistica tendenza delle particelle a ridistribuirsi da regioni dove sono molto concentrate a regioni dove la concentrazione è più bassa.

Tensione

La tensione è l'abilità di generare una corrente elettrica. Se un generatore di tensione, come ad esempio una batteria, viene posto in un circuito elettrico, più alta sarà la sua tensione maggiore sarà la corrente elettrica sviluppata. In un circuito funzionante, la differenza di tensione tra due punti qualsiasi determina la quantità di corrente in circolo (ad ogni punto può essere assegnato un livello di tensione). La differenza di tensione tra due punti può essere misurata attraverso l'uso di un voltmetro.

Il valore assoluto della tensione non ha alcun significato. Un voltmetro è in grado di misurare la differenza di tensione tra due località di un circuito e non esiste un valore di tensione in un unico punto. Convenzionalmente viene assegnato a qualche elemento, scelto arbitrariamente, una tensione pari a zero e successivamente vengono ricavati gli altri valori di tensione.

Lo stesso principio vale per la tensione nelle cellule. In un tessuto elettricamente attivo, la differenza di tensione tra due punti qualsiasi può essere misurata attraverso l'inserimento di un elettrodo in ogni punto e collegando entrambi gli elettrodi a uno strumento di misura. L'affermazione che la differenza di potenziale attraverso la membrana di una cellula è pari a -70mV può essere verificata mettendo elettrodi all'interno e all'esterno della cellula: assume significato solo se viene considerata la differenza tra due regioni ed è indifferente se viene assegnato il valore 70mV all'interno e 0mV all'esterno o viceversa.

In termini matematici, la definizione di tensione ha come base il concetto di campo elettrico E , un campo vettoriale che assegna una grandezza e una direzione ad ogni punto nello spazio. In molte situazioni il campo elettrico è un campo conservativo¹⁹, il che vuol dire che può essere

¹⁹ Esiste una funzione scalare (il potenziale elettrico) per cui il valore non dipende dal

espresso come gradiente di una funzione scalare V , cioè $\mathbf{E} = -\nabla V$. Questo campo scalare V è indicato come la distribuzione di tensione. I campi elettrici possono essere trattati come conservativi se non c'è una significativa influenza di campi magnetici, situazione applicabile ai tessuti biologici. Dato che il campo elettrico è il gradiente della distribuzione di tensione, rapidi cambiamenti di tensione all'interno di una piccola regione implicano un forte campo elettrico e, viceversa, se la tensione rimane all'incirca la stessa su di una grande regione, i campi elettrici in quella regione devono essere deboli. Un forte campo elettrico, equivalente a un forte gradiente di tensione, implica che venga esercitata una grande forza su tutte le particelle cariche che si trovano all'interno della regione in esame.

Usando la terminologia elettrica, la membrana cellulare si comporta come una combinazione di una resistenza²⁰ e di un condensatore²¹. La resistenza deriva dal fatto che la membrana impedisce il movimento degli ioni, il condensatore deriva dal fatto che la barriera della membrana è talmente sottile che l'accumularsi di cariche da un lato crea una forza elettrica che attrae particelle con carica opposta verso il lato opposto. La capacità²² di una cellula è circa stabile al valore di $2\mu\text{F}/\text{cm}^2$ (non è condizionata dai canali ionici e dalle pompe ioniche presenti nella membrana). La conduttanza²³ è molto bassa ma è la caratteristica dominante, in più è molto variabile a differenza della capacità.

cammino percorso ma è solo in funzione degli estremi.

20 Componente elettrico che ha la caratteristica di convertire l'energia elettrica in energia termica.

21 Componente elettrico che immagazzina energia in un campo elettrostatico, accumulando al suo interno una certa quantità di carica elettrica.

22 Dato un condensatore è la quantità di energia elettrica immagazzinabile dal componente.

23 Espressione quantitativa dell'attitudine di un conduttore ad essere percorso da corrente elettrica; è l'inverso della resistenza.

2.6.2 Potenziale di equilibrio

Il potenziale di equilibrio di uno ione è il valore di tensione della membrana raggiunto il quale si crea un equilibrio elettrochimico tra la forza elettrica e la diffusione, in modo da non avere spostamenti attraverso la membrana cellulare. Questo significa che la tensione della membrana è esattamente opposta alla forza di diffusione dello ione e quindi la corrente netta dello ione è zero e non cambia.

Il potenziale d'equilibrio $E_{eq,Ione}$ di un particolare ione si calcola con l'**equazione di Nernst**:

$$E_{eq,Ione} = \frac{RT}{zF} \ln \frac{[Ione]_{out}}{[Ione]_{in}}$$

dove

- R è la costante universale dei gas, pari a $8,314472 \frac{J}{^{\circ}K \text{ mol}}$;
- T è la temperatura, misurata in kelvin ($^{\circ}K = ^{\circ}C + 273.15$);
- z è il numero di cariche elementari dello ione interessato dalla reazione;
- F è la costante di Faraday, uguale a $96,485339 \frac{J}{mV * mol}$
- $[Ione]_{out}$ è la concentrazione extracellulare dello ione, misurata in $\frac{mol}{m^3}$;
- $[Ione]_{in}$ è la concentrazione intracellulare dello ione.

Anche se gli ioni hanno la stessa carica (K^+ e Na^+), questi hanno differenti valori per il potenziale d'equilibrio dato dal rapporto tra le concentrazioni esterne e interne. Per esempio, nei neuroni il potenziale d'equilibrio per K^+ (E_{eq,K^+}) è pari a circa $-90mV$ sapendo che di

media si ha un rapporto tra la concentrazione esterna e interna pari a $\frac{4 \times 10^{-3} \text{ mol/m}^3}{117 \times 10^{-3} \text{ mol/m}^3}$, mentre per Na^+ ($E_{\text{eq,Na}^+}$) è approssimativamente 30mV con un rapporto tra concentrazioni pari a $\frac{63 \times 10^{-3} \text{ mol/m}^3}{20 \times 10^{-3} \text{ mol/m}^3}$.

2.6.3 Potenziale di riposo

Quando la membrana di una cellula per un lungo periodo non ha cambiamenti significativi si dice che ha raggiunto il suo potenziale di riposo. Esso indica sia il potenziale delle cellule non eccitabili che quello delle eccitabili, nel caso in cui non vi sia eccitazione.

Le interazioni che generano il potenziale di riposo E_m sono modellate secondo l'**equazione di Goldman**:

$$E_m = \frac{RT}{zF} \ln \frac{P_K[K^+]_{\text{out}} + P_{\text{Na}}[\text{Na}^+]_{\text{out}}}{P_K[K^+]_{\text{in}} + P_{\text{Na}}[\text{Na}^+]_{\text{in}}}$$

dove:

- P_{Ione} è la permeabilità della membrana allo specifico ione;
- $[\text{Ione}]_{\text{out}}$, $[\text{Ione}]_{\text{in}}$ sono la concentrazione esterna e interna alla cellula dello ione;
- R , T , F sono rispettivamente la costante universale dei gas, la temperatura assoluta e la costante di Faraday.

Questa equazione ha una forma simile a quella di Nernst, in quanto si basa sulle cariche degli ioni nel sistema, così come sul rapporto tra le concentrazioni tra l'esterno e l'interno: l'equazione di Nernst, dove viene preso in considerazione un solo ione, è un caso particolare dell'equazione di Goldman. Essenzialmente l'equazione di Goldman esprime il potenziale di membrana come una media pesata (sulla permeabilità)

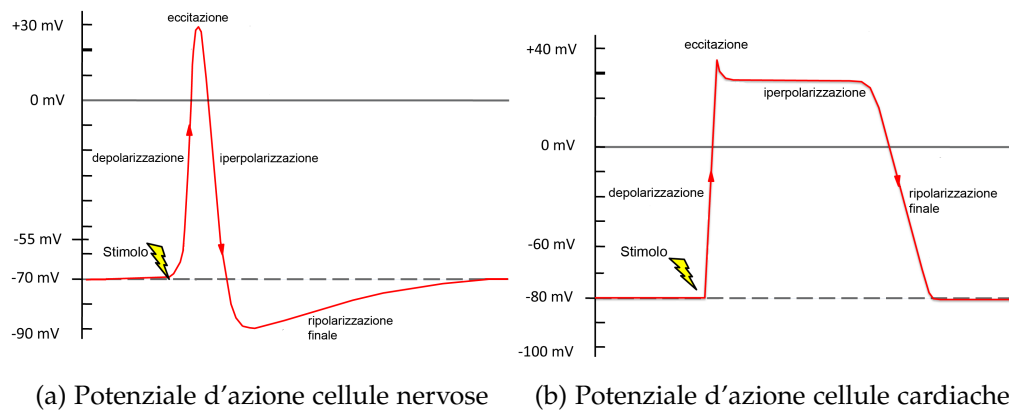


Figura 10: Diversi potenziali d'azione.

di potenziali di equilibrio degli ioni. Nella maggior parte delle cellule animali, la permeabilità del K^+ è molto più alta rispetto alla permeabilità del Na^+ (una cellula eccitabile, a riposo, ha una permeabilità al Na^+ pari al 5% rispetto al K^+). Il valore medio del potenziale di riposo di una cellula animale è di circa $-75mV$.

2.7 POTENZIALE D'AZIONE

Il potenziale d'azione, come accennato, è il fenomeno che prevede un rapido cambiamento di potenziale tra l'interno e l'esterno della membrana cellulare. La durata dell'evento è di circa $2ms$, seguiti da un periodo refrattario, prima assoluto²⁴, quindi relativo²⁵; infine si ristabilisce il potenziale di riposo.

Il potenziale d'azione del sistema nervoso, figura 10a, permette la

²⁴ Tutti i canali per Na^+ sono in uno stadio di inattività e non possono essere aperti poiché dovrebbero prima essere chiusi, quindi non si può generare mai un potenziale d'azione.

²⁵ Nel periodo refrattario relativo, invece, un certo numero di canali per Na^+ si sono chiusi e possono quindi essere aperti di nuovo per far passare ioni di Na^+ : perciò con uno stimolo notevolmente forte si potrebbe generare un nuovo potenziale d'azione.

trasmissione di informazioni fra cellule: il potenziale d'azione si trasmette simultaneamente a tutta la membrana della cellula e dunque anche alle diramazioni più distanti, come gli assoni²⁶.

Se la membrana si depolarizza fino a -55mV (detto *valore soglia*) si ha l'apertura dei canali voltaggio dipendenti per Na^+ , che fanno sì che ioni Na^+ entrino nella cellula, innescando così il "ciclo di Hodgkin" (figura 11), depolarizzando ulteriormente la membrana fino al valore d'equilibrio per Na^+ di 35mV .

Raggiunto questo valore di ΔV , oltre all'invio di segnali di stimolo dalla cellula eccitata alle cellule collegate a quest'ultima, si attivano i canali voltaggio dipendenti per K^+ , che permettono la fuoriuscita del K^+ fino al valore di equilibrio per K^+ di -90mV . A questo punto entra in azione la pompa ionica Na^+/K^+ che, contrastando il gradiente, riporta la membrana al valore di riposo (spostando per ogni 3 ioni Na^+ all'esterno 2 ioni K^+ all'interno).

Il ciclo di Hodgkin è un evento che avviene durante la fase ascendente dello stimolo neuronale o subito dopo aver raggiunto il valore soglia per innescare la propagazione del segnale. È un esempio di feed-back positivo: inizia con l'aumento della conduttanza al sodio che determina una depolarizzazione, la quale aumenta maggiormente la conduttanza.

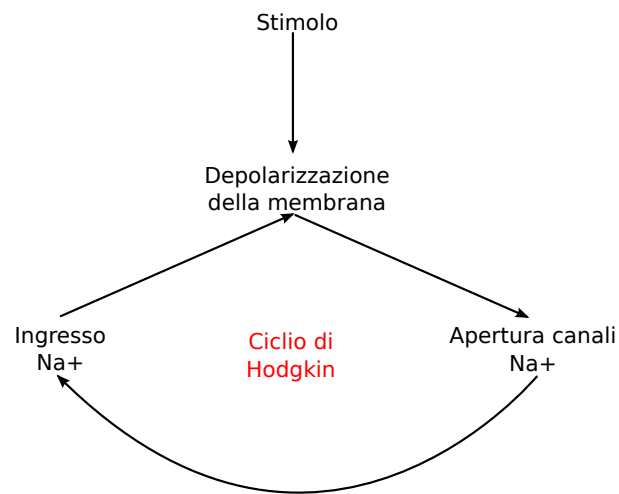


Figura 11: Schema del ciclo di Hodgkin.

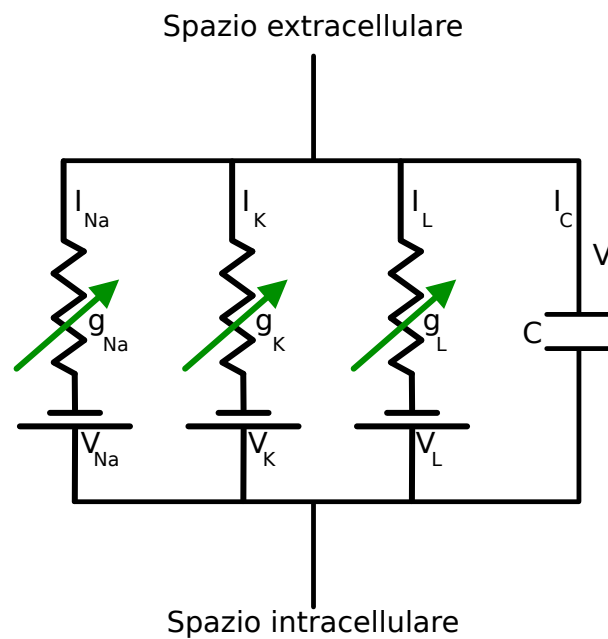


Figura 12: Componenti del modello di Hodgkin-Huxley. Esso mostra la caratteristica biofisica della membrana cellulare, rappresentata dal condensatore C . I canali ionici voltage dipendenti e i canali di perdita sono rappresentati rispettivamente dalle conduttanze g_n e g_L . Il gradiente elettrochimico che muove gli ioni è rappresentato dalle batterie V_n .

2.7.1 Circuito equivalente

Il circuito equivalente, mostrato in figura 12, è costituito da un condensatore in parallelo a tre vie, ciascuna composta di una batteria in serie a un potenziometro. La capacità del condensatore è determinata dalle proprietà della membrana ed è nota essere circa $2\mu\text{F}/\text{cm}^2$. Ciascuno dei tre percorsi paralleli viene da uno degli ioni e dalla presenza dei canali *di perdita*. La tensione di ogni percorso è determinata dalla concentrazione dello ione su ogni lato della membrana. La conduttanza di ogni percorso in qualsiasi momento è determinata dallo stato dei canali ionici permeabili a quello ione.

Questo modello è chiamato **modello di Hodgkin-Huxley**, dal nome dei due scienziati che nel 1952 hanno descritto il comportamento degli assoni dei calamari giganti, scoprendo quali meccanismi ionici stiano alla base della creazione e della propagazione del potenziale d'azione. Per questo nel 1963 hanno ricevuto il Premio Nobel per la fisiologia o la medicina.

26 Prolungamento principale che si distacca dal corpo cellulare del neurone e può contattare altri neuroni vicini, ovvero neuroni o altre cellule (ad esempio muscoli) anche molto distanti. Nella parte terminale dell'assone e delle sue collaterali si formano le terminazioni sinaptiche, che realizzano il contatto funzionale con altri neuroni.

BIOKLAIM, UN LINGUAGGIO PER MODELLI BIOLOGICI

Le algebre di processo [28, 29, 30, 31] sono un insieme di rigorosi linguaggi con una semantica ben definita che permette di descrivere e verificare proprietà di sistemi di comunicazione concorrenti. Possono essere visti come modelli matematici, considerando gli agenti che interoperano continuamente l'uno con l'altro in un ambiente comune. Gli agenti possono essere oggetti reali, oppure artefatti rappresentati da hardware o sistemi software.

Le algebre di processo forniscono un numero di costruttori per la creazione di sistemi e sono dotate di semantica operativa che descrive l'evoluzione del sistema.

In questi ultimi anni l'approccio delle algebre di processo è stato usato per dare descrizioni di sistemi biologici. L'idea è quella di rappresentare le macromolecole, i muri portanti dei sistemi biologici, come termini di algebre di processo. Infatti una macromolecola è una grande molecola che ha stati interni e attività esterne.

Una macromolecola quindi può essere descritta tramite un automa. Attribuire stati e transizioni a questi elementi permette di guardare un evento chimico con un'altra ottica: i legami tra macromolecole, o i loro cambiamenti interni, sono tutti connotati da un passaggio tra stati, il quale, tipicamente, è guidato da interazioni esterne. Molti linguaggi e calcoli hanno proposto modelli formali per specificare, simulare e analizzare sistemi biologici [32, 33, 34, 35], principalmente basati sulle

algebre di processo CCS e CSP.

KLAIM[7] (*Kernel Language for Agents Interaction and Mobility*) è un linguaggio sperimentale ideato specificatamente per descrivere sistemi distribuiti, costituiti da componenti mobili che interagiscono attraverso molteplici spazi delle tuple (messaggi), ognuno dei quali è identificato da un indirizzo.

In questo capitolo prenderemo in considerazione BIOKLAIM, una variante di KLAIM specificatamente pensata per la descrizione di sistemi biologici.

3.1 KLAIM

Il paradigma di comunicazione KLAIM si basa sulla nozione di comunicazione fondata su un unico spazio di tuple condiviso, suggerito da Linda[36], e la estende. Uno spazio delle tuple è un multi-insieme di tuple che rappresenta sequenze di unità di informazione. Le tuple sono anonime e possono essere recuperate da processi per mezzo di meccanismi di pattern-matching usando i template. La comunicazione tra processi è asincrona: produttore e consumatore di una tupla non devono sincronizzarsi.

Il modello di comunicazione di Linda è stato originariamente proposto per la programmazione concorrente su macchine isolate. KLAIM estende il paradigma di Linda considerando molteplici spazi delle tuple, possibilmente distribuiti. Il modello di comunicazione che ne deriva ha una serie di proprietà che lo hanno reso attraente per il global computing. Il modello consente il *time uncoupling* (il tempo di vita del dato è indipendente dal tempo di vita del processo produttore), il *destination uncoupling* (il produttore di un dato non ha bisogno di conoscere il futuro uso o

la destinazione finale del dato prodotto) e lo *space uncoupling* (processi comunicanti necessitano di conoscere una singola interfaccia, cioè le operazioni nello spazio delle tuple).

Il paradigma di programmazione di KLAIM enfatizza una chiara separazione tra il livello computazionale e il livello di coordinazione/amministrazione della rete. Intuitivamente vengono programmate unità computazionali (processi e agenti), viene descritta la distribuzione iniziale dei processi e degli elementi riguardanti la computazione e infine viene dato il completo controllo sui cambiamenti di configurazione della rete, cambiamenti che possono essere causa di addizione o sottrazione di componenti software. I processi possono essere posizionati in diverse località di una rete, in cui i dati possono essere archiviati e recuperati.

In KLAIM l'infrastruttura di rete è chiaramente distinguibile dai processi utente ed è esplicitamente modellata; le località possono essere dinamicamente create e comunicate. La strutturazione delle applicazioni in termini di processi e di coordinazione fornisce un'astrazione pulita e potente, utile a definire i comportamenti e i meccanismi di applicazione.

Un sistema KLAIM, chiamato rete, è un insieme di nodi ognuno dei quali identificato da una località fisica. Una località fisica può essere vista come l'indirizzo di un nodo della rete. Ogni nodo ha una componente computazionale, formata da una serie di processi in esecuzione in parallelo, un ambiente di allocazione e dei dati presenti nel suo spazio delle tuple. I processi interagiscono tra di loro sia localmente che in remoto, inviando e recuperando delle tuple da e verso un altro spazio delle tuple. I processi possono fare riferimento sia a località fisiche che logiche. Mentre una località fisica ha un significato globale, la località logica ha un significato locale e la sua interpretazione dipende dal nodo in cui i processi vengono eseguiti. Come già detto, ogni agente è dotato

di un ambiente di allocazione dove vengono riportate le corrispondenze tra località logica e località fisica. Quando un processo utilizza una località logica, viene consultato l'ambiente di allocazione per recuperare l'indirizzo fisico associato.

È stata presentata un'estensione di KLAIM: in essa gli agenti definiti rappresentano processi stocastici[37]. Nell'estensione proposta, un'azione è guidata da una distribuzione esponenziale. Il modello operativo risultante è quindi una catena di Markov a tempo continuo (CTMC).

3.2 CONCETTI DI TEORIA DELLA PROBABILITÀ

Viene riportata una breve introduzione alle catene di Markov a tempo continuo. Una trattazione completa può essere trovata in [38]. In secondo luogo verranno presentati i *Rate Transition System* (RTS) come proposti in [39, 40].

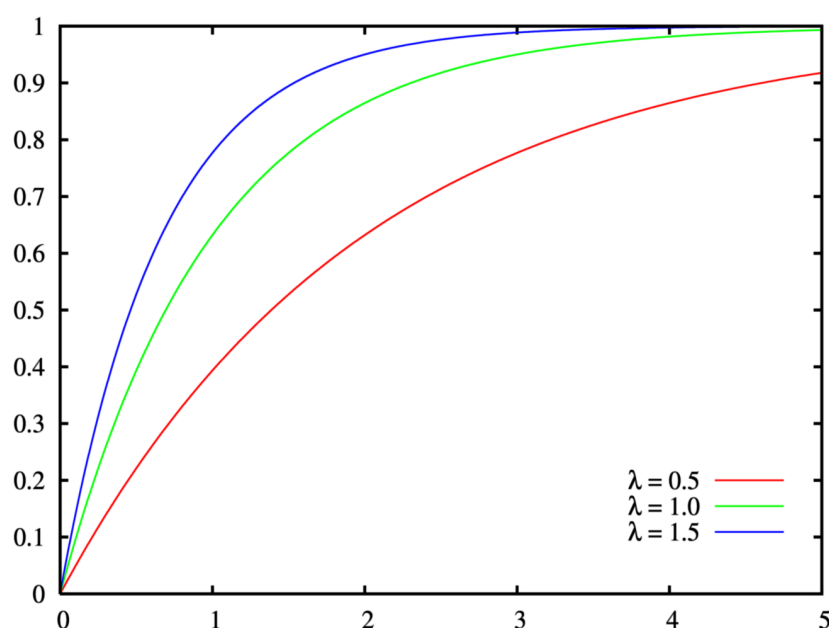


Figura 13: Funzione di distribuzione cumulativa per la distribuzione esponenziale. Immagine proveniente da Wikipedia.

