

Programación de Escenarios Biológicos en Nanocomunicaciones Moleculares



- Yesenia Cevallos
- Paola G. Vinuesa-Naranjo
- Deysi Inca B.
- Elvis Pilligua
- Albert Espinal
- Elba Bodero-Poveda
- Andrés Santiago Cisneros-Barahona
- María Isabel Uvidia-Fassler
- Diana Coello-Fiallos
- Cristian Vacacela Gómez
- Talía Tene
- Ramiro F. Isa-Jara
- Karina Inés Paredes-Páliz

Programación de Escenarios Biológicos en Nanocomunicaciones Moleculares

Primera Edición

Yesenia Cevallos
Paola G. Vinueza-Naranjo
Deysi Inca
Elvis Pilligua
Albert Espinal
Elba Boderó-Poveda
Andrés Santiago Cisneros-Barahona
María Isabel Uvidia-Fassler
Diana Coello-Fiallos
Cristian Vacacela Gómez
Talía Tene
Ramiro F. Isa-Jara
Karina Inés Paredes-Páiz

Revisión Técnica:

Editorial CEDIA

Corrección de Estilo:

Editorial CEDIA

Diseño de portada: José Luis Hermida

Diagramación: Gabriela Gómez Siranaula

Coordinación:

Laura Malache S. - Editorial CEDIA

Una publicación de la Editorial CEDIA,
arbitrada por pares académicos de doble ciego.

The logo for CEDIA, consisting of the word "cedia" in a bold, lowercase, blue sans-serif font.**CEDIA**

Gonzalo Cordero 2-111 y
J. Fajardo
Cuenca - Ecuador
cedia.edu.ec

Primera edición

ISBN: 978-9942-7317-4-6

DOI: <https://doi.org/10.48661/k9vx-6017>

Citación:

Cevallos, Y., Vinueza Naranjo, P. G., Inca, D., Pilligua-Vélez, E., Espinal, A., Boderó-Poveda, E. M., Cisneros Barahona, A. S., Uvidia Fassler, M. I., Coello Fiallos, D. C., Vacacela Gómez, C., Tene, T., Isa-Jara, R. F., & Paredes Páliz, K. I. (2026). Programación de escenarios biológicos en nanocomunicaciones moleculares (Versión 1). Cuenca, Ecuador: CEDIA. <https://doi.org/10.48661/k9vx-6017>

Riobamba, Ecuador
Abril de 2026

Supervisores Técnicos en las áreas de especialidad de esta obra:

Jesús Alonso, Norberto M. Ramos-Calero
Universidad de las Palmas de Gran Canaria, España

Jennyfer Granizo
Hospital Metropolitano, Ecuador

Esmeralda Terán, Fabián Cruz
Universidad de las Palmas de Gran Canaria, España

Yoskira Naylett Cordero de Jiménez
Universidad Tecnológica Empresarial de Guayaquil, Ecuador

Danilo Chávez
Escuela Politécnica Nacional, Ecuador

Román Alcides Lara-Cueva
Universidad de las Fuerzas Armadas, Ecuador

Edwin Valarezo Añazco
Escuela Superior Politécnica del Litoral

Silvana Guitarra, Luis Miguel Procel
**Instituto de Micro y Nanoelectrónica,
Universidad San Francisco de Quito, Ecuador**

Alejandra Pozo, Daniel Haro, Lorena Molina, Blanca Maygualema,
Carlos Ramiro Peñafiel-Ojeda, Antonio Meneses
Universidad Nacional Chimborazo, Ecuador

Todos los derechos reservados.

Prohibida la reproducción total o parcial de esta obra, por cualquier medio o método sin autorización por escrito del editor.

Yesenia Cevallos

Ingeniera en Electrónica y Telecomunicaciones,
Escuela Politécnica Nacional (Ecuador)

Magister en Tecnologías de la Información,
Universidad Técnica de Ambato (Ecuador)

Doctora en Ciencias y Tecnologías de los Sistemas Complejos,
Università della Calabria (Italia)

Miembro Senior Computational Life Sciences,
Nanotechnology Council, IEEE (USA)

Miembro del grupo internacional de trabajo para la aprobación del
estándar de nanocomunicaciones IEEE 1906.1.1

Representante internacional ante el Comité Técnico Internacional

Molecular, Biological and Multi-Scale
Communications, IEEE (USA, Europa, Asia)

Par Revisora, SPRINGER e IEEE

Par Revisora/Editora, FRONTIER IN BIOENGINEERING

CCNA, CCNAI, CISCO SYSTEMS (USA)

Docente/Investigadora, Universidad Nacional
de Chimborazo (Ecuador)

Profesora invitada no titular, Instituto de Micro y Nanoelectrónica,
Universidad San Francisco de Quito (Ecuador)

Investigadora externa, Universidad ECOTEC (Ecuador)

Investigadora no titular invitada, Instituto Universitario para el Desarrollo
Tecnológico y la Innovación en Comunicaciones (IDeTIC), Universidad de
las Palmas de Gran Canaria (España)

Paola G. Vinueza-Naranjo

Ingeniera en Sistemas y Computación,
Universidad Nacional de Chimborazo (Ecuador)

Doctora en Telecomunicaciones, Università La Sapienza di Roma (Italia)

Editora Académica. IEEE-UNACH Council

Par Revisora, SPRINGER, ELSEVIER e IEEE

Docente/Investigadora, Universidad Nacional de Chimborazo (Ecuador)

Investigadora invitada no titular, Grupo de Investigación ETEL,
Universidad de las Américas (Ecuador)

Doctorando en Nanocomunicaciones Moleculares, Instituto Universitario
para el Desarrollo Tecnológico y la Innovación en Comunicaciones
(IDeTIC), Universidad de las Palmas de Gran Canaria (España)

Deysi Inca

Ingeniera en Electrónica y Computación,
Escuela Superior Politécnica del Chimborazo (Ecuador)

Magister en Comunicación de Redes,
Pontificia Universidad Católica (Ecuador)

Docente/Investigadora, Universidad Nacional de Chimborazo (Ecuador)

Elvis Pilligua

Ingeniero en Electrónica y Telecomunicaciones,
Universidad Nacional de Chimborazo (Ecuador)

Doctorando en Nanocomunicaciones Moleculares, Instituto Universitario
para el Desarrollo Tecnológico y la Innovación en Comunicaciones
(IDeTIC), Universidad de las Palmas de Gran Canaria (España)

Albert Espinal

Ingeniero en Computación, Escuela Superior
Politécnica del Litoral (Ecuador)

Magister en Sistemas de Información Gerencial,
Escuela Superior Politécnica del Litoral (Ecuador)

Doctor en Ingeniería, Universidad Nacional de Cuyo (Argentina)

CCNA, CCNP, CCAI, Cisco Networking Academy

Main contact Cisco ASC / ITC, Corporación Ecuador Sin Brecha Digital.
CEO, SOL & TECH Soluciones y Tecnología

Docente/Investigador, Escuela Superior Politécnica del Litoral (Ecuador)

Elba Boderó-Poveda

Ingeniera en Sistemas Informáticos,
Escuela Superior Politécnica de Chimborazo (Ecuador)

Magíster en Tecnologías de la Información,
Universidad Técnica de Ambato (Ecuador)

Doctora Ph.D. en Ciencias Informáticas,
Universidad Nacional de La Plata (Argentina)

Miembro distinguido, Junta Directiva del
Consortio Iberoamericano de Ciencia y

Tecnología - ISTEAC, Universidad Nacional de La Plata (Estados Unidos)
Editora asociada, Revista Digital de Ciencia,

Ingeniería y Tecnología NOVASINERGIA,
Facultad de Ingeniería, UNACH (Ecuador)

Editora jefa, Revista ESPRINT, Investigación (Ecuador)

ESPRINT, ISSN 2960-8317 (Ecuador)

Docente/investigadora, Universidad Nacional de Chimborazo (Ecuador)

Andrés Santiago Cisneros-Barahona

Ingeniero en Electrónica y Computación,
Escuela Superior Politécnica de Chimborazo (Ecuador)

Magister en Interconectividad de Redes,
Escuela Superior Politécnica de Chimborazo (Ecuador)

Doctor en Tecnología Educativa, Universitat Rovira i Virgili (España)

Investigador, Centro de Investigación en Modelos de
Gestión y Sistemas de Información (CIMOGSYS),
Escuela Superior Politécnica de Chimborazo (Ecuador)

Investigador, Centro de Investigación en Ingeniería, Diseño y Saberes An-
cestrales (CIIDSA), Universidad Nacional de Chimborazo (Ecuador)

Docente/Investigador, Universidad Nacional de Chimborazo (Ecuador)

María Isabel Uvidia-Fassler

Ingeniera en Sistemas Informáticos,
Escuela Superior Politécnica del Chimborazo (Ecuador)

Magister en Gerencia Informática,
Pontificia Universidad Católica (Ecuador)

Máster Universitario en Análisis y Visualización de Datos
Masivos / Visual Analytics and Big Data,
Universidad Internacional de la Rioja (España)

Docente/Investigadora, Universidad
Nacional de Chimborazo (Ecuador)

Diana Coello-Fiallos

Biofísica, Escuela Superior Politécnica de Chimborazo (Ecuador)
PhD en Física y Tecnologías Cuánticas, Università Della Calabria (Italia)
Investigadora Senior, Escuela Superior
Miembro del Surface Nanoscience Group (Italia)
Miembro del Instituto per la Tecnologie delle Membrane del CNR (Italia)
Profesora ocasional tiempo completo, Escuela Superior Politécnica
de Chimborazo (Ecuador)

Cristian Vacacela Gómez

Biofísico, Escuela Superior Politécnica de Chimborazo (Ecuador)
Doctor en Física y Tecnologías Cuánticas, Università della Calabria (Italia)
Docente/Investigador, Escuela Superior Politécnica de
Chimborazo (Ecuador)
Investigador Asociado II, Universidad ECOTEC (Ecuador)
Editor Académico/Revisor en Revistas Internacionales
de Nanomateriales y Nanocomunicaciones

Talía Tene

Biofísica, Escuela Superior Politécnica de Chimborazo (Ecuador)
Doctora en Física y Tecnologías Cuánticas,
Università della Calabria (Italia)
Docente/Investigadora, Universidad Técnica Particular de Loja (Ecuador)

Ramiro F. Isa-Jara

Ingeniero en Electrónica, Control y Redes Industriales, Escuela Superior Politécnica de Chimborazo (Ecuador)

Doctor en Ingeniería orientación Electrónica,
Universidad Nacional de Mar del Plata (Argentina)

Posdoctorado en Desarrollo Tecnológico y Social de Proyectos Complejos,
Universidad Tecnológica Nacional (Argentina)

Profesor e Investigador,
Escuela Superior Politécnica de Chimborazo (Ecuador)

Director del Laboratorio de IA "A.M. Turing" y Concejero de IEEE - EMBS,
Escuela Superior Politécnica de Chimborazo (Ecuador)

Investigador externo, Centro IREN de la
Universidad Tecnológica Nacional (Argentina)

Karina Inés Paredes Páliz

Licenciada en Ciencias Biológicas,
Pontificia Universidad Católica del Ecuador (Ecuador)

Máster en Genética Molecular y Biotecnología,
Universidad de Sevilla (España)

Doctora en Biología Molecular y Biomedicina,
Universidad de Sevilla (España)

Docente/investigadora, Universidad Nacional de Chimborazo (Ecuador)

Dedicación y agradecimiento:

*A Dios, por sustentar nuestras vidas en
cada instante en todo...*

*Y a nuestros seres queridos, por su amor y
apoyo constantes...*

ÍNDICE

01

Pg. 22

Redes de Área Corporal BAN (Body Area Network)

- 1.1 Análisis general de las nanocomunicaciones en redes BAN
- 1.2 Redes de área corporal BAN
- 1.3 Nanomedicina y redes BAN²

02

Pg. 70

Entornos de programación para el análisis y simulación en Comunicaciones Moleculares

- 2.1 Contexto del uso de herramientas digitales en el análisis de Comunicaciones Moleculares
- 2.2 Lenguajes de programación basados en XML (eXtensible Markup Language) para Comunicaciones Moleculares
- 2.3 Simuladores de uso común en Comunicaciones Moleculares

03

Pg. 128

Lenguaje Extensible de Marcado para Comunicaciones Moleculares (MolComML- Molecular Communications Markup Language)

- 3.1 Plataformas de programación para el análisis, simulación y experimentación en Comunicaciones Moleculares
- 3.2 Análisis del Lenguaje Extensible de Marcado para Comunicaciones Moleculares MolComML
- 3.3 Experimentación en MolComML
- 3.4 Casos de estudio de simulaciones en MolComML

PRÓLOGO

Las Comunicaciones Moleculares (CM) son una fascinante arista de investigaciónⁱ en la que la biología, la ingeniería en telecomunicaciones (particularmente los sistemas de networking) y la estadística confluyen para proponer modelos comunicacionales sustentados por principios matemáticos, en base a simulaciones y la práctica experimental pertinente, con el fin de obtener aplicaciones que permitan un nuevo paradigma en varias áreas, entre las que se encuentran la industria, el medio ambiente, la seguridad militar, la ciencia de los nanomateriales, y principalmente la medicina. Actualmente la idea de combinar el potencial establecido mediante la conectividad entre personas, máquinas, y dispositivos genera entornos digitales de control y administración de sistemas electrónicos micro/macro en forma remota a través de Internet y eventualmente tecnologías futuras 6G y 7G+. Así, se han originado sistemas de comunicación con aplicaciones avanzadas y automatizadas que involucran la omnipresencia de la tecnología en las actividades cotidianas de las personas. Adicional al escenario mencionado, mecanismos de conectividad con el mundo biológico a nanoescala se han determinado para transformar el interior de un ser vivo en una fuente de transmisión/recepción de información, posibilitando aplicaciones médicas de una connotación tan impresionante que en tiempos pasados se hubieran considerado ficticias.

El uso de nanobots (o nanomáquinas) biológicos, artificiales o bio-híbridos inyectados dentro de un ser viviente permite la detección y actuación de señales de información biológicas reportadas a dispositivos de transmisión a microescala, tales dispositivos a su vez alertan a conmutadores de gama macro, de modo que mediante estos sistemas de comunicación la calidad de vida de pacientes cuyo estado de salud debe ser monitoreado permanentemente pueda mejorar enormemente.

ⁱ Project description: [MAMOKO – Vernetzung und Sicherheit digitaler Systeme \(forschung-it-sicherheit-kommunikationssysteme.de\)](https://www.forschung-it-sicherheit-kommunikationssysteme.de).

Consecuentemente, las TIC (Tecnologías de Información y Comunicación) como una herramienta en la medicina se enfocan en la actualidad en la adquisición, almacenamiento, análisis, modelado e intercambio de información mediante plataformas computacionales y de Internet. Las CM son capaces de detectar y controlar tal información y su propagación *in-vivo* a través de señales bioquímicas en el cuerpo del paciente, específicamente en sus células y su contenido. Este es el caso en el desarrollo y propagación del cáncer, por ejemplo, el cual según se cree, se produce por anomalías en la transmisión de información molecular por la diferenciación y proliferación celular. Al respecto, la ciencia se encuentra en los inicios de soluciones a estos problemas gracias a la equivalencia entre las teorías informacionales de los sistemas de comunicación convencionales y los sistemas de CM, de concretarse dichos hechos se obtendrían uno de los mayores avances tecnológicos del siglo XXI.

Como es notorio, las CM requieren el uso conjunto de otras disciplinas científicas para que sus avances tecnológicos puedan tener aplicaciones reales y de verdadero beneficio. Por tanto, así han aparecido las redes BAN (Body Area Network) las cuales están constituidas por las denominadas redes in-body y las WBAN (Wireless Body Area Network). Las primeras se refieren a los componentes comunicacionales de un sistema de transmisión molecular, en el interior del cuerpo humano en tanto las segundas hacen alusión a la conectividad entre un dispositivo central (hub) y sensores (muy ligeros en cuanto al peso se refiere y de bajo consumo energético) funcionando muy cerca del cuerpo, ya sea en prendas de vestir, accesorios colocados bajo la piel o de tejidos.

Los avances de la nano y biotecnología han sustentado el empleo de nanomáquinas con fines terapéuticos que operan a nivel intra o intercelular (donde los sistemas de CM gobiernan) en el cuerpo humano. Las redes de comunicación formadas mediante las mencionadas nanomáquinas BANⁱⁱ (Body Area Nanonetworks) permiten aplicaciones muy importantes en la nanomedicina. En este escenario nanomáquinas con fines medicinales comparten información para llevar a cabo tareas de cómputo, operaciones lógicas, y toma de decisiones en el tratamiento de enfermedades complejas.

Cuando una combinación de dispositivos electrónicos y de origen nanométrico se usan en el interior, sobre el cuerpo humano, o en sus proximidades a través de prendas o accesorios se forma una red comunicacional cuyos elementos operan como sensores y/o actuadores, entonces, esta infraestructura da origen a una red BAN. Comúnmente,

ⁱⁱ Project description: [Robots to fix underground pipes and help cut road-works - GOV.UK \(www.gov.uk\)](http://www.gov.uk).

conmutadores a macroescala como gateways, se emplean para conectar los dispositivos a Internet en este tipo de prendas. Por tanto, se pueden obtener aplicaciones altamente sofisticadas en varios campos de la salud entre las cuales las más importantes se determinan a partir de parámetros fisiológicos (como son temperatura corporal, glucosa, latidos cardíacos, movimientos corporales, electrocardiogramas, electroencefalogramas, electromiografías, fotopletismografías, respuesta galvánica de la piel y presión arterial) de pacientes, por lo que, el análisis de este tipo de información realizado en tiempo real, permite mejorar la calidad de vida de personas con enfermedades complicadas.

Para lograr potenciar estas innovadoras aplicaciones entre otras, en el campo médico para beneficio de la humanidad, es imprescindible realizar procesos vinculados a la experimentación, que permiten conseguir la práctica necesaria para evitar errores y consecuencias indeseadas al intentar emplear los resultados adecuados de la experimentación sobre todo cuando se trata de la salud de las personas. En la actualidad, no obstante, la experimentación conlleva un gran esfuerzo en cuanto a presupuesto se refiere (en especial porque la tecnología asociada a las nanocomunicaciones es costosa), en ese caso, es muy común la utilización de simuladores para no incurrir en altos gastos en la investigación de las CM y sus eventuales aplicaciones.

Precisamente, en función de la trascendencia de las CM y su enorme impacto en la ciencia en razón de sus probables aplicaciones, a nivel mundial se han creado una gran cantidad de herramientas informáticas para análisis, simulación y eventual experimentación en sistemas moleculares de comunicación, la mayoría incompatibles entre sí, así es que IEEE (Institute of Electrical and Electronics Engineers) ha generado los estándares de nanocomunicaciones 1906.1 y 1906.1.1 con el propósito de posibilitar la compatibilidad y rehusar la información de una investigación en otraⁱⁱⁱ. Igualmente, el proyecto CIRCLE^{iv} (Coordinating European Research on Molecular Communications), ha intentado establecer un lenguaje estandarizado (que se puede interpretar como la contraparte europea de IEEE para lograr la estandarización de herramientas digitales en CM) en cuanto a la programación, simulación y experimentación en CM a través del lenguaje de programación MolComML (Molecular Communication Markup Language). En razón de la trascendencia que MolComML

ⁱⁱⁱ Estos estándares han sido estudiados a profundidad en nuestros libros titulados: Comunicaciones Moleculares: Un análisis desde el paradigma protocolario de IEEE (<https://repositorio.cedia.edu.ec/handle/123456789/5>) y Estándares de Nanocomunicaciones Biológicas: Fundamentos y Simulación (<https://repositorio.cedia.edu.ec/items/2d965364-d67c-40a1-b5ff-3dc03fbc6eba>).

^{iv} [CIRCLE: Coordinating European Research on Molecular Communications \(ku.edu.tr\)](http://ku.edu.tr).

representa para el establecimiento de un “formato común” de programación para elementos comunicacionales en sistemas de transmisión molecular, en este libro, se analizarán de forma amplia este y otros lenguajes de marcado, además de los simuladores de mayor difusión y utilización frecuente en CM.

Esta obra académica contribuirá a la difusión de las CM en idioma español, cuya literatura en este idioma es básicamente inexistente.

En consistencia con el contenido del libro, el mismo requiere el conocimiento ingenieril pertinente a programación informática, sistemas de comunicaciones, redes de computadoras, sistemas de comunicación molecular biológicos y el manejo de las teorías de la información, matemáticas y técnicas probabilísticas subyacentes. De este modo, el libro se enfoca en estudiantes de ingeniería, maestría, doctorado o posdoctorado en las áreas científicas previamente mencionadas.

La organización del libro se establece de la siguiente manera:

En el capítulo 1 se examinará minuciosamente las redes BAN y sus aplicaciones en el campo médico. Este análisis le permitirá al lector tener en cuenta la importancia de las CM en aquellos casos en los que se puede mejorar la calidad de vida de las personas.

En el Capítulo 2 se investigan a detalle los principales lenguajes de marcado y simuladores de uso común en CM.

En el Capítulo 3 se discutirá pormenorizadamente MolComML y se presentarán varias simulaciones de CM en el contexto de aplicaciones médicas.

**Redes de
Área Corporal BAN
(Body Area Network)**

01

Las redes BAN se componen de una infraestructura con dispositivos localizados en el interior, sobre, o en las proximidades del cuerpo humano (a través de prendas o accesorios de tipo wearable), y comúnmente conmutadores a macroescala (como gateways) se emplean para conectar tales dispositivos con Internet, y de este modo establecer aplicaciones altamente especializadas en varias aristas científicas, entre las más significativas se tienen a aquellas en el campo de la medicina. De esta manera, se pueden monitorear parámetros fisiológicos del cuerpo y signos vitales como temperatura corporal, glucosa, latidos cardíacos, movimientos del cuerpo, electroencefalogramas, electrocardiogramas, electromiografías, fotopleetismografía, respuesta galvánica de la piel y presión arterial. Esta variedad de elementos decidores del estado de salud de un paciente y su análisis pertinente en tiempo real puede mejorar drásticamente la calidad de vida de una persona. Tomando en cuenta el potencial de las redes BAN y su inherente relación con las comunicaciones biológicas o Comunicaciones Moleculares (CM^v), en este capítulo se estudia a detalle esta temática.

^v Las nanocomunicaciones que tiene lugar en el interior del cuerpo humano, se refieren a las CM.

1.1

ANÁLISIS GENERAL DE LAS NANOCOMUNICACIONES EN REDES BAN

Las nanocomunicaciones han coadyuvado la aparición de aplicaciones innovadoras en algunos campos de la tecnología, principalmente en la medicina, además los adelantos en la biotecnología y la ciencia de los nanomateriales han permitido que tales propuestas se realicen mediante el uso de simuladores o incluso de forma experimental (generalmente en condiciones controladas). Por lo que resulta imprescindible la utilización de herramientas de análisis provenientes de las teorías comunicacionales en ingeniería, la matemática, la estadística, y la biología para efectivizar los programas médicos en referencia [1-17].

El empleo de diferentes tipos de nanomáquinas o nanobots con fines terapéuticos ha viabilizado la detección y eliminación de tumores en el cuerpo humano. Sin embargo, es necesario admitir que, a pesar de las enormes ventajas establecidas mediante el uso de estos nanocomponentes, su propio tamaño y el entorno en el que se propagan las nanomáquinas (el interior de los seres vivos es fundamentalmente un canal de transmisión ruidoso) define un grado importante de no confiabilidad y no total control en dichas aplicaciones. Para gestionar estos dos últimos aspectos se recurre a las ventajas generadas en el empleo de redes de nanocomunicaciones, en donde operan conjuntos de nanomáquinas (incluso de clase heterogénea) en una estructura denominada redes BAN² (Body Area Nanonetwork), la cual determina [18]:

1. Coordinación cooperativa en las tareas de las nanomáquinas.
2. Eficiencia al reunir información de biosensores.
3. Correlaciones de información en su ingreso a biosensores y toma de decisiones.
4. Transmisión de información a entidades externas.

En general, las redes BAN se contemplan como el paradigma bajo el cual se potencia la conectividad permanente en el tiempo sin importar la localización física. Lo anterior es factible debido a la evolución de las comunicaciones móviles e inalámbricas y su orientación hacia una dimensión que conste en una comunicación absolutamente personalizada, es decir, se propende el intercambio de información en el interior, alrededor y fuera del cuerpo humano. No obstante, las verdaderas comunicaciones personales ocurren como se ha indicado en el párrafo anterior a través de nanoredes para producir

el intercambio de información entre elementos o dispositivos en el interior de seres humanos. Esto ocurre por medio de mecanismos a nivel celular y molecular. A nanoescala, la transferencia de información ocurre mediante moléculas o estructuras moleculares con propiedades específicas como fluoróforos fotoactivos o canales de rodopsinas, anticuerpos (proteínas), bacterias móviles u ondas de iones a diferentes escalas como se ilustra en la Figura 1.1 [19].

Como es natural, resulta desafiante el desarrollo de este nuevo tipo de comunicaciones en aspectos relacionados con fuentes de poder, retardo en la transmisión de la información, entre otros; los aspectos referidos requieren ser direccionados desde un enfoque técnico, por ejemplo, el diseño de antenas, la electrónica subyacente, los materiales a utilizar, interfaces, software, etcétera. Por otra parte, también se tendrá que tomar en cuenta factores sociales y psicológicos de las personas que empleen tales terapias para sus tratamientos [19].

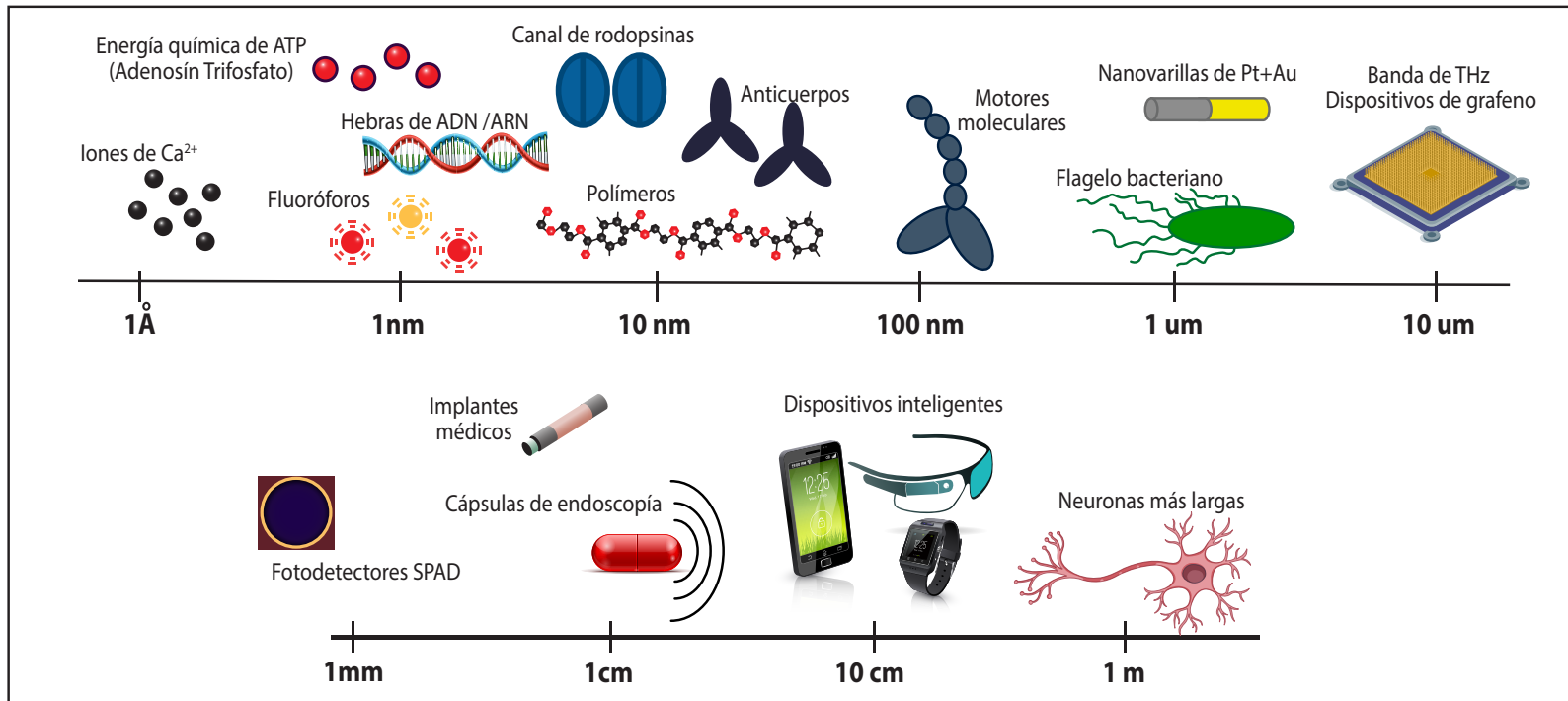


Figura 1.1. Escala de comunicación de mecanismos y dispositivos para aplicaciones médicas sofisticadas en CM.

Adicionalmente, ya que estos sistemas de transmisión/recepción, se convertirán en una parte integral del cuerpo, indudablemente modificará el estado de vida de las personas [19].

Existe en la literatura cuatro escenarios para nanocomunicaciones con fines terapéuticos para redes BAN² y se particularizan en la Tabla 1.1, y estos son [18, 19]:

1. Comunicaciones a distancias de nanómetros basadas en FRET (Forster Resonance Energy Transfer).
2. CM en un grupo de fenómenos a escala nano a micrométrica.
3. Comunicaciones EM (Electromagnéticas) operando a una escala milimétrica.
4. Comunicaciones fotoacústicas ultrasónicas en distancias de cientos de micrómetros.

Tabla 1.1. Estimación de parámetros para mecanismos de comunicaciones a nano/microescala [19].

Mecanismos	Rango	Throughput	Velocidad de la señal	Etapa Tecnológica
FRET	15 nm	25 Mbit/s	1 m/s	Experimentos de laboratorio
Propagación de ondas de calcio	300 μm ⁽¹⁾	1 bit/s	30 μm/s	Experimentos de laboratorio
Polímeros	Pocos μm	Pocos Kbit/s	Inferior a 1 mm/s	Análisis teórico
Motores moleculares	Hasta 50 μm	1 carga/motor ⁽²⁾	1 μm/s	Experimentos de laboratorio
Bacterias	Alrededor de 1 mm	Hasta 1 Mbit/bacteria	Hasta 20 μm/s	Experimentos de laboratorio
Nanovarillas	10 mm	128 Kbit/nanovarilla	Hasta 20 μm/s	Experimentos de laboratorio
Neuronas	1 m	1 Kbit/s	Hasta 120 m/s	Experimentos de laboratorio
Micro dispositivos basados en ondas Electromagnéticas	5 mm	500 Kbit/s	1,5 x 10 ⁸ m/s ⁽³⁾	Análisis Teórico

(1) Se asume una alineación favorable de células en las que los iones de calcio se propagan, usualmente el rango es más pequeño.

(2) La carga puede consistir en liposomas o vesículas que contienen hebras de ADN (Ácido Desoxirribonucleico), puntos cuánticos o silicio microfabricado.

(3) En sangre humana; esto puede diferir en otros tejidos dependiendo del índice de refracción; en el grafeno es 100 veces más pequeña.

El primer enfoque de nanocomunicaciones mencionado, fundamenta los fenómenos FRET los cuales permiten que la energía se pueda transferir entre moléculas de una manera no-radioactiva. Por lo que, el nanotransmisor y el nanoreceptor son dos moléculas vecinas. La molécula emisora se estimula hasta un estado de energía alto por radiación externa o por una reacción química, y envía su energía al receptor en un tiempo corto (en ηs) en un rango de distancias que se encuentran en el orden de nanómetros (es decir entre moléculas a distancias máximas de $20\ \eta\text{m}$ de separación). En términos de una transmisión FRET, esta se puede usar para una modulación de modo “on-off”, donde en “1” se produce una transferencia energética, en tanto que en “0” no ocurre ninguna transferencia. Además de su limitado rango FRET no es particularmente confiable. En lugar de enviar una señal al receptor, la molécula transmisora puede perder su energía por medio de un fotón; esta situación se puede mejorar con el empleo de un sistema FRET de múltiples entradas y múltiples salidas, y también múltiples moléculas en ambos extremos de la transmisión [19].

La segunda aproximación de análisis en el paradigma de las nanocomunicaciones en el rango de nano a micrómetros se relaciona con las comunicaciones biológicas en células de organismos vivientes a través de mensajeros moleculares que son moléculas de información, y que se denominan además portadoras moleculares. Esta clase de comunicación es bio-compatible pues existe en la naturaleza misma. Uno de los mecanismos más ampliamente considerados en esta forma de comunicación es la difusión molecular, y comúnmente se investigan las portadoras de información en ondas de iones de calcio emitidos por una célula. Los iones en sí son pequeñas partículas (de tamaño entre 100 a $200\ \mu\text{m}$), que se desplazan a distancias de hasta $300\ \mu\text{m}$, asumiendo una alineación favorable de células vivas en la difusión. Algunas partículas más grandes como los polímeros dan lugar a procesos de broadcasting, es decir, por difusión física, llevando información codificada en sus estructuras. Consecuentemente, los bits “0” y “1” se pueden codificar por la alteración de los polímeros mediante átomos de hidrógeno o de flúor. Debido a, que el fenómeno de difusión es lento, la velocidad promedio de los iones de calcio no excede los $30\ \mu\text{m/s}$ [19].

Adicionalmente, son frecuentes los estudios moleculares de fluidos en presencia de arrastre y de motores moleculares los cuales son estructuras de $100\ \eta\text{m}$ aproximadamente, como es el caso de la dineína y la kinesina, que transportan información en hebras de ARN (Ácido Ribonucleico) o como una secuencia de péptidos en una vesícula. Teniendo en cuenta que los motores moleculares se pueden entender en términos de telecomunicaciones como un medio de transmisión guiado, ya que estos, se desplazan a lo largo de filamentos de proteína que se conocen como microtúbulos, son mucho más lentos (alrededor

de 1 $\mu\text{m/s}$) que sus contrapartes en procesos de difusión, pero de todas maneras los motores moleculares son formas de transporte más confiables [19].

El tamaño de las portadoras moleculares^{vi} más grandes se encuentra en el rango de 1 μm , entonces técnicas de transporte activo se usan en este caso. La codificación de bits en estas condiciones se puede efectuar mediante hebras de ADN de un plásmido localizado en el interior de una bacteria o asociado a un nanomotor catalítico, por mencionar algunos casos [19].

La tercera aproximación en el análisis de comunicaciones a nanoescala se establece en la miniaturización de técnicas de transmisión de ondas EM. La escala de interés para dispositivos en este tipo de comunicación es aproximadamente de 10 μm , siendo este tamaño superior al de las moléculas en sí, por lo que tamaños reducidos de dispositivos operando a microescala, restringen las frecuencias de comunicación a rangos de THz. La utilización de la frecuencia más baja y favorable, por consiguiente, se enmarca entre 0,1 y 10 THz (rango de frecuencias escogidas por ser no invasivas ni ionizantes), en antenas de grafeno capaces de radiar eficientemente ondas EM en esta banda, cuya frecuencia de resonancia es dos veces más baja que aquellas metálicas del mismo tamaño. En la superficie de tales nuevos materiales las ondas EM se propagan como un polaritón de plasmón superficial, y bajo ciertas condiciones su velocidad puede ser 100 veces más baja que en el vacío. En estas circunstancias, la longitud de onda está en el orden de 1 μm [19].

Por consiguiente, en la banda de THz existen muchos desafíos, porque a estas frecuencias la propagación se pierde en el interior de los tejidos corporales (conformados primariamente por agua), lo cual limita el rango de un enlace a solo pocos milímetros, también las pérdidas por absorción (en la sangre y tejidos del cuerpo) son considerables, produciendo pérdidas por trayectoria que exceden los 120 dB para distancias de unos pocos milímetros. La resonancia de moléculas estimuladas por ondas de THz genera la conversión de energía electromagnética en cinética, la que es absorbida por estas moléculas, y por tanto, limitan la transmisión. Además, otra condición que no se puede pasar por alto en esta clase de transmisiones, es la variación molecular impuesta en este entorno como resultado de la radiación electromagnética en la misma banda de frecuencia, manifestándose esta situación en forma

^{vi} En nanocomunicaciones moleculares las moléculas de información son consideradas como portadoras de señales moleculares. Esto implica, que a diferencia de sus pares de telecomunicaciones, la información que se propaga a través de moléculas se convierte en la propia portadora informacional, por lo que no requiere de una instancia adicional para el sistema de comunicación molecular en uso.

de ruido coloreado con una potencia de magnitud que depende de la distancia de comunicación y del número de moléculas entre las antenas en los extremos de transmisión y de recepción. Igualmente, dado que la absorción molecular es condicionada por la presencia de una señal en el medio de transmisión, el ruido correspondiente resulta en un pulso disperso, por tanto, se tendrá que asegurar la separación suficiente entre pulsos sucesivos para prevenir la interferencia entre símbolos ISI (Inter-Symbol Interference) [19].

Esta tercera perspectiva en las comunicaciones a nivel nano puede ser abordada además a través de antenas plasmónicas operando en los rango infrarrojo y visible de luz, donde estas antenas y detectores de fotones individuales funcionan como transmisores y receptores, respectivamente. Adicionalmente, las ondas ópticas enfrentan dificultades en comunicaciones al interior del cuerpo humano, pero cuentan con la ventaja de que las pérdidas por absorción molecular son mínimas en el rango óptico del espectro electromagnético. En este tipo de propagación, la refracción a través de las células es dominante y la reflexión por el entorno de comunicación afecta la transmisión provocando trayectorias múltiples de señales de información; sin embargo, debido a que los glóbulos rojos (compuestos básicamente por agua generan pérdidas por trayectoria en ondas EM), son elementos esenciales en la estructura corporal humana que enfocan la luz a ser propagada por medio de ellos, y reducen las pérdidas por trayectoria en esta clase de transmisión; tales características de los glóbulos rojos (de acuerdo a la forma y orientación de las células) posibilitan diagnosticar enfermedades y la presencia de patógenos mediante la identificación de la respuesta impulsiva del canal de comunicación [19].

El cuarto enfoque de las transmisiones a escala nanométrica lo constituyen las ondas de ultrasonido como portadoras de información, las cuales presentan una propagación más favorable en tejidos corporales y cubren distancias desde varios μm a pocos cm. Considerando el tamaño reducido a microescala de nodos (que procesan la información), se imponen limitaciones en el empleo de transductores estándares de ultrasonido basados en el efecto piezoeléctrico y vibraciones mecánicas. Así, el efecto fotoacústico se define como una opción viable para la generación de ultrasonido desde dispositivos a nanoescala. Las ondas acústicas se producen por mecanismos termoelásticos, donde la luz que incide en la superficie de un material es absorbida, la rápida presencia de calor causa además su breve expansión lo cual permite la presencia de ondas de ultrasonido en el material. Por lo que, los nano-láseres disponibles se pueden usar para transmisiones optoacústicas, en tanto, que resonadores ópticos se pueden utilizar como detectores en el extremo de recepción [19].

Para asegurar apropiadamente una fuente de poder en el caso de dispositivos electromagnéticos y de ultrasonido, nanocables de óxido de zinc se proponen para recolectar la energía, por su facultad para producir voltaje. Algunos cientos de estos nanocables se requerirían para suplir una sola nanomáquina, cada cable posee una longitud de $2\ \mu\text{m}$, y de $100\ \text{nm}$ de diámetro. Existen otras propuestas en las que la alimentación para una nanomáquina se realiza por generación remota de ultrasonido, el mismo que al ser utilizado por estas nanomáquinas se convierte en energía eléctrica a través de nanoelementos piezoeléctricos [19].

De todos los enfoques de nanocomunicaciones revisados, uno de los más prometedores lo constituyen las CM, esto se debe a su accionar en procesos naturales del cuerpo, por ejemplo, en el sistema inmunológico en el cual, los leucocitos se comunican con otros para conjuntamente proceder como sensores en la detección y eliminación de patógenos peligrosos, proveyendo un mecanismo prodigioso de defensa. Por otra parte, los sistemas de CM ya han sido usados. Así, se han desarrollado sistemas de comunicación artificial célula-a-célula en mamíferos para gobernar la población celular utilizando elementos de señalización de óxido nítrico. Estos sistemas además, pueden actuar como una herramienta funcional para terapias genéticas en redes artificiales de regulación genética. Como es notable, las redes BAN² sustentadas en CM implican importantes aplicaciones médicas [18].

Los cuatro paradigmas de nanocomunicaciones referidos se pueden emplear con señales EM en gateways para la comunicación entre elementos de interfaces a niveles macro-nano. No obstante, de manera primordial en la actualidad, el esquema preponderante de funcionalidad real para aplicaciones tecnológicas y médicas se centra en las nanocomunicaciones mediante ondas EM y de tipo molecular. Consecuentemente, estos dos últimos formatos de comunicación constituyen las directrices esenciales para el desarrollo de los estándares de comunicaciones a nanoescala de IEEE (Institute of Electrical and Electronics Engineering) 1906.1 y 1906.1.1 [1-9].

1.2

REDES DE ÁREA CORPORAL BAN

Con la factibilidad de monitorear el entorno que les rodea, las redes BAN pueden servir en aplicaciones de cuidado de la salud, militares, deportes, entretenimiento, entre otras [19-24].

Estas redes conjugan diversos tipos de comunicaciones, como se ilustra en la Figura 1.2, a continuación se mencionan algunas de estas formas de transmisión [19]:

- Entre dispositivos en el interior del cuerpo humano (como sensores implantables, cuya denominación en inglés es in-body).
- Entre dispositivos dentro de la piel (cuya denominación en inglés es into-body) y dispositivos de tipo wearable.
- Entre dispositivos de tipo wearable (cuya denominación en inglés es on-body).
- Entre dispositivos de tipo wearable y puntos de acceso (Access Point) externos (cuya denominación en inglés es off-body).
- Entre dispositivos en diferentes redes BAN (cuya denominación en inglés es body-to-body).

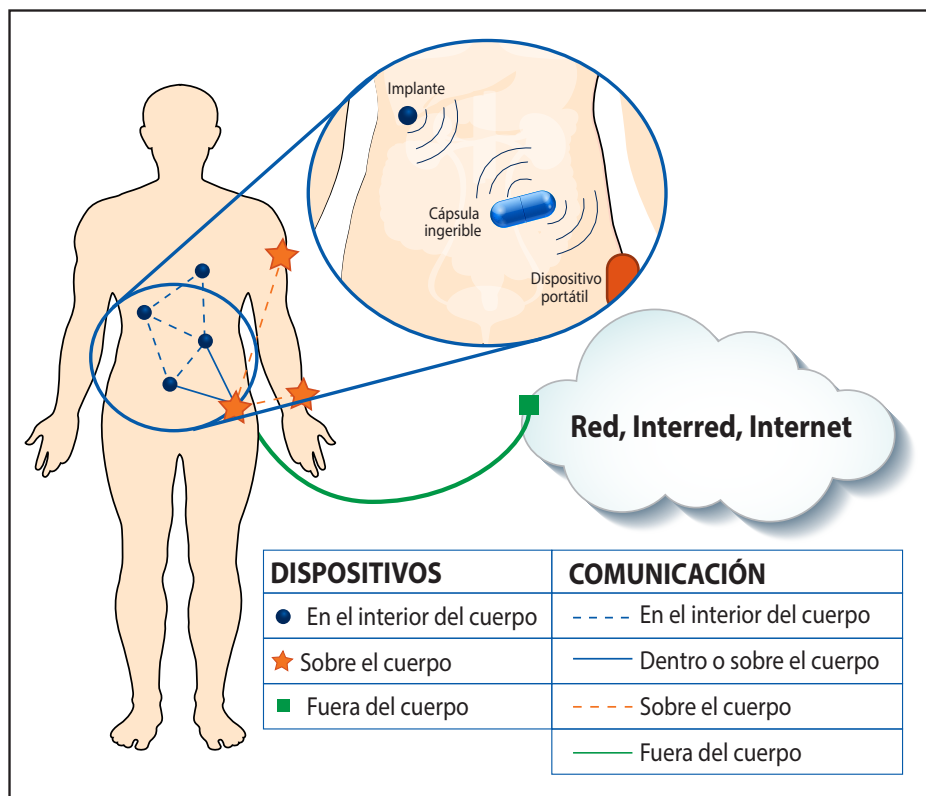


Figura 1.2. Escenarios de Comunicación en redes BAN.

Las redes BAN pueden utilizar diferentes tecnologías de comunicación, las que incluyen transmisiones guiadas, textiles inteligentes, acoplamiento inductivo ICC (Inductive Coupling Communication), acoplamiento del cuerpo humano BCC (Body Channel Communication), y transmisiones de radio. Mientras que algunas de estas tecnologías son universales y empleadas para todos los tipos de redes BAN, otras se restringen a ciertos escenarios, o dependen de la aplicación [19, 21].

Cuando se utilizan comunicaciones guiadas existen ciertas ventajas entre las que se encuentran, la seguridad y la confiabilidad. Algunas de las desventajas se refieren a que la estructura alamburada ocasiona limitaciones en el movimiento de sus usuarios y existe propensión a fallas en el medio guiado, por el trenzado que se puede ocasionar en el cable producido por el movimiento constante si el usuario es muy dinámico; por consiguiente, tales sistemas son apropiados solo en aplicaciones en las que los usuarios vistan un tipo de traje específico y su actividad de movimiento sea reducida, este es el caso de pilotos y conductores de vehículos. Los inconvenientes principales con sistemas cableados se pueden compensar a través del uso de textiles inteligentes, los que permiten la integración de fuentes de energía, la comunicación en sí y la circuitería de sensores en prendas lavables [19, 20, 21].

El canal de transmisión en ICC se establece entre dos bobinas acopladas magnéticamente, ya que el voltaje inducido en la bobina receptora es inversamente proporcional al cubo de la distancia, el acoplamiento solo existe en distancias muy cortas, en el orden de cm. La calidad del canal utilizado depende altamente de la alineación entre las bobinas y es independiente del tejido de alrededor. Entonces, las transmisiones de clase ICC son adecuadas para comunicaciones de tipo into-body y entre dispositivos externos e implantes (que son dispositivos médicos que se colocan en el interior del cuerpo) cercanos a la superficie del cuerpo humano, o entre implantes cercanos. Una ventaja adicional de las transmisiones ICC radica en la posibilidad de usar transferencia de potencia inducida, donde la misma bobina puede proveer tanto el enlace de comunicación como el suministro de potencia [19-21, 23].

En las transmisiones BCC el emisor y el receptor se acoplan con el cuerpo del usuario y lo emplean como medio de comunicación; en este tipo de comunicaciones se distinguen el acople capacitivo y el galvánico. En el primer caso, los extremos de transmisión se acoplan al cuerpo humano a través de enlaces capacitivos, creados por electrodos que están en contacto con la piel, mientras otros electrodos que son flotantes se acoplan con el entorno para proveer un camino de regreso y cerrar el circuito. En el segundo caso, las comunicaciones galvánicas BCC emplean las propiedades iónicas de los fluidos corporales para la propagación de señales en las que los electrodos de transmisión y recepción están en contacto con el cuerpo propiamente. A diferencia

de los sistemas BCC capacitivos se limitan a comunicaciones on-body, las transmisiones galvánicas se pueden utilizar para comunicaciones in/into-body. Los transmisores acoplados al cuerpo en frecuencias superiores a 50 MHz no son cuasi estáticos (en los dos tipos de sistemas) y se produce la propagación de ondas guiadas. Los dispositivos en este caso se diseñan para estimular ondas superficiales fuertes que pueden viajar a lo largo del cuerpo humano [19-21].

Adicionalmente, en las comunicaciones ultrasónicas se usan ondas de sonido a frecuencias superiores a 20 KHz para propagar información. Las ondas de ultrasonido ocasionan una atenuación mucho menor en el cuerpo que sus contrapartes EM, convirtiéndolas en mejores portadoras de información para comunicaciones in/into-body. Otra ventaja de este tipo de ondas radica en la seguridad, pues incluso se usan en imágenes médicas sin que se reporten efectos secundarios en el cuerpo humano. Sin embargo, se debe notar que los dispositivos para transmisiones de ultrasonido tienen que estar en contacto con el cuerpo en razón de que estas ondas no se pueden propagar por gases. Así, esta forma de transmisión es pertinente para comunicaciones in/into-body [19-21].

Las comunicaciones de ultrasonido proveen velocidades de transmisión hasta de 100 Kbit/s, pero el uso de esquemas de modulación como 64 QAM (Quadrature Amplitude Modulation), o multiplexación espacial MIMO (Multiple Input Multiple Output) se han propuesto para alcanzar velocidades superiores a 10 Mbit/s. No obstante, la compleja estructura de los transceptores y su consumo energético asociado a tales esquemas podrían estar fuera del alcance de los dispositivos implantables por su tamaño reducido y restricciones en fuentes de energía [19-21].

Además del sonido, la luz se puede utilizar como portadoras de información en redes BAN. Transmisiones ópticas inalámbricas se pueden emplear para establecer enlaces entre implantes y nodos de tipo on-body, las cuales alcanzan hasta 50 Mbit/s a través de 4 mm en tejidos gruesos. Las comunicaciones ópticas transdérmicas son realizables con dispositivos implantables (varios centímetros en el interior de tejidos corporales) en ventanas del espectro electromagnético con una longitud de onda entre 700 y 950 nm, en las que se tiene baja absorción [19, 20].

Un aspecto crítico para considerar en las transmisiones ópticas, es la restricción de potencia de los dispositivos implantables, y la duración de sus fuentes de energía mejoraría mediante comunicaciones con retro reflexión transdermal en la transmisión de la información desde el implante o un dispositivo externo. En este caso la fuente lumínica se produce mediante un dispositivo de tipo on-body con menores restricciones, y el implante modula la luz reflejada por un dispositivo

con un sistema mecánico microelectrónico MEMS (Micro-Electrical Mechanical System) es decir, con un retro reflector modulador miniatura MMR (Miniature Modulating Retroreflector). A más de sus únicas características para transmisiones ópticas transdermales, la inmunidad a la interferencia EM y la inherente seguridad debido al limitado rango de propagación, son algunas de las ventajas de esta tecnología [19, 20].

Una de las mejores opciones para redes BAN, las constituyen las ondas de radio debido al gran desarrollo que han tenido estas tecnologías, entre las que se encuentran: Bluetooth, Bluetooth Low Energy (BLE), Zigbee y Ultra-Wide band (UWB). Estas formas de transmisión inalámbrica se han adoptado para la capa física en el estándar existente para redes BAN, es decir, el estándar IEEE 802.15.6. La Tabla 1.2 denota los diferentes medios de comunicación, frecuencias de uso, alcance del enlace, así como la velocidad de transmisión de tecnologías para redes BAN [19-21].

Tabla 1.2. Principales formatos de comunicación para redes BAN [19-21].

Tipo de enlace físico	Sistema/mecanismo	Frecuencia [GHz]	Rango [m]	Velocidad de Transmisión [Mbps]
Medio guiado	Textiles inteligentes	-	-	0,25
Medio inalámbrico	Comunicaciones ópticas	10 ⁵	0,05	50
	Acoplamiento corporal inductivo	< 0,02	0,2	2,5
	Ultrasonido	< 0,3	0,5	0,1
	Acoplamiento corporal capacitivo	< 0,2	1	10
	Acoplamiento corporal galvánico	< 0,05	1	0,064
	Bluetooth LE	2,5	10	1
	WiGig (Wireless Gigabit)	60	10	7000
	Banda ultra ancha UWB	3-10	50	480
	Bluetooth	2,5	100	3
	Wifi	2,5 ; 5,8	100	100
Zigbee	2,5	200	0,25	

Las redes Bluetooth se diseñaron para intercambio de información de audio y datos entre dispositivos personales, pero su rango de cobertura y velocidad de transmisión las convierten en adecuadas para comunicaciones de tipo off-body. Bluetooth LE apareció posteriormente con características de eficiencia energética, baja latencia y como una alternativa a bajo costo, siendo compatible con los dispositivos Bluetooth y además funcional para redes de sensores. Zigbee ofrece otra variedad de tecnología de baja energía y costo reducido para implementaciones en redes BAN a expensas de su baja velocidad de transmisión [19-21, 24]. Permitiendo comunicaciones de salto múltiple para evitar obstrucciones en la propagación de la señal, Zigbee puede extender la cobertura y mejorar la confiabilidad en la transmisión [19-21].