3.2.1 Distribuzione esponenziale

Una variabile casuale continua X è *esponenziale* con parametro $\lambda > 0$ se la sua funzione di distribuzione cumulativa¹ (CDF), figura 13, è della forma:

$$F_X(d) \stackrel{\text{def}}{=} \mathbb{P}\{X \leq d\} = 1 - e^{-\lambda \cdot d} \text{ per } d \geq 0.$$

¹ Viene utilizzata per descrivere fenomeni quantitativi o comunque descritti con valori misurati su scale: ordinali (che possiedono una relazione d'ordine totale, ovvero i valori possono essere ordinati e si può stabilire se uno è "maggiore", uguale o "minore" di un altro), intervallari (dove c'è la possibilità di calcolare la differenza tra due valori, quindi anche di ordinarli) o proporzionali (dove c'è la possibilità di calcolare la proporzione tra due valori, quindi anche la loro differenza).

Viene indicata solitamente con $F(x)$ e rappresenta il numero di osservazioni del fenomeno che cadono prima del valore x .

Il numero reale λ è detto *rate* di X e determina univocamente una distribuzione esponenziale. Il valore atteso di X è $\frac{1}{\lambda}$, vale a dire che la durata media è di $\frac{1}{\lambda}$ unità di tempo, mentre la varianza è pari a $\frac{1}{\lambda^2}$.

Le distribuzioni esponenziali possiedono importanti caratteristiche:

1. la proprietà di essere prive di memoria, ovvero:

$$\mathbb{P}\{X > t + d | X > t\} = \mathbb{P}\{X > d\};$$

2. date due variabili casuali esponenziali X e Y , rispettivamente con rate λ e μ , la variabile esponenziale $\min(X, Y)$ è, a sua volta, esponenzialmente distribuita con rate $\lambda + \mu$;

La seconda proprietà viene anche chiamata *race condition* delle distribuzioni esponenziali. Viene chiamata così perché risulta essere una “gara” tra una serie di possibili eventi selezionabili: l’evento più veloce viene eseguito.

Nel caso in cui siano presenti due variabili aleatorie X e Y , rispettivamente con rate λ e μ , la probabilità che X abbia la meglio su Y è determinata da:

$$\mathbb{P}\{X = \min(X, Y)\} = \frac{\lambda}{\lambda + \mu}.$$

3.2.2 Catene di Markov a tempo continuo

Una CTMC è una coppia (S, \mathbf{R}) dove S è un insieme numerabile di stati e \mathbf{R} è una matrice che assegna rate non negativi a coppie di stati, tale che $\forall s \in S : \sum_{s' \in S} \mathbf{R}[s, s']$ converge.

(S, \mathbf{R}) descrive un processo stocastico dove, per qualsiasi stato $s \in S$, quando $\sum_{s' \in S} \mathbf{R}[s, s'] > 0$, la probabilità di selezionare una transizione

uscente da s nel tempo t è $1 - e^{-\sum_{s' \in S} \mathbf{R}[s, s'] \cdot t}$, cioè il tempo di residenza in s è esponenzialmente distribuito con rate $\sum_{s' \in S} \mathbf{R}[s, s']$, e la probabilità di selezionare una transizione da uno stato s ad uno stato s' è $\frac{\mathbf{R}(s, s')}{\sum_{s'' \in S} \mathbf{R}[s, s'']}$. Nel caso in cui $\sum_{s' \in S} \mathbf{R}[s, s'] = 0$, allora s è chiamato *absorbing state*, dato che se un processo entra nello stato s rimarrà sempre in s . Nel caso in cui $\mathbf{R}[s, s] > 0$ si dice che s presenta *self-loops*. È possibile rimuovere o aggiungere self-loops senza che i risultati dell'analisi transiente e steady-state vengano alterati.

Proposizione 3.2.1. *Il comportamento transiente di una CTMC $C = (S, \mathbf{R})$ con $\mathbf{R}[\bar{s}, \bar{s}] > 0$ coincide per qualche $\bar{s} \in S$ con quello di una CTMC $\tilde{C} = (S, \tilde{\mathbf{R}})$, tale che $\tilde{\mathbf{R}}[s, s'] \stackrel{\text{def}}{=} 0$ se $s = s'$ e $\tilde{\mathbf{R}}[s, s'] \stackrel{\text{def}}{=} \mathbf{R}[s, s']$ altrimenti.*

Dimostrazione. Vedere [41]. □

3.2.3 Rate Transition System

Un Rate Transition System [40] è una generalizzazione di un *Labelled Transition System*² (LTS), specificatamente costruito per descrivere comportamenti stocastici delle algebre di processo, inoltre è anche uno strumento per generare CTMC associate a un dato sistema. La relazione di transizione associa ad ogni stato e ad ogni azione una funzione che collega ogni stato a un numero reale non negativo. Formalmente:

Definizione 3.1 (Rate Transition System). Un RTS è definito da una tripla (S, A, \rightarrow) , dove S è un insieme di stati, A è un insieme di etichette, \rightarrow

² Un LTS è una tripla $(S, \Lambda, \rightarrow)$ dove S è un insieme di stati, Λ è un insieme di etichette e $\rightarrow \subseteq S \times \Lambda \times S$ è una relazione ternaria di transizioni etichettate. Se $p, q \in S$ e $\alpha \in \Lambda$, allora $(p, \alpha, q) \in \rightarrow$ si scrive: $p \xrightarrow{\alpha} q$. Le etichette possono rappresentare diversi aspetti a seconda di ciò che viene studiato.

è un sottoinsieme di $S \times A \times \sum_S$ dove \sum_S è l'insieme $[S \rightarrow \mathbb{R}_{\geq 0}]$ delle funzioni totali da S a $\mathbb{R}_{\geq 0}$.

Con $\mathcal{R}, \mathcal{R}_1, \mathcal{R}', \dots$ verranno indicati gli RTS, mentre $\mathfrak{P}, \mathfrak{L}, \mathfrak{R}, \dots$ saranno gli elementi di \sum_S ; in più la notazione $s \xrightarrow{\alpha} \mathfrak{P}$ indica $(s, \alpha, \mathfrak{P}) \in \rightarrow$.

Se $s_1 \xrightarrow{\alpha} \mathfrak{P}$ e $\mathfrak{P}(s_2) = v \in \mathbb{R}_{>0}$ allora s_2 è raggiungibile da s_1 tramite l'esecuzione di α e la durata di tale esecuzione è caratterizzata da una variabile casuale la cui distribuzione è contraddistinta da una distribuzione esponenziale negativa con rate v . $\mathfrak{P}(s_2) = 0$ significa che s_2 non è raggiungibile da s_1 attraverso α . In seguito verrà usato il simbolo \emptyset per indicare la funzione costante 0, mentre $[s_1 \mapsto v_1, \dots, s_n \mapsto v_n]$ indica una funzione che associa v_i a s_i e 0 a tutti gli altri stati. \sum_S possiede alcuni operatori come:

- $+$: $\sum_S \times \sum_S \rightarrow (S \rightarrow \mathbb{R}_{\geq 0})$ con $(\mathfrak{P} + \mathfrak{L})s \stackrel{\text{def}}{=} (\mathfrak{P}s) + (\mathfrak{L}s)$;
- \oplus : $\sum_S \rightarrow 2^S \rightarrow \mathbb{R}_{\geq 0}$ con $\oplus \mathfrak{P} C \stackrel{\text{def}}{=} \sum_{s \in C} (\mathfrak{P}s)$, per $C \subseteq S$;
- $\oplus \mathfrak{P}$ indica $\oplus \mathfrak{P}(S)$.

Definizione 3.2. Sia $\mathcal{R} = (S, A, \rightarrow)$ un RTS, allora:

- \mathcal{R} è ben definito se e soltanto se $\forall s \in S, \alpha \in A$ e $\mathfrak{P} \in \sum_S$ tale che $s \xrightarrow{\alpha} \mathfrak{P}$ si ha: $\exists x : \oplus \mathfrak{P} \leq x$;
- \mathcal{R} è *image-finite*³ se e soltanto se $\forall s \in S, \alpha \in A$ e $\mathfrak{P} \in \sum_S$ tale che $s \xrightarrow{\alpha} \mathfrak{P}$ oppure $\mathfrak{P} = \emptyset$ oppure $\mathfrak{P} = [s_1 \mapsto \lambda_1, \dots, s_n \mapsto \lambda_n]$;
- \mathcal{R} è completamente stocastico se e soltanto se $\forall s \in S, \alpha \in A, \mathfrak{P}$ e $\mathfrak{L} \in \sum_S$ si ha: $s \xrightarrow{\alpha} \mathfrak{P}, s \xrightarrow{\alpha} \mathfrak{L} \Rightarrow \mathfrak{P} = \mathfrak{L}$.

Verranno considerati solo RTS *ben definiti*.

³ Ogni stato raggiunge solo un numero finito di stati successivi.

In generale, dato un RTS (S, A, \rightarrow) , si cerca la CTMC composta dagli stati raggiungibili da un sottoinsieme C di S soltanto attraverso le azioni in $A' \subseteq A$. A questo scopo vengono introdotte le seguenti definizioni:

Definizione 3.3. Per insiemi $C \subseteq S$ e $A' \subseteq A$ l'insieme degli stati raggiungibili a partire da C attraverso A' , indicato con $\text{Der}(C, A')$, è il più piccolo insieme tale che:

- $C \subseteq \text{Der}(C, A')$;
- se $s \in \text{Der}(C, A')$ ed esiste $\alpha \in A'$ e $\mathcal{L} \in \sum_S$ tale che $s \xrightarrow{\alpha} \mathcal{L}$ allora $\{s' | \mathcal{L}(s') > 0\} \subseteq \text{Der}(C, A')$.

Definizione 3.4. Sia $\mathcal{R} = (S, A, \rightarrow)$ un RTS completamente stocastico, per $C \subseteq S$, la CTMC di C , quando si considerano soltanto le azioni nell'insieme finito $A' \subseteq A$, è definita come $\text{CTMC}[C, A'] \stackrel{\text{def}}{=} (\text{Der}(C, A'), \mathbf{R})$, dove $\forall s_1, s_2 \in \text{Der}(C, A')$: $\mathbf{R}[s_1, s_2] \stackrel{\text{def}}{=} \sum_{\alpha \in A'} \mathfrak{P}^\alpha(s_2)$ con $s_1 \xrightarrow{\alpha} (P)^\alpha$.

3.3 BIOKLAIM

BIOKLAIM è un'estensione di KLAIM appositamente creata per trattare modelli che descrivano sistemi biologici. Come in KLAIM i sistemi sono descritti come un insieme di nodi, ognuno dei quali identifica una parte del sistema descritto. In BIOKLAIM viene seguito l'approccio *molecole come tuple*, ovvero le tuple sono usate per rappresentare le molecole, mentre i processi identificano le reazioni chimiche possibili nel sistema in un particolare istante.

Per poter descrivere al meglio le reazioni chimiche, le azioni di KLAIM sono state arricchite per permettere il recupero multiplo di tuple con una singola operazione su di uno o più spazi delle tuple.

3.3.1 *Categorie sintattiche*

Si possono distinguere le seguenti categorie sintattiche:

- $\text{Val} = \{v, v', v_1, \dots\}$ è l'insieme dei valori (dati di base);
- $\text{LLoc} = \{l, l', l_1, \dots\}$ è l'insieme delle località logiche. Viene assunto che la località $\text{self} \in \text{LLoc}$.
- $\text{PLoc} = \{s, s', s_1, \dots\}$ è l'insieme delle località fisiche.
- $\text{Proc-var} = \{X, X', X_1, \dots\}$ è l'insieme delle variabili di processo.

Tutti gli insiemi sono numerabili e mutuamente disgiunti. Inoltre è presente l'insieme delle località, della forma $\text{Loc} = \text{LLoc} \cup \text{PLoc} = \{l, l', l_1, \dots\}$.

Per le sequenze viene adottata la notazione $\vec{\cdot}$, per esempio $\vec{v} = v_1, v_2, \dots, v_n$ è una sequenza appartenente a Val . Data una sequenza $\vec{s} = s_1, s_2, \dots, s_n$, $\{\vec{s}\}$ è l'insieme degli elementi della sequenza \vec{s} , ad esempio $\{\vec{s}\} = \{s_1, \dots, s_n\}$. Sequenze di un solo elemento e i singoletti sono indicati con l'elemento che contengono, ad esempio $\vec{s} = s'$ è indicato con s' e $\{s\}$ con s . La sequenza vuota è invece indicata con ϵ .

3.3.2 *Reti e processi*

I sistemi BIOKLAIM consistono in reti e processi. La rete più elementare è la rete nulla, indicata con \mathbf{o} . Una rete composta da un singolo nodo E con una località fisica s è indicata con $s :: E$. In generale, le reti sono espresse come più nodi composti in parallelo.

I nodi (E, E_1, \dots) sono processi in esecuzione in una località, dati rappresentati dalle tuple $\langle \vec{v} \rangle$ memorizzate nella località, oppure la composizione

di più nodi ($E_1|E_2$). I processi sono formati dal processo terminato **nil**, un insieme di azioni casuali ritardate e da costruttori algebrici standard, come l'azione prefissa, la scelta, la composizione parallela e l'istanziamento di un processo.

Quindi i processi sono costruiti seguendo questa grammatica:

$$N ::= \mathbf{o} \mid l :: E \mid N \mid N$$

$$E ::= P \mid \langle \vec{v} \rangle \mid E_1|E_2$$

$$P ::= \mathbf{nil} \mid (A, \lambda).E \mid P + P \mid P \mid P \mid X$$

Il processo $(A, \lambda).P$ esegue un'azione A con una durata casuale distribuita esponenzialmente con rate $\lambda \in \mathbb{R}_{>0}$. Da notare che l'esecuzione di un'azione può attivare nuovi elementi nella località dove il processo è in esecuzione.

3.3.3 Tuple e templates

Come detto in precedenza, i processi interagiscono l'uno con l'altro attraverso le tuple inserite in una specifica località. Una tupla è una sequenza di parametri attuali, ovvero valori, e viene recuperata tramite l'utilizzo di template e del pattern-matching.

Un template può essere un campo di una tupla (quindi un valore) oppure \star . I template hanno la seguente sintassi:

$$t ::= v \mid \star$$

Il paradigma di pattern-matching è molto semplice: un template corrisponde a una tupla se entrambi hanno lo stesso numero di campi ed i campi corrispondenti coincidono; due valori coincidono soltanto se sono indentici, mentre \star combacia con qualsiasi valore.

I template possono essere composti tramite l'operatore \otimes per selezionare più tuple. I pattern (π) e le tuple (T) sono definite come segue:

$$\pi ::= \vec{t} \mid \vec{t} \otimes \pi \qquad T ::= \vec{v} \mid \vec{v} \otimes T$$

Un pattern $\vec{t} \otimes \pi$ coincide con una sequenza $\vec{v} \otimes T$ se e soltanto se \vec{t} coincide con \vec{v} e π combacia con T .

3.3.4 Azioni

Un processo ha la possibilità di scrivere una tupla \vec{v} nella località ℓ attraverso l'azione **out**(\vec{v})@ ℓ . Con l'azione **in**(π)@ ℓ un processo può recuperare un'insieme di tuple che coincidono con il pattern π dalla località ℓ . L'azione **read**(π)@ ℓ è simile a **in**(π)@ ℓ tranne che le tuple recuperate non vengono eliminate dallo spazio delle tuple in ℓ . L'azione **eval**(P)@ ℓ crea un processo P nella località ℓ .

Ad ogni azione può essere associata una guardia. Questa è un'espressione booleana in grado di abilitare e disabilitare l'azione collegata. Se un'azione non ha guardia, questa verrà sempre considerata come una possibile opzione tra le azioni che, in quell'istante, saranno disponibili.

La struttura delle guardie rispetta la grammatica:

$$G ::= \text{true} \mid B \bowtie B \mid !G \mid G \wedge G$$

$$B ::= \langle \vec{v} \rangle @s \mid B \text{ bop } B \mid \text{uop } B \mid v$$

dove \bowtie corrisponde a uno degli operatori di confronto $\{<, >, \leq, \geq\}$, $v \in \mathbb{R}_{\geq 0}$ mentre bop e uop sono gli operatori binari e unari su valori in \mathbb{R} (ad esempio: bop = +, *, /, -, ... e uop = sin, cos, ln, log, ...).

Infine, le azioni sono costruite secondo la seguente grammatica:

$$A ::= \{G\} \text{ out}(\vec{v})@l \mid \{G\} \text{ in}(\pi)@l \mid \{G\} \text{ read}(\pi)@l \mid \{G\} \text{ eval}(P)@l$$

3.3.5 Specifiche

Una specifica BIOKLAIM S è una tripla $(\mathcal{E}, \Delta \vdash N)$ dove:

- \mathcal{E} è una funzione che associa a ogni nodo nella rete N un ambiente di allocazione;
- Δ è un insieme di definizioni di processi ($X \triangleq P$);
- N è una rete che descrive sia la struttura che il comportamento del sistema.

Per associare le località logiche alle località fisiche vengono usati ambienti di allocazione: formalmente un ambiente di allocazione ρ è una funzione totale da Loc a PLoc .

Si dice che una specifica BIOKLAIM $\mathcal{E}, \Delta \vdash N$ è *ben formata* se e soltanto se è corretta sintatticamente e:

- per ogni $s \in \text{PLoc}$, se $\mathcal{E}(s) = \rho$ allora:
 - $\forall s' \in \text{PLoc}: \rho(s') = s'$;
 - $\rho(\text{self}) = s$
- per ogni $X \triangleq P \in \Delta$, la variabile di processo X appare *guardata*, ovvero preceduta da un'azione, in P oppure come argomento di una **eval**;
- i processi usano soltanto località che esistono.

Da ora in poi verrà assunto che le specifiche siano ben formate, verranno omessi sia \mathcal{E} sia Δ dalla specifica, quando la loro definizione è chiara dal contesto. S sarà il nome dell'insieme delle specifiche BIOKLAIM.

3.4 SEMANTICA OPERAZIONALE DI BIOKLAIM

La semantica operativa delle specifiche BIOKLAIM è definita attraverso l'uso del RTS $\mathcal{R}_{bk} = (\text{Spec}, \Lambda, \mapsto)$; Spec è l'insieme delle specifiche BIOKLAIM mentre Λ è l'insieme delle etichette per le transizioni γ , definite come segue:

$$\begin{aligned} \gamma ::= & s_1 : \mathbf{o}(\vec{v})@s_2 \mid s_1 : \mathbf{e}(P)@s_2 \mid s_1 : \mathbf{i}(T)@\vec{d}s_2 \mid s_1 : \mathbf{r}(T)@\vec{d}s_2 \\ & \mid s_1 \triangleright T@s_2 \mid s_1 \blacktriangleright T@s_2 \mid s_1 \triangleleft T@s_2 \mid s_1 \blacktriangleleft T@s_2 \end{aligned}$$

dove $s_1 : \mathbf{o}(\vec{d})@s_2$ e $s_1 : \mathbf{e}(P)@s_2$ sono le azioni di output e di eval, $s_1 : \mathbf{i}(T)@s_2$ e $s_1 : \mathbf{r}(T)@\vec{d}s_2$ identificano le azioni di input e read recuperando le tuple in T. Tali etichette definiscono le interazioni interne.

Le etichette $s_1 \triangleright T@s_2$ e $s_1 \blacktriangleright T@s_2$ concretizzano la possibilità che un processo esegua un'azione di input o read. Queste azioni si sincronizzano con $s_1 \triangleleft T@s_2$ e $s_1 \blacktriangleleft T@s_2$, che descrivono la disponibilità delle tuple in T.

La relazione di transizione \mapsto è definita in termini della relazione $\rightarrow_{\mathcal{F}_N}$ come segue:

$$\frac{N \xrightarrow{\alpha}_{\mathcal{F}_N} P}{N \xmapsto{\alpha} P}$$

Tale relazione, formalmente definita in tabella 1, è parametrizzata rispetto alla funzione \mathcal{F} utilizzata per valutare le guardie presenti nei processi. Questa funzione è definita induttivamente sulla sintassi delle reti come segue:

$$\begin{aligned} \mathcal{F}_0(\langle \vec{v} \rangle@s) &= 0 \\ \mathcal{F}_{\langle \vec{v} \rangle@s}(\langle \vec{v} \rangle@s) &= 1 \\ \mathcal{F}_{N_1|N_2}(\langle \vec{v} \rangle@s) &= \mathcal{F}_{N_1}(\langle \vec{v} \rangle@s) + \mathcal{F}_{N_2}(\langle \vec{v} \rangle@s) \end{aligned}$$

La funzione risulta utile per valutare, data una rete N , la soddisfacibilità di una guardia, che viene definita seguendo le regole della grammatica definita nella precedente sezione, come segue:

$$\mathcal{F}_N \models \text{true}$$

$$\mathcal{F}_N \models B_1 \bowtie B_2 \iff v[B_1]_{\mathcal{F}_N} \bowtie v[B_2]_{\mathcal{F}_N}$$

$$\mathcal{F}_N \models !G \iff \mathcal{F}_N \not\models G$$

$$\mathcal{F}_N \models G_1 \wedge G_2 \iff \mathcal{F}_N \models G_1 \wedge \mathcal{F}_N \models G_2$$

Il teorema successivo garantisce la trattabilità della semantica proposta (riportata nelle tabelle 1, 2 e 3):

Teorema 3.4.1. \mathcal{R}_{bk} è un (\sum_{Spec}) RTS completamente stocastico e image-finite.

3.5 SPECIFICARE PROPRIETÀ PER IL GLOBAL COMPUTING

Per fare valutazioni riguardanti alcuni aspetti di un sistema viene in aiuto il *model checking*, tecnica completamente automatica per la prova di proprietà[42]: le proprietà sono formulate come formule di una logica temporale. Per le CTMC esiste CSL (Continuous Stochastic Logic [43, 44]), un'estensione stocastica di CTL che permette di specificare proprietà probabilistiche del tipo: “la probabilità di raggiungere uno stato entro t unità di tempo, visitando soltanto stati concessi, è di almeno 0,92”.