Las redes UWB son probablemente una de las óptimas tecnologías de radio para redes BAN, debido a las ventajas asociadas al gran ancho de banda y a su baja densidad de potencia (debajo de 41,3 dBm/MHz) lo cual las define como apropiadas para operar en entornos sensibles a la radiación electromagnética y en las proximidades al cuerpo humano por los bajos niveles de exposición. Adicionalmente, la gran profundidad de adentramiento convierte la propagación UWB como idóneas para comunicaciones in/into-body [19-21].

Las ondas milimétricas también forman parte de las diversas opciones para redes de área corporal por su gran capacidad y baja interferencia con otras redes BAN. Por otra parte, las pérdidas de propagación favorecen el equilibrio en el funcionamiento de una red BAN y aspectos relacionados con su seguridad, porque las señales que se emiten a través de ondas milimétricas se confinan al interior del área próxima al usuario. Concomitantemente, tales pérdidas desafían el mantener señales portadoras de información a niveles requeridos para cumplir la calidad de la transmisión pertinente [19-21, 24].

En tanto, que las redes BAN representan realmente el futuro de las comunicaciones en seres humanos, existen grandes retos relacionados a la seguridad del usuario en sí, la seguridad de la información misma, su utilización simple y el confort del usuario propiamente. La movilidad de dicho usuario constituye además un aspecto que influye en la calidad del canal de transmisión; el cuerpo humano también ejerce afectación en las características de propagación de las antenas trabajando en sus proximidades, en especial en el caso de antenas inmersas en tejidos humanos [19]. Estos aspectos se han de orientar en todos los niveles de diseño: transmisión física, acceso al medio, enrutamiento, y capas superiores [19-21, 24].

A pesar de los grandes retos que involucra el desarrollo de redes BAN, el monitoreo y comunicación de sus usuarios es una práctica que se realiza comúnmente en hospitales modernos y en atletas. Su potencial

verdadero se extiende a la cooperación que se puede originar entre nanoredes y redes BAN, efectivizando el transporte de información desde un tejido en el interior del cuerpo humano hacia el mundo exterior; en este escenario la nanored transferirá información a un implante, donde una BAN transportaría la información a un punto de acceso tipo off-body desde donde empezaría el proceso con el resto de los dispositivos de comunicación de internetworking [19-21, 24].

Las interfaces para redes BAN se puede determinar por la investigación de propiedades físicas de moléculas o materiales entre los cuales se encuentran [19]:

- a) Luz de estimulación en canales de rodopsinas.
- b) Fluminiscencia por BRET (Bioluminescence Resonance Energy Transfer).
- c) El rol de ATP (Adenosine Triphosphate).
- d) Fotodetectores y SPR (Surface Plasmon Resonance).
- e) Comunicaciones micro/macro de ondas EM.

De las diversas formas factibles en las que las nanocomunicaciones interactúan con redes BAN, las inercialmente óptimas serían las CM porque esa es la manera natural por la que el interior del cuerpo humano se comunica, además, sistemas de CM se han implementado exitosamente [18].

1.3

NANOMEDICINA Y REDES BAN²

La nanomedicina utiliza el comportamiento molecular como herramienta para diagnosticar, establecer terapias y prevenir enfermedades. La nanomedicina es un campo cada vez más creciente en la conjugación de los avances presentes en la nano y biotecnología para viabilizar aplicaciones prometedoras. Las aplicaciones actuales y futuras de la nanomedicina se pueden caracterizar en tres grupos principales [18-20].

En el primer grupo se incluyen a elementos como nanoporos y puntos cuánticos como se detalla en la Figura 1.3. Estos elementos pueden interactuar directamente en el fenómeno biológico (por ejemplo, una

sola célula o células de orgánulos), como es el caso del establecimiento de un bucle de retroalimentación de glucosa fundamentado en nanoporos para el encapsulamiento de células de insulina [18, 20].

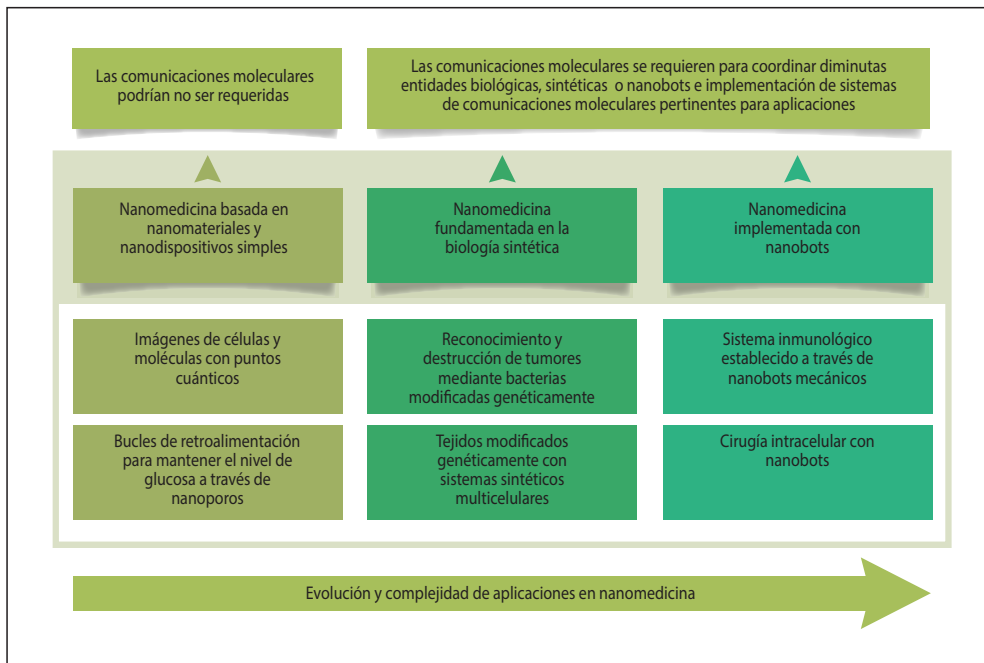


Figura 1.3. Clasificación de aplicaciones de la nanomedicina en tres grupos y sus correspondientes herramientas a nanoescala categorizadas por el tiempo de evolución y complejidad.

El segundo grupo se trata de herramientas sobresalientes de la biología sintética que se adaptan o sintetizan desde la naturaleza misma, esto ocurre en bacterias genéticamente alteradas y en células sintéticas. Entre los ejemplos de esta clasificación está un conjunto de bacterias modificadas para el reconocimiento y eliminación de tumores en el cuerpo humano [18].

El tercer grupo lo integran aplicaciones de nanomedicina futuristas (como son los nanobots mecánicos o bio-híbridos), que podrían cooperar con equipamiento de nanomedicina. Aquí se debe mencionar aplicaciones de cirugías intracelulares mediante tecnologías inalámbricas efectuadas por nanobots mecánicos [18-20].

En las aplicaciones que se observan en la Figura 1.3, uno de los aspectos más críticos radica en el control y la confiabilidad al desarrollar la aplicación, estos aspectos se gestionan de mejor manera por

medio de las redes BAN² utilizando CM con nanomáquinas terapéuticas. Una BAN² puede coordinar tareas que se han de realizar por parte de nanomáquinas terapéuticas heterogéneas para alcanzar comportamientos altamente sofisticados e incrementar el número de posibilidades de suceso. De este modo, en la Figura 1.4 células artificiales y nanobots comparten información molecular para coordinar y ejecutar procesos de alto nivel en la administración de medicamentos (drug delivery) y del sistema inmunológico basado en tareas [18].

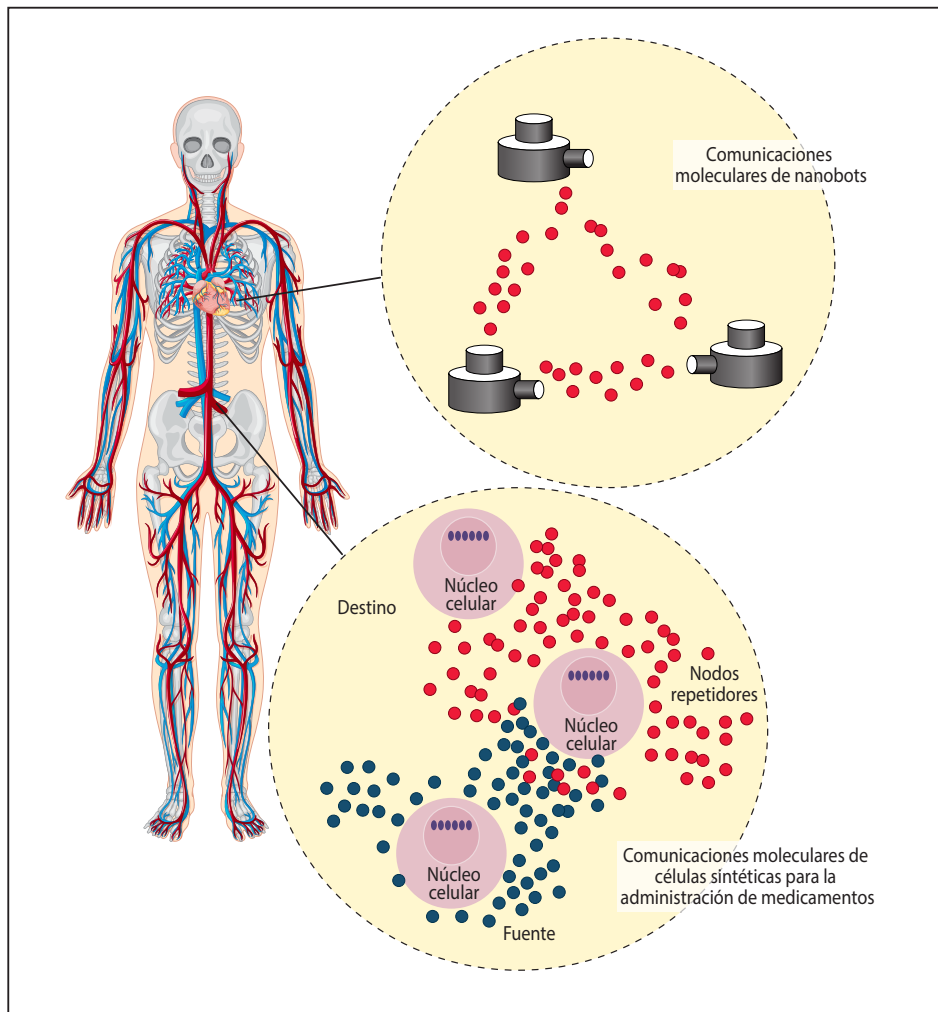


Figura 1.4. Dos diferentes redes BAN² de células artificiales y nanobots a través de CM para efectuar aplicaciones de administración de medicamentos y de sistema inmunológico basado en tareas.

Las nanoredes son indispensables para funciones vitales que se fundamentan en CM, como la percepción del quórum bacteriano en el cual las bacterias integran una nanored y cada una de ellas produce, emite, y recibe moléculas de hormonas mediante mensajeros moleculares. Este proceso biológico determina que cada bacteria se sincronice con todas las actividades de su colonia, y modifique por tanto, el estudio de la colonia en respuesta a un estímulo externo [18, 20].

1.3.1 Aplicaciones de nanomedicina mediante Comunicaciones Moleculares en redes BAN

Como se indica en la Figura 1.3, la biología sintética sustentada por nanobots realiza aplicaciones complejas que han transformado el tratamiento de enfermedades complicadas. En estas terapias las CM en las BAN² desarrollan operaciones lógicas y computacionales y toman decisiones en dichos tratamientos. Algunas propuestas de estos tratamientos se describen rápidamente en los siguientes párrafos [18].

En la biología sintética sistemas de CM se designan para controlar el comportamiento de la población de células sintéticas en el tiempo y el espacio. Gracias a la difusión y recepción de moléculas mensajeras las CM pueden conseguir una predicción dinámica y robusta de la población. En este aspecto, la literatura propone un sistema de CM para realizar predicciones de patrones espaciales de células modificadas genéticamente. La señalización de la acil-Homoserina lactona AHL (Acyl-Homoserine Lactone) se sintetiza en células emisoras y se propaga a su entorno, provocando que células receptoras con redes de genes de detección de bandas respondan a un umbral de detección de concentración de AHL. La red en referencia se comporta como si tuviera conectividad en una topología de anillo. La aplicación más inmediata de este sistema de CM radica en la ingeniería de tejidos y en la fabricación de biomateriales y biosensores [18].

Otro diseño importante de sistemas de CM de tipo célula-a-célula es aquel en el que se han empleado elementos de señalización de óxido de nitrógeno para incrementar la funcionalidad de circuitos genéticos sintéticos usados en la coordinación y programación de poblaciones celulares modificadas genéticamente; esta aplicación de nanomedicina se convierte en un instrumento muy versátil para terapias genéticas [18].

Similarmente, nanobots que fundamentan aplicaciones de nanomedicina permiten funciones de control y confiabilidad en redes BAN². La acción conjunta de los nanobots puede mejorar la probabilidad de operaciones pertinentes con éxito. Las CM claramente logran que los nanobots puedan correlacionar fácilmente los resultados de la obser-

vacación de parámetros al actuar como sensores y la correspondiente transmisión de información a microsistemas externos diseñados con fines de monitoreo del estado de salud [18, 20].

De manera general, el eje central de las aplicaciones de las nanoredes se establece en el contacto y control desde el mundo a macroescala durante todo el tiempo de operación de las mencionadas nanoredes. Entonces, los probables desarrollos de las comunicaciones al nivel nano radican en el diagnóstico médico y procesos de cirugía. Un sistema para diagnosticar se puede constituir por ejemplo, mediante nanopartículas desplegadas en el interior de tejidos. La información recolectada por el sistema se puede transmitir fuera del cuerpo al personal médico a través de las conexiones de las redes BAN en tiempo real, y por tanto, para el análisis correspondiente también en tiempo real. Similarmente, se pueden usar nanomáquinas para cirugías remotas en las que los cirujanos acceden a la información recolectada para tomar decisiones [19-21].

El almacenamiento de información es otra área que se beneficia grandemente de las nanocomunicaciones, particularmente en el formato de las CM. Actualmente debido a la gran demanda mundial, los medios típicos de almacenamiento de información están saturados, de allí que, la ciencia ha considerado nuevas alternativas para almacenar datos en elementos biológicos. Por lo que, las hebras de ADN y ARN que mantienen información del genoma humano en cadenas de nucleótidos son altamente efectivas como medios para conservar la información. Teóricamente, en cada nucleótido se puede almacenar 2 bits de información porque existen cuatro tipos de nucleótidos en hebras de ADN y ARN. Ya se han efectuado procedimientos biológicos de almacenamiento y recuperación de información exitosamente en moléculas de ADN, consiguiendo en promedio 1,57 bits/nucleótido lo cual corresponde a una impresionante densidad de 215 Peta bytes/g de ADN [2, 4, 19, 20].

Estos sistemas se podrían emplear en uniformes inteligentes del personal militar o policial en incidentes bélicos articulados con sistemas de sensores embebidos, capaces de monitorear signos vitales del cuerpo o detectar heridas de bala para protección y seguridad. Los nanosensores que funcionan al interior del cuerpo humano propiciarían diagnósticos rápidos, localización y clasificación precisa de daños en tejidos en caso de heridas, con lo que se conseguiría una rápida acción por parte del personal médico pertinente [19, 20].

Otras aplicaciones consistirían en el monitoreo del estado de salud de atletas de alto rendimiento, esto incluiría la medición de parámetros fisiológicos como latidos cardíacos, consumo de energía, porcentaje

de grasa, contenido de agua en el cuerpo, etcétera. El monitoreo y control en tiempo real representarían un alto nivel de seguridad para el profesional en estas circunstancias [19, 20].

Las redes BAN se pueden usar además para una serie de aplicaciones médicas relacionadas con la prevención, diagnóstico y tratamiento de una serie de enfermedades; este es el caso de desórdenes de tipo mental y físico, discapacidades, cuidado neonatal, salud infantil y embarazo, cuidado de personas ancianas, diabetes, enfermedades cardíacas, cáncer, epilepsia, paraplejia, problemas de visión, asma y alergia, entre otras [20, 21, 24].

1.3.2 Parámetros de comunicación multicapa en nanomedicina y en redes BAN²

1.3.2.1 Arquitectura para redes BAN²

Una red BAN inalámbrica, WBAN (Wireless Body Area Network) se compone de un conjunto de nodos establecidos por biosensores inalámbricos interconectados entre sí (formados por dispositivos de tipo wearable, implantes médicos y nanonodos propiamente) de forma común en una topología de estrella o multi-salto. Estos nodos típicamente se encuentran sobre el cuerpo humano, implantados en su interior, o establecidos de modo que monitoreen signos vitales y parámetros fisiológicos sin interrumpir la rutina cotidiana de un paciente. Entre estos signos vitales se contemplan electrocardiogramas ECG (Electrocardiography), electroencefalograma EEG (Electroencephalography), respuesta galvánica de la piel GSR (Galvanic Skin Responce), flujo sanguíneo, nivel de glucosa, presión arterial, nivel de oxígeno en la sangre, entre otros. Con los avances de la nanotecnología, el progreso en redes WBAN ha evolucionado para integrar sistemas de salud basados en nanosensores con funciones de detección de señales, actuadores, realizadores de operaciones de cómputo y almacenamiento. Por tanto, un ecosistema completo de una red o interred WBAN incluye nanonodos ejecutando tareas *in-vivo* remotamente como se presenta en la Figura 1.5 [22, 24].

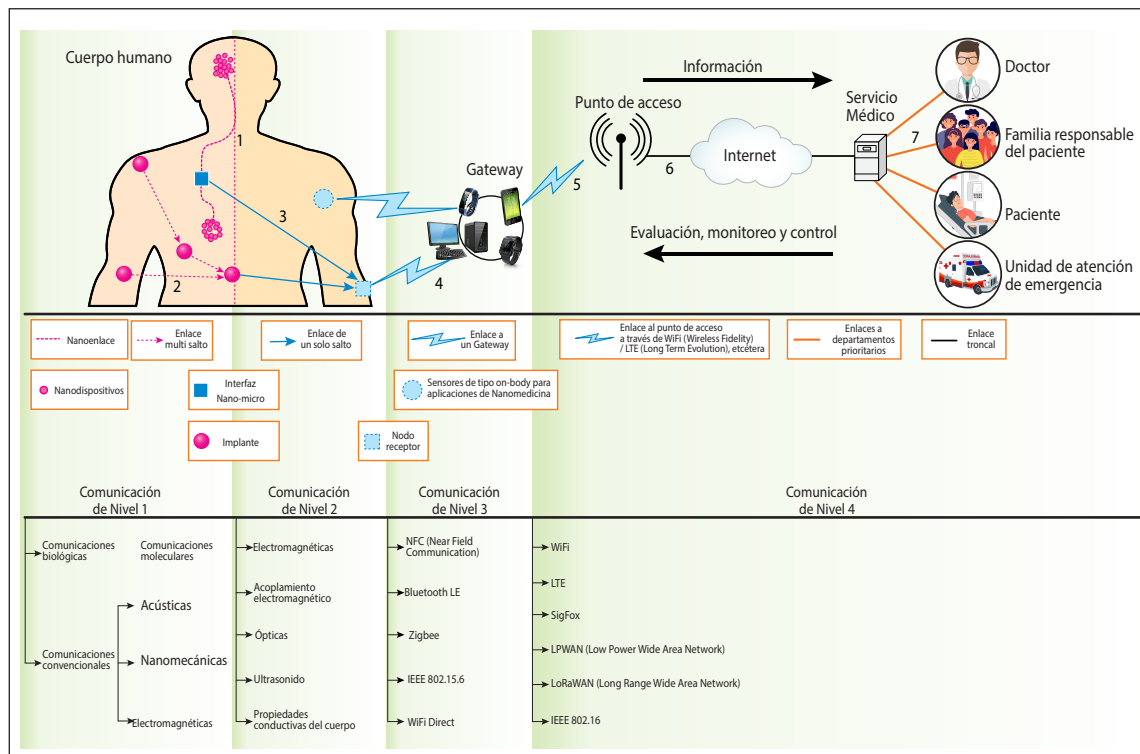


Figura 1.5. Perspectiva completa de una red o interred WBAN para sistemas de nanomedicina con acceso remoto establecida en 4 niveles de comunicación. En el Nivel 1 el transmisor y el receptor residen en el interior del cuerpo humano y se identifican con los enlaces numerados como 1 y 2. El Nivel 2 se establece cuando al menos uno de los dispositivos de transmisión reside en el interior del ser humano y se identifican con los enlaces numerados como 3. En el Nivel 3 al menos uno de los dispositivos de comunicación es de tipo off-body y son identificados con los enlaces de número 4. Finalmente, todos los enlaces de transmisión después del dispositivo de comunicación identificado como gateway se consideran de Nivel 4 y se identifican con el enlace número 5.

Una arquitectura completa de redes WBAN se identifica en la Figura 1.5 incluyendo transmisiones de clase in-body y off-body. Las transmisiones on-body involucran dispositivos de tipo wearable y otros sensores para mediciones de SG, EGG, glucosa en la sangre y presión arterial. Los elementos de comunicaciones en la Figura 1.5 se estructuran de la siguiente manera [22]:

- 1.** Nanodispositivos. Los nanodispositivos se constituyen como una de las entidades más pequeñas en el ecosistema de nanomedicina, y pueden ejecutar funciones básicas a nanoescala como aquellas de cómputo, almacenamiento, actuación y comunicación además de desempeñar el papel de sensor.
- 2.** Nanoenlaces. Estos se refieren a enlaces entre nanodispositivos y nano-micro interfaces.
- 3.** Nano-micro interfaces. Conectan los nanodispositivos en el interior del cuerpo humano con nodos receptores, y estos últimos cuentan con conexiones al dispositivo de tipo off-body.
- 4.** Implantes. Son dispositivos médicos que se colocan en el interior del cuerpo para el monitoreo de ciertas enfermedades, adquisición de signos vitales o para identificación biométrica.
- 5.** Nodos receptores. Actúan como concentradores de información (hubs) en redes WBAN y por ende, colectan la información desde dispositivos in-body para retransmitirla al servidor de aplicaciones médicas y viceversa.
- 6.** Sensores on-body. Son sensores de utilización sobre la piel o dentro de prendas de vestir interiores dispuestas para medir y monitorear signos vitales como son ECG, EEG, presión arterial, glucosa en la sangre y nivel de oxígeno.
- 7.** Gateway. Es el dispositivo que se emplea para la conexión entre la red WBAN y el servidor de aplicaciones médicas. Un teléfono inteligente o cualquier otro dispositivo como un computador o elemento de IoT (Internet of Things) que se enlaza a una estación base con tecnología 3G o 4G, por ejemplo, es un dispositivo gateway.
- 8.** Punto de acceso. Representa una base de comunicaciones móviles o un punto de acceso en la red LAN (Local Area

Network) o Wifi (Wireless Fidelity), que permite dirigir el tráfico generado por los sensores al servidor de aplicaciones médicas.

9. Servidor de aplicaciones médicas. Está constituido como una base de datos, que almacena información de los sensores en la infraestructura de conectividad WBAN, con el objetivo de análisis posterior y las acciones consiguientes. Aquí, se puede incluir el monitoreo de signos vitales y de información clínica en tiempo real, para que pacientes y médicos se puedan conectar y determinar alertas en caso de ocurrir notificaciones de criticidad dirigidos a doctores o miembros de familia que son responsables de los pacientes.

Los nodos (o sensores) WBAN localizados al interior del cuerpo, en la piel, sobre y alrededor del cuerpo humano, por tanto, monitorean información vital de un paciente y comúnmente emplean comunicaciones inalámbricas para alcanzar un nodo maestro (master node). El primer nodo maestro reúne la información de otros nodos en la red y la direcciona a los dispositivos de servicios médicos. Por lo regular, estos sensores miden parámetros precisos del paciente y del sensor-actuador para establecer decisiones adecuadas [21].

Es frecuente en las arquitecturas WBAN clasificar a los sensores en dos categorías de acuerdo con la clase de señal a medir. El primer grupo cuantifica señales continuas cuyo procesamiento se realiza en tiempo real, como son, las gráficas de electrocardiogramas, y sensores de acelerómetros. El segundo grupo procesan señales de variación discreta en el tiempo que no son inmediatas, por ejemplo, en sensores de presión arterial y sensores del tracto respiratorio. En la Tabla 1.3 se proveen algunos aspectos generales de las redes WBAN [21].

Tabla 1.3. Características generales de las redes WBAN [21].

Atributos	Valores
Tamaño de la red	Máximo 100 dispositivos
Densidad de la red	2-4 redes/m ²
Tiempo de configuración de la red	< 1 s/dispositivo
Consumo de potencia	< 1 mW/Mbit/s
Tiempo de inicio	< 100 η s
Latencia extremo a extremo	10 ms
Distancia	2 m estándares, y en algunos casos especiales 5 m

Los nodos anteriormente referidos pueden contar con diversas topologías, entre ellas, estrella, malla y árbol. Sin embargo, la más utilizada es la topología en estrella, en la cual el nodo central es identificado como nodo máster. De acuerdo con la aplicación, en algunas ocasiones se debe realizar una conexión con el nodo máster para completar el proceso de transferencia de información. En tales aplicaciones, se consideran principalmente movimientos relativos de algunas partes del cuerpo humano con relación a otras, y este aspecto es fundamental para el despliegue posicional de los nodos [21].

En la Figura 1.6 se esquematiza un ejemplo de la colocación de estos nodos en el cuerpo humano. Los nodos transmisores de sonido e imagen han de ser colocados cuidadosamente en vista de los nervios sensitivos de la cabeza. Adicionalmente, es imprescindible tener presente la sensibilidad de la tasa de absorción específica SAR (Specific Absorption Rate) del ojo. Los nodos localizados en el torso y hacia la cabeza no se deben mover demasiado. No obstante, los nodos en las proximidades de ciertas partes corporales como: piernas, brazos, torso y cabeza se pueden mover más unos con respecto a otros [21].

En la Tabla 1.4 se explican posibles comunicaciones entre nodos ubicados en el cuerpo humano. Estos nodos poseen una potencia de procesamiento muy bajo y son capaces de producir energía para todo el cuerpo, es decir en transmisiones de tipo in-body y on-body [21].