Un vantaggio di questo approccio è di essere completamente formale: le descrizioni informali sono soggette ad essere difficilmente misurabili e possono essere anche mal interpretate, portando la ricerca di proprietà in una situazione di errore. In più, con l'aiuto di un model checker stocastico, è possibile verificare automaticamente se un requisito sia soddisfatto dal sistema specificato ed è importante sottolineare che attraverso tali

strumenti non si ricava solo una risposta affermativa o negativa, ma anche valori di probabilità.

In [45] è stata proposta una logica, MoSL, che permette di utilizzare come riferimento per i sistemi una variante stocastica di KLAIM. Il punto di partenza è una variante basata su azioni di CSL, proposta per la prima volta in [46], che sia adatta alle azioni di KLAIM. Le caratteristiche che contraddistinguono questa logica nei confronti di CSL sono:

- proposizioni atomiche possono riferirsi a località dove il dato e il processo risiedono;
- le azioni sono generalizzate a *specificatori di azioni* che si comportano come pattern per caratterizzare insiemi di azioni.

MoSL fornisce, inoltre, operatori per descrivere proprietà risultanti dalla produzione e dal consumo di risorse.

Dato che le entità base di BioKLAIM sono le azioni, la logica è sia *action-based* che *state-based*, in opposizione alle logiche solamente *state-based*, come LTL e CTL, e a quelle soltanto *action-based*, come ACTL. Per essere in grado di fare riferimento al carattere distribuito dei sistemi, la logica ha alcuni costrutti relativi alla natura spaziale del sistema, ispirati a [47].

Il risultato è una logica con le seguenti caratteristiche:

- è una logica temporale che permette la descrizione dell'evoluzione dinamica del sistema;
- è sia *action-based* che *state-based*;
- è una logica *real-time*;
- è una logica probabilistica che permette di esprimere non soltanto proprietà funzionali, ma anche proprietà legate a performance e affidabilità;

- è una logica spaziale che si riferisce alla struttura spaziale del sistema specificato.

3.5.1 Sintassi

Formule di stato base

Le formule di stato base sono costruite tramite gli operatori di *consumo* (\rightarrow) e di *produzione* (\leftarrow), che permettono di formalizzare proprietà riguardanti la presenza di risorse (tuple e processi) e reazioni del sistema al posizionamento di nuove risorse in uno stato.

Una formula di consumo:

$$Q@i \rightarrow \Phi$$

vale per una rete ogni volta che esiste un processo Q in esecuzione in una località i , e il resto della rete soddisfa Φ .

La formula:

$$\langle \vec{V} \rangle @i \rightarrow \Phi$$

vale ogni volta che una tupla \vec{v} corrispondente a \vec{V} risiede nella località i e il resto della rete soddisfa Φ . Il risultato della sostituzione del pattern-matching viene usato per valutare Φ . Un uso tipico dell'operatore di consumo è contare. La proprietà: "due tuple, corrispondenti a \vec{v} , sono presenti in i " è formalizzata come:

$$\langle \vec{v} \rangle @i \rightarrow \langle \vec{v} \rangle @i \rightarrow \text{true}.$$

In modo del tutto simile è possibile garantire anche "ci sono almeno tre istanze del processo Q , due delle quali nella località i_1 e la terza nella località i_2 ":

$$Q@i_1 \rightarrow Q@i_2 \rightarrow Q@i_1 \rightarrow \text{true}.$$

Una formula di produzione:

$$Q@i \leftarrow \Phi$$

vale se la rete soddisfa Φ ogni qual volta il processo Q venga eseguito in una località i .

La formula:

$$\langle \vec{v} \rangle @i \leftarrow \Phi$$

vale se la rete soddisfa Φ ogni qual volta la tupla \vec{v} risieda nella località i . Le formule di produzione sono molto utili per le specifiche del contesto del sistema. Per esempio, data una rete N , ci può essere interesse nello studiare la reazione di un sistema se un determinato processo Q viene eseguito in una località i di N , in particolare potrebbe essere provato che la rete può continuare a soddisfare una determinata formula Φ quando Q è stato eseguito: tale richiesta può essere formulata controllando se N soddisfa $Q@i \leftarrow \Phi$. La proprietà viene direttamente controllata sulla rete N : in altre parole, non è necessario andare a modificare il modello che viene utilizzato per provare la formula, nel caso specifico andando ad aggiungere Q al sistema.

Le formule di produzione vengono usate per specificare come il sistema reagisce a nuove richieste di servizio. Per esempio $\langle S_2 \rangle @A \leftarrow \Phi$ vale quando Φ è soddisfatta dopo che è stata ricevuta una richiesta di servizio di tipo S_2 .

In breve la grammatica per le formule base di stato è:

$$\mathfrak{N} ::= Q@i \rightarrow \Phi \mid \langle \vec{V} \rangle @i \rightarrow \Phi \mid Q@i \leftarrow \Phi \mid \langle \vec{v} \rangle @i \rightarrow \Phi$$

Inoltre vengono adottate alcune abbreviazioni come $Q@i$ per $Q@i \rightarrow \text{true}$ e $\langle \vec{V} \rangle @i$ per $\langle \vec{V} \rangle @i \rightarrow \text{true}$.

Specificatori di azioni e insiemi di azioni

È necessario distinguere due classi di formule: formule di stato ($\Phi, \Phi', \Phi_1, \dots$) e formule di cammino ($\varphi, \varphi', \varphi_1, \dots$). Le azioni sono specificate dall'insieme degli specificatori di azioni: questi ultimi si attengono alla seguente grammatica (ξ_i è uno specificatore):

$$\Delta ::= T \mid \{\} \mid \{\xi_1, \dots, \xi_n\}.$$

T sta per “qualsiasi insieme” e può essere usato quando non è imposto alcun requisito sulle azioni. Un insieme di specificatori di azioni è soddisfatto da un'azione se quest'ultima soddisfa almeno uno degli elementi dell'insieme. Gli specificatori di azioni sono dei template per le azioni. Hanno la forma:

$$\xi ::= g : O(\vec{F}, g) \mid g : I(\vec{F}, g) \mid g : R(\vec{F}, g) \mid g : E(Q, g)$$

dove g è una località. Ad esempio l'azione $\text{loc}_1 : O(\text{GO}, \text{loc}_2)$ è soddisfatta soltanto dall'azione compiuta da un processo nella località loc_1 che scrive la tupla GO nella località loc_2 .

Formule di cammino

Per le formule di cammino la sintassi standard è l'*Until* di CTL: $\Phi \mathcal{U} \Psi$. Per riuscire a riferirsi anche alle azioni eseguite lungo un cammino viene impiegata una variante.

A tale scopo, l'operatore until viene parametrizzato rispetto a due insiemi di azioni. Un cammino soddisfa $\Phi \Delta \mathcal{U}_\Omega \Psi$ quando uno stato soddisfacente Ψ è raggiungibile attraverso un cammino composto da stati che soddisfano Φ e, durante il passaggio degli stati, che lo compongono le azioni che il processo compie soddisfano Δ ; successivamente il processo entra in uno stato soddisfacente Ψ tramite un'azione che soddisfa Ω .

Infine viene aggiunto anche un limite temporale alle formule di cammino attraverso il parametro t . In base alla definizione di until data in precedenza, adesso viene imposto che lo stato Ψ dovrebbe essere raggiunto entro t unità di tempo. Se $t = \infty$ il limite temporale è trascurabile. Un cammino soddisfa $\Phi \Delta \mathcal{U}^{<t} \Psi$ se lo stato iniziale soddisfa Ψ (al tempo 0) oppure viene raggiunto nel cammino, entro il tempo t attraverso stati soddisfacenti Φ , uno stato soddisfacente Ψ e, mentre il processo si muove attraverso gli stati soddisfacenti Φ , le azioni che vengono eseguite soddisfano Δ .

La sintassi delle formule di cammino è la seguente:

$$\varphi ::= \Phi \Delta \mathcal{U}_{\Omega}^{<t} \Psi \mid \Phi \Delta \mathcal{U}^{<t} \Psi.$$

$\Phi \Delta \mathcal{U}^{<t} \Psi$ non è equivalente a $\Phi \Delta \mathcal{U}_{\top}^{<t} \Psi$ poiché la seconda formula assume che almeno una transizione sia compiuta per raggiungere lo stato soddisfacente Ψ , mentre per la prima non è richiesto.

Un esempio di formula di cammino è:

$$\text{true } \top \mathcal{U}_{loc_1:O(GO, loc_2)}^{<\infty} \text{true}$$

questa formula afferma che dopo un po' di tempo la tupla verrà posta nella località loc_2 dalla località loc_1 . La formula $\text{true } \top \mathcal{U}_{loc_1:O(GO, loc_2)}^{<t} \text{true}$ richiede che l'azione venga completata entro un tempo t ed è differente dalla formula $\text{true } \top \mathcal{U}_{\top}^{<t} \langle GO \rangle @ loc_1$ che afferma che la tupla GO deve essere presente nella località loc_1 entro t dopo almeno una transizione.

Formule di stato

Le proprietà inerenti gli stati sono formulate secondo le formule di stato. Esistono tre categorie di formule di questo tipo: la prima include le formule presentate in 3.5.1, la seconda le istruzioni relative alla probabilità che i cammini soddisfino una proprietà e la terza categoria include le

proprietà "steady-state". In generale una formula può essere composta da sottoformule di diverse categorie. Sia φ una proprietà imposta sui cammini, lo stato s soddisfa la proprietà $\mathcal{P}_{\bowtie p}(\varphi)$ ogni volta che la probabilità per tutti i cammini che partono da s che soddisfano φ soddisfa il limite $\bowtie p$. \bowtie è un operatore binario di confronto ($<, >, \leq, \geq$) e $p \in [0, 1]$. Per esempio la proprietà $\mathcal{P}_{>0,99}(\text{legal } \top \mathcal{U}_T^{\leq 31,2} \text{goal})$ significa che la probabilità di raggiungere uno stato goal entro 31,2 unità di tempo, soltanto attraverso il cammino di stati legal e con al massimo una transizione, supera 0,99.

Le proprietà "steady-state" si riferiscono al sistema che ha raggiunto l'equilibrio. Uno stato s soddisfa $\mathcal{S}_{\bowtie p}(\Phi)$ se, partendo da s , la probabilità di raggiungere uno stato che soddisfi Φ a regime $\bowtie p$. Per esempio la formula:

$$\mathcal{S}_{\geq 0,2}(\langle \text{GO} \rangle @ \text{loc}_1)$$

afferma che, a steady-state, la probabilità di trovare GO nella località loc_1 è almeno 0,2.

Le formule di stato sono costruite secondo la grammatica:

$$\Phi ::= \text{true} \mid \text{false} \mid \neg \Phi \mid \Phi \vee \Phi \mid \mathcal{P}_{\bowtie p}(\varphi) \mid \mathcal{S}_{\bowtie p}(\Phi)$$

3.5.2 Semantica

I cammini giocano un ruolo centrale nella definizione formale della semantica di MoSL^+ . La definizione di cammino è:

Definizione 3.5. Sia $\mathbb{A} = (S, \text{ACT}, \mapsto)$ una CTMC con azioni etichettate (AMC). Un cammino π di \mathbb{A} è una sequenza

$$s_0(\gamma_0, t_0) s_1(\gamma_1, t_1) \dots$$

tale che valgano le condizioni:

- $s_j \in S$, $t_j \in \mathbb{R}_{>0}$ e $s_j \xrightarrow{\gamma_j, \lambda} s_{j+1}$ per qualche $\lambda > 0, \forall j \geq 0$;
- π è minimale, cioè è infinito oppure esiste un numero naturale l tale che s_l è absorbing (non esistono s, γ) e λ tale che $s_l \xrightarrow{\gamma_j, \lambda} s$.

Vengono introdotti cinque operatori, riportati in tabella 4. $\text{len}(\pi)$ restituisce la lunghezza di un cammino come il numero di transizioni che contiene; $\text{st}(\pi, j)$ restituisce il j -esimo stato del cammino π ; $\text{ac}(\pi, j)$ restituisce l'etichetta della j -esima transizione nel cammino; $\text{dl}(\pi, j)$ restituisce il ritardo attuale della j -esima transizione, se esiste, nel cammino. $\pi(t)$ restituisce lo stato raggiunto nel cammino dopo che sono passate t unità di tempo. Per ogni stato s di una AMC, \mathbb{A} viene definito $\text{Paths}_{\mathbb{A}}(s)$ che risulta essere l'insieme di tutti i cammini $s_0(\gamma_0, t_0)s_1(\gamma_1, t_1) \dots$ di \mathbb{A} con $s_0 = s$.

Definizione 3.6. Una specifica BIOKLAIM (β_0, N_0, \vec{D}) soddisfa una formula di stato Φ , scritto $(\beta_0, N_0, \vec{D}) \models_{\text{BK}} \Phi$ se e solo se $s_0 \models_{\mathbb{A}} \Phi$, con $\models_{\mathbb{A}}$ definita in tabella 5, dove s_0 è lo stato di $\mathbb{A} = \text{AMC}(\beta_0, N_0, \vec{D})$ corrispondente allo stato iniziale.

Formule di stato

La relazione di soddisfacibilità per le formule di stato sfrutta il pattern matching. Per questo fine viene utilizzata la funzione $\text{match}()$ definita in [42]. Θ risulta essere una sostituzione $[d_1/w_1, \dots, d_n/w_n]$ con $w_i \neq w_j$ per $i \neq j$ che sostituisce w_j con d_j per $0 \leq j \leq n$, con \square si esprime la sostituzione vuota.

La funzione $\text{match}()$ è una funzione parziale che restituisce la sostituzione generata dal combaciare del suo primo argomento, un template, con il secondo, una tupla.

```

bio
localities: ...;
environments: ...;
messages: ...;
agents: #un commento
...
system:
...
end

```

Listing 3.1: Struttura di una specifica BioKLAIM.

Insiemi di specificatori di azioni

La tabella 6 mostra la definizione di $\models_{\mathbb{A}}$.

Formule di cammino

La definizione della relazione di soddisfacibilità per le formule di cammino è presentata in tabella 7. Da notare che nella definizione di $\Phi_{\Delta} \mathcal{U}_{\Omega}^{\leq t} \Psi$ l'unica sostituzione che viene usata per rimpiazzare le variabili con i valori è quella generata dal matching dell'azione dell'ultima transizione prima dello stato Ψ e Ω (Θ_{k-1}). Ciò che precede la suddetta transizione viene scartato; in questo modo non c'è la necessità di contare o utilizzare una pila.

3.6 L'AMBIENTE bksim

È stato creato un tool capace di interagire con i sistemi specificati tramite BioKLAIM: bksim, che permette ad un utente, data una specifica,

```
<nome_associato_al_rate_1> -> <valore_rate_1>;  
...  
<nome_associato_al_rate_n> -> <valore_rate_n>;
```

Listing 3.2: Struttura delle corrispondenze nome associato/rate in BIOKLAIM.

di poterla eseguire, simulare e verificare proprietà stocastiche su di essa.

3.6.1 *Dichiarazione di una specifica e sua esecuzione*

Il tool fornisce un ambiente di esecuzione interattivo di una specifica: per caricarla è necessario eseguire il comando `load '<file_specifica>'`. Inoltre è possibile raggruppare i rate delle azioni presenti nell'intera specifica, mettendoli in corrispondenza con dei nomi, in un file secondario che può essere caricato a parte mediante `load mapping '<file_mapping>'`. Avere la possibilità di creare dei file di corrispondenze, che hanno una struttura mostrata nel listing 3.2, consente di poter realizzare un'architettura intercambiabile che può tornare utile in situazioni di ricerche riguardanti mutazioni che vanno a incidere sui rate delle reazioni chimiche. Completato il caricamento della specifica, con il comando `run` viene fatta partire l'esecuzione.

Una specifica BIOKLAIM, la struttura che presenta è riportata nel listing 3.1, è costituita da cinque sezioni:

1. `localities` contiene le dichiarazioni delle località utilizzate nel sistema (sia fisiche che logiche);
2. in `environments` sono definiti gli ambienti di allocazione;
3. `messages` è la lista delle tuple che appaiono nel sistema;


```
physical nome_loc_fisica; #dichiarazione località fisica
logical nome_loc_logica; #dichiarazione località logica
```

Listing 3.3: Dichiarazioni di località in BIOKLAIM.

4. `agents` contiene tutte le dichiarazioni degli agenti;
5. `system` presenta la rete di nodi con la configurazione iniziale del sistema specificato.

I commenti sono caratterizzati dall'essere preceduti dal simbolo “#” e si estendono fino alla fine della riga, questi sono permessi ovunque.

Le parole chiave di BIOKLAIM, che non possono essere utilizzate in altro modo se non come verrà mostrato, sono:

```
agents - be - let - localities - environments - system - bio
messages - end - true - false - physical - logical - in
```

Località

Come detto la prima sezione di una specifica è usata per la dichiarazione delle località fisiche e logiche usate nel sistema. La dichiarazione dei due tipi di località avviene come mostrato nel listing 3.3.

Ambienti di allocazione

In BIOKLAIM ogni nodo è fornito di un ambiente di allocazione che fa corrispondere le località logiche alle località fisiche. La dichiarazione avviene come mostrato nel listing 3.4. Un ambiente non può essere vuoto e possono essere fatte corrispondere soltanto località logiche dichiarate con località fisiche a loro volta dichiarate.

```

nome_loc_fisica:
    [nome_loc_logica_1 -> nome_loc_fisica_1 ,
    ...
    nome_loc_logica_n -> nome_loc_fisica_n] ;

```

Listing 3.4: Ambiente di allocazione in BIOKLAIM.

```

messages: nome_tupla_1, nome_tupla_2, ..., nome_tupla_n;

```

Listing 3.5: Dichiarazione tuple in BIOKLAIM.

Tuple

Le tuple che partecipano alle interazioni del sistema vengono dichiarate come mostrato nel listing 3.5

Processi

La quarta parte di una specifica BIOKLAIM ha come compito quello di dichiarare gli agenti che appariranno nel sistema considerato. Ogni agente è univocamente identificato da un nome, la sua dichiarazione segue lo schema mostrato nel listing 3.6 e il suo body è composto da uno o più processi.

Un altro aspetto importante nelle specifiche è la parola chiave `bio`, la quale sta a significare che i rate delle azioni presenti nel sistema saranno

```

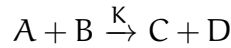
let <nome_agente> be
    <body>
end

```

Listing 3.6: Dichiarazione agenti in BIOKLAIM.

proporzionali al numero di tuple che soddisfano il template dell'azione.

La reazione



viene associata al processo BIOKLAIM

```
P -> in(A | B)@self:K.[C] | [D] | P
```

Una reazione chimica è rappresentata da un processo che recupera, con un rate di reazione calcolato come $K \cdot |A| \cdot |B|^4$, le tuple corrispondenti agli elementi interessati dalla stessa e produce elementi. Questo tipo di calcolo dei rate corrisponde alla legge di azione di massa [2.2.1].

3.6.2 Simulazione

bksim permette anche di eseguire simulazioni, dopo che si è caricato il file della specifica e l'eventuale file di corrispondenze. Il risultato di una simulazione è un insieme di file e ognuno di questi descrive come il numero di una determinata tupla varia nel tempo. Per indicare al tool quali tuple dovranno essere registrate durante la simulazione, è necessario caricare il file di simulazione tramite il comando `load experiments '<nome_file_esperimenti>'`. Questo file ha la struttura presentata nel listing 3.7. Il campo `<etichetta>` presenta un nome che identifica il particolare dato che viene registrato, `Expr` invece è una funzione che mette in relazione, attraverso gli operatori binari e unari su \mathbb{R} , una o più tuple del sistema. Ogni tupla presente nella funzione `Expr` ha la forma: `[tupla_del_sistema]@<una_località>. <tupla_del_sistema>` è una tupla presente nel modello (nel caso in cui la tupla non sia dichiarata viene segnalato l'errore), mentre `<una_località>` può essere sia il

⁴ $|A|$ (come $|B|$) corrisponde alla cardinalità di A (B).

```
<etichetta> : Expr
```

Listing 3.7: Dichiarazione dati simulazione in BIOKLAIM.

nome di una località fisica che il carattere *. Nel primo caso vengono contate le tuple presenti nella località specificata, nel secondo le tuple presenti in tutte le località.

Come ultimi parametri necessari a configurare una simulazione, l'utente deve specificare il limite di unità di tempo per il quale la simulazione termina, l'intervallo di campionamento del sistema e il numero di iterazioni che devono essere compiute. Per impostarli basta servirsi di `set deadline = <valore>` per il limite temporale, `set dt = <valore>` per il campionamento e `set iterations = <valore>` per il numero di iterazioni.

Infine per poter lanciare la simulazione basterà scrivere `simulate` nella console di `bksim`.

3.6.3 *Verifica proprietà*

Il tool `bksim` permette, dopo aver caricato una specifica, di provare alcune proprietà su di un modello. Questa caratteristica fa sì di poter affermare l'emergere di determinati aspetti, come la presenza di anomalie nel comportamento del modello studiato, e inoltre dà la possibilità di comprendere che relazione c'è tra dati conosciuti e l'andamento di determinate reazioni.

Non si ha soltanto a che fare con formule per garantire la dependability: si affiancano a queste anche formule che possano mostrare che una particolare situazione (come la fine di una serie di reazioni) accade, ad

esempio, quando risulta mancare una determinata sostanza.

bksim consente l'uso di questo strumento per mezzo di tre comandi:

`plot '<formula_di_cammino>' with [<inizio>:<fine>:<dt>] on '<nome_file>':`

disegna l'andamento della probabilità di una formula di cammino, che segue la sintassi e la semantica presentata in 3.5. La verifica parte dall'istante <inizio> fino a <fine>, muovendosi ad intervalli <dt> e riportando i risultati sul file <nome_file>. Nel caso in cui non venga specificato il nome del file, ne viene creato uno con nome "output.dat".

`prob '<formula di cammino>' with [<parametri>]:`

restituisce la probabilità di una formula di cammino. Opzionalmente viene permesso di specificare dei parametri per la procedura di valutazione della formula.

`check '<formula>' with [<parametri>]:`

restituisce una valutazione netta (vero/falso) della formula passata. Anche qui c'è la possibilità di specificare dei parametri.

3.7 JSAM

Come strumento di ausilio nel processo di stesura, verifica e simulazione, che viene affrontato durante una ricerca, può essere impiegato JSAM, 14, un'applicazione Java che raccoglie, in un'interfaccia utente, tutte le funzioni che BIOKLAIM fornisce. Per tutte le operazioni JSAM si appoggia a bksim, presentato in 3.6.

Il programma, appena lanciato, mostra all'utente una serie di schermate contenute dentro una scrivania: ognuna di queste schermate ha un ruolo ben stabilito.

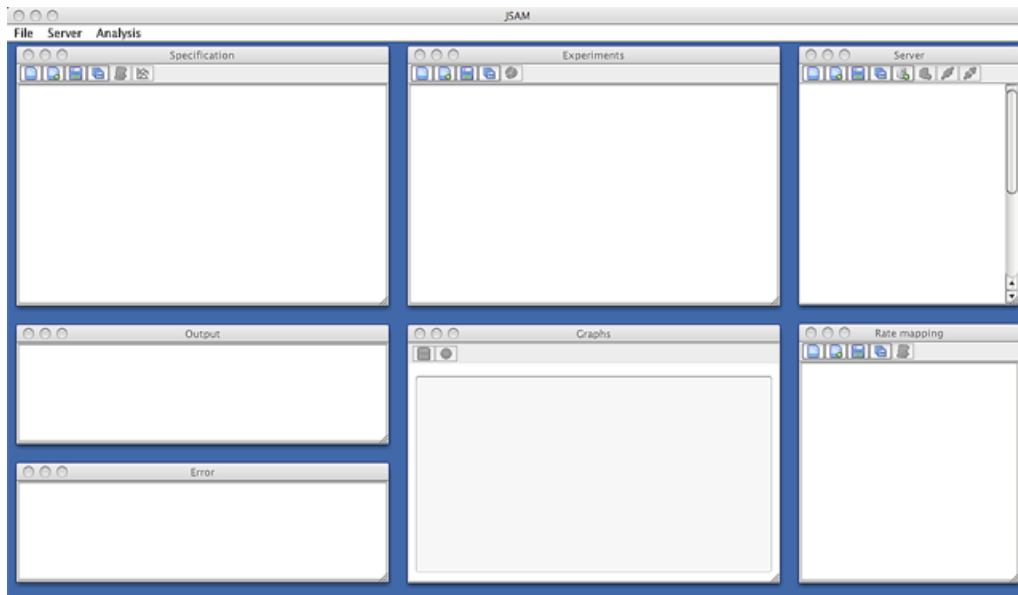


Figura 14: Schermata di come si presenta JSAM.

- In Server l'utente inserisce l'indirizzo (di rete o locale) di più macchine, dove è stato lanciato il tool bksim con il parametro `-net` e l'eventuale `-port <numero_porta>` per modificare la porta di ascolto, per poi utilizzarle per ridistribuire il carico computazionale delle operazioni svolte.
- In Specification, Experiments e Rate mapping vengono caricate rispettivamente la specifica, i dati da registrare durante le simulazioni e l'eventuale file di mapping su cui lavorare.
- Output ed Error sono delle console dove vengono stampati i messaggi inviati dai server, nel primo caso a esito positivo, nel secondo se si è presentato qualche problema.
- Graphs riporta il grafico di ciò che viene analizzato.

Dopo aver configurato i server, l'utente può analizzare una specifica BIOKLAIM. Innanzitutto devono essere caricati la specifica da studiare,

gli esperimenti e il file di mapping a lei associati (se ce ne sono), poi, collegando i server presenti nella lista, la specifica può essere inviata ai vari server, dove sarà valutata. A questo punto all'utente è permesso di preparare una serie di simulazioni o di verifiche di proprietà e in seguito all'inizio dell'esecuzione degli esperimenti i risultati delle operazioni saranno mostrati, sotto forma di grafico, nella finestra Graphs. In seguito sarà possibile salvare i dati per poi utilizzarli all'esterno del programma.

$$\begin{array}{c}
\frac{\mathcal{E}(s_1)(\ell) = s_2 \quad \mathcal{F}_N \models G}{s_1 :: (\{G\} \text{out}(\vec{d})@ \ell, \lambda).E \xrightarrow{s_1 \cdot \mathbf{o}(\vec{d})@s_2}_{\mathcal{F}_N} [s_1 :: E \parallel s_2 :: \langle \vec{d} \rangle \mapsto \lambda]} \text{(OUT-G-TRUE)} \\
\\
\frac{\mathcal{E}(s_1)(\ell) = s_2 \quad \mathcal{F}_N \not\models G}{s_1 :: (\{G\} \text{out}(\vec{d})@ \ell, \lambda).E \xrightarrow{s_1 \cdot \mathbf{o}(\vec{d})@s_2}_{\mathcal{F}_N} \emptyset} \text{(OUT-G-FALSE)} \\
\\
\frac{\mathcal{E}(s_1)(\ell) = s_2 \quad \mathcal{F}_N \models G}{s_1 :: (\{G\} \text{eval}(Q)@ \ell, \lambda).P \xrightarrow{s_1 \cdot \mathbf{e}(Q)@s_2}_{\mathcal{F}_N} [s_1 :: P \parallel s_2 :: Q \mapsto \lambda]} \text{(EVAL-G-TRUE)} \\
\\
\frac{\mathcal{E}(s_1)(\ell) = s_2 \quad \mathcal{F}_N \not\models G}{s_1 :: (\{G\} \text{eval}(Q)@ \ell, \lambda).P \xrightarrow{s_1 \cdot \mathbf{e}(Q)@s_2}_{\mathcal{F}_N} \emptyset} \text{(EVAL-G-FALSE)} \\
\\
\frac{\mathcal{E}(s_1)(\ell) = s_2 \quad \text{match}(\pi, T) \quad \mathcal{F}_N \models G}{s_1 :: (\{G\} \text{in}(\pi)@ \ell, \lambda).P \xrightarrow{s_1 \triangleright T@s_2}_{\mathcal{F}_N} [s_1 :: P \mapsto \lambda]} \text{(P-IN-G-TRUE)} \\
\\
\frac{\mathcal{E}(s_1)(\ell) = s_2 \quad \text{match}(\pi, T) \quad \mathcal{F}_N \not\models G}{s_1 :: (\{G\} \text{in}(\pi)@ \ell, \lambda).P \xrightarrow{s_1 \triangleright T@s_2}_{\mathcal{F}_N} \emptyset} \text{(P-IN-G-FALSE)} \\
\\
\frac{\mathcal{E}(s_1)(\ell) = s_2 \quad \text{match}(\pi, T) \quad \mathcal{F}_N \models G}{s_1 :: (\{G\} \text{read}(T)@ \ell, \lambda).P \xrightarrow{s_1 \blacktriangleright T@s_2}_{\mathcal{F}_N} [s_1 :: P \mapsto \lambda]} \text{(P-READ-G-TRUE)} \\
\\
\frac{\mathcal{E}(s_1)(\ell) = s_2 \quad \text{match}(\pi, T) \quad \mathcal{F}_N \not\models G}{s_1 :: (\{G\} \text{read}(T)@ \ell, \lambda).P \xrightarrow{s_1 \blacktriangleright T@s_2}_{\mathcal{F}_N} \emptyset} \text{(P-READ-G-FALSE)} \\
\\
\frac{}{s_2 :: \langle \vec{v} \rangle \xrightarrow{s_1 \triangleleft \vec{v}@s_2}_{\mathcal{F}_N} [\mathbf{o} \mapsto 1]} \text{(T-IN)} \qquad \frac{}{s_2 :: \langle \vec{v} \rangle \xrightarrow{s_1 \blacktriangleleft \vec{v}@s_2}_{\mathcal{F}_N} [s_2 :: \langle v \rangle \mapsto 1]} \text{(T-READ)} \\
\\
\frac{\begin{array}{c} N_1 \xrightarrow{s_1 \triangleleft \vec{v}@s_2} \mathcal{P}^1 \quad N_1 \xrightarrow{s_1 \triangleleft T@s_2} \mathcal{Q}^2 \quad N_1 \xrightarrow{s_1 \triangleleft \vec{v} \otimes T@s_2} \mathcal{P}^3 \\ N_2 \xrightarrow{s_1 \triangleleft \vec{v}@s_2} \mathcal{Q}^1 \quad N_2 \xrightarrow{s_1 \triangleleft T@s_2} \mathcal{Q}^2 \quad N_2 \xrightarrow{s_1 \triangleleft \vec{v} \otimes T@s_2} \mathcal{Q}^3 \end{array}}{N_1 \parallel N_2 \xrightarrow{s_1 \triangleleft \vec{v} \otimes T@s_2} \mathcal{P}^1 \parallel \mathcal{Q}^2 + \mathcal{P}^2 \parallel \mathcal{Q}^1 + \mathcal{P}^3 \parallel N_2 + N_1 \parallel \mathcal{Q}^3} \text{(T-IN2)} \\
\\
\frac{\begin{array}{c} N_1 \xrightarrow{s_1 \triangleleft \vec{v}@s_2} \mathcal{P}^1 \quad N_1 \xrightarrow{s_1 \triangleleft T@s_2} \mathcal{Q}^2 \quad N_1 \xrightarrow{s_1 \triangleleft \vec{v} \otimes T@s_2} \mathcal{P}^3 \\ N_2 \xrightarrow{s_1 \triangleleft \vec{v}@s_2} \mathcal{Q}^1 \quad N_2 \xrightarrow{s_1 \triangleleft T@s_2} \mathcal{Q}^2 \quad N_2 \xrightarrow{s_1 \triangleleft \vec{v} \otimes T@s_2} \mathcal{Q}^3 \end{array}}{N_1 \parallel N_2 \xrightarrow{s_1 \triangleleft \vec{v} \otimes T@s_2} \mathcal{P}^1 \parallel \mathcal{Q}^2 + \mathcal{P}^2 \parallel \mathcal{Q}^1 + \mathcal{P}^3 \parallel N_2 + N_1 \parallel \mathcal{Q}^3} \text{(T-READ2)} \\
\\
\frac{l :: P \xrightarrow{\gamma} \mathcal{P} \quad l :: Q \xrightarrow{\gamma} \mathcal{Q}}{l :: P + Q \xrightarrow{\gamma} \mathcal{P} + \mathcal{Q}} \text{(PLUS)} \qquad \frac{l :: E_1 \parallel l :: E_2 \xrightarrow{\gamma} \mathcal{P}}{l :: E_1 | E_2 \xrightarrow{\gamma} \mathcal{P}} \text{(P-PAR)} \\
\\
\frac{N_1 \xrightarrow{\gamma} \mathcal{P} \quad N_2 \xrightarrow{\gamma} \mathcal{Q} \quad \gamma \notin \{s_1 : \mathbf{i}(T)@s_2, s_1 : \mathbf{r}(T)@s_2, s_1 \triangleleft \vec{v} \otimes T@s_2, s_1 \blacktriangleleft \vec{v} \otimes T@s_2\}}{N_1 \parallel N_2 \xrightarrow{\gamma} \mathcal{P} \parallel N_2 + N_1 \parallel \mathcal{Q}} \text{(N-PAR)}
\end{array}$$

Tabella 1: Semantica operativa di BIOKLAIM (Tabella 1)

$$\begin{array}{c}
\frac{
\begin{array}{c}
N_1 \xrightarrow{s_1:\mathbf{i}(\mathbf{T})@s_2} \mathcal{P}^s \quad N_1 \xrightarrow{s_1 \triangleright \mathbf{T}@s_2} \mathcal{P}^i \quad N_1 \xrightarrow{s_1 \triangleleft \vec{\mathbf{d}}@s_2} \mathcal{P}^o \\
N_2 \xrightarrow{s_1:\mathbf{i}(\mathbf{T})@s_2} \mathcal{Q}^s \quad N_2 \xrightarrow{s_1 \triangleright \mathbf{T}@s_2} \mathcal{Q}^i \quad N_2 \xrightarrow{s_1 \triangleleft \vec{\mathbf{d}}@s_2} \mathcal{Q}^o
\end{array}
}{N_1 \parallel N_2 \xrightarrow{s_1:\mathbf{i}(\mathbf{T})@s_2} \mathcal{P}^s \parallel N_2 + N_1 \parallel \mathcal{Q}^s + \mathcal{P}^i \parallel \mathcal{Q}^o + \mathcal{P}^o \parallel \mathcal{Q}^i} \text{ (IN)} \\
\\
\frac{
\begin{array}{c}
N_1 \xrightarrow{s_1:\mathbf{r}(\mathbf{T})@s_2} \mathcal{P}^s \quad N_1 \xrightarrow{s_1 \blacktriangleright \mathbf{T}@s_2} \mathcal{P}^i \quad N_1 \xrightarrow{s_1 \blacktriangleleft \mathbf{T}@s_2} \mathcal{P}^o \\
N_2 \xrightarrow{s_1:\mathbf{r}(\mathbf{T})@s_2} \mathcal{Q}^s \quad N_2 \xrightarrow{s_1 \blacktriangleright \mathbf{T}@s_2} \mathcal{Q}^i \quad N_2 \xrightarrow{s_1 \blacktriangleleft \mathbf{T}@s_2} \mathcal{Q}^o
\end{array}
}{N_1 \parallel N_2 \xrightarrow{s_1:\mathbf{r}(\mathbf{T})@s_2} \mathcal{P}^s \parallel N_2 + N_1 \parallel \mathcal{Q}^s + \mathcal{P}^i | \mathcal{Q}^o + \mathcal{P}^o | \mathcal{Q}^i} \text{ (READ)} \\
\\
\frac{X \stackrel{\Delta}{=} P \in \Delta \quad l :: P \xrightarrow{\gamma} \mathcal{P}}{l :: X \xrightarrow{\gamma} \mathcal{P}} \text{ (REC)}
\end{array}$$

Tabella 2: Semantica operativa di BIOKLAIM (Tabella 2)

$$\begin{array}{c}
\mathbf{o} \xrightarrow{\gamma} \emptyset \text{ (F-O)} \qquad s_1 :: \mathbf{nil} \xrightarrow{\gamma} \emptyset \text{ (F-NIL)} \\
\\
\frac{\gamma \neq s_1 : \mathbf{o}(\vec{v})@E(s_1)(\ell)}{s_1 :: (\mathbf{out}(\vec{v})@l, \lambda).P \xrightarrow{\gamma} \emptyset} \text{ (F-OUT)} \qquad \frac{\gamma \neq s_1 : \mathbf{e}(Q)@E(s_1)(\ell)}{s_1 :: (\mathbf{eval}(Q)@l, \lambda).P \xrightarrow{\gamma} \emptyset} \text{ (F-EVAL)} \\
\\
\frac{\gamma \neq s_1 \triangleright \mathbf{T}@s_1 \vee \neg \text{match}(\pi, \mathbf{T})}{s_1 :: (\mathbf{in}(\pi)@s_2, \lambda).P \xrightarrow{\gamma} \emptyset} \text{ (F-IN)} \qquad \frac{\gamma \neq s_1 \blacktriangleright \mathbf{T}@s_1 \vee \neg \text{match}(\pi, \mathbf{T})}{s_1 :: (\mathbf{read}(\pi)@s_2, \lambda).P \xrightarrow{\gamma} \emptyset} \text{ (F-READ)} \\
\\
\frac{\gamma \neq s_1 \triangleleft \mathbf{T}@s_2 \quad s_1 \blacktriangleleft \mathbf{T}@s_2}{s_2 :: \langle \vec{\mathbf{d}} \rangle \xrightarrow{\gamma} \emptyset} \text{ (F-TUPLE)}
\end{array}$$

Tabella 3: Semantica operativa di BIOKLAIM (Tabella 3)

Per il cammino $\pi = s_0(\gamma_0, t_0)s_1(\gamma_1, t_1) \dots$, il numero naturale j e $t \in \mathbb{R}_{\geq 0}$

$$\begin{aligned}
 \text{len}(\pi) &\stackrel{\text{def}}{=} \begin{cases} \infty, & \text{se } \pi \text{ è infinito} \\ l, & \text{altrimenti, dove } s_l \text{ è absorbing di } \pi \end{cases} \\
 \text{st}(\pi, j) &\stackrel{\text{def}}{=} \begin{cases} s_j, & \text{se } 0 \leq j \leq \text{len}(\pi) \\ \text{non definito,} & \text{altrimenti} \end{cases} \\
 \text{ac}(\pi, j) &\stackrel{\text{def}}{=} \begin{cases} \gamma_j, & \text{se } 0 \leq j < \text{len}(\pi) \\ \text{non definito,} & \text{altrimenti} \end{cases} \\
 \text{dl}(\pi, j) &\stackrel{\text{def}}{=} \begin{cases} t_j, & \text{se } 0 \leq j \leq \text{len}(\pi) \\ \infty, & \text{se } j = \text{len}(\pi) \\ \text{non definito,} & \text{altrimenti} \end{cases} \\
 \pi(t) &\stackrel{\text{def}}{=} \begin{cases} \text{st}(\pi, \text{len}(\pi)), & \text{se } t > \sum_{j=0}^{\text{len}(\pi)-1} t_j \\ \text{st}(\pi, m), & \text{altrimenti, dove } m = \min\{j \mid t \leq \sum_{k=0}^j t_k\} \end{cases}
 \end{aligned}$$

Tabella 4: Operatori sui cammini.

$s \models_{\mathbb{A}} \text{true}$	
$s \models_{\mathbb{A}} \neg \Phi$	se e solo se $s \models_{\mathbb{A}} \Phi$ non vale
$s \models_{\mathbb{A}} \Phi \vee \Psi$	se e solo se $s \models_{\mathbb{A}} \Phi$ o $s \models_{\mathbb{A}} \Psi$
$s \models_{\mathbb{A}} \mathcal{S}_{\bowtie p}(\Phi)$	se e solo $\lim_{t \rightarrow \infty} \mathcal{P}\{\pi \in \text{Paths}_{\mathbb{A}}(s) \mid \pi(t) \models_{\mathbb{A}} \Phi\} \bowtie p$
$s \models_{\mathbb{A}} \mathcal{P}_{\bowtie p}(\varphi)$	se e solo se $\mathbb{P}\{\pi \in \text{Paths}_{\mathbb{A}}(s) \mid \pi \models_{\mathbb{A}} \varphi\} \bowtie p$ se e solo se esiste N, ρ, P e Θ tale che valgono: <ol style="list-style-type: none"> 1. $N_s \equiv N \mid i ::_{\rho} P$;
$s \models_{\mathbb{A}} Q@i \rightarrow \Psi$	<ol style="list-style-type: none"> 2. $\text{match}(Q, P) = \Theta$; 3. $(\beta_0, N, \vec{D}) \models_{\text{BK}} \Psi\Theta$.
	se e solo esiste N, ρ, \vec{f} e Θ tale che valgono: <ol style="list-style-type: none"> 1. $N_s \equiv N \mid i ::_{\rho} \langle \vec{f} \rangle$;
$s \models_{\mathbb{A}} \langle \vec{F} \rangle @i \rightarrow \Psi$	<ol style="list-style-type: none"> 2. $\text{match}(\vec{F}, \vec{f}) = \Theta$; 3. $(\beta_0, N, \vec{D}) \models_{\text{BK}} \Psi\Theta$.
	se e solo se esistono N, E e ρ tale che valgono: <ol style="list-style-type: none"> 1. $N_s \equiv N \mid i ::_{\rho} E$;
$s \models_{\mathbb{A}} Q@i \leftarrow \Psi$	<ol style="list-style-type: none"> 2. $(\beta_0, N_s \mid i ::_{\rho} Q, \vec{D}) \models_{\text{BK}} \Psi$
	se e solo se esiste N, E , e ρ tale che valgono: <ol style="list-style-type: none"> 1. $N_s \equiv N \mid i ::_{\rho} E$;
$s \models_{\mathbb{A}} \langle \vec{f} \rangle @i \leftarrow \Psi$	<ol style="list-style-type: none"> 2. $(\beta_0, N_s \mid i ::_{\rho} \langle \vec{f} \rangle, \vec{D}) \models_{\text{BK}} \Psi$.

Tabella 5: Relazione di soddisfacibilità per le formule di stato. La funzione $\text{match}()$ è definita nella tabella 4 di [42].

$$\begin{array}{ll}
\gamma, \Theta \models_{\mathbb{A}} T & \\
\gamma, \Theta \models_{\mathbb{A}} \{\xi_1, \dots, \xi_n\} & \text{se e solo se esiste } j, 0 < j \leq n, \text{ tale che } \gamma, \Theta \models_{\mathbb{A}} \xi \\
\gamma, \Theta \models_{\mathbb{A}} \xi & \text{se e solo se } \text{match}(\xi, \gamma) = \Theta
\end{array}$$

Tabella 6: Relazione di soddisfacibilità per gli specificatori di azioni.

se e soltanto se esiste $k, 0 < k \leq (\text{len}\pi)$ tale che valgono:

$$\begin{array}{ll}
\pi \models_{\mathbb{A}} \Phi_{\Delta} \mathcal{U}_{\Omega}^{\leq t} \Psi & \begin{array}{l}
1. \ t > \sum_{j=0}^{k-1} \text{dl}(\pi, j); \\
2. \text{ esiste un } \Theta_{k-1} \text{ tale che valgono:} \\
\quad \bullet \ \text{st}(\pi, k-1) \models_{\mathbb{A}} \Phi; \\
\quad \bullet \ \text{ac}(\pi, k-1), \Theta_{k-1} \models_{\mathbb{A}} \Omega; \\
\quad \bullet \ \text{st}(\pi, k) \models_{\mathbb{A}} \Psi \Theta_{k-1} \\
3. \text{ se } k > 1 \text{ allora esiste } \Theta_0, \dots, \Theta_{k-2} \text{ tale che} \\
\quad \forall j, 0 \leq j \leq k-2 \text{ valgono:} \\
\quad \bullet \ \text{st}(\pi, j) \models_{\mathbb{A}} \Phi; \\
\quad \bullet \ \text{ac}(\pi, j), \Theta_j \models_{\mathbb{A}} \Delta.
\end{array}
\end{array}$$

se e solo se $\text{st}(\pi, 0) \models_{\mathbb{A}} \Psi$ oppure esiste $k, 0 < k \leq (\text{len}\pi)$, tale che valgono:

$$\begin{array}{ll}
\pi \models_{\mathbb{A}} \Delta \mathcal{U}^{\leq t} \Phi & \begin{array}{l}
1. \ t > \sum_{j=0}^{k-1} \text{dl}(\pi, j); \\
2. \ \text{st}(\pi, k) \models_{\mathbb{A}} \Psi; \\
3. \text{ esiste } \Theta_0, \dots, \Theta_{k-1} \text{ tale che:} \\
\quad \bullet \ \text{st}(\pi, j) \models_{\mathbb{A}} \Phi; \\
\quad \bullet \ \text{ac}(\pi, j), \Theta_j \models_{\mathbb{A}} \Delta.
\end{array}
\end{array}$$

Tabella 7: Relazione di soddisfacibilità per le formule di cammino.