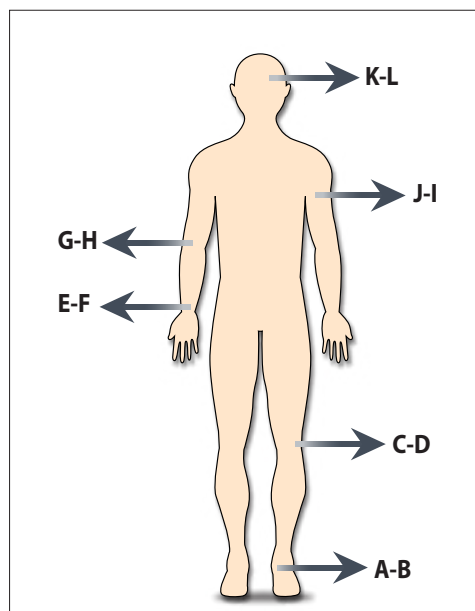


Figura 1.6. Ejemplo de colocación de nodos en el cuerpo humano para aplicaciones WBAN.

Tabla 1.4. Distribución de elementos en redes WBAN para aplicaciones in-body y on-body [21].

ENLACE ENTRE ZONAS	DESCRIPCIÓN
A-B	A través del tobillo
C-D	A través del muslo
E-F	A través de la mano
G-H	A través de la muñeca
I-J	Torso, desde el frente hacia atrás
K-L	De oído izquierdo a oído derecho

Para aplicaciones médicas y no médicas existe otra arquitectura para redes WBAN, y de acuerdo con el tipo de demanda de tráfico de información, estas aplicaciones se pueden usar para la comunicación de eventos emergentes o casos con tráfico normal (en aquellas circunstancias donde simplemente se realicen monitoreos de rutina). Si la demanda de tráfico se requiere para la transmisión de información en casos especiales, el nodo máster inicializa la comunicación. Este mismo caso ocurre cuando la transmisión de información debe tener prioridad, por ejemplo, para efectuar recomendaciones de diagnóstico y su eventual empleo como requerimiento de operaciones médicas. Asimismo, la infraestructura de comunicación en la red WLAN deberá permitir la transmisión de información como resultado de eventos médicos esporádicos. La gestión del tráfico emergente empieza con un nodo iniciador que detecta un umbral (nivel en la magnitud de una variable de información que es captada por un sensor) superior al predefinido con anterioridad. Esta clase de tráfico no se crea a intervalos regulares, lo cual generalmente es impredecible [21].

La información de tráfico catalogado como normal es frecuente en intervalos de condiciones convencionales, en los que no se producen eventos críticos en el tiempo. Este es el caso del estado de salud común de un paciente y su tratamiento correspondiente para muchas enfermedades, por ejemplo, desórdenes gastrointestinales y neurológicos, cáncer, rehabilitación, discapacidad, enfermedades cardíacas y otras. Los datos de tráfico normal se recolectan, y se transfieren al nodo máster para su procesamiento pertinente. Dependiendo de los requerimientos de la aplicación, el nodo máster puede incluir un enlace de radio de mayor capacidad para transferir información de emergencia, y un enlace adicional para múltiples conexiones de capa física. Cuando las aplicaciones así lo definen, más nodos máster se conectan a servidores de telemedicina y de aplicaciones médicas [21].

La escalabilidad en redes WBAN se limita al interior del cuerpo de una persona, pero redes WSN (Wireless Sensor Networks) pueden monitorear el entorno con una mayor cobertura. La Tabla 1.5 indica casos de uso de aplicaciones médicas y no médicas para redes WBAN [21].

Tabla 1.5. Casos de utilización de sensores en redes WBAN [21].

Tipo de Aplicación	Nodos sensores	Tipo de sensor	Ancho de banda [Hz]	Exactitud [bits]	Velocidad de transmisión	Ciclo de trabajo, porcentaje de tiempo	Potencia de consumo	Calidad de servicio	Privacidad
Aplicaciones in-body	Sensor de glucosa	Base para tirillas para glucómetro	0-50	16	Algunos Kbps	<1%	Extremadamente baja	Si	Alta
	Marcapasos	Acelerómetro	0-500	12	Algunos Kbit/s	<1%	Baja	Si	Alta
	Cápsulas de endoscopia	Píldoras que contienen una cámara diminuta	–	–	>2 Mbit/s	<50%	Baja	Si	Media
Aplicaciones médicas on-body	Electrocardiograma	Electrodos para la piel / para el pecho	100-1000	12	3 Kbit/s	<10%	Baja	Si	Alta
	Electromiografía	Electrodos para la piel	0-10000	16	320 Kbps	–	–	–	–
	Temperatura	Pruebas de temperatura o parches para la piel	0-1	8	120 bit/s	–	–	–	–
	Porcentaje de oxígeno en la sangre	Oxímetros de pulso	–	–	32 Kbit/s	<1%	Baja	Si	Alta
	Presión arterial	Monitor en brazaletes	0-1	8	<10 bit/s	<1%	Alta	Si	Media
Aplicaciones no médicas on-body	Música para auriculares	Sonido	–	–	1,4 Mbit/s	alto	Relativamente alta	Si	Baja
	Monitorización de objetos olvidados	Monitoreo	–	–	256 Kbit/s	medio	Baja	No	Baja
	Redes Sociales	Sensores combinados	–	–	<200 Kbit/s	<1%	Baja	No	Alta

Como en todas las redes de computadoras, las redes WBAN también deben cumplir requerimientos de seguridad, entre los que se encuentran [20, 22, 24]:

- 1.** Confidencialidad. El contenido de la información solo tendrá que estar disponible para el personal autenticado, el cual estará pertinentemente autorizado. Además, deben existir mecanismos de seguridad posteriores al almacenamiento de la información y para su confidencialidad. Este será también el caso para todas las fases de procesamiento de la información.
- 2.** Integridad. La información en sí y los dispositivos que la contengan no se podrán modificar por entidades no autorizadas. Es preciso verificar el origen de la información para completar el esquema de su integridad.
- 3.** Disponibilidad. La información propiamente y sus dispositivos de almacenamiento deben permitir el acceso permanente a entidades autorizadas, de modo que no ocurran interrupciones por vulneraciones.

También aspectos claves en la seguridad de las redes WBAN son aquellos relacionados con [22]:

- a)** Autenticación (en especial de firmware de dispositivos implantados).
- b)** Denegación de servicios (denial of service).
- c)** Acciones de manipulación, modificación o adición de la información por parte del personal no autorizado (data tampering).
- d)** Encriptación de la información (en particular en formas biológicas de datos, como es el caso del ADN).

El reto de la seguridad y privacidad de la información en nanocomunicaciones es mayor para las CM. Particularmente, los resultados generados en entornos experimentales, así como en simulación de tejidos humanos y de sangre artificial, no se pueden reproducir en redes reales *in-vivo*, no obstante, se investigan casos en los que las calibraciones de soluciones salinas usadas para simular señales fisiológicas podrían utilizarse en pruebas de nanocomunicaciones *in-vivo* [22].

Por otra parte, la escasez de dispositivos médicos *in-vivo* e *in-vitro* como nodos receptores, implantes médicos y nanonodos, determinan

problemas en la implementación de redes WBAN. Adicionalmente, tipos de bacterias y recolectores de energía de los alrededores de los implantes y también de implantes sin bacterias, son dificultades por considerar en los enlaces de este tipo de redes (a pesar de soluciones basadas en bacterias externas para implantes dentro del cuerpo humano) [22].

1.3.2.2 Parámetros físicos en redes BAN²

En esta sección se abordarán brevemente algunos parámetros de capas^{vii} física, enlace de datos y redes, en sistemas teóricos de CM.

1. Modelo del canal de comunicación y velocidad de transmisión. De acuerdo con la forma de propagación de la información en portadoras moleculares se tienen principalmente dos tipos de transporte: pasivo y activo [18].

En el transporte pasivo la información se emite desde el transmisor al receptor por difusión de portadoras moleculares (Figura 1.7). Algunas de las moléculas transferidas interactúan electroquímicamente con receptores biológicos en el extremo destino, y por interacción se realiza el proceso de recepción de información. En el transporte activo generalmente^{viii} la información molecular se comunica por medio de motores moleculares, en estructuras moleculares de proteína o microtúbulos (Figura 1.8) [18].

A continuación, se analizan de manera concisa estas formas de transmisión molecular.

^{vii} Salvo que se indique lo contrario, la referencia a capas estratificadas de redes de computadoras corresponderá al modelo ISO/OSI (International Organization for Standardization/Open Systems Interconnection).

^{viii} Adicionalmente existe transporte activo en procesos moleculares de difusión-advección, transporte bacteriano, uniones Gap [6, 16, 24-26].

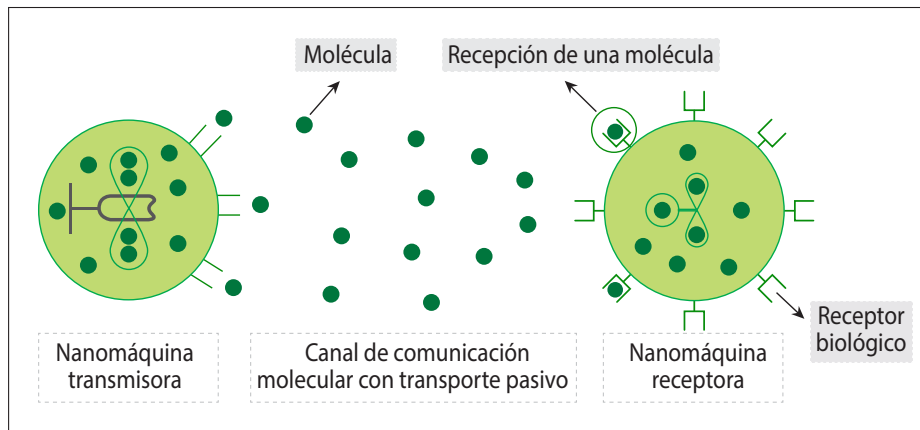


Figura 1.7. Canal molecular con transporte pasivo.

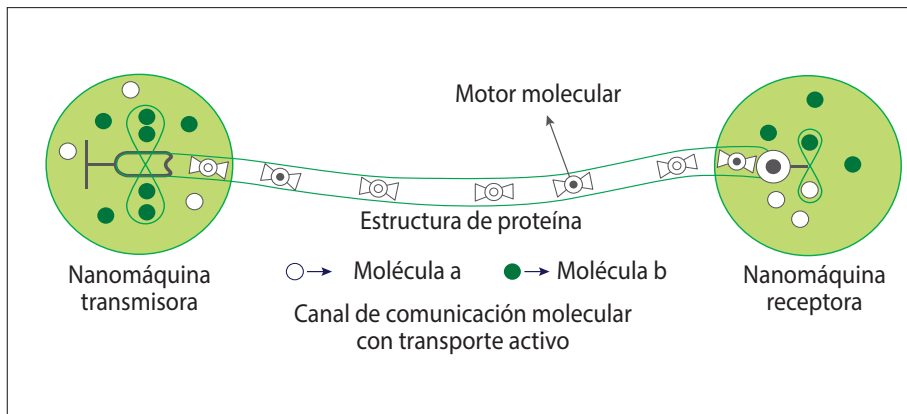


Figura 1.8. Canal molecular con transporte activo.

- a) CM con transporte pasivo. En redes inalámbricas tradicionales los mensajes de información se codifican de manera binaria (dos niveles 0 o 1), estos bits además corresponden a dos diferentes niveles de señal. Análogamente, un canal de transmisión molecular con transporte pasivo frecuentemente se modela como un canal binario, también, en estas condiciones, el transmisor genera una molécula o un número de ellas en el inicio de un periodo de tiempo de duración fija para propagar el bit-1, generalmente. Si la molécula o una concentración de estas alcanza el receptor, durante el periodo de tiempo, el extremo comunicacional destino recibe dicho bit. Caso contrario, erróneamente se recibe el bit-0 [18].

En tanto, para transmitir el bit-0 el emisor no envía molécula alguna en el período de tiempo. Sin embargo, las moléculas

emitidas en el periodo previo (correspondientes al bit-1) pueden alcanzar el receptor en forma tardía cuando se transmite el bit-0, lo cual genera errores en la comunicación. Para mitigar estos errores, el canal binario se modifica a dos periodos de duración. Específicamente, se transmiten duplas de bits 00, 01, 10, 11, para propagar los bit-0 y bit-1. El canal resultante se conoce como canal de 4 entradas/2 salidas en la Figura 1.9 [18].

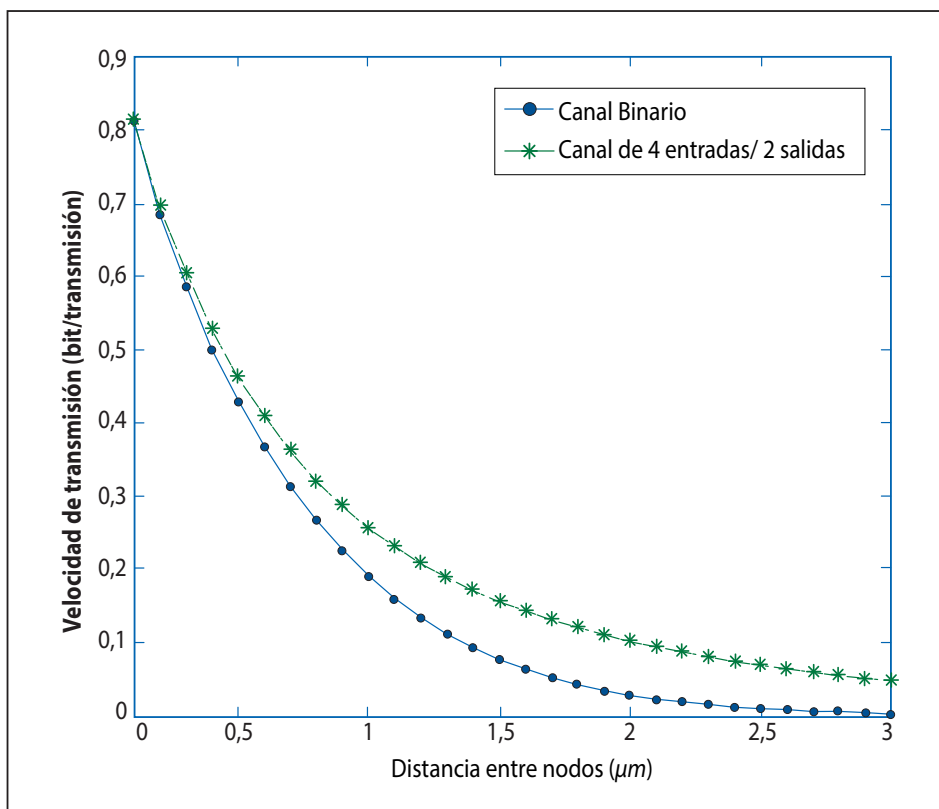


Figura 1.9. Velocidad de transmisión en un canal binario de 4 entradas/2 salidas con respecto a la variación de la distancia entre 2 nodos desde el transmisor al receptor. El canal de 4 entradas/2 salidas supera al canal binario ya que mitiga los errores originados por la expansión de la duración de cada transmisión en relación con el proceso de comunicación de un solo bit. Para la distancia más corta entre nodos (es decir 0,1-0,5 μm), el canal de 4 entradas/2 salidas, mejora levemente la velocidad de transmisión, mientras su superioridad es evidente para distancias relativamente largas. Este resultado revela que el canal de 4 entradas/2 salidas, resulta más apropiado para aplicaciones de nanomedicina en las que las nano-máquinas no se encuentran relativamente cerca [18].

b) CM con transporte activo. El rendimiento de las comunicaciones con transporte activo se caracteriza a través de parámetros basados en la media de la probabilidad de error. Las moléculas cargadas por el transmisor en motores moleculares en transporte activo se observan en la Figura 1.8, en la que se asume que por degradación la información no alcanza el receptor, con un error de probabilidad p_e si n moléculas diferentes se usan, cada una de ellas puede arribar al destino con una probabilidad de $(1 - p_e)$ o no puede alcanzar el receptor con una probabilidad p_e . Por lo que, en sí un canal molecular de transporte activo se puede modelar como un canal de borrado *n-ario* (erasure channel) y la tasa de transmisión vendrá dada por $(1-p_e) \log n$ [18].

En la Figura 1.10 la velocidad de transmisión del canal de borrado *n-ario* se muestra con diversos valores p_e de acuerdo con la variación del número de diferentes moléculas (es decir n) utilizadas en el canal. La velocidad de transmisión claramente se puede incrementar en un factor de n . La comparación numérica de las Figuras 1.9 y 1.10 denota que el canal de transporte activo provee una velocidad de transmisión más alta que sus contrapartes pasivas. Sin embargo, en el transporte activo siempre se requiere una infraestructura (en este caso elementos proteicos y motores moleculares), en tanto, en el transporte pasivo no se tiene afectación impuesta por estas restricciones, y pueden resultar más adecuados para aplicaciones de nanomedicina en las que las nanomáquinas terapéuticas pueden no necesitar altas velocidades (a nivel molecular) y solamente se precisa la propagación de información de tal manera que su contenido sea simple [18].

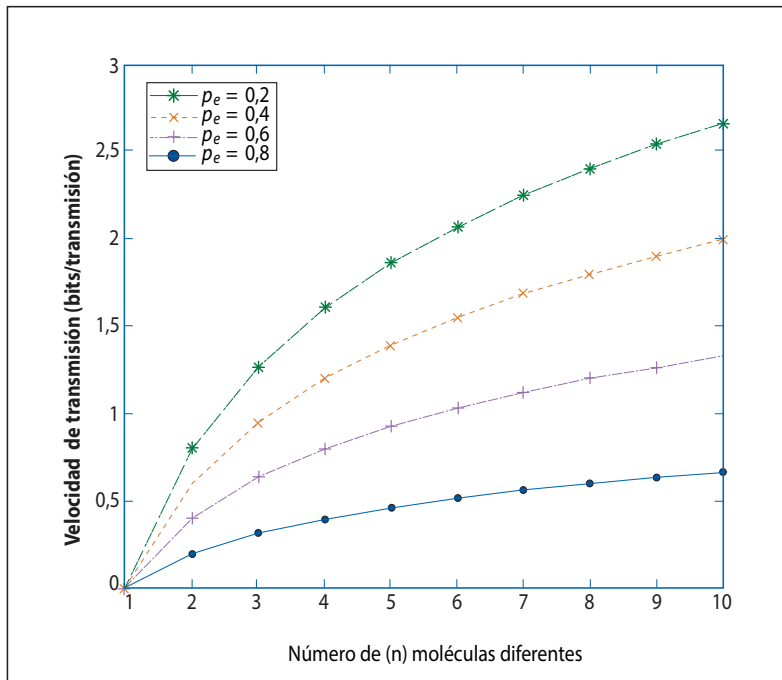


Figura 1.10. Velocidad de transmisión de un canal de borrado $n - ario$ con variación del error de probabilidad del canal pertinente a p_{e_i} , para comunicaciones con variación del número de moléculas diferentes.

2. Ruido en el canal de transmisión molecular. El efecto del ruido en el rendimiento de un sistema de CM se debe investigar considerando varias condiciones en el extremo de transmisión, en el canal propiamente, y de acuerdo con la técnica empleada para la codificación y decodificación de las portadoras de información. Por otra parte, el ruido biológico inherente en los procesos intra e intercelulares tiene que tomarse en cuenta. Adicionalmente, la capacidad del canal molecular será determinada mediante la distribución de la probabilidad de ruido. De hecho, las señales de ruido estadísticamente están correlacionadas con la señal de entrada al canal [6]. Así, la señal de entrada y el ruido se analizarán en conjunto a las características teóricas de la información en el canal de propagación. También, debido a que las moléculas emitidas se difunden continuamente en el medio de transmisión hasta el receptor se produce una contribución temporal de una señal de ruido, lo cual establece un factor importante a considerarse en las aplicaciones de las redes BAN [18].

- 3.** Velocidad de transmisión y tasa de bits errados. Los parámetros como tipo de aplicación, velocidad de transmisión y área de cobertura de la información, deben tomarse en cuenta para la definición del enlace molecular. La información además se puede transmitir en bloques de la misma clase y en altas velocidades durante requerimientos precisos. La confiabilidad de la transmisión para garantizar la adecuada recepción de la información en aplicaciones médicas viene dada por el parámetro BER (Bit Error Rate). Para dispositivos médicos, la confiabilidad se determina por la velocidad de la información. Bajas velocidades se pueden
- 4.** Baja latencia. Los mensajes considerados prioritarios se tienen que transmitir tan rápido como sea posible, lo cual puede ser difícil de gestionar porque en el caso de la utilización de CM por ejemplo, se podría tardar 12 horas en llegar a la estación de análisis, por lo que, resulta evidente que la propagación de información fuera del cuerpo humano es mucho más rápida que en su interior. Este tipo de comunicación deberá restringir al mínimo tecnologías basadas en ultrasonido u ondas de radio en el rango de THz para disminuir sustancialmente la latencia. Este hecho se sustenta también en que se tendrá que procurar el menor consumo de energía al interior del cuerpo para evitar que las nanomáquinas dejen de funcionar y además para tratar de no estresar innecesariamente al cuerpo. Para lograr cumplir tal objetivo se pretende que existan múltiples gateways de modo que cada nanored reconozca su propio gateway pertinentemente, sin embargo, un solo gateway podrá ser responsable de la comunicación de varias nanoredes [25, 26].
- 5.** Confiabilidad. La confiabilidad no es un problema fuera del cuerpo, no obstante, es complicado determinar si un mensaje alcanzó su destino comunicacional. La confiabilidad se puede mejorar a través del incremento de la probabilidad de que al menos una copia realizada de la información arribe al receptor, por ejemplo, mediante mecanismos de flooding o enviando el mensaje a múltiples gateways. El uso de acuses de recibo es difícil de efectuar en particular, en el caso de CM, pero una posibilidad de emular tales acuses se puede conseguir en la capa aplicación, es decir, por la transmisión de mensajes de comandos a la nanored y la adecuada determinación de si esto produce algún efecto en los valores de parámetros al ser controlados en el cuerpo o los modifica. Un gateway puede consecuentemente transmitir un mensaje y solicitar a los sensores correspondientes de la red BAN que informen de cambios ocurridos en las mediciones que han

y solicitar a los sensores correspondientes de la red BAN que informen de cambios ocurridos en las mediciones que han efectuado. Si ninguna respuesta se obtiene, el gateway puede reenviar el mensaje [25, 26].

1.3.2.3 Operación multicapa de redes BAN²

1. Capa Física. Generalizando las tareas de la capa física en redes BAN, se debe indicar que es la capa que define los mecanismos de transmisión de bits mediante un enlace inalámbrico hasta el receptor. Como es notable en este capítulo, algunas bandas de frecuencia se han investigado para la transmisión física de información. Por lo que, mientras la banda de 404 - 405 MHz parece ser favorable para comunicaciones MICS (Medical Implant Communications Service) usadas para sensores implantados, las bandas ISM (Industrial Scientific and Medical) definidas como 13,5 MHz; 400 MHz; 900 MHz; 2,4 GHz y 3,1 - 10,6 GHz (banda UWB) aparentan ser adecuadas para sensores de tipo on-body. Sin embargo, las señales de radio al circular por el cuerpo humano se difractan. De esta manera, las pérdidas de trayectoria son mucho mayores cuando los nodos transmisores y receptores se colocan en el mismo lado del cuerpo. Más aun, el entorno de propagación dinámico causado por el movimiento corporal y el desvanecimiento por multitraectoria generan problemas en la validación de los modelos empíricos del canal de propagación. Para la banda ISM de 2,4 GHz, la distribución Rician y el patrón de decaimiento exponencial se pueden emplear para describir a pequeña escala el desvanecimiento y los modelos de pérdidas por trayectoria en el perímetro alrededor del cuerpo humano, respectivamente. Para canales on-body, una señal de radio en la banda ISM 13,5 MHz presenta una pérdida por trayectoria similar a la del espacio libre [24-26].

En cuanto a la tecnología de radio, una de las más prometedoras es la banda UWB y su operación en la banda libre de 3,1 - 10,6 GHz caracterizada por una emisión de densidad espectral de potencia relativamente baja, la convierte en apropiada para comunicaciones a distancias cortas con velocidades de transmisión de hasta 480 Mbps. Los entornos de hospitales se benefician grandemente de estas características, dado que las interferencias de radio con equipos médicos son absolutamente indeseables [24-26].

La capa física trata los mecanismos bio-fisiológicos de transmisión, propagación, y recepción de información. En esencia, las señales de información biológicas a nanonivel son moléculas, de modo que los procesos físicos/químicos de su comportamiento deberán tenerse presente en esta capa, incluyendo formas de

ruido, interferencia, y atenuación molecular. Aspectos para interfaces y hardware distintivos en esta capa se relacionan con [10, 16, 24-26]:

- a)** Hardware. Los elementos fundamentales en CM son emisor, canal de comunicaciones, receptor, moléculas de información, y moléculas secundarias.
- b)** Información/mensajes. En redes WBAN la forma convencional de comunicación ocurre a 2,4 o a 60 GHz en la banda milimétrica. En redes WBAN no convencionales, la capa física emite mensajes moleculares en portadoras de información que son las mismas moléculas de información, a diferencia de sus contrapartes de telecomunicaciones. Las moléculas de información pueden ser proteínas, vesículas, o bacterias.
- c)** Técnicas de modulación. La modulación en CM consiste en el proceso de transformar una o más características de la física y/o química de las moléculas de modo de poder interpretar estas modificaciones, a través de un proceso de control de la información en lo que se refiere a su interpretación al llegar al receptor. Las principales técnicas de modulación se establecen mediante la transmisión de una molécula o concentraciones de estas (lo que representaría la técnica de modulación en amplitud convencional), un tipo de molécula diferente en cada emisión, o el tiempo de emisión molecular (equivalente a la modulación en frecuencia en los sistemas de telecomunicaciones) constituyen las principales técnicas de modulación en CM. Adicionalmente, es muy común que en algunos casos, se emplee una combinación de las técnicas de modulación descritas [10].
- d)** Modelos del canal de propagación. Los modelos de propagación en canales de CM se dividen básicamente en transporte pasivo y activo, en el primer caso se encuentran los modelos de difusión libre en tanto que en el segundo se tienen, la difusión con arrastre, el transporte bacteriano, los motores moleculares y las uniones Gap.
- e)** Fuentes de ruido. El ruido presente en CM se origina fundamentalmente en los procesos de emisión, propagación aleatoria, en el canal de transmisión propiamente, y en el extremo de recepción. Generalmente en todas estas etapas de la comunicación, el ruido es el resultado de reacciones químicas entre moléculas, la degradación molecular en sí y la interacción de otras fuentes de transmisión molecular.