DUE CASI DI STUDIO: MICROGLIA E CELLULE ECCITABILI

Il principale scopo di un'indagine sul comportamento di sistemi biologici è riuscire a strutturare al meglio un modello dal quale si possa dedurre l'emergere di un comportamento. Lo studio tradizionale dei sistemi biologici richiede metodi di riduzione nei quali le quantità di dati sono raccolte per categoria, come ad esempio le concentrazioni di sostanze chimiche nel tempo in risposta ad un determinato stimolo.

I computer sono di primaria importanza per analizzare tali dati e costruire modelli a essi relativi: l'obiettivo è la creazione di modelli di risposta agli stimoli interni e ambientali, al fine di trovare punti deboli o comprendere il funzionamento del sistema preso in esame. Tale approccio spesso comporta lo sviluppo di sistemi automatici come ricostruzione di sistemi dinamici: ad esempio una rete cellulare può essere modellata matematicamente attraverso nozioni provenienti dalla chimica e dalla fisiologia. A questo scopo, le algebre di processo si sono rivelate uno strumento efficace per la descrizione di sistemi complessi, come quelli biologici, a partire dalla descrizione degli elementi che lo compongono.

In questo capitolo verrà utilizzato BioKLAIM per descrivere: un modello che descrive il funzionamento del recettore CB₂ sito nelle cellule della microglia, descritto a livello biologico in 2.3, e in secondo luogo dei modelli che, con livelli di astrazione diversi, illustrano il comportamento delle cellule eccitabili riportato in 2.6.

4.1 MICROGLIA

Di seguito viene presentata e commentata una serie di esperimenti *in silicio*, finalizzati a visualizzare e a mettere in luce alcune proprietà riguardanti l'attivazione degli NK o meno, nel caso in cui sia necessario il loro intervento per combattere una neuroinfiammazione. Oltre all'interazione con il recettore CB₂, è stato anche strutturato il processo di smaltimento (da parte degli enzimi) dei substrati endocannabinoidi, così da poter avere una visione dell'intero sistema endocannabioide.

È possibile identificare due modelli di partenza: uno che descrive le cellule della microglia che non presentano alterazioni genetiche (detto anche *wild type*) e un secondo, invece, dove è presente una mutazione che comporta una costante di disassociazione tra il CB₂ e il 2-AG più alta, rendendo il legame tra queste due proteine meno probabile. In questo secondo scenario viene considerata la presenza di un farmaco che faccia da agonista totale per il CB₂ (come 2-AG), atto a sopperire questa mutazione genetica. Ogni individuo ha una qualche mutazione genetica, più o meno marcata, che lo differisce da un altro, quindi il modello *wild type* può essere considerato un approccio teorico.

Da questo contesto è possibile studiare l'efficienza del farmaco nel modificare i valori, riportandoli a quelli del sistema privo di mutazioni, comprendendo meglio anche la sua migliore concentrazione per ottenere i risultati attesi. In più è importante ricordare che in media i costi di esperimenti *in vitro* ed *in vivo*, siccome comportano numerose serie di prove in laboratorio, risultano essere elevati: precedendoli con prove *in silicio* è possibile eliminare le eventuali casistiche poco significative (o del tutto inutili), consentendo così di abbattere i costi della ricerca.

4.1.1 *Il modello*

Modello privo di mutazione

Il modello privo di mutazione che regola le interazioni è strutturato con la specifica BioKLAIM mostrata nel listing 4.1. Questa è costruita seguendo i concetti riportati in 2.3.

Il modello è realizzato con l'uso di tre località, due fisiche e una logica: *outside*, *inside* e *self*. Come i nomi suggeriscono *outside* corrisponde allo spazio esterno alla cellula mentre *inside* a quello interno. *self*, come è possibile vedere dall'ambiente di assegnazione, viene associato, in ogni località, alla stessa località di appartenenza dell'ambiente di assegnazione: di conseguenza *self* risulterà un modo più intuitivo di riferirsi in una determinata località a se stessa. Le tipologie di tuple in gioco nel sistema sono nove:

- Ag: corrisponde a una molecola di 2-AG, l'agonista totale del nostro sistema;
- Aea: risulta essere una molecola di AEA, l'agonista parziale del nostro sistema;
- R: equivale a un recettore;
- Faah: corrisponde ad una molecola di FAAH, l'enzima responsabile dello smaltimento della AEA;
- Magl: rispecchia una molecola di MAGL, l'enzima che distrugge il 2-AG;
- Aa: corrisponde a una molecola di AA, il prodotto dell'idrolisi catalizzata dagli enzimi di entrambi i substrati (AEA e 2-AG);

```

bio

localities:
physical outside;
physical inside;
logical self;

environments:
outside: [self -> outside];
inside: [self -> inside];

messages: Aea, Ag, Aa, Pge, Plus, R, Cox, Faah, Magl;

agents:
let ReceptorManager be
  X -> read( Ag | R )@self:agcb.[Ag, Plus] | X
      + read( Aea | R )@self:aeacb.[Aea, Plus] | X
end

let InOutCanal be
  X -> in( Aea )@inside:atm.[Aea] | X
      + in( Ag )@inside:atm.[Ag] | X
end

let OutInCanal be
  X -> in( Aea )@outside:atm.[Aea] | X
      + in( Ag )@outside:atm.[Ag] | X
end

let EnzymesManager be
  X -> in( Aea | Faah )@self:aeafaah.[Faah] | [Aa] | X
      + in( Ag | Magl )@self:agmagl.[Magl] | [Aa] | X
      + in( Aa | Cox )@self:aacox.[Cox] | [Pge] | X
end

system:
outside::[Ag]{35000} || outside::[Aea]{50000}
|| outside::[R]{5} || outside::ReceptorManager[X]{1}
|| outside::InOutCanal[X]{1} || inside::OutInCanal[X]{1}
|| inside::EnzymesManager[X]{1} || inside::[Faah]{89000}
|| inside::[Magl]{383500} || inside::[Cox]{250000}
end

```

Listing 4.1: La specifica BioKLaIM di una cellula della microglia.

- Cox: corrisponde a una molecola di COX-2, enzima che smaltisce l'AA;
- Pge: corrisponde a una molecola di PGE, prodotto della degradazione dell'AA, attuata da COX-2 ;
- Plus: è una tupla di "servizio" utilizzata per memorizzare l'avvenuto legame di un substrato (2-AG o AEA) con un recettore.

Gli agenti presenti nella specifica descrivono le reazioni che interessano i substrati. ReceptorManager è l'agente che al suo interno ha il processo che caratterizza i legami dei due agonisti con il CB₂. Viene usata l'azione **read** poiché, a legame avvenuto, i substrati che hanno partecipato all'interazione non si distruggono ma rimangono pronti, in un'interazione successiva, a legarsi ancora per attivare o inibire gli NK. InOutCanal e OutInCanal, come già i nomi suggeriscono, sono degli agenti che trasportano i substrati dall'interno all'esterno della cellula e viceversa. Per ultimo EnzymesManager è l'agente che distrugge i substrati in gioco producendo, dopo due passaggi, PGE.

Infine viene riportata la configurazione iniziale: i substrati Aea e Ag vengono posti, insieme ai recettori R, all'esterno della cellula insieme agli agenti ReceptorManager e InOutCanal. All'interno della cellula invece risiedono gli enzimi insieme agli agenti OutInCanal e EnzymesManager. Dato che il fenomeno da analizzare è la produzione e l'attivazione delle interleuchine (2.3.3), si assume che la cellula sia già attivata, di conseguenza i substrati vengono posti direttamente all'esterno.

Come è possibile vedere, per i rate delle azioni è stato utilizzato un file di mapping, presentato nel listing 4.2.

I valori presenti all'interno della specifica, ovvero il numero di tuple associate ai substrati e agli enzimi, sono le concentrazioni cellulari di

```
aeafaah → 60000000.0;  
aeacb → 350000.0;  
  
agmagl → 110000.0;  
agcb → 15000000.0;  
  
aacox → 24000000.0;  
  
atm → 700000.0;
```

Listing 4.2: Il file di mapping per il modello microglia non mutato.

queste proteine, dato che, in questo studio, il volume totale è stato considerato unitario; inoltre i valori presenti nel file di mapping rappresentano il reciproco della costante di disassociazione, quindi a valori più grandi sarà associata una maggiore affinità tra gli elementi in considerazione. Tutti i valori numerici vengono da concentrazioni studiate in letteratura: [17, 48, 49, 50]: i dati presentati sono tutti riportati alla stessa unità di misura, ovvero $\frac{\text{pM}}{\text{g}}$ (pico mole per grammo).

Modello con mutazione

Il modello con mutazione che regola le interazioni è strutturato con la specifica BioKLAIM mostrata nel listing 4.3.

Come è possibile osservare, questa è pressoché identica alla specifica senza mutazione. Tra le cose che differiscono nei due modelli si può notare la presenza di una decima tupla, *Drug*, che rappresenta una molecola di farmaco. Questa viene trattata esattamente come le tuple rappresentanti i substrati AEA e 2-AG. Inoltre è da notare che il processo che descrive il possibile legame tra CB2 e farmaco produce, dopo il legame, lo stesso risultato prodotto dal processo che descrive l'interazione tra CB2 e 2-AG: questo perché il farmaco considerato ha la stessa funzione del 2-AG, ovvero di agonista totale.

```

bio

localities:
physical outside;
physical inside;
logical self;

environments:
outside: [self → outside];
inside: [self → inside];

messages: Aea, Ag, Drug, Aa, Pge, Plus, R, Cox, Faah, Magl;

agents:
let ReceptorManager be
  X → read( Aea | R )@self:aeacb.[Aea, Plus] | X
    + read( Ag | R )@self:agcb.[Ag, Plus] | X
    + read( Drug | R )@self:drugcb.[Ag, Plus] | X #Regola che gestisce la mutazione del fattore 1/Ki del
      recettore con Ag
end

let InOutCanal be
  X → in( Aea )@inside:atm.[Aea] | X
    + in( Ag )@inside:atm.[Ag] | X
    + in( Drug )@inside:atm.[Drug] | X
end

let OutInCanal be
  X → in( Aea )@outside:atm.[Aea] | X
    + in( Ag )@outside:atm.[Ag] | X
    + in( Drug )@outside:atm.[Drug] | X
end

let EnzymesManager be
  X → in( Aea | Faah )@self:aeafaah.[Faah] | [Aa] | X
    + in( Ag | Magl )@self:agmagl.[Magl] | [Aa] | X
    + in( Drug | Magl )@self:drugmagl.[Magl] | [Aa] | X
    + in( Aa | Cox )@self:aacox.[Cox] | [Pge] | X
end

system:
outside::[Ag]{35000} || outside::[Aea]{50000} || outside::[Drug]{18000}
|| outside::[R]{5} || outside::ReceptorManager[X]{1}
|| outside::InOutCanal[X]{1} || inside::OutInCanal[X]{1}
|| inside::EnzymesManager[X]{1} || inside::[Faah]{89000}
|| inside::[Magl]{383500} || inside::[Cox]{250000}
end

```

Listing 4.3: La specifica BioKLAIM di una cellula mutata della microglia.

```
aeafaah → 60000000.0;  
aeacb → 350000.0;  
  
agmagl → 110000.0;  
agcb → 5000000.0;  
  
aacox → 24000000.0;  
  
atm → 700000.0;  
  
drugmagl → 110000.0;  
drugcb → 20000000.0;
```

Listing 4.4: Il file di mapping per il modello microglia mutato.

La configurazione iniziale del sistema è identica a quella nel caso in cui si tratti il modello senza mutazione a meno dell'inizializzazione della tupla Drug. Anche qui è stato usato un file dei rate separato, mostrato nel listing 4.4.

Le uniche differenze che si notano con il file di mapping del modello non mutato risultano essere: un valore del rate di legame tra 2-AG e CB2 più piccolo rispetto al precedente (questa è la mutazione) e la presenza di nuove corrispondenze, associate al fattore di legame del farmaco, del CB2 e dell'enzima, necessarie per completare i dati associati alla presenza del farmaco.

Gli esperimenti *in silicio* che verranno compiuti, come detto, si basano sulla ricerca della migliore concentrazione di farmaco per riportare i valori del sistema il più vicino possibile a quelli teorici. Questo tipo di studio viene compiuto, ad esempio, per farmaci come il JWH-015¹ del quale è nota l'affinità con il recettore ma, per una corretta somministrazione che sopperisca al problema, è necessario conoscerne anche la migliore quantità da immettere nel sistema.

¹ È una sostanza chimica che agisce come agonista totale. Questo si lega quasi 28 volte più fortemente a CB2 che CB1, ed è risultato utile nel trattamento del dolore e dell'infiammazione.

```

Natural_Killer_Attivato: [Ag, Plus]@outside
Natural_Killer_Bloccato: [Aea, Plus]@outside
Prostaglandine: [Pge]@inside

```

Listing 4.5: I dati di cui registrare l'andamento durante la simulazione.

In generale, con delle piccole modifiche al modello mutato, è inoltre possibile lo studio di altre casistiche come, per fare un esempio, la situazione nella quale si è a conoscenza del farmaco e della quantità somministrata e si vuole comprendere l'incidenza che questo ha. Un altro studio possibile è voler riconoscere la mutazione che caratterizza un sistema, sempre essendo a conoscenza del tipo di farmaco e della quantità somministrata.

La simulazione

Il fine della simulazione del sistema è quello di mostrare l'andamento delle concentrazioni di NK attivati, NK inibiti e PGE sintetizzate dagli enzimi. Per entrambi il modelli i dati da raccogliere sono gli stessi e sono mostrati nel listing 4.5.

I parametri per ogni simulazione sono:

- `deadline = 0.00003`, dato che l'andamento caratteristico dell'attivazione degli NK si ha entro questo valore;
- `dt = 0.0000000001`, in modo da avere l'intervallo analizzato diviso in 300001 valori;
- `iterations = 1000`, così da avere un comportamento medio del modello.

Sono state eseguite simulazioni per il sistema non mutato e per il sistema mutato con diverse concentrazioni di farmaco: ogni simulazione ripor-

```
plot "true {true}U<=$o (![Aea]@outside -> true && ![Ag]@outside -> true)" with [0:0.0003:0.0000001]
```

Listing 4.6: La formula di cammino da verificare con il Model Checker.

ta, istante per istante, l'andamento medio e i valori di $\text{media} \pm \text{varianza}$ delle tuple analizzate.

Le tempistiche legate al processo di simulazione, su di un Intel[®] Core Quad Processor Q6600² equipaggiato con 4 Gb di memoria, sono mediamente di:

160 secondi ad iterazione per il sistema non mutato;

80 secondi ad iterazione per il sistema mutato in assenza di farmaco;

140 secondi ad iterazione per il sistema mutato con valori di farmaco che vanno da $10^{-8} \frac{M}{g}$ a $2 * 10^{-8} \frac{M}{g}$.

La verifica di proprietà

La proprietà da verificare in questo modello riguarda la ricerca della probabilità che il sistema abbia raggiunto il deadlock. Il deadlock è dato dal fatto che tutti i substrati AEA e 2-AG vengono trasportati dall'esterno all'interno della cellula, non partecipando più alle interazioni con il recettore CB2 e, quindi, non attivando (o inibendo) altri NK.

La formula da provare è la stessa per tutti i modelli, questa è mostrata nel listing 4.6.

Essa, riprendendo le nozioni spiegate in 3.5 e 3.6.3, restituisce il valore di probabilità che le tuple Aea e Ag non risiedano, con il passare del

² <http://processorfinder.intel.com/DetailsPrinterFriendly.aspx?sSpec=SLACR>

tempo, all'esterno della cellula³. Questa formula è stata provata su una serie di specifiche che descrivono il modello senza e con mutazione: per quest'ultimo caso con varie concentrazioni di farmaco, per avere una prova dell'incidenza che il farmaco ha sull'intero sistema.

Per quanto riguarda le performance del processo di verifica della proprietà, sempre sulla stessa macchina utilizzata per le simulazioni, i valori sono mediamente:

240 secondi ad iterazione per il sistema non mutato e per il mutato con concentrazioni di farmaco che vanno da $10^{-8} \frac{M}{g}$ a $2 * 10^{-8} \frac{M}{g}$;

120 secondi ad iterazione per il sistema mutato in assenza di farmaco.

4.1.2 Risultati

Successivamente verranno mostrati i risultati degli esperimenti *in silicio* relativi ai modelli, mutati o meno, della microglia. Alla base di questi c'è l'intenzione di poter ricercare la migliore concentrazione di farmaco da somministrare, oltre che di mostrare il funzionamento della cellula nel caso si sia attivata per combattere qualche agente patogeno.

Simulazione

Per la simulazione sono stati preparati otto scenari: uno per il modello non mutato, mentre i rimanenti sette per poter vedere quanto una mutazione e valori di concentrazioni di farmaco differenti vadano a modificare il comportamento dell'intero sistema.

³ Rispetto a MoSL⁺, \$0 risulta essere una macro: indica un insieme di formule dove, da formula a formula, cambia \$0 in base al range definito dai valori tra parentesi quadre dopo il with. Il grafico che si ottiene è quello che associa al valore di \$0 la relativa probabilità.

Ogni simulazione mostra l'andamento dei tre valori monitorati: attivazione/blocco degli NK e produzione di PGE. Delle otto simulazioni vengono riportati, in grafico (figure 15, 16), il comportamento del sistema non mutato e di quello mutato in assenza di farmaco.

Dato che la mutazione interessa soltanto la costante di dissociazione tra il recettore CB2 e il 2-AG, l'andamento delle curve riguardanti il blocco degli NK e la produzione di PGE risulta identico per tutte le simulazioni. Per questo motivo vengono riportati in un unico grafico (figura 17) i vari andamenti dell'attivazione degli NK.

I risultati ottenuti hanno lo stesso andamento di quelli riportati in letteratura ([17, 48, 49, 50]), soprattutto per quanto riguarda la durata della reazione.

Come viene evidenziato dai dati raccolti, il valore della varianza di tutti i modelli proposti è circa $\pm 23\%$ del valor medio. Inoltre si può osservare che la concentrazione di farmaco che meglio sopprime la mutazione del sistema varia da $1,7 \times 10^{-8} \frac{\text{M}}{\text{g}}$ a $1,8 \times 10^{-8} \frac{\text{M}}{\text{g}}$, attraverso il model checking verrà confermata questa affermazione.

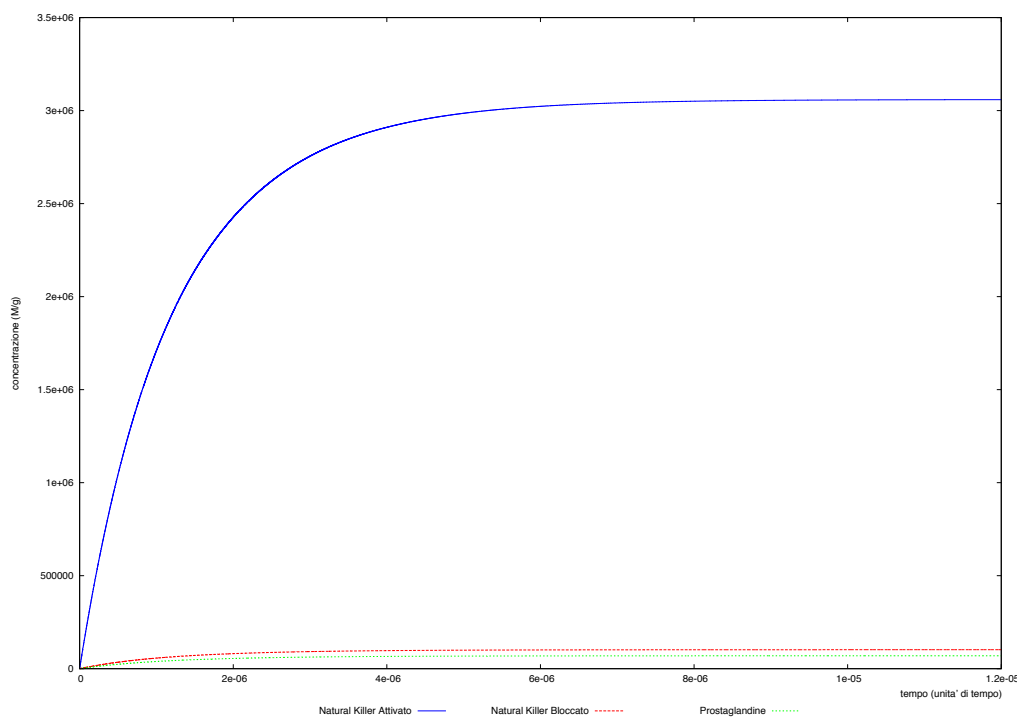


Figura 15: Comportamento del sistema senza mutazione (wild type).