- f) Capacidad del canal. De manera práctica se puede asumir que los canales de CM no son canales sin memoria. Por lo que la capacidad del canal será diferente para diversos modelos de propagación. En general, si x^n representa una secuencia de n símbolos de transmisiones sucesivas, y y^n denota los símbolos recibidos correspondientemente, y entonces, la capacidad del canal será (los términos en la Ecuación 1.1 son tomados del inglés):

$$C = \lim_{n \rightarrow \infty} \ln \sup_{x^n} I(X^n; Y^n)$$

2. Capa Enlace de Datos. En redes BAN los nodos usualmente comparten un canal inalámbrico común para las transmisiones. Un canal compartido se controla con el empleo de las funcionalidades de la capa enlace de datos, en particular se utilizan las tareas de administración de acceso al medio de la subcapa MAC (Medium Access Control) para la gestión de un protocolo de contención de la transmisión, este aspecto resulta ser apropiado para redes BAN, sin embargo, en razón de que los sensores podrían entrar en un modo de inactividad hasta contar con el tiempo pertinente para poder comunicar su información, se tiene que garantizar la actividad permanente de tales sensores sin que esto implique un gasto energético innecesario en sus baterías. Precisamente, este es el caso de varios protocolos para esta subcapa, cuya funcionalidad ha resultado ser eficiente [27-30]. Adicionalmente, en la literatura se han propuesto varios protocolos basados en el dinamismo de periodos de tiempo para la transmisión de información de sensores en redes BAN [27-30].

De manera similar como ocurre en las telecomunicaciones, la capa enlace de datos debe enmascarar problemas que se suscitan a nivel físico, por lo que las funciones de esta capa son [10, 16, 24-26]:

- a) Manejo de errores. Para detectar los mensajes transmitidos, el número de moléculas recibidas tiene que ser superior a cierto umbral de modo que se pueda realizar su detección correspondiente en forma adecuada. No obstante, la emisión de un gran número de moléculas incrementa la interferencia entre moléculas de periodos de tiempo anteriores, generando una mayor tasa de ruido. Para evitar errores en la comunicación se debe tratar de ajustar apropiadamente la velocidad de transmisión. La detección y corrección de errores se puede conseguir añadiendo moléculas de redundancia a las moléculas de información propiamente. Por

ejemplo, cuando se añade una secuencia de ADN adicional a una molécula de ADN previa, se permite la detección de errores y la corrección hacia adelante.

- b) Direccionamiento.** Es el proceso por el cual se especifica un destino molecular. Usualmente el direccionamiento se realiza por el tipo de molécula emitida en el extremo transmisor.
- c) Sincronismo.** El sincronismo de los extremos comunicacionales en sistemas de CM es complicado debido al gran retardo y jitter presente. Sin embargo, existen investigaciones proponiendo mecanismos factibles de sincronismo molecular, algunos de los más representativos se encuentran en [31-35].
- d) Control de acceso al medio.** La interferencia entre múltiples pares de transmisores y receptores se puede evitar utilizando diferentes tipos de moléculas para cada par transmisor-receptor, aunque esta técnica no resulta realmente efectiva cuando la cantidad de dichos pares es demasiado numerosa. En esos casos, la multiplexación ofrece una alternativa razonable para acceder al medio de comunicación.
- e) Control de flujo.** Si las reacciones químicas en el receptor son notablemente menores que en el extremo de transmisión, entonces es imprescindible implementar el control de flujo para evitar el desborde de información en el extremo de recepción molecular.

3. Capa Red. Esta capa es la responsable del envío efectivo, eficiente, y oportuno de paquetes desde un nodo fuente a uno destino, generalmente mediante un número determinado de nodos intermedios. En esta capa se tendrán que determinar además el establecimiento y mantenimiento de rutas al receptor pertinente [26].

En la literatura existen procesos de enrutamiento estático a través de la transmisión de información de bacterias con moléculas de direccionamiento que les guían al receptor particular al cual se dirige la comunicación. El nodo que actúa como router toma la bacteria y emplea procesos químicos estadísticamente definidos para retransmitir la información empleando una bacteria que es orientada por un gradiente químico hasta otro router determinado. El proceso de detección de quórum bacteriano es otra forma de enrutamiento [10]. En [36] se analizan varias propuestas de protocolos de enrutamiento para redes WBAN basados en temperatura.

1.3.2.4. Software de simulación para redes BAN

- a) NS-3. El simulador de software abierto de eventos discretos NS-3 (Network Simulator-3) permite escalabilidad y extensiones por medio de implementaciones modulares y una arquitectura central que se fundamenta en programación orientada a objetos. El simulador soporta modelos desarrollados en C++, y usa scripts en el lenguaje Python. NS-3 soporta virtualización en red y pruebas reales de integración y se distribuye con muchas actualizaciones de manuales de usuario, además cuenta con una estructura estadística múltiple [26]. NS-3 será analizado en el siguiente capítulo, Sección 2.3.5.

Para implementar un enlace molecular en NS-3 se deben establecer los procesos que se indican a continuación [6]:

1. Creación del ayudante P1906.
2. Creación de nodos.
3. Creación del medio de propagación y del componente de movimiento.
4. Creación de otros componentes y sus dispositivos.
5. Determinación de la posición y el modelo de movimiento del transmisor hacia el receptor.
6. Conexión de dispositivos, medios, componentes, y entidades.
7. Creación de la unidad de datos del protocolo PDU (Protocol Data Unit).
8. Transferencia de información.

A continuación, se debe realizar el proceso de comunicación entre componentes y se deben implementar las clases que ejecutan el marco de referencia de los estándares IEEE de nanocomunicaciones, es decir, los estándares 1906.1 y 1906.1.1 [6]. Estas clases administran los componentes de comunicación de los estándares en mención. Así, las clases para los componentes *Message Carrier*, *Perturbation*, *Motion*, *Field* y *Specificity*, y sus elementos se tendrán que utilizar en la topología de red que se desee implementar, teniendo en cuenta explícitamente las características del

enlace entre los extremos de comunicación y las características físicas que permite la transmisión de información entre el transmisor y receptor moleculares.

- b) OMNeT++.** También con arquitectura abierta OMNeT++ es un simulador de eventos discretos, y su IDE (Integrated Development Environment) se fundamenta en Eclipse, facilitando el ensamble gráfico y la configuración de simulaciones como una alternativa para archivos en texto plano. En el entorno de programación existen dos opciones, en una de ellas la interfaz basada en línea de comandos se apunta a la ejecución por lotes en máquinas especializadas; en tanto que en el entorno gráfico se soporta de mejor manera las interacciones con los componentes de la simulación que se ejecuta, permitiendo el monitoreo directo alterando estados internos [26].

OMNeT++ refuerza una separación estricta del código en lo que se refiere a comportamiento y descripción. El comportamiento del código (es decir, la especificación del código en cuanto a la gestión de módulos simples y el envío de mensajes que estos operan, además de cómo los canales manejan estos mensajes) está escrito en C++. El código descriptivo (es decir, cómo el código declara la estructura de módulos/canales y mensajes) se almacena en texto plano en los archivos *Message Definition (msg)*, y *Network Description (ned)*, respectivamente. Las configuraciones del tiempo de ejecución de módulos se encuentran en un archivo de inicialización. Con el comportamiento del código contenido en un programa de C++, los componentes de OMNeT++ pueden interactuar fácilmente con terceras librerías y se puede depurar empleando utilitarios, por tanto, estas características establecen la rapidez en la reacción y desarrollo de aplicaciones de calidad [26].

Estos dos tipos de software denotan un paradigma de programación para aplicaciones de nanocomunicaciones^{ix} flexibles y de alta calidad. La ventaja de NS-3 radica en el hecho de que es una plataforma versátil para el modelamiento de características de nanocomunicaciones y particularmente de CM en lo que respecta a varios tipos de transmisores, canales de propagación, y receptores. Igualmente, NS-3 posibilita la programación de emisores en los que se puede definir diferentes formas de modulación molecular (amplitud, tipo de moléculas y tiempo de envío). En cuanto se refiere al canal de transmisión se pueden usar transporte activo y pasivo de moléculas, y en el extremo destino se pueden emplear receptores biológicos activos y pasivos [26].

^{ix} OMNeT++ se usa principalmente para programación de nanocomunicaciones inalámbricas, en las que las antenas trabajan a frecuencias en el rango de THz.

La integración de los aspectos de funcionamiento de modelos de nanocomunicaciones in-body y de comunicaciones por ondas de radio en redes BAN proporciona enormes posibilidades en el rendimiento de sistemas de CM, pero también establecen alta complejidad en aplicaciones de nanomedicina [26].

En la literatura no se menciona explícitamente a OMNeT++ como simulador de CM, sin embargo, como se ha analizado las CM constituyen una parte importante de las redes BAN o WBAN en las que OMNeT++ es uno de los simuladores más empleados [37-44].

En cuanto se refiere a simuladores o lenguajes de marcado propiamente para CM se tiene principalmente a:

- BiNS2 (Biological and Nano-Scale Communication Simulator).
- N3Sim (Simulation framework for diffusion-based molecular communication nanonetworks).
- CalComSim (Calcium Signaling Molecular Communication Sim).
- COMSOL Multiphysics.
- NS-2 (Network Simulator-2) y NS-3 (Network Simulator-3).
- MolComML (Molecular Communications Markup Language).
- BNSim (Bacteria Network Simulator).
- NCSim (Bacteria Nanonetworks).
- HLA (High-level Architecture)
- dMCS (Distributed Molecular Communication Simulator)
- MUCIN (MolecUlar CommunicatIoN)
- AcCoRD (Actor-based Communication via Reaction-Diffusion)
- N4Sim (Nervous NaNoNetwork Simulator)
- TouchCom (Touchable Communication)
- SBML (Systems Biology Markup Language).
- SBOL (Synthetic Biology Open Language).
- CellML (Cellular Modeling Markup Language).
- NeuroML (Neuroscience Markup Language).

Estos programas para estudio de sistemas de CM se analizarán en el Capítulo 2.



Referencias Bibliográficas

- [1] Y. Cevallos, L. Tello-Oquendo, D. Inca, D. Ghose, A. Z. Shirazi, and G. A. Gomez, "Health Applications Based on Molecular Communications: A Brief Review," in *2019 IEEE International Conference on E-health Networking, Application & Services (HealthCom)*, 2019, pp. 1–6.
- [2] Y. Cevallos *et al.*, "A brief review on dna storage, compression, and digitalization," *Nano Commun. Netw.*, vol. 31, p. 100391, 2022.
- [3] Y. Cevallos, L. Molina, A. Santillán, F. De Rango, A. Rushdi, and J. B. Alonso, "A digital communication analysis of gene expression of proteins in biological systems: A layered network model view," *Cognit. Comput.*, vol. 9, no. 1, pp. 43–67, 2017.
- [4] Y. Cevallos *et al.*, "On the efficient digital code representation in DNA-based data storage," in *Proceedings of the 7th ACM International Conference on Nanoscale Computing and Communication*, 2020, pp. 1–7.
- [5] Y. Cevallos, L. Tello-Oquendo, D. Inca, C. Palacios, and L. Rentería, "Genetic Expression in Biological Systems: A Digital Communication Perspective," *Open Bioinforma. J.*, vol. 12, no. 1, 2019.
- [6] Y. Cevallos *et al.*, "Comunicaciones Moleculares Un análisis desde el paradigma protocolario de IEEE," 2023.
- [7] Y. Cevallos *et al.*, "Modelamiento comunicacional de la expresión genética y el transporte de proteínas mediante un sistema de transmisión digital extremo a extremo," 2022.
- [8] Y. Cevallos *et al.*, "Modeling Gene Expression and Protein Delivery as an End-to-End Digital Communication System," *Open Bioinforma. J.*, vol. 14, no. 1, 2021.
- [9] M. Gong, L. Pan, T. Song, and G. Zhang, *Bio-inspired Computing -- Theories and Applications: 11th International Conference, BIC-TA 2016, Xi'an, China, October 28-30, 2016, Revised Selected Papers*, no. pt. 1. Springer Singapore, 2017. [Online]. Available: https://books.google.it/books?id=_HbbDQAAQBAJ

- [10] Y. Cevallos *et al.*, “Analysis of Layer’s Tasks in Molecular Communication: Application, Transport, Network, and Link Layers,” in *Molecular Communications: An Analysis from Networking Theories Perspective*, Springer, 2023, pp. 13–61.
- [11] D. Bi, A. Almpanis, A. Noel, Y. Deng, and R. Schober, “A survey of molecular communication in cell biology: Establishing a new hierarchy for interdisciplinary applications,” *IEEE Commun. Surv. & Tutorials*, 2021.
- [12] L. Kong *et al.*, “A Survey for Possible Technologies of Micro/nano-machines Used for Molecular Communication within 6G Application Scenarios,” *IEEE Internet Things J.*, 2023.
- [13] M. Egan *et al.*, “Toward interdisciplinary synergies in molecular communications: Perspectives from synthetic biology, nanotechnology, communications engineering and philosophy of science,” *Life*, vol. 13, no. 1, p. 208, 2023.
- [14] L. Chouhan and M.-S. Alouini, “Interfacing of Molecular Communication System with Various Communication Systems over Internet of Every Nano Things,” *IEEE Internet Things J.*, 2023.
- [15] W. Haselmayr *et al.*, “Integration of molecular communications into future generation wireless networks,” *Proc. 6G Wirel. Summit*, pp. 1–2, 2019.
- [16] M. D. N. Anjum and H. Wang, “Molecular Communication for Wireless Body Area Networks,” 2019.
- [17] I. F. Akyildiz, M. Pierobon, and S. Balasubramaniam, “Moving forward with molecular communication: From theory to human health applications [point of view],” *Proc. IEEE*, vol. 107, no. 5, pp. 858–865, 2019.
- [18] B. Atakan, O. B. Akan, and S. Balasubramaniam, “Body area nanonetworks with molecular communications in nanomedicine,” *IEEE Commun. Mag.*, vol. 50, no. 1, pp. 28–34, 2012.

- [19] P. Kulakowski, K. Turbic, and L. M. Correia, "From nano-communications to body area networks: A perspective on truly personal communications," *IEEE Access*, vol. 8, pp. 159839–159853, 2020.
- [20] S. Javaid, S. Zeadally, H. Fahim, and B. He, "Medical sensors and their integration in wireless body area networks for pervasive health-care delivery: A review," *IEEE Sens. J.*, vol. 22, no. 5, pp. 3860–3877, 2022.
- [21] F. R. Yazdi, M. Hosseinzadeh, and S. Jabbehdari, "A review of state-of-the-art on wireless body area networks," *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 11, 2017.
- [22] M. Usman, M. R. Asghar, I. S. Ansari, and M. Qaraqe, "Security in wireless body area networks: From in-body to off-body communications," *IEEE Access*, vol. 6, pp. 58064–58074, 2018.
- [23] L. Khaloopour, M. Mirmohseni, and M. Nasiri-Kenari, "Silent Target Localization Using Molecular Diffusion".
- [24] M. Yaghoubi, K. Ahmed, and Y. Miao, "Wireless body area network (WBAN): A survey on architecture, technologies, energy consumption, and security challenges," *J. Sens. Actuator Networks*, vol. 11, no. 4, p. 67, 2022.
- [25] F. Dressler and S. Fischer, "Connecting in-body nano communication with body area networks: Challenges and opportunities of the Internet of Nano Things," *Nano Commun. Netw.*, vol. 6, no. 2, pp. 29–38, 2015.
- [26] S. González-Valenzuela, X. Liang, H. Cao, M. Chen, and V. C. M. Leung, "Body area networks," *Auton. Sens. Networks Collect. Sens. Strateg. Anal. Purp.*, pp. 17–37, 2013.
- [27] M. Kaur, R. Bajaj, and N. Kaur, "A review of mac layer for wireless body area network," *J. Med. Biol. Eng.*, pp. 1–38, 2021.
- [28] S. Misra, P. K. Bishoyi, and S. Sarkar, "i-MAC: In-body sensor MAC in wireless body area networks for healthcare IoT," *IEEE Syst. J.*, vol. 15, no. 3, pp. 4413–4420, 2020.

- [29] K. Ray, V. Pal, G. Singal, and S. Moulik, "Fuzzy-MAC: An FIS based MAC protocol for a multi-constrained traffic in wireless body area networks," *Comput. Commun.*, vol. 195, pp. 451–462, 2022.
- [30] I. S. Srinivasa Rao and V. Sivakumar, "6LoWPAN MAC layer parameters optimization using evolutionary algorithm based ANN topology in wireless body area networks," *J. Intell. Fuzzy Syst.*, no. Preprint, pp. 1–25, 2023.
- [31] V. Jamali, A. Ahmadzadeh, and R. Schober, "Symbol synchronization for diffusion-based molecular communications," *IEEE Trans. Nanobioscience*, vol. 16, no. 8, pp. 873–887, 2017.
- [32] V. Jamali, A. Ahmadzadeh, and R. Schober, "Symbol synchronization for diffusive molecular communication systems," in *2017 IEEE International Conference on Communications (ICC)*, 2017, pp. 1–7.
- [33] L. Huang, L. Lin, F. Liu, and H. Yan, "Clock synchronization for mobile molecular communication systems," *IEEE Trans. Nanobioscience*, vol. 20, no. 4, pp. 406–415, 2020.
- [34] T.-Y. Tung and U. Mitra, "Synchronization error robust transceivers for molecular communication," *IEEE Trans. Mol. Biol. Multi-Scale Commun.*, vol. 5, no. 3, pp. 207–221, 2019.
- [35] L. Lin, W. Li, R. Zheng, F. Liu, and H. Yan, "Diffusion-based reference broadcast synchronization for molecular communication in nanonetworks," *IEEE Access*, vol. 7, pp. 95527–95535, 2019.
- [36] S. Jalili Marandi, M. Golsorkhtabaramiri, M. Hosseinzadeh, and S. Jafarali Jassbi, "IoT based thermal aware routing protocols in wireless body area networks: Survey: IoT based thermal aware routing in WBAN," *IET Commun.*, vol. 16, no. 15, pp. 1753–1771, 2022.
- [37] K. V. S. R. Murthy, S. Kumar, and M. K. Singh, "Recent Developments in Electronics and Communication Systems: Proceedings of the First International Conference on Recent Developments in Electronics and Communication Systems (RDECS-2022)," 2023.

- [38] C. Smera and J. Sandeep, "Networks simulation: Research based implementation using tools and approaches," in *2022 IEEE 3rd Global Conference for Advancement in Technology (GCAT)*, 2022, pp. 1-7.
- [39] M. M. Alam *et al.*, "Realistic simulation for body area and body-to-body networks," *Sensors*, vol. 16, no. 4, p. 561, 2016.
- [40] S. Verma, S. Kaur, and A. D. Gupta, "A Brief Survey on Simulators for Designing Routing Protocol for Wireless Sensor Network," in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2023, pp. 1-6.
- [41] G. H. Adday, S. K. Subramaniam, Z. A. Zukarnain, and N. Samian, "Investigating and Analyzing Simulation Tools of Wireless Sensor Networks: A Comprehensive Survey," *IEEE Access*, 2024.
- [42] W. H. W. Hassan, S. Sarang, D. M. Ali, G. M. Stojanovic, W. N. S. W. Muhammad, and N. Ya'acob, "Adaptive Medium Access Control Protocol for Dynamic Medical Traffic with Quality of Service Provisioning in Wireless Body Area Network," *IEEE Access*, 2024.
- [43] S. Alahakoon, M. M. Alobaedy, and Y. Fazea, "Conceptual Framework for the Optimization of Edge Device Workload Allocation in Wireless Body Area Networks," in *2024 International Symposium on Networks, Computers and Communications (ISNCC)*, 2024, pp. 1-5.
- [44] R. Sangeetha and U. D. Gandhi, "Priority Based Energy Efficient MAC Protocol by Varying Data Rate for Wireless Body Area Network.," *Comput. Syst. Sci. Eng.*, vol. 48, no. 2, 2024.

**Entornos de programación
para el análisis y simulación en
Comunicaciones Moleculares**

02

Las Comunicaciones Moleculares (CM), constituyen un área de investigación que combina los campos científicos de la biología y la ingeniería. Así, todas las publicaciones científicas en CM examinan señales químicas como portadoras de información en la biología celular empleando las teorías comunicacionales propias de los sistemas de transmisión convencionales [1, 2]. Este paradigma científico, ha permitido que mediante la simulación y el uso de los conceptos de comunicación mencionados previamente, se pueda analizar la transmisión, propagación y recepción de información molecular en sistemas biológicos con aplicaciones potenciales, fundamentalmente en tratamientos médicos para mejorar la calidad de vida de las personas. Este hecho ha generado el apareamiento de una cantidad importante de herramientas para la programación de tales aplicaciones. Por lo tanto, en este capítulo se analizarán los principales lenguajes de marcado y simuladores para CM.

2.1

CONTEXTO DEL USO DE HERRAMIENTAS DIGITALES EN EL ANÁLISIS DE COMUNICACIONES MOLECULARES

Debido a la gran cantidad de programas para el análisis y aplicación de sistemas de CM, se ha alcanzado un nivel de heterogeneidad muy alto entre las diferentes plataformas digitales de programación de CM. Esta situación ocurre porque la mayoría de estas herramientas digitales son incompatibles entre sí, lo que produce que no se puedan replicar los resultados de investigaciones en otras plataformas, sin posibilitar entonces la usabilidad y la eficiencia de investigaciones previas [1, 2]. Lo cual ha estimulado la imprescindible necesidad de un “lenguaje” común para describir y configurar escenarios complejos de sistemas de transmisión biológicos para reusar los resultados entre investigaciones, y obtener la compatibilidad apropiada entre diversos tipos de software en el desarrollo de aplicaciones en CM a nivel académico, investigativo e industrial [3-8].

En ese sentido, como se ha indicado en el Capítulo 1, IEEE (Institute of Electrical and Electronics Engineers) mediante los estándares de nanocomunicaciones 1906.1 y 1906.1.1 con el empleo de NS-3 (Network Simulator-3) y el proyecto CIRCLE, a través del uso de MolComML (Molecular Communications Markup Language), han fomentado iniciativas para tratar de alcanzar usabilidad y compatibilidad, no obstante, estas dos iniciativas no se han adoptado completamente, por lo que, actualmente existe una gran variedad de herramientas digitales para el análisis de sistemas de CM. En este capítulo se estudian brevemente los lenguajes de marcado y simuladores más comunes en el contexto y la terminología frecuente que se utiliza para la comprensión de sistemas de nanocomunicaciones a nivel molecular [3-8].

Precisamente, en la Figura 2.1 se muestra un escenario de CM genérico con componentes de transmisión/recepción que generalmente usan análisis numérico y/o síntesis experimental [1, 9].

En este ejemplo, se propone un protocolo bidireccional, similar a TCP (Transmission Control Protocol) que funciona en dos capas de comunicación formando una arquitectura de transmisión. Esta arquitectura incluye los siguientes elementos de comunicación [1, 9]:

- 1.** Transmisor (elemento circular de color celeste, contiene receptores biológicos).
- 2.** Destino comunicacional o receptor (elemento circular de tonalidad verde y crema que contiene receptores biológicos).
- 3.** Dos tipos de señales de información representadas en la Figura 2.1 mediante círculos y esferas de colores naranja y rojo respectivamente. En el transmisor se efectúa la comunicación en forma dúplex, es decir se emite una señal de información y se recibe otra que opera como control (retroalimentación o acuse de recibo). En el extremo destino, los receptores biológicos captan las señales y producen el acuse de recibo correspondiente. El transmisor y el destino comunicacional se encuentran en un entorno líquido (fluido) que se puede abstraer como “espacio ilimitado”. En este escenario, se tiene entonces una comunicación bidireccional a través de diferentes tipos de moléculas que se propagan, por ejemplo, mediante mecanismos de difusión en el entorno. La direccionalidad molecular desde el transmisor al receptor se usa para transportar la información en sí, mientras que la transferencia molecular desde el receptor al transmisor sirve como mensajes de control para regular la velocidad de liberación de moléculas de información desde el transmisor al entorno circundante.

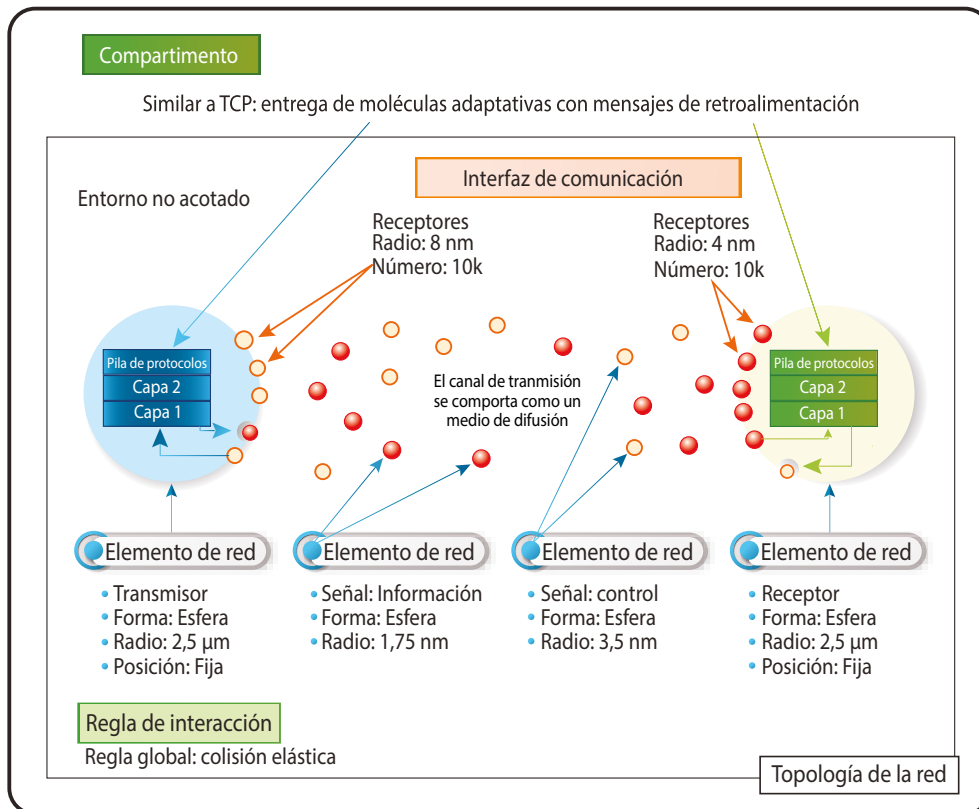


Figura 2.1. Escenario ilustrativo de CM.

En la Figura 2.1 los elementos de comunicación son fundamentales en una arquitectura molecular como ocurre también en los sistemas de telecomunicaciones. Estos elementos se crean e interconectan para definir una simulación o configuración experimental. Cada elemento se puede establecer en diferentes niveles de abstracción. Por ejemplo, un elemento podría ser una entidad conceptual, o estar relacionado con algunas características físicas para la propagación molecular. Además, cada elemento de comunicación posee un conjunto de atributos generales que se pueden extender mediante propiedades particulares con ciertos atributos (elementos de red) [1]. Por tanto, los elementos de comunicación pueden ser referidos también como elementos de red, los cuales pueden intercambiar información utilizando una interfaz de comunicación. Dichas interfaces cuentan con varias propiedades que describen el tipo de señales transmitidas y recibidas, su afinidad, la dirección de la comunicación, y son reconocidas en el extremo destino. También se puede definir un subconjunto de parámetros específicos para detallar las propiedades de cada interfaz [1, 9].

Por otra parte, en la Figura 2.1 los compartimentos se diseñan como entornos o medios de propagación en constante agitación. Se carac-

terizan por un tipo y forma particular donde pueden ubicarse especies (como es el caso de sustancias químicas y/o células modeladas como elementos de red) [1, 9].

Un escenario de simulación puede contener múltiples compartimentos, incluso del mismo tipo, organizados en una arquitectura jerárquica anidada o cercanos entre sí. En ambos casos, las conexiones entre los diferentes compartimentos están representadas por vías que son definidas mediante reglas de interacción. Cada elemento de red en un escenario de simulación está ubicado dentro de un compartimento, y, su asociación a elementos de red responde a tales reglas de interacción. Una regla de interacción puede ser global, es decir, válida en todas partes del escenario de simulación, o una regla más específica, solo en ciertas partes de este escenario. Los elementos y compartimentos de red deben estar asociados con al menos una regla de interacción [10].

Otro parámetro importante de simulación es la sección referente a la topología de red que se emplea para detallar el desplazamiento relativo de los elementos de red en los entornos de simulación. Además, se trabaja comúnmente con una pila de protocolos que se asocia con el estado interno de los elementos de red. La pila de protocolos de comunicación en el ejemplo de la Figura 2.1 se ha diseñado en dos capas, la Capa 1 en este caso consta de propiedades de la capa física del networking tradicional, y así, se encarga de transmitir/recibir ráfagas de información para codificar/detectar símbolos; la Capa 2 debe decodificar mensajes, decidir el nivel de señal para limitar la congestión del receptor y enviar señales de retroalimentación [10].

Aunque los componentes básicos de comunicación descritos en el ejemplo anterior son idénticos a aquellos que los sistemas de transmisión convencionales usan, es imprescindible notar que las CM realmente son un nuevo paradigma de comunicación, cuyas características de emisión/propagación/recepción de información se fundamentan en la forma en la que la biología permite la comunicación de sus elementos constitutivos, los cuales se diferencian claramente de las ondas electromagnéticas. La Figura 2.2 especifica estas diferencias [2].



Figura 2.2. Diferencia entre los sistemas de telecomunicaciones y sistemas de CM [2].

2.2

LENGUAJES DE PROGRAMACIÓN BASADOS EN XML (eXtensible Markup Language) PARA COMUNICACIONES MOLECULARES

XML es un metalenguaje que se utiliza para el almacenamiento de datos en forma legible y el intercambio de información estructurada entre diferentes plataformas [11, 12].

Además, XML cumple un rol importante en la actualidad dado que posibilita la compatibilidad entre sistemas para compartir información de una manera segura, confiable y sencilla. Existen diferentes lenguajes de programación para CM que se fundamentan en XML (componentes en la Figura 2.3), uno de los más empleados es SBML (Systems Biology Markup Language) [13], este lenguaje modela procesos comunicacionales en seres vivos a través de enfoques computacionales en sistemas biológicos. SBML propicia la representación de diferentes clases de fenómenos biológicos, como vías de señalización celular, redes reguladoras, entre otras, facilitando el intercambio de modelos entre diferentes plataformas de software [14].

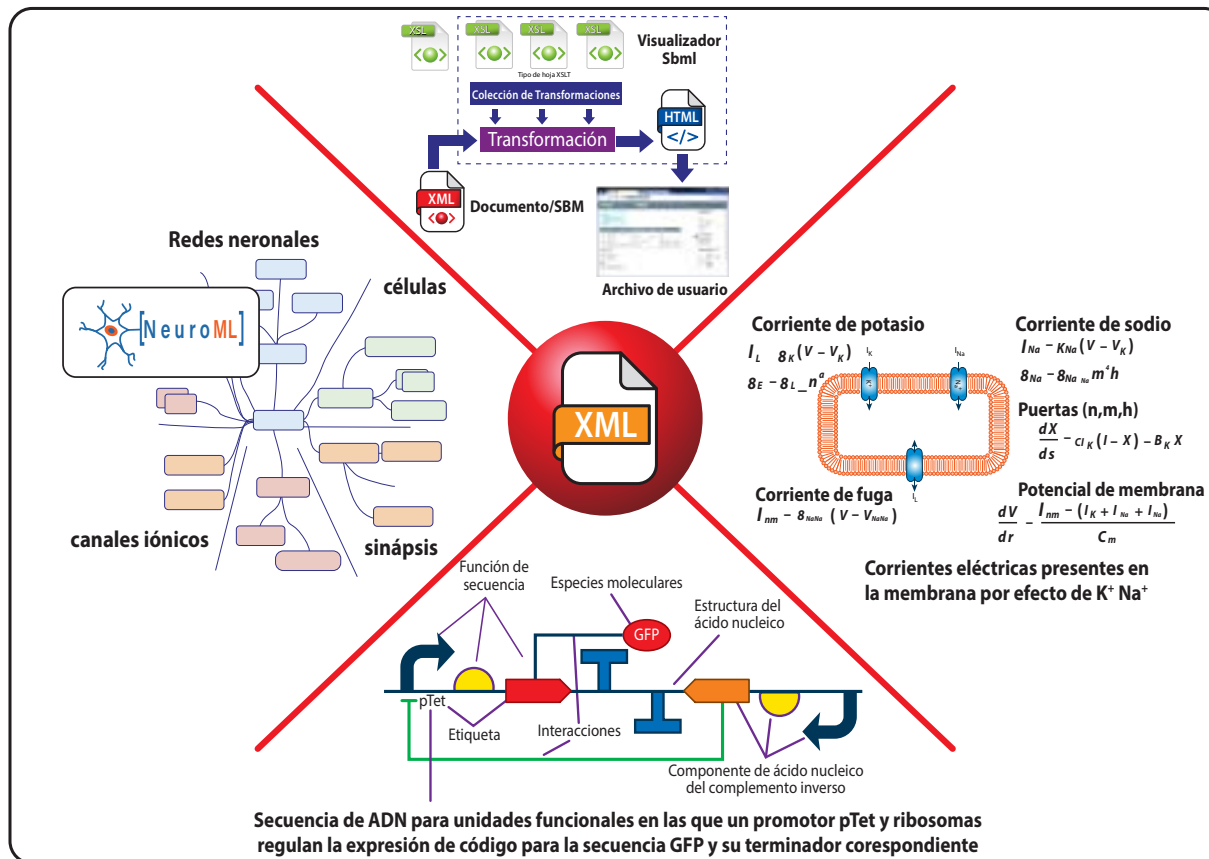


Figura 2.3. Lenguajes basados en XML para CM.

SBML usa UML (Unified Modeling Language) y este a su vez se emplea para representar información en XML. A continuación, se enlista algunos componentes que ayudan en la definición de un modelo en SBML [15, 16]:

1. **Compartimento.** Características de un volumen finito que se encuentra en constante agitación en el que se agregan sustancias biológicas para que existan reacciones químicas.
2. **Especie.** Sustancia o entidad química que participa en una reacción. Algunos ejemplos de especies son los iones, entre los que se encuentran los de calcio y moléculas como aquellas de ATP (Adenosine Triphosphate).
3. **Reacción.** Declaración que detalla alguna transformación, proceso de transporte o unión que puede modificar una o más especies. Las reacciones cumplen leyes de velocidad asociadas que determinan su comportamiento.
4. **Parámetro.** Cantidad que cuenta con un nombre simbólico. SBML proporciona la capacidad de particularizar parámetros globales de un modelo, así como los parámetros específicos de una reacción.
5. **Definición de unidad.** Nombre para diferenciar secciones en un modelo, permitiendo establecer unidades predeterminadas y combinaciones de unidades a las que se les darán nombres abreviados..
6. **Regla.** Expresión matemática que se suma a un modelo de ecuaciones construidas a partir de un conjunto de reacciones. Se emplean para especificar valores de parámetros, establecer restricciones entre cantidades, etcétera.