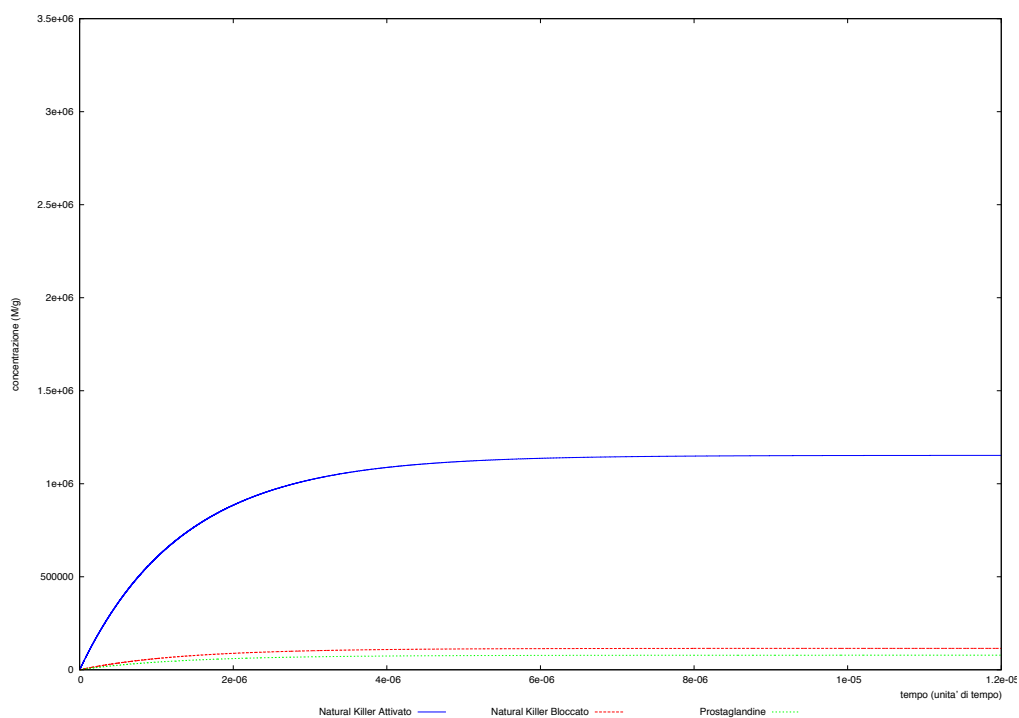


Figura 16: Comportamento del sistema mutato in assenza di farmaco.

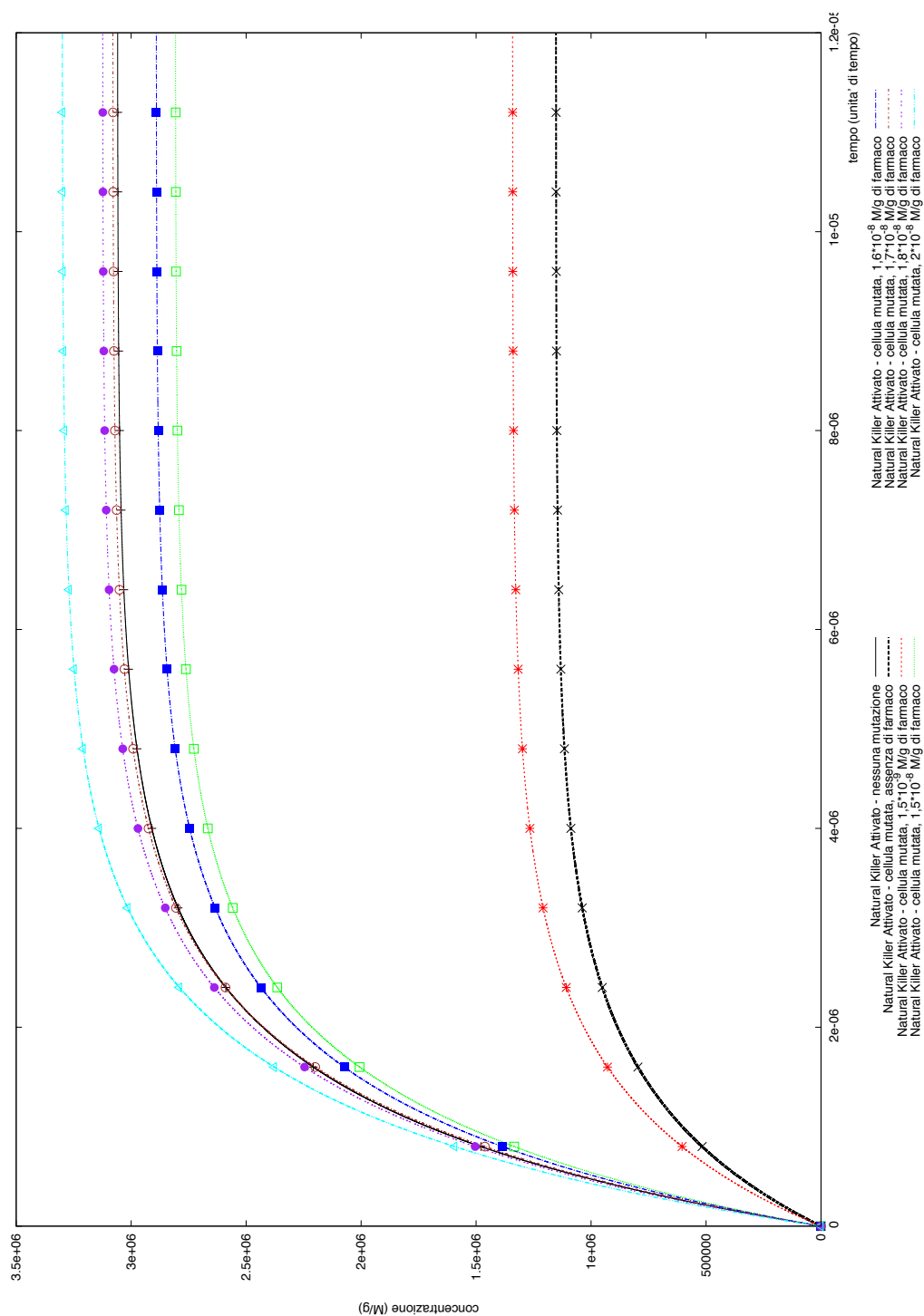


Figura 17: Andamenti dell'attivazione degli NK nei vari modelli.

Verifica di proprietà

Il model checking è stato impiegato per investigare sulla probabilità che i substrati AEA e 2-AG siano presenti o meno, all'esterno della cellula. Questa probabilità viene riportata in figura 18.

Al variare della concentrazione del farmaco, il model checker consente di verificare quanto l'andamento del sistema mutato differisca da quello non mutato. Tale differenza si può notare attraverso la probabilità di raggiungere il deadlock: questa risulta più alta nei sistemi rappresentanti cellule mutate che quelle non mutate. Anche attraverso questa verifica, le concentrazioni di farmaco che sopperiscono alla mutazione vanno da $1,7 \times 10^{-8} \frac{M}{g}$ a $1,8 \times 10^{-8} \frac{M}{g}$.

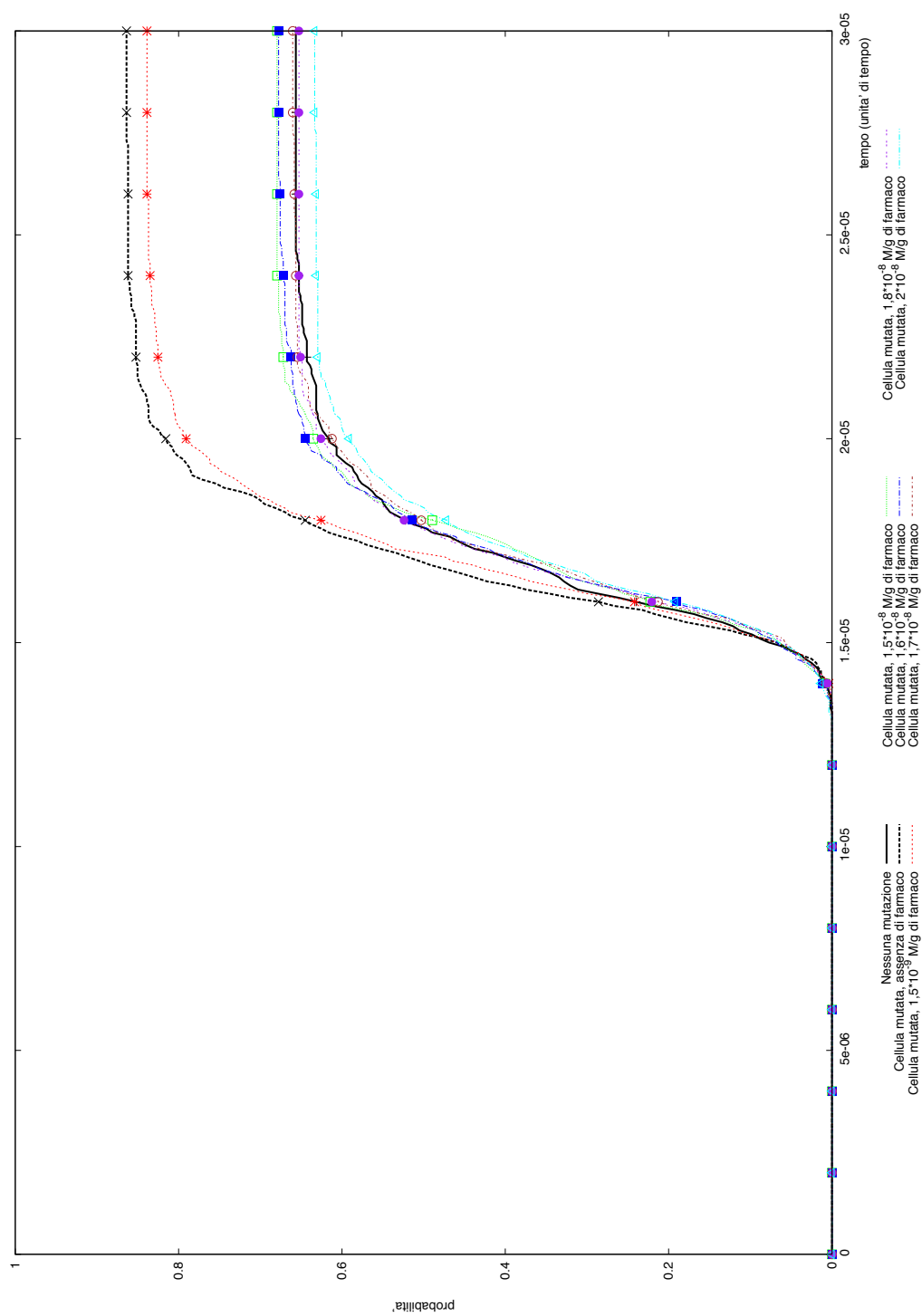


Figura 18: Probabilità per i vari modelli che i substrati AEA e 2-AG non siano presenti all'esterno della cellula al variare del tempo.

4.2 CELLULE ECCITABILI

In questa sezione vengono proposti i modelli BioKLAIM e gli esperimenti *in silicio* riguardanti le cellule eccitabili. Il fine di questi esperimenti è di mostrare come i tessuti di cellule eccitabili si comportano in relazione al potenziale d'azione. A questo riguardo sono stati creati due modelli: il primo tratta la variazione del potenziale astruendo da cosa la modifica nel tempo, un secondo invece riporta il comportamento del potenziale delle cellule eccitabili in funzione degli ioni presenti all'interno e all'esterno della cellula. Il motivo per il quale vengono presentati due modelli è da ricercare nel fatto che, durante la ricerca di letteratura riguardante l'argomento, è stato più facile procedere per gradi, passando da un modello all'altro soltanto dopo aver raggiunto una confidenza con il problema analizzato. Le osservazioni sono tutte basate sulla propagazione degli stimoli nel tessuto.

Le guardie sulle azioni sono state introdotte proprio per essere di aiuto nella creazione dei modelli per le cellule eccitabili, dato che è risultato necessario caratterizzare sottoinsiemi di azioni attivate in ogni fase del ciclo descritto dal potenziale d'azione. Un'altra caratteristica introdotta in BioKLAIM per poter dare una descrizione appropriata del comportamento di questo sistema è la possibilità di esprimere i risultati di una simulazione come funzione di più parametri del modello.

4.2.1 Il modello

Inizialmente verranno mostrate le specifiche che descrivono un'unica cellula, successivamente verrà presentato come le cellule vengono composte per creare tessuti.

```

bio

localities:
physical c1;
physical c2;

logical self;
logical next;

environments:
c1: [self → c1, next → c2];

messages: Ddp, Signal, Stimulated, Excited, Ignore;

agents:
let Cell be
  X → in( Signal )@self:1.0.[Stimulated] | [Ignore] | X1 #stimolo rilevato.

  X1 → {[Ignore]@c1 => 1.0} in( Signal )@self:10000000.0.X1
    + {[Ddp]@c1 < 160.0 && [Stimulated]@c1 => 1.0} read( Ddp )@self:0.112.[Ddp] | X1
    + {[Ddp]@c1 => 160.0 && [Stimulated]@c1 => 1.0} in( Stimulated )@self:1.0.[Excited] | [Signal]@c2
      | X1
    + {[Ddp]@c1 => 40.0 && [Stimulated]@c1 <= 0.0 && [Excited]@c1 => 1.0} in( Ddp )@self:0.12.X1
    + {[Ddp]@c1 < 40.0 && [Stimulated]@c1 <= 0.0} out( Ddp )@self:1.0.X2

  X2 → {[Ignore]@c1 => 1.0} in( Signal )@self:10000000.0.X2
    + {[Ddp]@c1 < 60.0} out( Ddp )@self:1.0.[Ddp] | X2
    + {[Ddp]@c1 => 60.0} in( Ddp | Excited | Ignore )@self:1.0.[Ddp] | X
end

system:

c1::[Signal]{1} || c1::[Ddp]{60} || c1::Cell[X]{1}
end

```

Listing 4.7: La specifica BIOKLAIM di una cellula eccitabile.

Il modello che regola le interazioni che descrivono il fenomeno del potenziale d'azione è quello riportato nel listing 4.7.

Questo modello è realizzato con l'uso di quattro località, due logiche e due fisiche: `c1`, `c2`, `self` e `next`. La prima corrisponde alla cellula dove avviene il potenziale d'azione, la seconda, in questo caso, è soltanto la località dove viene inviato il segnale per far partire una nuova reazione. Le località logiche, come è possibile vedere dall'ambiente di assegnazione, sono utilizzate per riferirsi alle due località fisiche, per facilitare la lettura della specifica: `self` è la località dove risiede l'interazione, `next`

è la località dove verrà inviato l'impulso per far scaturire una nuova stimolazione. Le tuple del sistema sono cinque:

- `Ddp`: rappresenta 1mV di potenziale;
- `Signal`: corrisponde al segnale di stimolo iniziale che attiva la reazione;
- `Stimulated`: è la tupla che rappresenta l'inizio della reazione di eccitazione;
- `Excited`: caratterizza una cellula nel momento in cui ha raggiunto l'eccitazione;
- `Ignore`: ha il compito di ignorare tutte le possibili richieste di attivazione della reazione durante il potenziale d'azione, poiché non è contemplabile la possibilità che venga considerata una successiva attivazione della reazione.

L'agente presente nella specifica caratterizza il comportamento nelle fasi di depolarizzazione e iperpolarizzazione. La configurazione iniziale è abbastanza semplice: l'agente `Cell` viene posto, insieme alla tupla `Signal`, nella località `c1`. Inoltre è da notare la presenza di 60 tuple `Ddp` sempre nella località `c1`: questa è dovuta al fatto che il potenziale di riposo risulta essere più alto del potenziale di equilibrio per il K^+ , quindi è necessario far partire il sistema con delle tuple.

È da notare come l'uso delle guardie permetta di caratterizzare situazioni eccezionali (come ad esempio in questo caso il raggiungimento del potenziale di equilibrio per il Na^+) attraverso un'azione messa in scelta con le altre, senza dover cercare strade meno intuitive e più farraginose: sarà `BIOKLAIM` ad attivarla nel momento in cui le condizioni espresse nella guardia diventeranno vere.

Come accennato verranno presentati due modelli per le cellule eccitabili, uno (il precedente) più semplice e un secondo che tratta il fenomeno del potenziale d'azione come un processo di scambio ionico tra esterno e interno della cellula eccitabile. Questo secondo modello è riportato nel listing 4.8.

All'interno della specifica, nel commento iniziale, viene riportata l'idea che sta alla base dell'intero sistema, il potenziale d'azione è contraddistinto dalla presenza di due lavori che agiscono: il lavoro elettrico, caratterizzato dalla formula $L_e = z * F * E$, e il lavoro chimico $L_c = R * T * \ln \frac{[Ione_{out}]}{[Ione_{in}]}$. Il primo è dovuto al campo elettrico, dovuto dalla presenza degli ioni, che si sviluppa tra l'interno e l'esterno della cellula, il secondo invece è associato alle concentrazioni ioniche sempre presenti tra l'esterno e l'interno della cellula; l'equazione di Nernst (2.6.2) deriva dall'equivalenza di questi due lavori.

Nel momento in cui i due lavori raggiungono l'equilibrio, $L_c = -L_e$, non c'è più movimento dello ione preso in considerazione e quindi il sistema si porta in deadlock. Soltanto grazie alle pompe ioniche una cellula che ha terminato il processo di eccitazione, dopo una fase detta di ripristino, può essere pronta a sviluppare un nuovo potenziale d'azione.

Le località in questo modello sono cinque: tre fisiche e due logiche. `inside` corrisponde all'interno dell'unica cellula dove avviene il fenomeno da registrare, `outside` è l'esterno della cellula, mentre `inside_1` rappresenta l'interno della cellula che riceverà il segnale di attivazione dalla cellula precedente. Le località logiche, come si può vedere anche dall'ambiente di assegnazione, hanno lo stesso significato di quelle nel modello precedente. In questo modello le tuple che partecipano al sistema sono nove:

- `Na`: rappresenta una molecola di Na^+ ;


```

bio

localities:
physical inside;
physical outside;
physical inside_1;

logical self;
logical next;

environments:
inside: [self → inside, next → inside_1];

messages: Na, K, Signal, NaChan, KChan, Pump, Ddp, Ready, Ignore, Excited;

# F = 96.485 J / (V * mol) [Cost. Faraday]
# R = 8.314 J / (K * mol) [Cost. Gas]
# T = 310.15 K [Temperatura umana]
# z = numero di valenza ioni (in questo caso 1)

# Lavoro Elettrico:
# Le = z * F * E
# Lavoro Chimico:
# Lc = R * T * ln([I]@outside / [I]@inside)
# Equilibrio:
# Lc = -Le

agents:
let Cell be
  X → in( Signal )@self:1.0.[NaChan] | [Ignore] | X1 #stimolo rilevato.

  X1 → {[Ignore]@inside => 1.0} in( Signal )@self:1000000000000.0.X1
    + {[NaChan]@inside > 0.0 && (8.314*310.15*ln([Na]@outside / [Na]@inside)) > -(96.485*30)} in( Na )
      @outside:0.65.[Na] | X1
    + {[NaChan]@inside > 0.0 && (8.314*310.15*ln([Na]@outside / [Na]@inside)) <= -(96.485*30)} in(
      NaChan )@self:10.0.[KChan] | [Signal]@next | [Excited] | X1
    + {[KChan]@inside > 0.0 && (8.314*310.15*ln([K]@outside / [K]@inside)) < (96.485*90)} in( K )@self
      :10.0.[K]@outside | X1
    + {[KChan]@inside > 0.0 && (8.314*310.15*ln([K]@outside / [K]@inside)) => (96.485*90)} in( KChan )
      @self:0.7.[Pump] | X3

  X3 → eval( Cell[X4] )@self:100000.0.X2

  X4 → {[Ignore]@inside => 1.0} in( Signal )@self:1000000000000.0.X4
    + {[Pump]@inside > 0.0 && [Ddp]@inside < 15.0} out( Ddp )@self:100000.0.X4
    + {[Pump]@inside <= 0.0} in( Ddp )@self:1.0.X4
    + {[Pump]@inside > 0.0 && [Ready]@inside <= 0.0 && [Ddp]@inside => 15.0} out( Ready )@self:
      10000.0.X4

  X2 → {[Ignore]@inside => 1.0} in( Signal )@self:1000000000000.0.X2
    + {[Pump]@inside > 0.0 && [K]@inside < 117.0} in( K )@outside:10000.0.[K]@self | X2
    + {[Pump]@inside > 0.0 && [Na]@outside < 63.0} in( Na )@self:10000.0.[Na]@outside | X2
    + {[Pump]@inside > 0.0 && [K]@inside => 117.0 && [Na]@outside => 63.0} in( Ignore | Pump | Ready )
      @self:10000.0.X

end

system:
inside::[Signal]{1} || inside::Cell[X]{1} ||
inside::[K]{117} || outside::[K]{4} ||
inside::[Na]{20} || outside::[Na]{63}
end

```

Listing 4.8: La specifica BioKLAIM di una cellula eccitabile.

- K: corrisponde alla molecola di K^+ ;
- NaChan: comunica l'apertura dei canali voltaggio dipendenti per Na^+ ;
- KChan: comunica l'apertura dei canali voltaggio dipendenti per K^+ ;
- Pump: comunica l'attivazione della pompa Na^+/K^+ ;
- Ddp: rappresenta 1mV utile per rappresentare il comportamento della pompa ionica;
- Ready: ha il compito di fare da semaforo per il processo che riporta il sistema nella configurazione iniziale;
- Ignore: come nel modello precedente, ha il compito di ignorare tutte le possibili richieste di attivazione della reazione.

L'agente *Cell*, esattamente come nel modello precedente, contiene i processi che regolano il fenomeno. In questo caso le guardie e le azioni associate ad esse riguardano lo scambio di ioni dall'esterno all'interno e l'attivazione o meno delle strutture cellulari che cooperano. La configurazione iniziale pone le concentrazioni, espresse in nM, degli ioni interessati, inserisce l'agente *Cell* all'interno della cellula e pone la tupla *Signal* in modo da far scatenare il potenziale d'azione.

I valori dei rate associati alle azioni e il comportamento del modello che tratta la reazione considerando lo scambio ionico sono stati ricavati da ricerche in letteratura, principalmente da [24, 51, 52].

In questo caso gli esperimenti *in silicio*, come accennato, saranno basati sulla creazione di tessuti di cellule analizzando la propagazione dell'eccitazione. Questo tipo di esperimenti stanno alla base degli studi di particolari malattie come l'epilessia, le convulsioni o, in generale, tutte

quelle che sono legate a problemi dovuti a una serie di collegamenti tra cellule eccitabili tali da poter scatenare forti scariche elettriche che attraversano il tessuto. Per fare un esempio: l'attacco epilettico è caratterizzato da un'incontrollata scarica elettrica dovuta alla presenza di collegamenti tra cellule che immettono segnali di inizio eccitazione in zone che non dovrebbero essere eccitate, provocando forti dolori.

TESSUTI La creazione dei tessuti avviene componendo le singole cellule mostrate in precedenza; vengono presentate due configurazioni di collegamento:

SQUARE: ogni cellula si collega alle altre otto cellule a sé confinanti e, con il raggiungimento del picco di eccitazione, invia loro un impulso che scatena la depolarizzazione delle membrane.

LIST: questa configurazione collega ogni cellula ad un'unica altra cellula.

La prima configurazione è propria dei tessuti muscolari, mentre la seconda degli assoni.