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <sbml xmlns="http://www.sbml.org/sbml/level1"
3   level="1" version="2">
4   <model name="gene_network_model">
5     <listOfUnitDefinitions>
6       ...
7     </listOfUnitDefinitions>
8     <listOfCompartments>
9       ...
10    </listOfCompartments>
11    <listOfSpecies>
12      ...
13    </listOfSpecies>
14    <listOfParameters>
15      ...
16    </listOfParameters>
17    <listOfRules>
18      ...
19    </listOfRules>
20    <listOfReactions>
21      ...
22    </listOfReactions>
23  </model>
24 </sbml>
```

Figura 2.4. Ejemplo de una definición de modelos expresados en SBML [15].

En la Figura 2.4 se muestra un ejemplo de modelado SBML y su descripción. En la misma se puede observar las características comunes de un flujo de datos en XML. Este es un texto sin formato, donde cada elemento consta de un par de etiquetas de *inicio/final*, las mismas que se adjuntan por los caracteres '<' y '>'. También, algunos elementos pueden contener atributos en la forma: *atributo = 'valor'* [15, 16].

En la línea 1 de la Figura 2.4 se tiene una secuencia particular de caracteres (que empiezan con '<?xml') para declarar el resto del flujo de datos conforme al estándar de codificación XML. En la línea 2, el elemento *sbml*, encapsula una definición de modelo SBML. El primer atributo, *xmlns*, se requiere para que las herramientas puedan leer archivos en formato XML, esto con la finalidad de verificar la sintaxis de una definición particular en el esquema XML para SBML [15, 16].

El atributo de nivel en el elemento *sbml* identifica el nivel en uso de SBML. En *sbml*, debe haber exactamente un subelemento modelo, que a su vez puede tener un solo atributo opcional cuyo valor especifica el nombre del modelo (como se precisa en la línea 4). Un elemento del modelo puede contener varios subelementos; cada uno de estos actúa como un contenedor para un tipo diferente de componente en la definición de un modelo [15, 16].

En los siguientes párrafos se ejemplifican los aspectos antes referidos de SBML [16]. Se modelará un proceso celular eucariota con dos compartimentos para un circuito oscilatorio hipotético de un solo sentido [15, 16].

El modelo se muestra esquemáticamente en la Figura 2.5 y las ecuaciones de reacción para el modelo se encuentran en la Tabla 2.1, en este modelo altamente simplificado, el núcleo de la célula se representa como un compartimento y el citoplasma celular circundante como otro compartimento. Se supone la existencia de un gen *G* que codifica su propio represor y se activa por transcripción a una velocidad constante V_p , por un factor de transcripción ubicuo *U*. La activación transcripcional implica varias reacciones enzimáticas para la producción de ARNPól (Ácido Ribonucleico Polimerasa), su activación (a partir de material fuente *src*) y su desactivación (reciclado) [15, 16].

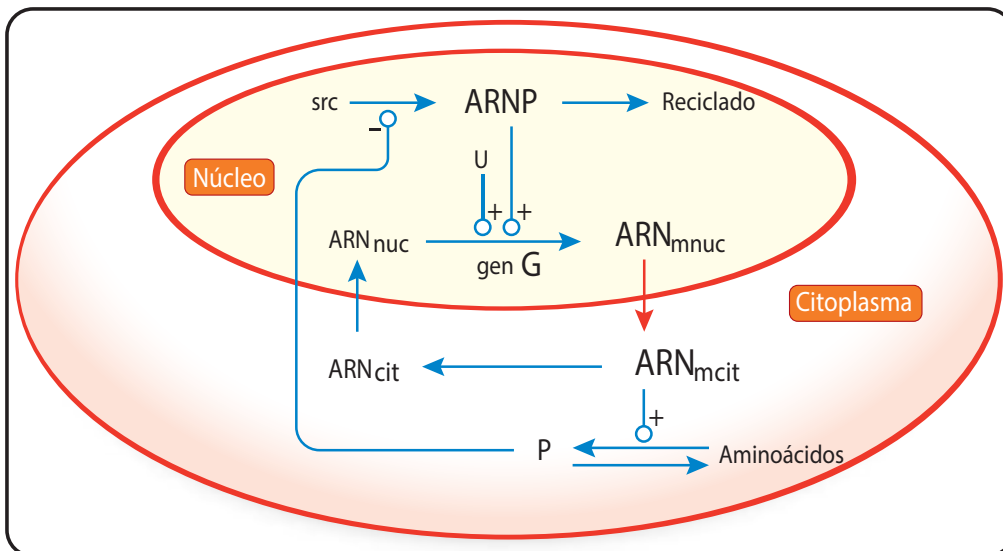


Figura 2.5. Diagrama esquemático del modelo de ejemplo de SBML [15].

El ARNm (Ácido Ribonucleico Mensajero) se transporta fuera del núcleo y hacia el citoplasma, donde se traduce al producto *P* del gen *G*, de los aminoácidos constituyentes y donde también está sujeto a degradación. El producto *P* viaja desde el citoplasma al núcleo para reprimir la transcripción adicional del gen *G*, pero está sujeto también a degradación. Eventualmente, la concentración de *P* se vuelve tan baja que *U* puede reactivar *G* y el ciclo se repite [15, 16].

Reacción	Ecuación
$src \rightarrow ARNP$	$V_i / (1 + P/K)$
$ARNP \rightarrow \text{reciclado}$	$V_{kd} \cdot ARNP$
$ARN_{nuc} \rightarrow ARN_{mnuc}$	$\frac{V_{m1} \cdot ARNP \cdot ARN_{nuc}}{K_{m1} + ARN_{nuc}}$
$ARN_{mnuc} \rightarrow ARN_{mcit}$	$K_{m1} \cdot ARN_{nuc}$
$ARN_{mcit} \rightarrow ARN_{cit}$	$\frac{V_{m2} \cdot ARN_{mcit} \cdot AA}{ARN_{cit} + K_{m2}}$
$ARN_{cit} \rightarrow ARN_{nuc}$	$K_2 + ARN_{cit}$
$AA \rightarrow P$	$\frac{V_{m3} \cdot ARN_{mcit} \cdot AA}{AA + K_{m3}}$
$P \rightarrow AA$	$(V_{m4} \cdot P) / (P + K_{m4})$

Tabla 2.1. Reacciones en el modelo del ejemplo de utilización de SBML en cuanto a los procesos: ARNmnuc (Ácido Ribonucleico mensajero en el núcleo). ARNmcit (Ácido Ribonucleico mensajero en el citoplasma). ARNcit (Ácido Ribonucleico en el citoplasma). ARNnuc (Ácido Ribonucleico en el núcleo). K y V son parámetros de valores dados [15].

Otro lenguaje XML para CM es SBOL (Synthetic Biology Open Language), este software se utiliza también para la simulación de escenarios de la biología sintética [17], y permite representar procesos moleculares con detalles profundos en la especificación e intercambio de información de diseños en procesos biológicos sintéticos. Al mismo tiempo, admite explícitamente especificaciones de diseños a través de una definición rigurosa de reglas y mejores prácticas en cuanto al uso de modelos de información biológica. Por lo que es adecuado para representar los componentes estructurales de diseño, que incluyen ADN (Ácido Desoxirribonucleico), ARN (Ácido Ribonucleico), proteínas y pequeñas moléculas [17].

CellML (Cellular Modeling Markup Language) también es otro lenguaje de CM basado en XML, es considerado un esquema informático abierto [18]. CellML se utiliza para el análisis e interpretación de modelos matemáticos fundamentados en computadora, y la correspondiente compartición de los modelos mencionados entre científicos, incluso cuando estos usen diferentes herramientas de software de construcción de modelos. CellML, es similar a SBML con la ventaja que proporciona mayor alcance para la modularidad y reutilización de modelos sin especificar las descripciones bioquímicas. CellML incluye lenguajes matemáticos y metadatos como son MathML (Mathematical Markup Language) y RDF (Resource Description Framework) [19].

NeuroML (Neuroscience Markup Language) es otro lenguaje fundamentado en XML, que se caracteriza por un formato efectivo para especificar e intercambiar modelos de neuronas y redes neuronales [20-23]. La estructura y el comportamiento de las descripciones de modelos de canales iónicos, sinapsis, células y redes neuronales se fundamentan en algunos modelos subyacentes escritos en LEMS (Low Entropy Model Specification), que es un lenguaje independiente y su uso radica en el modelamiento matemático jerárquico de entidades físicas [24]. Si bien los lenguajes de programación para CM que se sustentan en XML no son de fácil empleo, han viabilizado un mejor y mayor intercambio de elementos de modelado entre herramientas de software en la neurociencia computacional. Por lo que NeuroML, cuenta con dos interfaces de programación de aplicaciones API (Application Programming Interface), escritas en Python, que se utilizan para modificar modelos especificados en XML.

2.3

SIMULADORES DE USO COMÚN EN COMUNICACIONES MOLECULARES

El empleo de simuladores en diferentes áreas resulta fundamental en investigaciones y los correspondientes avances científicos. Las simulaciones conllevan a una emulación de una operación de un proceso del mundo real. En CM muchas plataformas de simulación han sido diseñadas, para el proceso de propagación y detección por ejemplo de una especie, además del análisis de características propias de las CM. Existen muchas limitaciones al respecto, una de ellas es la nano y microescala sobre la macroescala por la escasez de análisis experimental. Por esta razón, las simulaciones se han convertido en un factor importante en el análisis y estudio de numerosos parámetros en CM [25].

En los párrafos posteriores se discute y describe brevemente algunas de las plataformas de software para la simulación de CM (Tabla 2.2).

Tabla 2.2. Comparación de diversos simuladores para CM.

Simulador	BiNS2	N3Sim	CalComSim	COMSOL Multiphysics	Network Simulator - 3	BNSim	NCSim	dMCS	MUCIN	AcCoRD
Licencia	Abierta	Abierta	Abierta	Comercial	Abierta	Abierta	Abierta	Comercial	Abierta/ Comercial	Abierta
Propósito	Comunicación molecular basada en difusión	Comunicación molecular basada en difusión	Comunicación molecular dentro de tejidos	Plataforma diseñada para la simulación de múltiples procesos	Diseñado como simulador de nanoredes	Simulador de múltiples subprocesos para redes bacterianas	Comunicación molecular en bacterias	Transmisión molecular por difusión	Comunicación molecular basada en difusión	Comunicación molecular en bacterias
Interfaz de Usuario	Línea de comandos con visualización de propagación de partículas	Línea de comandos	Línea de comandos	Interfaz gráfica	Línea de comandos y herramientas de animación	Línea de comandos	Línea de comandos	Línea de comandos	Línea de comandos	Línea de comandos
Implementación	Java	Java	Python	No aplica	C++/Python	Java	C++ multi-proceso + Python	Java	MATLAB	C++
Despliegue	Tanto una máquina como un grupo	Sólo una nanomáquina	Sólo una nanomáquina	Tanto una máquina como un grupo	Sólo una nanomáquina	Sólo una nanomáquina	Punto a punto por defecto	Tanto una máquina como un grupo	Sólo una nanomáquina	Tanto una máquina como un grupo
Dimensión	3D	2D, 3D	1D - 3D	1D - 3D	3D	3D	3D	3D	1D- 3D	1D- 3D
Recepción	Receptores	Muestreo	Receptores	Receptores	Berg, Gillespie	Receptores	Muestreo	Absorción	Absorción	Receptores

2.3.1 BiNS (Biological and Nano-Scale Communication Simulator)

Es una herramienta de simulación para redes biológicas. BiNS, es un simulador de subprocesos múltiples para sistemas de CM, fue desarrollado por investigadores de la Universidad de Perugia. Su diseño personalizable (Figura 2.6), proporciona un conjunto de herramientas para la generación de objetos y su modelado para especificar el comportamiento de entidades biológicas, adicionalmente, incluye el manejo de colisiones y procesos de difusión con propagación de flujos de advección tanto en un entorno limitado como en entornos de espacios abiertos. BiNS es adaptable a cualquier tipo de partícula de información [26].

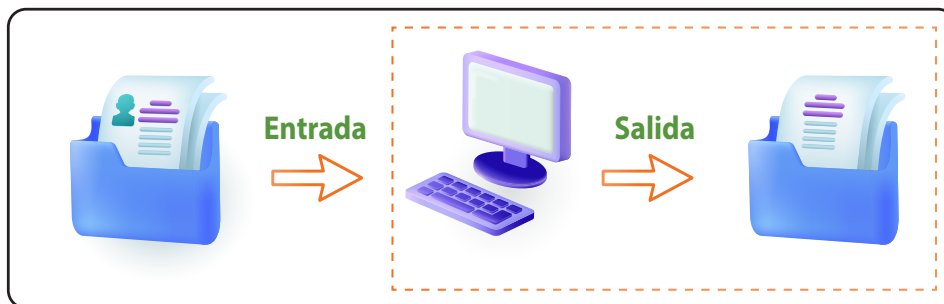


Figura 2.6. Diagrama del modelo de BiNS2 [26].

Los desarrolladores de BiNS, proporcionan una descripción de sus bibliotecas de simulación, en donde demuestran las capacidades del simulador. Como por ejemplo, el modelamiento de una sección de un ganglio linfático y la transferencia de información dentro de él. Entre las principales características que BiNS se tienen, un espacio tridimensional (3D) de operación, el dinamismo en la recepción de información molecular y su correspondiente afinidad en el extremo destino, diferentes portadoras de información, vida útil de las moléculas, seguimiento de moléculas individuales, colisiones y nodos receptores móviles [26].

BiNS se actualizó desde su primera versión que fue desarrollada en el año 2014, hasta convertirse en BiNS2 en 2020, ofreciendo nuevas funciones para simular entornos biológicos. Al igual que BiNS su nueva versión, BiNS2 está implementado en Java y consta de un tipo genérico de objetos de software, denominados *Nano Objects*, y un paquete multiproceso de simulación para CM [26].

BiNS2 se ha diseñado con el propósito de producir modularidad y personalización en el modelamiento del comportamiento de nodos de comunicación (transmisores, receptores o ambos), y portadoras de información que son entidades biológicas que se emiten por transmi-

sores y constituyen la señal de información en CM. Además, es posible configurar las propiedades del canal de comunicación simulado, por ejemplo, usando canales para la propagación en vasos sanguíneos y experimentos *in vitro* con la precisión deseada [26].

Los nodos y las portadoras de información se constituyen implementaciones específicas de objetos de programación de tipo *Nano Object* y, aunque comparten sus características generales, pueden exponer funciones muy diferentes. En particular, para cualquier escenario, es posible diferenciar el tipo de objeto que aparece en una simulación. Las mismas propiedades están disponibles para modelar portadoras como *Nano Objects* [26].

El simulador BiNS2 particulariza las propiedades del diseño molecular a través de un conjunto de herramientas para la creación de objetos, su modelado y gestión de colisiones para partículas mediante los algoritmos correspondientes, así como la propagación de estas partículas en entornos de difusión en espacios abiertos o restringidos [26].

Existen investigaciones en donde se han realizado simulaciones que se han centrado en la comunicación entre plaquetas y el endotelio a través de la difusión de nanopartículas, verificando resultados obtenidos de las simulaciones con datos experimentales [27].

2.3.2 N3Sim (Simulation framework for diffusion-based molecular communication nanonetworks)

N3Sim consta de una estructura (framework) de simulación fundamentada en Java que se utiliza solo para CM en difusión. Esta plataforma establece el análisis de redes moleculares que poseen varios transmisores y receptores [28]. El proceso de difusión a través del medio se modela como movimiento *Browniano*, que toma en consideración la inercia de las partículas y las colisiones entre ellas a medida que viajan a través del medio. En N3Sim, la simulación se basa en una arquitectura de tres capas que son: Capa de interfaz de usuario, Capa de datos y Capa de dominio [29, 30].

N3Sim está diseñado para simular un conjunto de nanomáquinas que se comunican entre sí a través de difusión molecular en fluidos [31]. La información que emiten las nanomáquinas transmisoras modula la velocidad a la que se liberan las moléculas que se modelan como partículas y se depositan en el medio de transmisión. Por ejemplo, el envío de un bit-1 se puede representar por la emisión de un conjunto de partículas, y la transmisión de un bit-0 se codificaría por la no emisión de tales partículas. Este tipo de comunicación crea variaciones en la concentración local de las partículas, que se propagan por el canal de

transmisión con movimiento Browniano y por las interacciones entre ellas. Los receptores pueden estimar la concentración de partículas en su cercanía (vecindad) en función del conteo de partículas en un volumen alrededor de su ubicación. A partir de esta medida, se puede decodificar la información transmitida [29-32].

En la Figura 2.7, se observa un diagrama de bloques de N3Sim, en donde se visualiza los pasos necesarios para ejecutar una simulación [29-32]:

1. El usuario especifica los valores de los parámetros de simulación en un archivo de configuración. Estos parámetros incluyen el número y posición de los transmisores y los receptores, la señal de información a emitir, el tamaño de las partículas transmitidas y el coeficiente de difusión del medio de propagación, entre otros parámetros. Un archivo de secuencia de comandos permite al usuario que pueda ejecutar múltiples simulaciones automáticamente usando solo un archivo de configuración, lo cual es útil para evaluar fácilmente la influencia de un parámetro específico (como es el caso de la cantidad de partículas que se han emitido) en la salida del sistema.
2. El simulador de difusión toma el archivo de configuración y los scripts de automatización como datos de entrada y realiza la simulación real del escenario de propagación. El simulador de difusión también calcula la posición y la velocidad de cada partícula en cada paso de tiempo de la simulación. Cuando la simulación finaliza, sus resultados se almacenan en archivos de receptor (uno por cada receptor), que contienen cada nivel de concentración cuantificada para cada receptor en función del tiempo.
3. Se puede disponer de otro conjunto de scripts para organizar los resultados de varios receptores y representarlos en un solo gráfico.

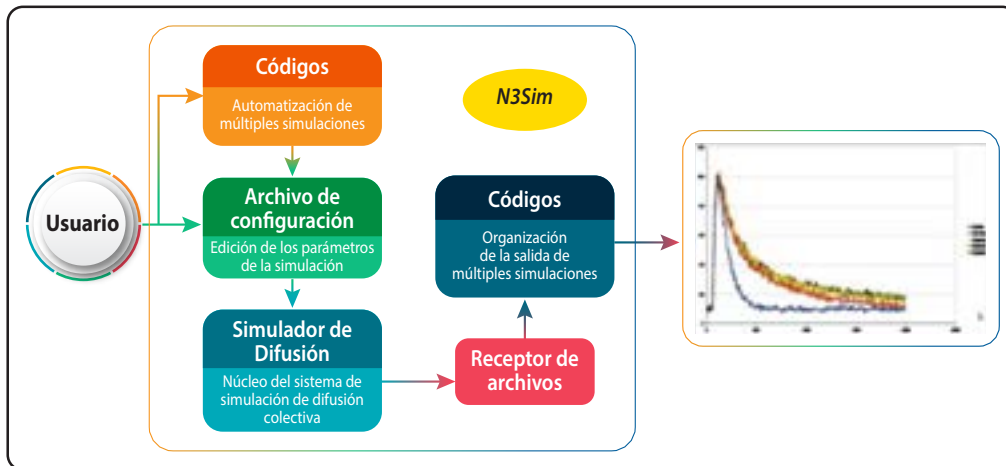


Figura 2.7. Diagrama de bloque de N3Sim [33].

N3Sim actualmente implementa tres tipos básicos de nodos: transmisores, receptores y recolectores, como se indica a continuación [29-32].

1. El transmisor se particulariza por su ubicación en el espacio de simulación y su tamaño, lo que determina su espacio de influencia (es decir, la región donde puede liberar partículas). Cada transmisor modula la información que se enviará en una forma de onda asociada, misma que define su patrón de liberación de partículas. Esta forma de onda se puede elegir entre varias formas predefinidas, entre las que se encuentran pulsos cuadráticos, pulsos Gaussianos o trenes de pulsos. Adicionalmente, el usuario puede definir una forma de onda personalizada.
2. El receptor se puede modelar como una esfera o un cubo capaz de cuantificar el número instantáneo de partículas dentro de su rango de detección, a partir del cual se puede estimar el nivel de concentración local. Este modelo es una idealización del mecanismo de unión ligando-receptor que se encuentra en la naturaleza. Varios parámetros del receptor se pueden ajustar de manera que, después de que el receptor haya detectado un grupo de partículas, este puede absorberlas o actuar como un receptor completamente transparente en relación a las mismas [32].

N3Sim también permite al usuario colocar múltiples transmisores y receptores en el entorno de simulación, habilitando una comunicación multipunto a multipunto. El usuario solo necesita definir las características de cada transmisor y receptor, es decir, su posición, forma y tamaño, que pueden ser dife-

rentes para cada uno de ellos. Esto posibilita el estudio de varios aspectos, como la viabilidad de las CM en difusión, o la naturaleza de la interferencia cuando dos o más transmisores están emitiendo información simultáneamente [32].

3. Finalmente, el tercer tipo de nodos en N3Sim se refiere a nodos de tipo recolector. Los recolectores tienen la capacidad de capturar moléculas de su entorno. Estas moléculas se almacenan posteriormente en un repositorio de capacidad fija, y se pueden emplear en futuras comunicaciones [32].

En N3Sim se implementan tres métodos principales de recolección de información [29-32]:

1. La recolección estándar es el método básico, en el que un nodo captura moléculas del entorno a la velocidad más alta posible hasta que el repositorio de moléculas se llene. El repositorio se libera tras una emisión de moléculas, y puede reanudar la operación de recolección.
2. Para la recolección de velocidad limitada se requiere especificar una frecuencia de recolección por la cual un nodo puede recolectar partículas dentro de su área de recepción. Por ejemplo, si un recolector está limitado a una velocidad del 80%, solo el 80% de las moléculas presentes en el entorno del recolector se capturan en cada momento. Esto permite controlar la velocidad a la que se “absorberán” las moléculas del entorno, debido a la operación de recolección.
3. Un nodo que utiliza el método de recolección ilimitada posee un depósito ilimitado y recolecta cada partícula dentro de su área de recepción. Por lo tanto, la recolección ilimitada se puede considerar como un caso especial de la tasa de recolección limitada, con una velocidad de recolección del 100%.

N3Sim se puede configurar para simular escenarios en un espacio bidimensional (2D) o tridimensional (3D). Uno de los parámetros del espacio de simulación es la concentración inicial de partículas en el medio circundante. Si es cero, se pueden simular tanto un espacio

acotado como uno no acotado. Sin embargo, si la concentración inicial de partículas es mayor que cero, el espacio de simulación debe estar acotado para evitar tener un número infinito de partículas [29-32].

Cuando se trabaja con un espacio acotado, se considera un espacio de simulación cúbico donde las partículas rebotan en los límites de dicho espacio. Se considera que este modelo es el más realista posible para aplicaciones de CM. Además, existe la posibilidad de simular un conjunto de nanomáquinas que se encuentran en comunicación al interior de un vaso sanguíneo y cada vez que una partícula impacta con la pared de este, entonces rebota. N3Sim incluye un mecanismo opcional que permite que varias partículas desaparezcan cuando alcanzan los límites del espacio de simulación, según las leyes de difusión [29-32].

Adicionalmente, se pueden colocar objetos en el escenario para simular obstáculos entre los extremos de transmisión. Por citar un caso, esto permite recrear un escenario en el que un grupo de bacterias atraviesa el medio circundante y obstruye el paso entre emisores y receptores [33].

Otro escenario en N3Sim consiste en el transporte molecular en un medio de difusión permitiendo el paso de nanoredes. Aquí, los transmisores codifican la información liberando moléculas en el medio circundante, modificando el estado de su concentración local. N3Sim modela el movimiento de estas moléculas según la dinámica Browniana, y también tiene en cuenta su inercia y las interacciones entre ellas. Los recolectores capturan moléculas del entorno circundante para reutilizarlas en comunicaciones posteriores. Los receptores decodifican la información detectando la concentración de partículas en sus alrededores [29-32].

Las ventajas de N3Sim son múltiples, entre estas están: la validación de modelos de canales de comunicación y la evaluación de nuevos esquemas de modulación [33]. Varios autores han desarrollado modelos analíticos de canales de transmisión con difusión [31, 34-36].

Estudios recientes consideran a las moléculas como recursos discretos, en transmisiones múltiples mediante recolección molecular [37]. Sin embargo, tales investigaciones precisan de una validación de sus modelos, a través de estudios experimentales. A pesar de los avances recientes en la biología sintética, una configuración experimental de CM resulta difícil de construir; en consecuencia, la simulación actualmente parece ser la opción más factible. Este simulador permitiría recrear un entorno de difusión y medir métricas relevantes para evaluar su rendimiento, la atenuación del canal de propagación y el retardo. Los resultados del simulador podrían luego compararse con aquellos de los modelos analíticos para evaluar su validez y crear nuevos modelos y más precisos [29-33].

N3Sim es el primer marco de simulación de código abierto para entornos de simulación con difusión que incluye casos de difusión normal y anómalos. También, en N3Sim se puede identificar propiedades importantes de CM en difusión [38], la evaluación de técnicas de modulación y la validación de un modelo de canal de transmisión con difusión [39]. Seguidamente se exhiben casos de cómo aparecen resultados de una simulación en varios escenarios (Figura 2.8 y Figura 2.9) [38, 39].

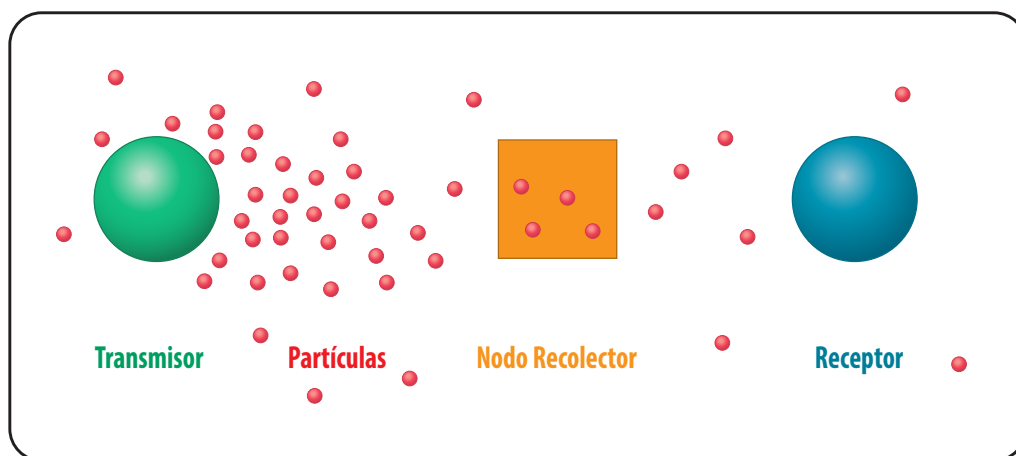


Figura 2.8. Configuración de CM en N3Sim que se compone de un transmisor, un receptor y un nodo de recolección [33].

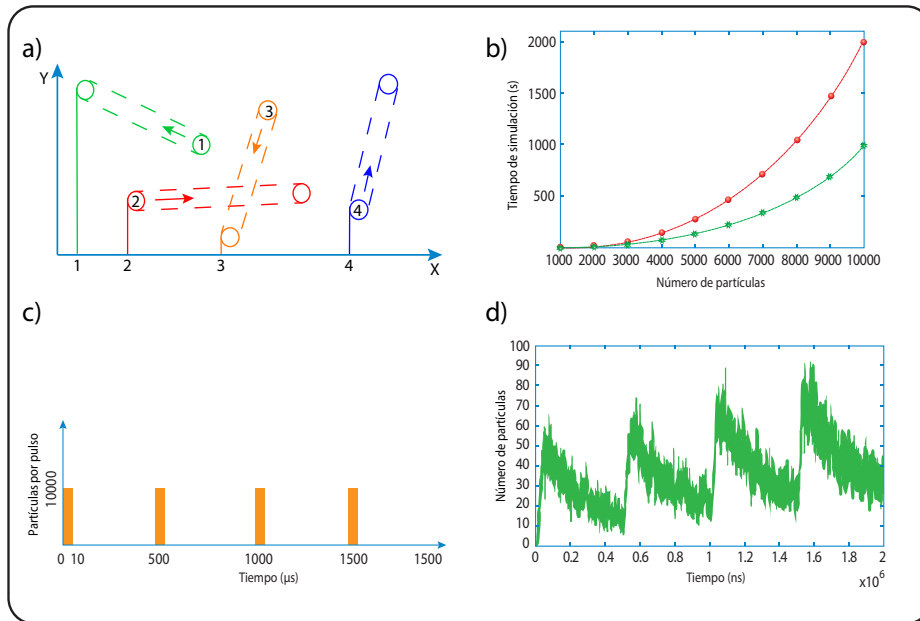


Figura 2.9. a) Simulación en N3Sim de CM para la detección de colisiones; b) Número de partículas transmitidas en un determinado tiempo; c) Pruebas de pulsos de forma cuadrada liberados por el nodo transmisor; d) Medición de la señal de información en el nodo receptor [33].

2.3.3 CalComSim (Calcium Signaling Molecular Communication Sim)

Es un simulador de sistemas de CM que analiza la señalización de calcio, y está diseñado para transmisiones en entornos 3D, tanto sintéticas como naturales que se encuentran dentro de tejidos humanos [40, 41]. Para estas simulaciones, se destina una integración de modelos biológicos en procesos de señalización gobernados por solucionadores estocásticos para el manejo de reacciones múltiples y paralelas en cada célula del tejido. Además, modela tres tipos diferentes de tejidos [40, 41]:

1. No excitables (tejidos que no pueden propagar corriente eléctrica).
2. Excitables (tejidos en los que puede circular la corriente eléctrica).
3. Híbridos (tejidos capaces de comunicarse con ambos tipos de tejidos, excitables y no excitables).

Entre la variedad de células a simular se tienen: epiteliales (tejidos no excitables), de músculo liso (excitables) y astrocitos (híbridos). Los modelos biológicos que se incorporan al simulador se centran en datos experimentales reales, que también incluyen el cierre estocástico y el comportamiento de apertura de uniones Gap para cada tipo de célula (Figura 2.10) [40, 41].

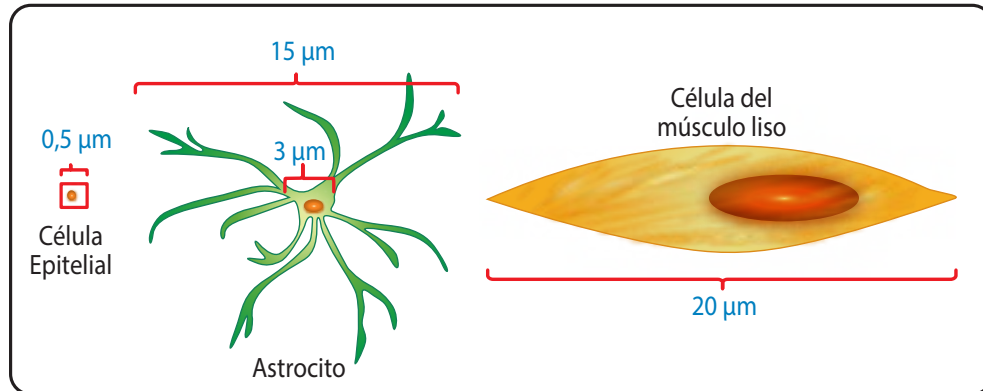


Figura 2.10. Comparación del tamaño entre células epiteliales, músculos lisos y astrocitos [41].