Per automatizzare il processo di creazione di specifiche riguardanti tessuti viene in aiuto un'applicazione C, *TissueManager*⁴. L'applicazione si aspetta che l'utente dichiari quale modello utilizzare (quello che considera, oppure no, le interazioni ioniche tra esterno e interno), la dimensione del tessuto (il numero di cellule per righe e colonne), il tipo di configurazione (**SQUARE** oppure **LIST**) e infine le cellule che inizialmente sono stimulate, dalle quali poi partirà la propagazione.

⁴ Reperibile a <http://code.google.com/p/klaim-excitable-builder/>. Oltre a questa è presente anche un'applicazione per sistemi Apple (*ExcitableBuilder*).

Delta_V: [Ddp]@c1

Listing 4.9: Dati da registrare per il modello che astrae l'attività ionica cellulare.

Delta_V:

```
(([NaChan]@inside)*( 105.5908433 - (((8.314*310.15)/(96.485))*ln([Na]@outside / [Na]@inside)))) +
([KChan]@inside)*( 46.0342038592 - (((8.314*310.15)/(96.485))*ln([K]@outside / [K]@inside)) ) +
([Pump]@inside) * (-90 + [Ddp]@inside))
```

Listing 4.10: Dati da registrare per il modello che considera l'attività ionica cellulare.

La simulazione

Le simulazioni riguardano sia la singola cellula, per poterne visionare il comportamento, sia i tessuti.

Nel caso della singola cellula viene raccolto l'andamento della differenza di potenziale e quindi, per avere una buona panoramica di come la variazione di potenziale si comporti mediamente, ognuno dei due modelli è stato simulato con 2000 iterazioni. Invece, nel caso dei tessuti, il numero di iterazioni è pari a 1, in questo modo si può vedere esattamente come avviene la propagazione nel tessuto.

I dati da raccogliere sono mostrati nel listing 4.9 per il primo modello, mentre nel listing 4.10 per il secondo.

Gli esperimenti proposti sono:

- singola cellula con modello che astrae dall'attività ionica cellulare;
- singola cellula con modello che tiene conto dell'attività ionica cellulare;
- tessuto 10×10 di cellule con modello che astrae dall'attività ionica cellulare, dove inizialmente è stimolata la cellula in posizione (0,0);

```
plot "true {true}U<=$o ([Excited]@inside -> true)" with [0:1000:0.001]
```

Listing 4.11: La formula di cammino da verificare con il Model Checker.

- tessuto 10×10 di cellule con modello che astrae dall'attività ionica cellulare, dove inizialmente è stimolata la cellula in posizione (5,5);
- tessuto 1×10 di cellule con modello che astrae dall'attività ionica cellulare, dove inizialmente è stimolata la cellula in posizione (0,0).

Il tempo di simulazione è in media di 0,01 secondi a iterazione per gli esperimenti basati su modelli di singole cellule e di 60 secondi per gli esperimenti che registrano il comportamento di un tessuto.

La verifica di proprietà

In questo caso il model checking è stato impiegato per poter affermare che i due modelli per le singole cellule hanno lo stesso comportamento per quanto riguarda il raggiungimento dell'eccitazione. La formula da provare è riportata nel listing 4.11.

Mostrare che i due modelli si eccitano secondo un comportamento simile permette di utilizzarne indifferentemente uno a piacimento, avendo quindi tempi di simulazione ridotti grazie al modello che astrae dall'attività ionica cellulare. In questo caso i tempi di verifica si aggirano, mediamente, intorno a 20 secondi ad iterazione.

4.2.2 *Risultati*

Qui verranno mostrati i risultati degli esperimenti *in silicio* precedentemente riportati. Per quanto riguarda la simulazione delle specifiche relative ai tessuti di cellule, i dati risultanti dalla simulazione sono stati

prima riorganizzati attraverso un programma Java, BioKlaim_Parser⁵. Tale programma, dai dati registrati dal tool durante la simulazione, genera nuovi documenti per poter avere delle “istantanee” dell’intero sistema, nei vari momenti, dalle quali poter disegnare l’andamento della propagazione della differenza di potenziale nel tessuto.

Per quanto riguarda l’analisi delle singole cellule e dei tessuti, i risultati ottenuti hanno lo stesso andamento di quelli riportati in letteratura ([24, 53]).

Simulazione

I risultati degli esperimenti proposti sono mostrati nelle figure 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29 e 30.

È possibile prendere visione del comportamento dei tre tessuti presentati visitando <http://code.google.com/p/klaim-excitable-builder/>, dove sono presenti dei video che registrano il fenomeno del potenziale d’azione per i tessuti presentati.

Verifica di proprietà

Questo tipo di investigazione, come precedentemente detto, è stato affrontato per poter affermare che i modelli che considerano l’attività ionica o meno sono in realtà pressoché sostituibili l’uno con l’altro. Viene analizzata la probabilità che avvenga l’eccitazione al variare del tempo, questa è riportata in figura 31.

Le curve, che rappresentano le due probabilità che i modelli si eccitino, hanno circa lo stesso andamento e questo consente di dichiarare che i modelli studiati risultano molto simili per quanto riguarda l’eccitazione.

⁵ Sempre reperibile a <http://code.google.com/p/klaim-excitable-builder/>.

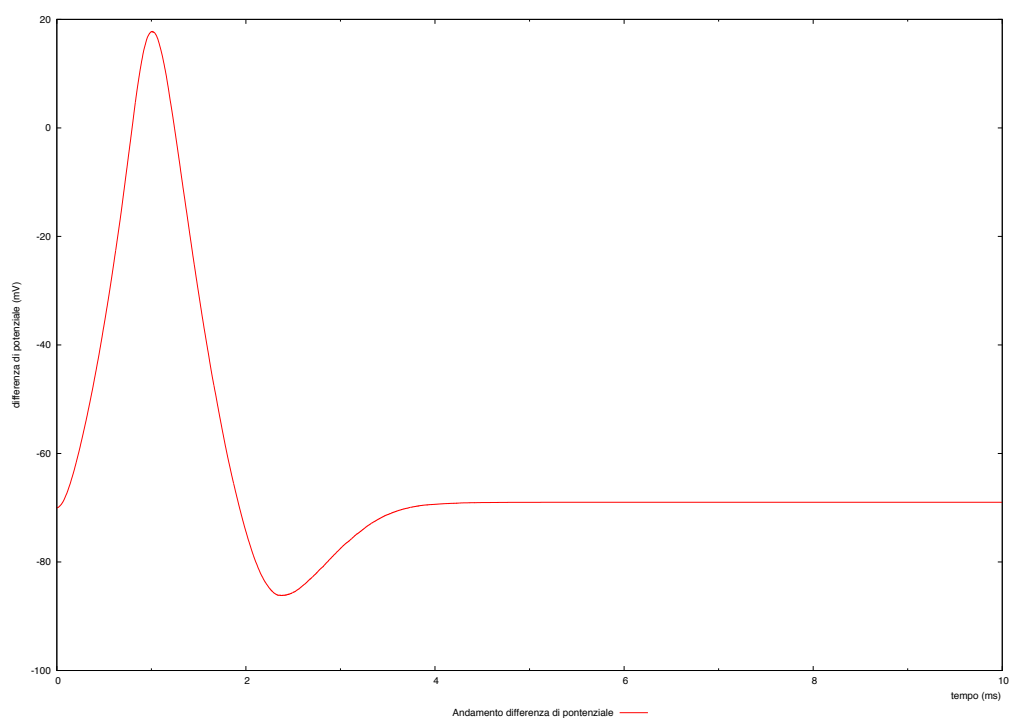


Figura 19: Andamento della differenza di potenziale nel modello di cellula che astrae dall'attività ionica cellulare.

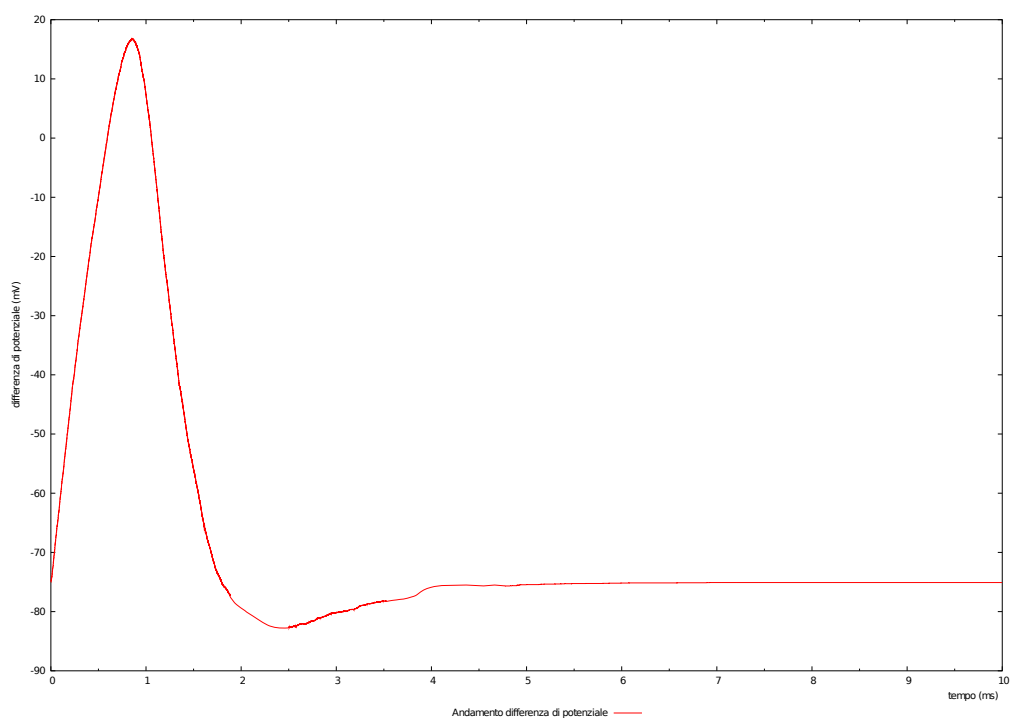


Figura 20: Andamento della differenza di potenziale nel modello di cellula che tiene conto dell'attività ionica cellulare

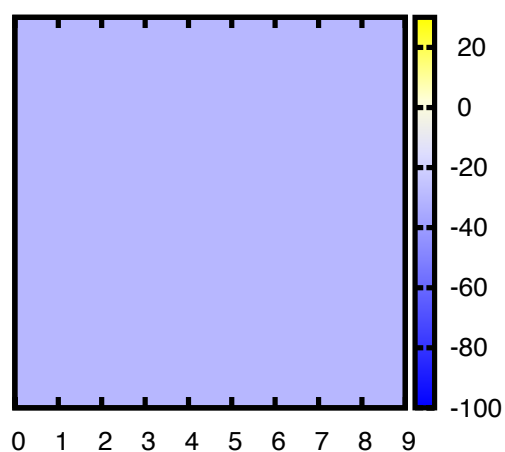


Figura 21: Propagazione della differenza di potenziale in un tessuto in configurazione LIST, 0 ms.

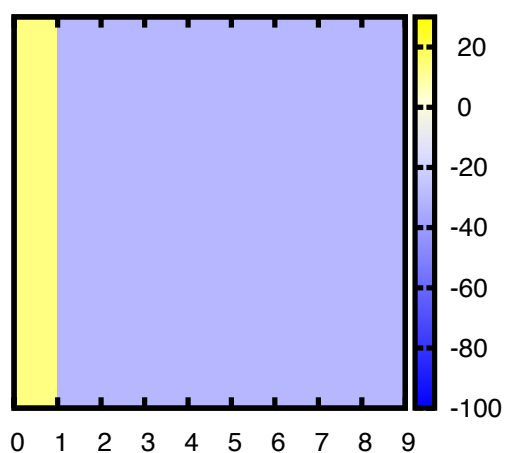


Figura 22: Propagazione della differenza di potenziale in un tessuto in configurazione LIST, 1 ms.

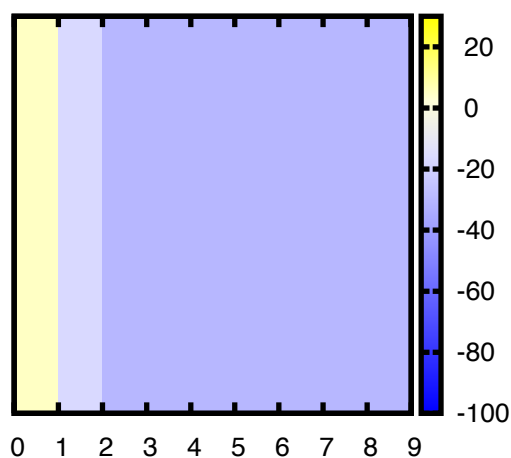


Figura 23: Propagazione della differenza di potenziale in un tessuto in configurazione LIST, 1 ms.

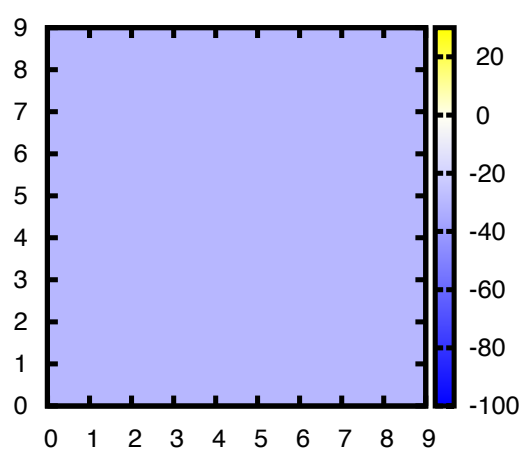


Figura 24: Propagazione della differenza di potenziale in un tessuto in configurazione SQUARE, 0 ms.

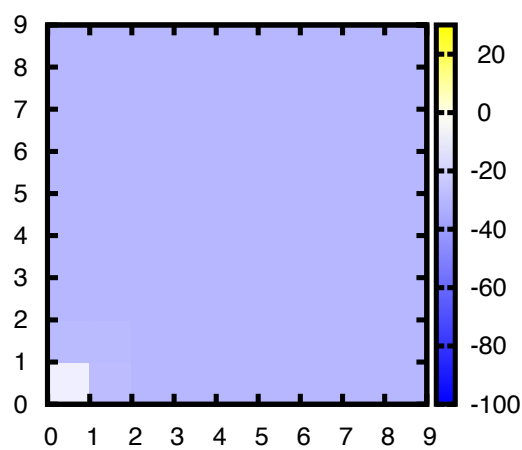


Figura 25: Propagazione della differenza di potenziale in un tessuto in configurazione SQUARE, 0 ms.

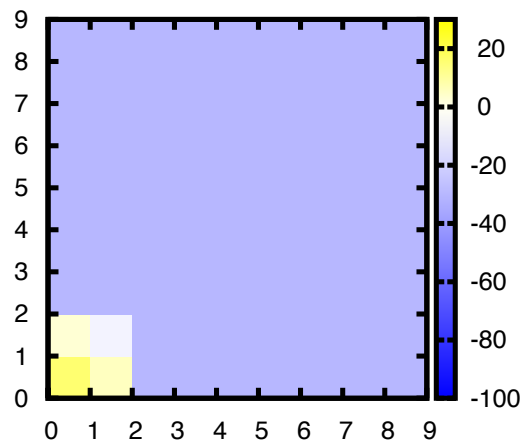


Figura 26: Propagazione della differenza di potenziale in un tessuto in configurazione SQUARE, 1 ms.

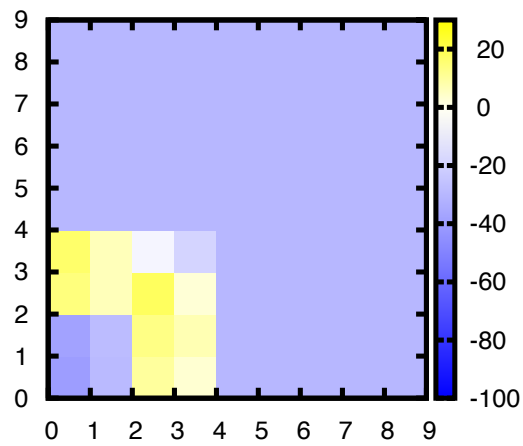


Figura 27: Propagazione della differenza di potenziale in un tessuto in configurazione SQUARE, 2 ms.

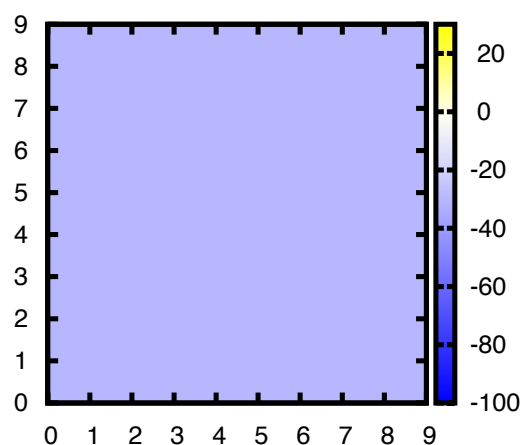


Figura 28: Propagazione della differenza di potenziale in un tessuto in configurazione SQUARE, 0 ms.

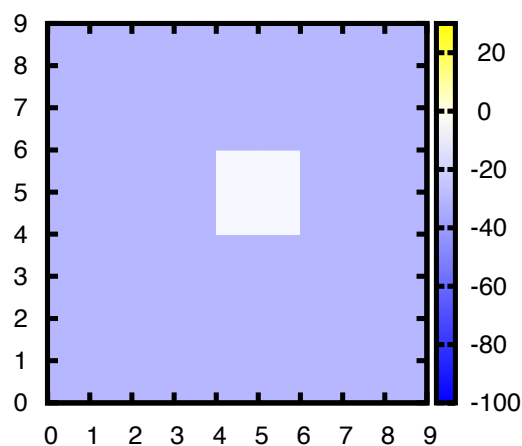


Figura 29: Propagazione della differenza di potenziale in un tessuto in configurazione SQUARE, 0 ms.

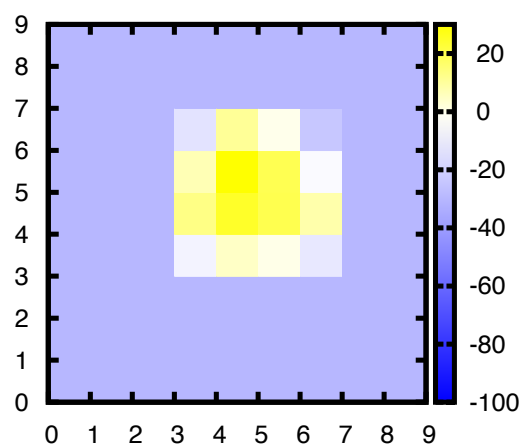


Figura 30: Propagazione della differenza di potenziale in un tessuto in configurazione SQUARE, 1 ms.

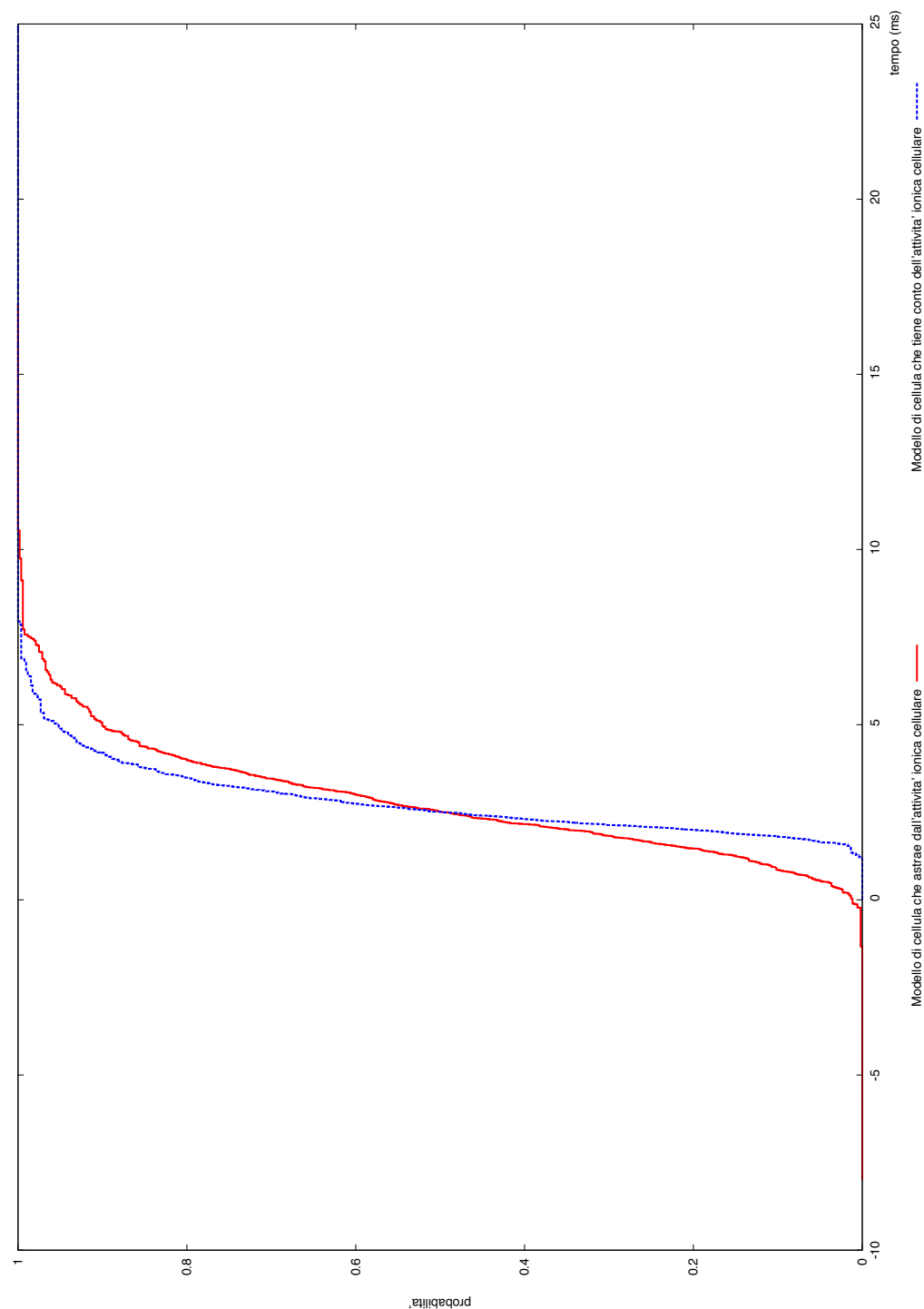


Figura 31: Probabilità per i due modelli che venga raggiunta l'eccitazione al variare del tempo.

BREVI APPROFONDIMENTI CONCLUSIVI

Dal 2000 in poi il termine Biologia Computazionale è stato usato in tutte le bioscienze in molti contesti, quindi non è così facile attribuirgli un'unica accezione. Un'ambizione molto marcata della Biologia Computazionale è quella di tentare di scoprire aspetti che coinvolgono buona parte delle entità in gioco, permettendo lo sviluppo di un comportamento molto più complesso, la cui descrizione teorica è ignota, risultante dall'analisi delle singole entità.

BioKLAIM ha lo scopo di essere di supporto per la Biologia Computazionale, permettendo la rappresentazione di un sistema come un'interazione tra componenti. Altri approcci seguiti per dare descrizioni di sistemi complessi sono gli Hybrid Automata [9, 10], i P-Systems [11], SBML [12] e Beta Binders [13].

5.1 COSA È LA BIOLOGIA COMPUTAZIONALE

La Biologia Computazionale può essere considerata da diversi punti di vista:

- un campo di studio: lo studio delle interazioni tra le componenti dei sistemi biologici, di come queste interazioni diano luogo alla funzione e al comportamento di tale sistema.
- un paradigma: messo in antitesi rispetto al paradigma riduzionista (un approccio a comprendere un sistema complesso riducendolo a

interazioni di sue parti, dove i risultati delle parti contribuiscono al risultato del sistema che le comprende) ma pienamente coerente con il metodo scientifico.

Per comprendere la distinzione tra i due paradigmi è sufficiente leggere queste citazioni:

“L’approccio riduzionista ha identificato con successo la maggior parte delle componenti e molte delle interazioni, ma, purtroppo, non offre convincenti concetti o metodi per capire come emergono le proprietà del sistema ... il pluralismo delle cause e degli effetti biologici nelle reti è meglio affrontato osservando, attraverso misure quantitative, più componenti simultaneamente e da una rigorosa integrazione dei dati con i modelli”

(Science)

“La System Biology¹ ... è del mettere insieme, piuttosto che del prendere a parte, l’integrazione piuttosto che di riduzione. Essa esige che si sviluppino modi di pensare l’integrazione, che siano altrettanto rigorosi come i nostri paradigmi riduzionisti, ma in modo diverso.”[54]

(Denis Nobel)

- una serie di protocolli operativi utilizzati per eseguire la ricerca: un ciclo composto da teoria, costruzione analitica o computazionale (per proporre specifiche ipotesi verificabili su di un sistema

¹ Disciplina che si interessa dei circuiti regolativi delle grandi funzioni biologiche, intese come dovute all’interazione di componenti di tali circuiti i quali, interagendo fra di loro, generano la funzione del sistema. La Biologia Computazionale appartiene a questa macro-disciplina.

biologico) validazione sperimentale, e, infine, il perfezionamento del modello di partenza utilizzando i risultati ottenuti.

- un'applicazione della teoria dei sistemi dinamici alla Biologia Molecolare.
- un fenomeno socio-scientifico: definito dalla strategia di perseguire l'integrazione di dati complessi attraverso interazioni di sistemi biologici provenienti da diverse fonti sperimentali, usando strumenti e personale interdisciplinare.

Questa varietà di punti di vista è esemplificativa del fatto che la Biologia Computazionale si riferisce a un cluster di concetti periferici e sovrapposti piuttosto che a un unico e ben delineato settore. Tuttavia il termine ha guadagnato popolarità nel 2007, negli istituti di Biologia, Informatica, Matematica e Fisica di tutto il mondo.

5.1.1 *Tecniche*

Poiché la Biologia Computazionale viene intesa come la capacità di ottenere, integrare e analizzare dati complessi provenienti da fonti sperimentali molteplici, tramite strumenti interdisciplinari, gli ambienti disciplinari tipici di questa materia sono, ad esempio, la Genomica, l'Epigenetica e la Trascrittomia: materie che descrivono e studiano il genoma² di un organismo sottoposto a stimoli, come questo possa subire modifiche e come

² Nella moderna Biologia Molecolare, il genoma è l'informazione ereditaria nella sua interezza che caratterizza questo. È codificato in DNA e, per molti tipi di virus, in RNA. Per capire meglio è possibile immaginarsi un'analogia tra il genoma umano, presente nel DNA, e un insieme di istruzioni presenti in una biblioteca contenente 46 libri (i cromosomi), dove ogni libro ha una dimensione variabile dalle 400 alle 3340 pagine (i geni) ed è composto da 48 a 250 milioni di lettere (adenina, timina, guanina e citosina,

venga trascritto per ottenere gli elementi “base” utilizzati dalle cellule. I dati restituiti da queste discipline vengono usati come valori per costruire, per mezzo della Biologia Computazionale, sistemi per analizzare il comportamento dei fenomeni da studiare. La Biologia Computazionale implica l’uso di simulazioni di sistemi biologici, come sottosistemi cellulari, per analizzare e visualizzare le connessioni di questi complessi processi cellulari.

Ultimamente (20 maggio 2010) l’operato in questo settore ha portato alla progettazione, alla sintesi e alla costruzione di cellule capaci di autoreplicarsi. Questo studio, descritto in [55], annuncia la costruzione della prima cellula batterica artificiale, risultato importante dal punto di vista scientifico. La cellula è di origine naturale (spesso vengono impiegati batteri) ma è completamente controllata da un DNA artificiale.

Il punto di arrivo sarà, molto probabilmente, una forma vivente interamente costruita in laboratorio e programmata per una funzione precisa: le cellule artificiali potrebbero essere impiegate per creare dei batteri “salva ambiente”, da utilizzare per liberare le acque e i terreni da sostanze inquinanti, oppure per assorbire anidride carbonica.

5.2 METODOLOGIA UTILIZZATA

Dal momento che questa tesi tratta un argomento interdisciplinare, spaziando tra Informatica e Biologia, il metodo per sviluppare i modelli proposti è stato quello scientifico induttivo.

Il metodo per sviluppare i modelli proposti è stato quello scientifico induttivo e il procedimento si è svolto ciclicamente attraverso i seguenti

le basi azotate). La biblioteca contiene più di 6 miliardi di lettere in totale e risiede nel nucleo di quasi ogni cellula presente nel corpo umano.

passaggi:

1. osservazione e ricerca di informazioni in letteratura;
2. raccolta dei concetti utili alla risoluzione del problema affrontato;
3. correlazione fra le grandezze in gioco nel sistema;
4. definizione di un modello su cui poter lavorare;
5. creazione di un modello matematico;
6. analisi dei risultati. Nel caso di esito negativo viene reiterato il processo facendo le dovute modifiche, altrimenti vengono formalizzati i risultati in una teoria.

Nella situazione presentata, per le prime due fasi, è stata utile la collaborazione con la Dott.ssa Daiana Minocci³, che, studiando il sistema endocannabinoide, ha reperito parte della letteratura e chiarito alcuni aspetti critici, utili alla descrizione dei modelli per il recettore CB2.

Per quanto riguarda le cellule eccitabili, le prime due fasi sono risultate meno complesse, in generale in letteratura sono presenti molti più studi e fonti. Lo studio delle cellule eccitabili, data la loro forte presenza in tutte le funzioni vitali, è stato analizzato in modo molto più approfondito rispetto allo studio del sistema endocannabinoide che, come già accennato, ha ricevuto interesse da parte della comunità scientifica soltanto da poco tempo.

³ Dottoranda in Neurobiologia e Clinica dei Disturbi Affettivi. Dipartimento di Psichiatria, Neurobiologia, Farmacologia e Biotecnologie, Università di Pisa.

5.3 DESCRIVERE SISTEMI CON BIOKLAIM

Il paradigma di trattare le molecole come tuple permette di poter usare i modelli descritti come sottosistemi nella costruzione di un modello più complesso, senza dover riscrivere le interazioni già definite. Tale tipo di approccio, consente a chi deve sviluppare e mantenere progetti anche di grandi dimensioni, di avere sempre sotto controllo tutti gli elementi, e le reazioni chimiche, interessati dallo studio in atto.

Un altro aspetto interessante risiede nelle guardie: utilizzandole è possibile avere il completo controllo sull'attivazione delle azioni dei processi, producendo modelli più accurati e simulazioni che rispecchino, in modo più preciso, la realtà da analizzare. Così risulta possibile regolare l'ingresso di nuovi elementi nel sistema, ad esempio: nei modelli che descrivono tessuti di cellule eccitabili, sarebbe possibile regolare la stimolazione di alcune cellule in base al raggiungimento di una condizione legata alla configurazione dell'intero sistema, descrivendo in questo modo la regolare stimolazione che si sviluppa nel tessuto muscolare del cuore.

Infine BIOKLAIM, oltre a simulare il modello sottoposto, dà la possibilità di verificare alcune proprietà utilizzando una logica che garantisca l'assenza di ambiguità in ciò che viene verificato. In questa maniera gli aspetti critici, da tenere sotto controllo, vengono analizzati con uno strumento potente e sicuro.

BIBLIOGRAFIA

- [1] W. Fontana and L. W. Buss, "'the arrival of the fittest': Toward a theory of biological organization.," *Bulletin of Mathematical Biology*, 1994. (3)
- [2] W. Fontana and L. W. Buss, "What would be conserved 'if the tape were played twice'.," *Proceedings of the National Academy of Sciences*, 1994. (3)
- [3] F. Varela, H. Maturana, and R. Uribe, "Autopoiesis: the organization of living systems, its characterization and a model," *BioSystems*, 1974. (3)
- [4] S. J. Gould, *This Wonderful Life*. W.W. Norton, 1989. (3)
- [5] F. Zhang, K. Vadakkan, S. Kim, L. J. Wu, Y. Shang, and M. Zhuo, "Selective activation of microglia in spinal cord but not higher cortical regions following nerve injury in adult mouse," *Molecular Pain*, 2008. (5)
- [6] G. Anastasi, S. Capitani, M. L. Carnazza, S. Cinti, R. D. Caro, R. F. Donato, V. F. Ferrario, L. Fonzi, A. T. Franzi, E. Gaudio, R. Geremia, G. G. Lanza, C. E. Grossi, F. E. Manzoli, G. Mazzotti, F. Michetti, S. Miscia, V. Mitolo, A. Montella, G. Orlandini, A. Paparelli, T. Renda, D. Ribatti, A. Ruggeri, P. Sirigu, A. Soscia, G. Tredici, M. Vitale, D. Zaccheo, G. Zauli, and S. Zecchi, *Trattato di anatomia umana*. Edi-Ermes, 2007. (4 e 7)

- [7] L. Bettini, V. Bono, R. De Nicola, G. Ferrari, D. Gorla, M. Loretì, E. Moggi, R. Pugliese, E. Tuosto, and B. Venneri, "The Klaim Project: Theory and Practice," in *Global Computing - Programming Environments, Languages, Security and Analysis of Systems*, vol. 2874 of LNCS, Springer-Verlag, 2003. (6 e 36)
- [8] M. H. Meisler and J. A. Kearney, "Sodium channel mutations in epilepsy and other neurological disorders," *Journal of Clinical Investigation*, 2005. (8)
- [9] T. A. Henzinger, "The theory of hybrid automata," *LICS*, pp. 278–292, 1996. (10 e 103)
- [10] N. A. Lynch, R. Segala, and F. W. Vaandrager, "Hybrid i/o automata," *Inf. Comput.*, vol. 185, pp. 105–157, 2003. (10 e 103)
- [11] G. Paun, "Computing with membranes," tech. rep., Turku Center of Computer Science, 1998. (10 e 103)
- [12] M. Hucka, A. Finney, H. M. Sauro, H. Bolouri, J. C. Doyle, and H. Kitano, "The systems biology markup language (sbml): a medium for representation and exchange of biochemical network models," *Bioinformatics*, 2003. (10 e 103)
- [13] C. Priami and P. Quaglia, "Beta binders for biological interactions," in *CMSB*, pp. 20–33, 2004. (10 e 103)
- [14] A. Hodgink and A. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," *The Journal of Physiology*, vol. 117, pp. 500–544, 1952. (11)
- [15] L. E. Limbird, *Cell Surface Receptors: A Short Course on Theory and Method*. M. Nijhoff, Boston, 1995. (14)

- [16] T. P. Kenakin, R. A. Bond, and T. I. Bonner, *Definition of pharmacological receptors*. Pharmacol. Rev., 1992. (14)
- [17] T. Sugiura, Y. Kobayashi, S. Oka, and K. Waku, "Biosynthesis and degradation of anandamide and 2-arachidonoylglycerol and their possible physiological significance," *Elsevier Science*, 2001. (16, 76, e 82)
- [18] J. Z. Long, D. K. Nomura, R. E. Vann, D. M. Walentiny, L. Booker, X. Jin, J. J. Burston, L. J. Sim-Selley, A. H. Lichtman, J. L. Wiley, and B. F. Cravatt, "Dual blockade of faah and magl identifies behavioral processes regulated by endocannabinoid crosstalk in vivo," *Proceedings of the National Academy of Sciences*, 2009. (18)
- [19] R. Kawata, S. Hyo, M. Araki, and H. Takenaka, "Expression of cyclooxygenase-2 and microsomal prostaglandin e synthase-1 in head and neck squamous cell carcinoma," *Auris Nasus Larynx*, 2009. (19)
- [20] F. He, Z. H. Qiao, J. Cai, W. Pierce, D. C. He, and Z. H. Song, "Involvement of the 90-kda heat shock protein (hsp-90) in cb2 cannabinoid receptor-mediated cell migration: A new role of hsp in migration signaling of a g protein-coupled receptor," *Molecular Pharmacology*, 2007. (19)
- [21] S. A. Rieder, A. Chauhan, U. Singh, M. Nagarkatti, and P. Nagarkatti, "Cannabinoid-induced apoptosis in immune cells as a pathway to immunosuppression.," *Elsevier Science*, 2009. (20)
- [22] M. Karsak, M. Cohen-Solal, J. Freudenberg, A. Ostertag, C. Morieux, U. Kornak, J. Essig, E. Erxlebe, I. Bab, C. Kubisch, M. C. de Vernejoul,

- and A. Zimmer, "Cannabinoid receptor type 2 gene is associated with human osteoporosis," *Human Molecular Genetics*, 2005. (21)
- [23] E. S. Onaivi, H. Ishiguro, J. P. Gong, S. Patel, P. A. Meozzi, L. Myers, A. Perchuk, Z. Mora, P. A. Tagliaferro, E. Gardner, A. Brusco, B. E. Akinshola, B. Hope, J. Lujilde, T. Inada, S. Iwasaki, D. Macharia, L. Teasenfitz, T. Arinami, and G. R. Uhl, "Brain neuronal cb2 cannabinoid receptors in drug abuse and depression: From mice to human subjects," *PLoS one*, 2008. (21)
- [24] M. F. Bear, B. W. Connors, and M. A. Paradiso, *Neuroscienze*. Elsevier, 2007. (23, 92, e 96)
- [25] D. C. Camerino, D. Tricarico, and J. F. Desaphy, "Ion channel pharmacology," *Neurotherapeutics*, 2007. (23)
- [26] A. S. Verkan and L. J. Galletta, "Chloride channels as drug targets," *Nat Rev Drug Discov*, 2009. (23)
- [27] D. C. Camerino, J. F. Desaphy, D. Tricarico, S. Pierno, and A. Liantonio, "Theraupetic approaches to ion channel diseases," *Adv. Genet.*, 2008. (23)
- [28] R. Milner, *A Calculus of Communicating Systems.*, vol. 92 of *Lecture Notes in Computer Science*. Springer-Verlag, 1980. (35)
- [29] R. Milner, *Communication and concurrency*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1989. (35)
- [30] S. D. Brookes, C. A. R. Hoare, and A. W. Roscoe, "A theory of communicating sequential processes," *J. ACM*, vol. 31, no. 3, pp. 560–599, 1984. (35)

- [31] J. Bergstra and J. Klop, "Process algebra for synchronous communication.," *Information and Control*, vol. 60, no. 1-3, pp. 109–137, 1984. (35)
- [32] L. Boulianne, M. Dumontier, and W. J. Gross, "A stochastic particle-based biological system simulator," in *SCSC: Proceedings of the 2007 summer computer simulation conference*, (San Diego, CA, USA), pp. 794–801, Society for Computer Simulation International, 2007. (35)
- [33] M. Calder and J. Hillston, "Process algebra modelling styles for biomolecular processes," *T. Comp. Sys. Biology*, vol. 11, no. 5750, pp. 1–25, 2009. (35)
- [34] Grima, Ramon, Schnell, and Santiago, "A mesoscopic simulation approach for modeling intracellular reactions," *Journal of Statistical Physics*, vol. 128, pp. 139–164, July 2007. (35)
- [35] C. Priami, A. Regev, E. Y. Shapiro, and W. Silverman, "Application of a stochastic name-passing calculus to representation and simulation of molecular processes," *Inf. Process. Lett.*, vol. 80, no. 1, pp. 25–31, 2001. (35)
- [36] D. Gelernter, "Generative communication in linda," *ACM Trans. Program. Lang. Syst.*, vol. 7, no. 1, pp. 80–112, 1985. (36)
- [37] R. De Nicola, J.-P. Katoen, D. Latella, M. Loret, and M. Massink, "Klaim and its Stochastic Semantics," tech. rep., 2006. (38)
- [38] V. Kulkarni, *Modelling and Analysis of Stochastic Systems*. Chapman & Hall, 1995. (38)
- [39] R. De Nicola, D. Latella, M. Loret, and M. Massink, "Rate-Based Transition Systems for Stochastic Process Calculi," *Automata, Language-*

- ges and Programming, 36th Internatilonal Collogquium, ICALP 2009, Rhodes, greece, July 5-12, 2009, Proceedings, Part II*, vol. 5556, pp. 435–446, 2009. (38)
- [40] B. Klin and V. Sassone, *Structural operational semantics for process calculi*. Springer, 2008. (38 e 41)
- [41] R. De Nicola, D. Latella, M. Loreti, and M. Massink, “On a uniform framework for the definition of stochastic process languages,” *LNCS vol.5825*, pp. 9–25, 2009. (41)
- [42] R. De Nicola, J.-P. Katoen, D. Latella, M. Loreti, and M. Massink, “Model checking mobile stochastic logic,” *Elsevier Science*, 2007. (49, 56, e 69)
- [43] A. Aziz, K. Sanwal, V. Singhal, and R. Brayton, “Model checking continuous time markov chains,” *ACM Transactions on Computational Logic 1*, pp. 162–170, 2000. (49)
- [44] C. Baier, J.-P. Katoen, and H. Hermanns, *Approximate symbolic model checking of continuous-time Markov chains*. Springer-Verlag, 1999. (49)
- [45] R. De Nicola, J.-P. Katoen, D. Latella, M. Loreti, and M. Massink, “Mosl: A stochastic logic for stoklaim,” Tech. Rep. ISTI-06-35, ISTI-CNR, 2006. (50)
- [46] H. Hermanns, J.-P. Katoen, J. Meyer-Kayser, and M. Siegle, *Towards model checking stochastic process algebra*, vol. LNCS 1945. Springer-Verlag, 2000. (50)
- [47] R. De Nicola and M. Loreti, “MoMo: A Modal Logic for Reasoning About Mobility,” in *FMCO*, pp. 95–119, 2004. (50)

- [48] G. Griffin, Q. Tao, and M. E. Abood, "Cloning and pharmacological characterization of the rat cb2 cannabinoid receptor," *The Journal Of Pharmacology and Experimental Therapeutics*, 1999. (76 e 82)
- [49] L. E. Keshet, *Mathematical Models in Biology*. Society for Industrial and Applied Mathematics, 2004. (76 e 82)
- [50] H. Lodish, A. Berk, S. Zipursky, P. Matsudaira, D. Baltimore, and J. Darnell, *Molecular Cell Biology*. Freeman @ Company Limited, W H, 2003. (76 e 82)
- [51] T. Dudev and C. Lim, "Determinants of k^+ and na^+ selectivity in potassium channles," *Journal of the American Chemical Society*, 2009. (92)
- [52] S. I. Borjesson, T. Parkkari, S. Hammarstrom, and F. Elinder, "Electrostatic tuning of cellular excitability," *Biophysical Journal*, vol. 98, pp. 396–403, 2010. (92)
- [53] E. Bartocci, *A Formal Framework for Modeling, Simulating and Analyzing Networks of Excitable Cells*. PhD thesis, Università degli studi di Camerino, 2008. (96)
- [54] D. Noble, *The Music of Life: Biology beyond the genome*. Oxford University Press, 2006. (104)
- [55] D. G. Gibson, J. I. Glass, C. Lartigue, V. N. Noskov, R.-Y. Chuang, M. A. Algire, G. A. Benders, M. G. Montague, L. Ma, M. Moodie, C. Merryman, S. Vashee, R. Krishnakumar, N. Assad-Garcia, C. Andrews-Pfannkoch, E. A. Denisova, L. Young, Z.-Q. Qi, T. H. Segall-Shapiro, C. H. Calvey, P. P. Parmar, C. A. Hutchison, III, H. O.

- Smith, and J. C. Venter, "Creation of a bacterial cell controlled by a chemically synthesized genome," *Scienceexpress*, 2010. (106)
- [56] L. Cardelli, "Brane calculi," *Proceedings of Workshop on Computational Methods in Systems Biology*, vol. 3082, pp. 257–278, 2005.
- [57] P. K. Janert, *Gnuplot in Action: Understanding Data with Graphs*. Manning, 2010.
- [58] A. D. Baxevanis and B. F. F. Ouellette, *Bioinformatics: A Pratical Guide to the Analysis of Gene and Proteins*. Wiley Publishing, Inc, 2005.
- [59] J. L. Snoep and H. V. Westerhoff, *From isolation to integration, a systems biology approach for building the Silicon Cell*. Springer, 2005.
- [60] T. F. Smith and M. S. Waterman, "Identification of common molecular subsequences," *Journal of Molecular Biology*, 1981.
- [61] V. Sperschneider, *Bioinformatics, Problem Solving Paradigms*. Springer, 2008.
- [62] J. M. Claverie and C. Notredame, *Bioinformatics For Dummies*. Wiley Publishing, Inc, 2007.