CalComSim, puede emplearse, por ingenieros de telecomunicaciones, biólogos, investigadores farmacéuticos, y en el diseño de nuevos medicamentos y tratamientos para enfermedades que surgen por la deficiencia de calcio [40, 41].

CalComSim ha sido implementado en Python y requiere las siguientes librerías: *Numpy*, *Scypy*, *Pypy*, y *Cpython*. Estas se usan para acceder a funciones matemáticas y científicas. *Pypy* es un intérprete alternativo para CalComSim. *Cpython*, se emplea para ejecutar bibliotecas en C que se pueden utilizar en códigos Python [42, 43]. CalComSim también cuenta con un algoritmo estocástico de *Gillespie*, algoritmo utilizado para resolver ecuaciones diferenciales parciales que modelan la señalización de calcio intracelular e intercelular. CalComSim, está controlado por una interfaz de línea de comandos [44, 45].

2.3.4 COMSOL Multiphysics

COMSOL Multiphysics (conocido como FEMLAB antes de 2005), es una plataforma comercial multipropósito diseñada para simular problemas basados en física, como objetos y procesos físicos a través de un flujo de trabajo unificado para aplicaciones eléctricas, mecánicas, de fluidos y químicas [46]. Su mayor uso ocurre en el campo de la electroquímica analítica, especialmente para análisis teóricos no estándar. La facilidad de implementación que ofrece este software puede ayudar al investigador a obtener rápidamente resultados teóricos sujetos a un rango de aproximaciones fácilmente controlados. Un problema de “multi-física” electroquímico, sería el análisis de celdas de combustible, que combina la dinámica de fluidos, el transporte de masa, la transferencia de calor y de carga. En la simulación de multi-física, se puede maximizar el conocimiento físico y el poder predictivo al describir con precisión todos los fenómenos relevantes [46].

COMSOL implementa el análisis de FEA (Finite Element Applied) para diferentes aplicaciones de física e ingeniería, y está disponible para diferentes sistemas operativos. Además, con COMSOL, se puede simular un sistema de administración de fármacos en CM [47].

COMSOL implementa ecuaciones valiéndose de interfaces de física. Estos son paquetes prediseñados de ecuaciones físicas y una selección de condiciones de contorno asociadas. En el análisis electroquímico se utiliza las siguientes interfaces físicas integradas: distribución de corriente primaria/secundaria/terciaria y electroanálisis. Estas interfaces brindan acceso completo a las ecuaciones físicas y permiten configuraciones físicas flexibles y controladas por el usuario para la descripción de sistemas electroquímicos [48].

En todos estos métodos preconstruidos, existe una gran flexibilidad para combinar y acoplar métodos con variables o ecuaciones definidas por el usuario, así como para modificar configuraciones numéricas predeterminadas [48].

Adicionalmente, COMSOL también modela interacciones de partículas biológicas a nivel molecular, que requieren la implementación de modelos específicos. Entre los que se tienen, la formación de ligandos-receptores que no es compatible con bibliotecas específicas y debe ser implementada por los usuarios. Se pueden emplear scripts de MATLAB (Matrix Laboratory) para implementar nuevos módulos [48].

COMSOL además admite la computación en clúster. Por lo tanto, cualquier trabajo de simulación se puede implementar en cualquier número de computadoras agrupadas [48].

Los autores en [49] señalan que los enfoques basados en COMSOL, conducen a excelentes resultados en circunstancias particulares, pero, dependen de la profunda experiencia de sus potenciales usuarios por lo que, pueden conducir fácilmente a predicciones erróneas en manos de novatos. Si bien la última observación es muy acertada, cualquier método numérico puede arrojar conclusiones poco confiables por parte de usuarios con experiencia electroquímica inadecuada para interpretar los resultados numéricos en un contexto apropiado.

Los errores asociados con la solución de ecuaciones de elementos finitos se pueden minimizar con cuidado en la configuración física del problema. Como es normal, con cualquier método numérico, se requiere de un análisis de convergencia para establecer la magnitud de los errores numéricos debido a la discretización. La facilidad de implementación de un modelo empleando interfaces preconstruidas en software comercial no exime al usuario de la responsabilidad de afirmar la razonabilidad física de sus resultados [33].

COMSOL viabiliza el control de ecuaciones físicas y la configuración numérica, al tiempo que proporciona valores automáticos predeterminados correspondientes para la configuración de solucionadores. El propósito de esto es permitir que los electroquímicos expertos realicen trabajos teóricos, sin exigir un conocimiento de gran experticia en el análisis numérico. En este entorno, se espera que la electroquímica se beneficie de la aplicación del software de elementos finitos [48]. En la Figura 2.11 aparece un ejemplo de la interfaz gráfica de COMSOL.

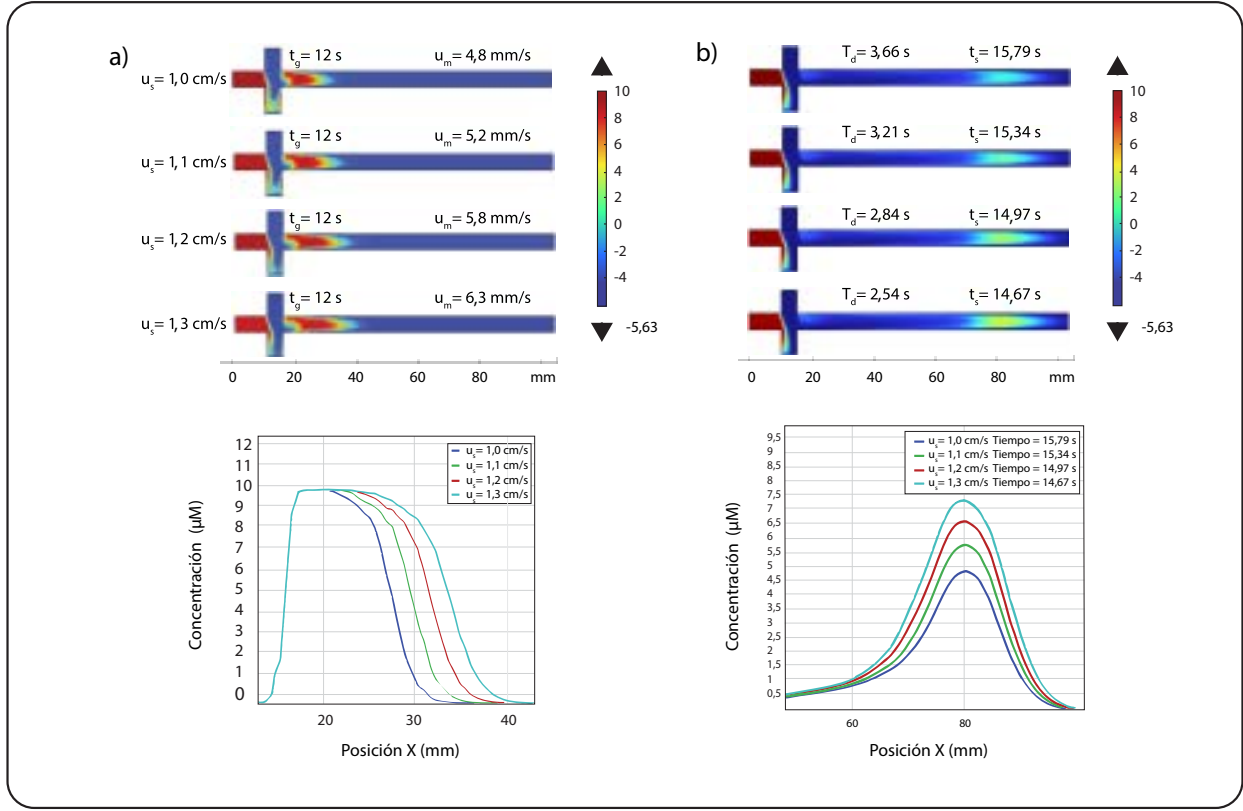


Figura 2.11. Ejemplo del uso de COMSOL Multiphysics en un caso de análisis molecular mediante la variación de velocidad de un fluido en la propagación de la información en el canal de transmisión. a) Pulso generado; b) Pulso propagado [50].

2.3.5 NS-2 (Network Simulator-2) y NS-3 (Network Simulator-3)

NS-2 y NS-3, son simuladores de redes de comunicación de eventos discretos para sistemas de interredes, que inicialmente no fueron desarrollados para CM. No obstante, sus estructuras flexibles han permitido implementar algunos elementos básicos de comunicaciones biológicas. NS-3 es la plataforma de software para simulaciones que establecen los estándares de nanocomunicaciones IEEE 1906.1 y 1906.1.1 [51].

NS-3, es la versión actualizada de NS-2, y no es compatible con su antecesora. El simulador se encuentra organizado en diferentes bibliotecas de software que pueden funcionar juntas. Los programas de usuario, escritos en lenguajes de programación C++ o Python, pueden adaptar estas bibliotecas o vincularse con ellas. Están disponibles animadores externos y herramientas de visualización y análisis de datos. Sin embargo, para explotar todo el potencial de NS-3, es preferible la interfaz de línea de comandos. El simulador está diseñado para ser ejecutado en una sola máquina. Los requisitos mínimos para ejecutar simulaciones básicas son un compilador *gcc* o *clang* y un intérprete de Python, además la granularidad de la simulación es definida por los usuarios [51]. En este libro no se estudia a profundidad NS-3, pues su análisis detallado se ha realizado en nuestros libros:

1. Comunicaciones Moleculares: Un análisis desde el paradigma protocolario de IEEE (<https://repositorio.cedia.edu.ec/handle/123456789/5>).
2. Estándares de Nanocomunicaciones Biológicas: Fundamentos y Simulación (<https://repositorio.cedia.edu.ec/items/2d965364-d67c-40a1-b5ff-3dc03fbc6eba>).

En NS-2 las redes inalámbricas de nanosensores se modelan empleando comunicaciones electromagnéticas en lugar de CM. Para CM el objetivo principal de este simulador consistía en identificar las reglas básicas para la robustez basada en GRN (Gene Regulatory Network) y la capacidad de un estado biológico para persistir a pesar de los errores de los componentes mediante la configuración de una plataforma de simulación NS-2 genérica, en lugar de desarrollar canales de CM más detallados [52, 53].

Se ha utilizado NS-2 como una plataforma basada en silicio para cuantificar la solidez de las redes biológicas. Particularmente, debido a que el propósito principal de una red de sensores inalámbricos es el transporte de información a nodos fuente específicos operando en condiciones de ruido y susceptibles a errores al igual que en las redes biológicas, se podría definir la factibilidad de diseño de este tipo de redes a través de la facultad de cada nodo en la red para entregar in-

formación con pérdidas mínimas. Antes de imaginar un modelo para cualquier sistema biológico funcional variable en el tiempo, es importante ilustrar el modelo preliminar para el sistema biológico en NS-2. Si bien actualmente no existen simuladores exclusivos para modelar una red molecular, los simuladores existentes se pueden ajustar para modelar la red requerida [52, 53].

Otro simulador de CM es NanoNS (Nanoscale Network Simulator), que es un simulador fundamentado en NS-2 para comunicaciones biológicas bajo difusión en entornos líquidos y movimiento térmico continuo de moléculas y nanonodos que se moldean como esferas. NanoNS, está orientado a objetos escritos en C++ con un intérprete OTcl (ObjectTcl). El propósito principal de esta plataforma es contar con una herramienta de simulación para lograr una mejor comprensión de nanoredes y facilitar el desarrollo de nuevas técnicas de comunicación, y también, proporcionar la validación de resultados teóricos. Este software incorpora módulos de simulación para varios paradigmas de comunicación a nanoescala, empleando un canal de transmisión de CM en difusión [52, 53].

Este simulador se sustenta en el movimiento Browniano [29, 30], como consecuencia del movimiento térmico constante de átomos, moléculas y sus reacciones químicas. Cuenta con una biblioteca *OTcl*, denominada *ns-mol.tcl*, para estructuras de nodos que se conocen como *NanoNode*, y se sustentan en una clase independiente de OTcl de *Node*; *NanoNode*, es una clase que implementa funciones específicas de moléculas portadoras de información a ser liberadas y transportadas a través de líquidos. La biblioteca *OTcl*, junto con otros componentes, parámetros y métodos de nanoredes para CM, facilita la programación de la simulación, la configuración de la topología de red y la configuración de los parámetros de la nanored [52, 53].

2.3.6 BNSim (Bacteria Network Simulator)

BNSim es una plataforma de simulación en Java y realiza múltiples subprocesos para analizar redes de bacterias. Estas redes interconectan escenarios de diseño bacteriano que intercambian información a nanoescala. BNSim integra tres métodos de simulación [54]:

1. Algoritmo de simulación estocástica Gillespie SSA (Stochastic Simulation Algorithm) [44]. Con este algoritmo se resuelven ecuaciones CME (Chemical Master Equation), para determinar la evolución temporal de un sistema químico; el tamaño extremadamente pequeño de los eventos hace que este enfoque no sea práctico para problemas a nivel de población.

Mediante CME se describe la ecuación de evolución temporal para la determinación de la densidad de probabilidad de las especies en las reacciones químicas. A pesar de la ventaja que CME proporciona, la evolución temporal de la distribución de probabilidad de todos los estados posibles a ocurrir en los eventos resulta a menudo matemáticamente intratable [5].

2. Ecuaciones diferenciales estocásticas SDE (Stochastic Differential Equations). Son empleadas para modelar sistemas químicos a gran escala con un nivel de aproximación controlado.
3. Algoritmo híbrido. Este código integra tanto el algoritmo SSA como los métodos basados en ecuaciones diferenciales. Los métodos aproximados que se sustentan en ecuaciones diferenciales estocásticas se utilizan para modelar el comportamiento de especies que están presentes en grandes cantidades y generan problemas de escalabilidad.

Existen diferentes simuladores para el análisis de bacterias en la actualidad, en la Figura 2.12 se observa un ejemplo de una comparativa con el simulador BNSim [54].

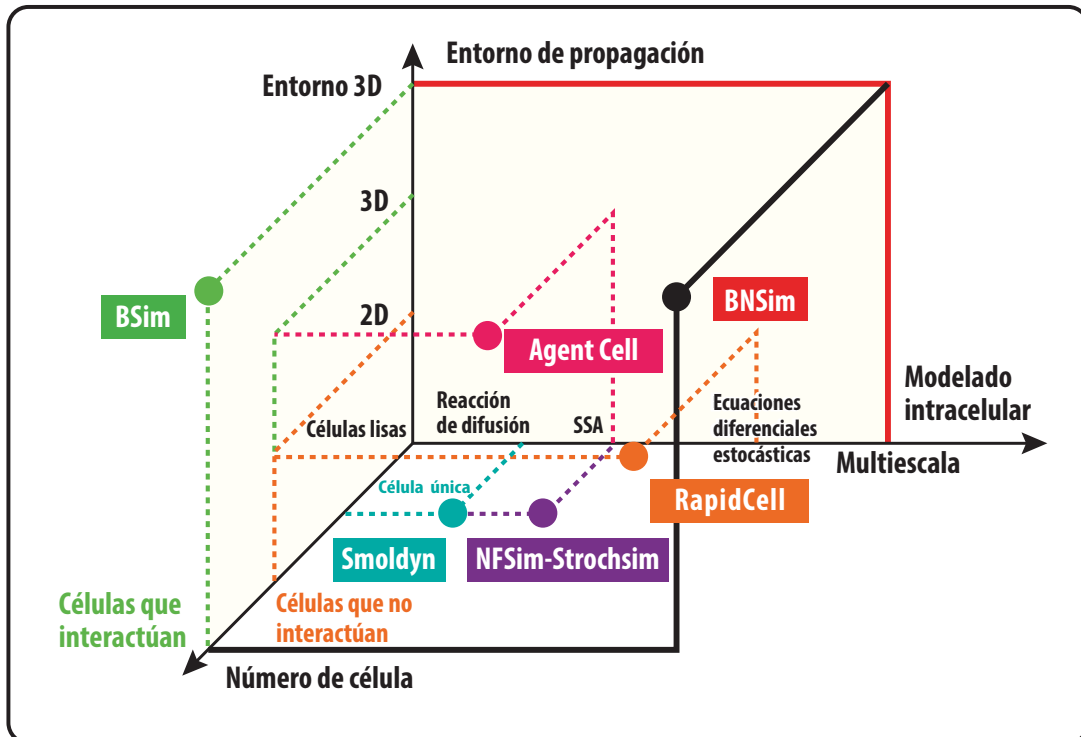


Figura 2.12. Comparación de BNSim con simuladores de estudios bacterianos actuales. BNSim simula los sistemas bioquímicos intracelulares con SSA de Gillespie, SDE, y algoritmos estocásticos multi-escalar. Al mismo tiempo, BNSim admite la simulación de una gran población de bacterias que interactúan en un entorno complejo 3D [54].

2.3.7 NCSim-Bacteria Nanonetworks

NCSim es un simulador para CM que emplea flagelos de bacterias para la transmisión de información (Figura 2.13). El enfoque principal de esta plataforma se centra en diferentes técnicas de codificación de mensajes. Se pueden simular varios enlaces simultáneos entre nanomáquinas, adicionalmente, NCSim admite políticas de implementación típicas, es decir regulaciones para ejecutar ciertos parámetros de la simulación, como políticas de implementación personalizadas. Este simulador incorpora modelos estocásticos para la movilidad bacteriana [55], y la transferencia de bacterias tipo plásmido/cromosoma con otras en procesos de conjugación [56].

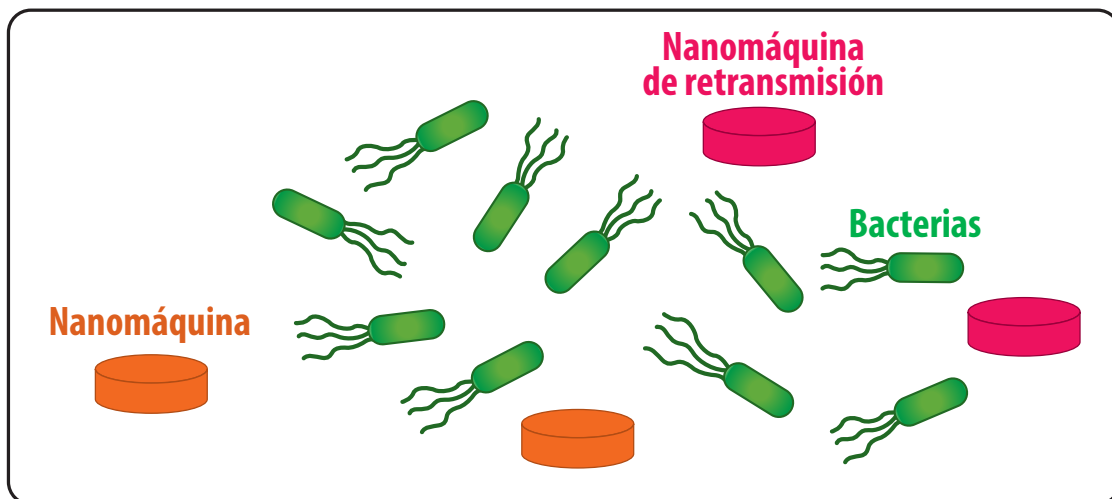


Figura 2.13. Ejemplos de nanoredes que se forman entre bacterias (los nodos naranja representan nodos de origen y destino, mientras que los nodos rosas constituyen nodos de retransmisión) [56].

Al presente, las simulaciones interactúan en un entorno 2D. La precisión de las métricas del simulador se puede evidenciar en la comparación con los resultados analíticos en escenarios simplificados. NCSim consta de tres módulos [56]:

1. PHY (Physical), componente físico de nanoredes bacterianas. Este módulo establece el despliegue, la movilidad y la conjugación de bacterias, además de la codificación/decodificación de mensajes.
2. Generador de escenarios y monitor de la simulación.

3. Generación de plots (imágenes/resultados). Herramienta de trazado destinada al posprocesamiento de información pertinente a la simulación sin procesar y a la subsecuente generación de gráficos correspondientes a la simulación.

El módulo PHY, cuenta con más recursos computacionales implementados en C++. Mientras que los dos últimos módulos están escritos en Python para simplificar el mantenimiento y la extensión de la simulación. El usuario interactúa con NCSim escribiendo pequeños scripts en Python para definir escenarios [54].

2.3.8 HLA (High Level Architecture)

La Oficina de Coordinación de Modelado y Simulación del Departamento de Defensa de Estados Unidos ha desarrollado HLA (High Level Architecture) en el periodo 1995-1996, para facilitar la integración de modelos de simulación distribuidos en una arquitectura común, cuya finalidad era la de cubrir aplicaciones de defensa, sin embargo, se ha usado en muchas áreas industriales y de investigación, para tiempo después ser estandarizado según la norma IEEE 1516-2010 [7]. El objetivo principal de HLA es definir modelos de componentes y sus interacciones. Estos componentes se denominan federados (federates) y permiten la reutilización y abstracción de software. Tales componentes se comunican a través de ITE (Infrastructure Time Execution), empleando protocolos estandarizados según la norma referida, para formar un modelo de simulación denominado federación [15].

Cada federación cuenta con un modelo FOM (Federation Object Model) que se crea de acuerdo con una plantilla OMT (Object Model Template) definidas en la documentación del estándar previamente mencionado [57]. HLA permite el diseño y desarrollo independiente de componentes y también la ejecución distribuida de la simulación. En la Figura 2.14 se indica una arquitectura HLA genérica. RTI (Run-Time Infrastructure) es la columna vertebral de la federación y proporciona servicios de sincronización, comunicación e intercambio de información entre los federados. Cada federado puede consistir en un evento independiente o simulaciones en tiempo real con interacción humana, sistemas o equipos *in vivo* [15].

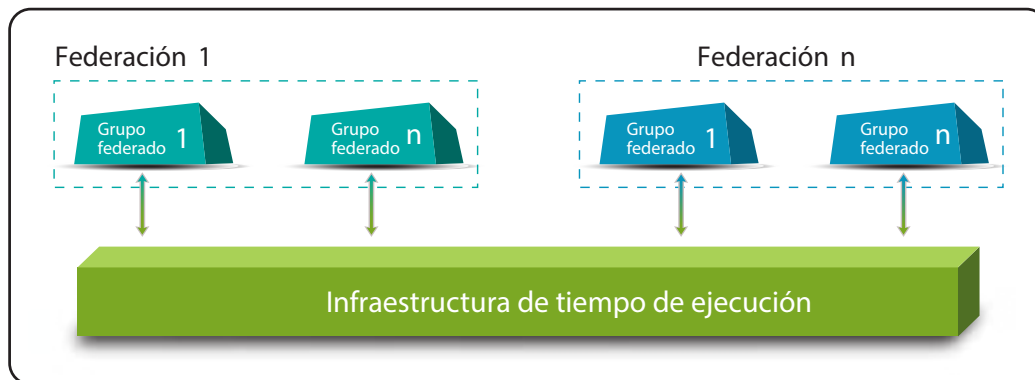


Figura 2.14. Arquitectura de HLA [58].

HLA no restringe el modelamiento en un federado, provee la interacción entre ellos, y ofrece seis clases de servicios RTI [15]:

1. Gestión de federaciones. Funcionalidad esencial requerida para crear y ejecutar una federación.
2. Gestión de declaraciones. Administra el intercambio de datos entre federados, usando la información proporcionada por los federados.
3. Gestión de objetos. Sirve para la creación, eliminación, identificación y otros servicios a nivel de objetos.
4. Gestión de propiedad. Es la transferencia dinámica de objetos/atributos durante la ejecución de la federación.
5. Gestión del tiempo. Sincronización del intercambio de información de la simulación en el tiempo de ejecución.
6. Gestión de distribución de datos. Es la guía de información entre federados durante la ejecución de la federación.

En [58] se presentan los principios de diseño de un simulador de CM que se centra en la escalabilidad, mediante la adopción de modelos de HLA (Figura 2.15).

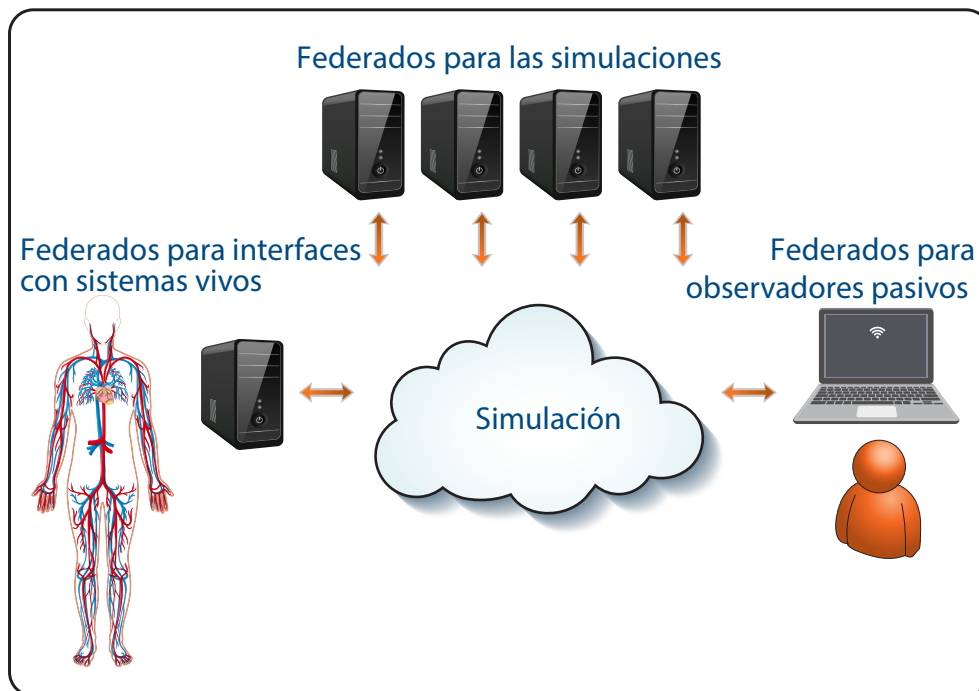


Figura 2.15. Principios de simulación desarrollados en base a HLA [58].

Estos modelos se utilizan para diseñar una herramienta de simulación distribuida, de modo que se puedan emplear diferentes opciones de escalabilidad para agregar potencia de procesamiento y reducir el tiempo de ejecución. Un simulador distribuido fundamentado en HLA hace posibles escenarios de simulación más grandes e interacción con sistemas que contienen organismos vivos [58].

A lo largo de los años, varios esfuerzos de investigación centraron su atención en la creación de soluciones de software para facilitar el desarrollo de simulaciones sustentadas en HLA, principalmente con el fin de contar con una cadena de herramientas integrada para crear y simular sistemas complejos. Para los usuarios de MATLAB/Simulink, se disponen diferentes paquetes de software y herramientas para implementar simulaciones de HLA, como Forwardsim HLA Toolbox para MATLAB [59], que ofrece una interfaz de usuario que sirve para diseñar y personalizar completamente federaciones de HLA. Otra herramienta que otorga a los desarrolladores gestionar de manera efectiva la estructura de *HLA Federate* es PITCH Developer Studio [60]. Este software permite a los programadores agregar funcionalidades HLA o de tipo DIS (Distributed Interactive Simulation) de forma rápida y sencilla a través de un módulo que genera código C++ o Java a partir de la estructura de *HLA Federate*.

HlaListener es otra herramienta que permite crear y probar fácilmente sistemas en *HLA Federate* [61]. Fue desarrollada por la Universidad de Alberta (Canadá) y proporciona una interfaz GUI (Graphical User Interface) simple con todas las funcionalidades comunes de HLA como: conectarse/desconectarse - hacia/desde RTI, enviar/recibir actualizaciones de las clases *InteractionClass* y *ObjectClass*, y la definición de puntos de sincronización. DMI (Danish Maritime Institute) también creó un marco de referencia de software a través de HLA con el propósito de simplificar el desarrollo de simulaciones en tiempo real en el dominio militar, y define una variedad de conceptos de simulación para respaldar la ejecución de funcionalidades disponibles en DMI [62].

FEDEF es otra plataforma específica desarrollado por Defense R&D Canada-Atlantic, que establece un conjunto de API (Application Programming Interface) para admitir los estándares DMSO 1.3 y HLA IEEE 1516-2000. Además, la plataforma otorga diferentes capacidades para simplificar muchas tareas de programación que normalmente se requieren cuando se diseña una federación en el dominio militar [63].

Esta aplicación consta de un equipo de desarrollo de HLA DKF (Development Kit Framework), para diseñar, producir y ejecutar una simulación distribuida completa de HLA. DKF es un framework independiente de propósito general, diseñado y desarrollado por SMASH-La (System Modeling And Simulation Hub-Laboratory) de la Universidad de Calabria (Italia) en cooperación con la División de Software, Robótica y Simulación en el centro espacial JSC (Johnson Space Center) de la NASA (National Aeronautics and Space Administration). Se implementa en Java y ha sido publicada bajo la política de código abierto LGPL (Lesser GNU Public License), que facilita el diseño de *HLA Federates* [64-68]. DKF facilita a los desarrolladores centrarse en los aspectos pertinentes a sus propios federados en lugar de ocuparse de los aspectos comunes de HLA, como la gestión del tiempo de simulación; la conexión/desconexión en/desde HLA RTI; la publicación, suscripción y actualización de los elementos *ObjectClass* e *InteractionClass* [65, 69].

En la Figura 2.16 se tiene un ejemplo de un escenario de simulación con HLA.

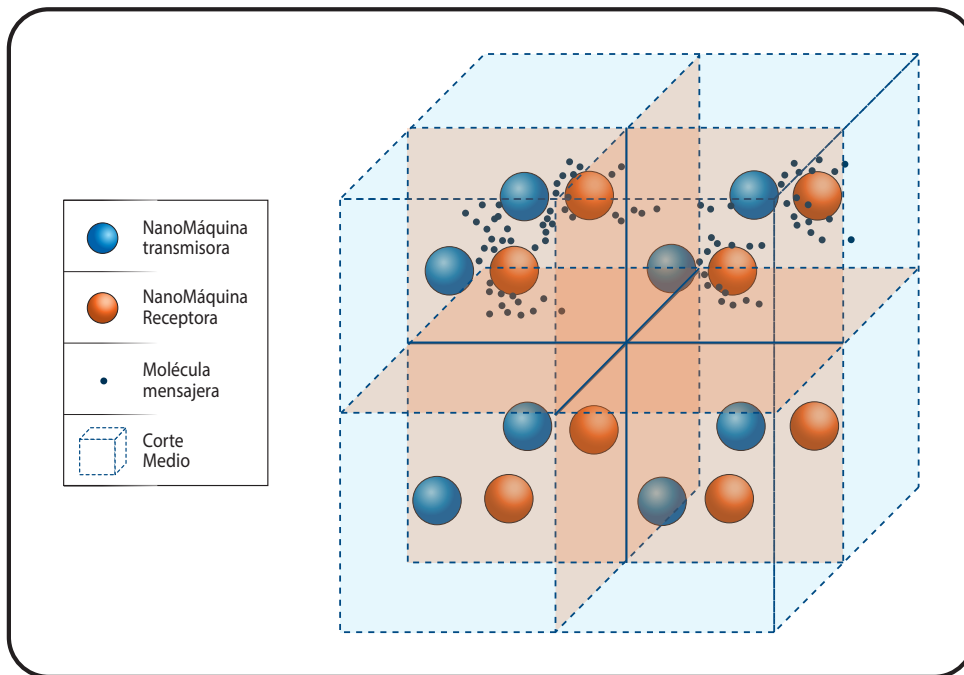


Figura 2.16. Escenario de simulación en HLA [58].

2.3.9 dMCS (Distributed Molecular Communication Simulator)

Su arquitectura se fundamenta en HLA [70]. Con dMCS, es posible explorar diferentes opciones de escalabilidad para acortar significativamente el tiempo de ejecución de la simulación. Esta arquitectura define componentes de simulación interoperables y reutilizables, que pueden ejecutarse en un entorno distribuido. Esto permite el desarrollo de escenarios de simulación complejos y a gran escala [43].

El modelo de simulación de dMCS se puede emplear para simular CM en difusión para redes corporales mediante células modificadas o nanomáquinas creadas artificialmente. El modelo de simulación se puede tornar menos complejo al realizar ciertas suposiciones, que pueden afectar los resultados, o solo se pueden considerar en escenarios de simulación simples. La arquitectura fundamentada en HLA permite la escalabilidad horizontal, lo que hace posibles escenarios de simulación muy grandes [43]. La especificación de la interfaz HLA define la forma en que se accede a los servicios, pero no define cómo realmente los servicios se implementan [71].

En dMCS, las simulaciones se conceptualizan como federaciones de CM. Se definen federaciones separadas para moléculas, nanomáquinas y el medio de transmisión. Por su parte, los federados se comunican entre sí a través de RTI basada en un modelo FOM común. La escalabilidad se obtiene distribuyendo tareas relacionadas mediante los atributos: *Molecule Federates*, *Nanomachine Federates* y el espacio 3D se asigna al atributo *Medium Federate* para la administración de cada componente involucrado en la simulación [70, 71].

En la Figura 2.17, se visualiza las federaciones de dMCS y a continuación se detalla su uso [70]:

1. **Federación de Moléculas (Molecule Federates).** Abstrae las moléculas en la simulación, y es la responsable de los movimientos moleculares en el medio de comunicaciones. Esta se suscribe a los atributos de Medium Federate, lo que significa que en caso de que los atributos de Medium Federate se modifiquen, esta información se comunica con Molecule Federate a través de RTI. Molecule Federate calcula la siguiente posición de las moléculas y actualiza los atributos. En la actualidad se dispone una implementación para el movimiento Browniano [30, 72], pero es posible implementar diferentes leyes de difusión. Ya que las moléculas se propagan en el espacio 3D, este movimiento se modela como tres desplazamientos independientes, uno para cada dimensión en el espacio 3D.
2. **Federación de Nanomáquinas (Nanomachine Federates).** Modela las nanomáquinas en la simulación, es responsable de la lógica de transmisión y recepción de moléculas. Se puede definir diferentes tipos de nanomáquinas transmisoras y receptoras con diversos comportamientos. Las diferentes implementaciones de una nanomáquina transmisora pueden liberar moléculas en grandes cantidades o secuencialmente, desde un solo punto o diferentes puntos. La implementación de una nanomáquina receptora se realiza de tal modo que, puede recibir moléculas en toda su superficie; en otro tipo de implementación se pueden recibir moléculas a través de receptores distribuidos en su superficie solamente. El comportamiento de la nanomáquina dependerá de los mecanismos de codificación y decodificación establecidos por las nanomáquinas en los extremos comunicacionales.
3. **Federación del medio de transmisión (Medium Federate).** Esta federación facilita la construcción de entornos 3D grandes en la simulación. El manejo de colisiones se gestiona en Medium Federate a través de interacciones de reversión de moléculas en pasos de tiempo intermedios.

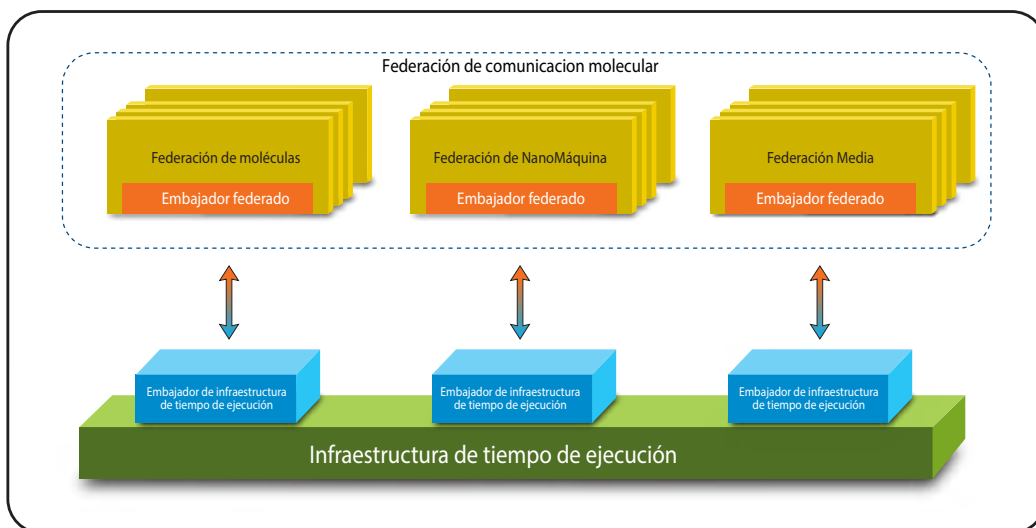


Figura 2.17. Arquitectura de simulación de dMCS [73].

En la Figura 2.18 se visualizan las interfaces entre federados mencionados previamente y su relación [70].

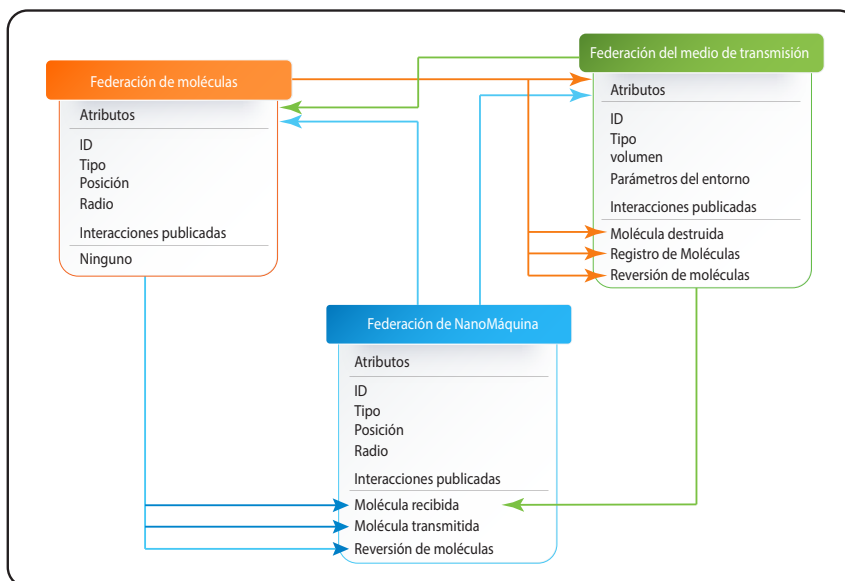


Figura 2.18. Interfaz entre federados en dMCS [58].

2.3.10 MUCIN (MolecUlar CommunicatiON)

MUCIN es un simulador modular de extremo a extremo, desarrollado en MATLAB, es extensible en términos de diferentes esquemas de modulación y es capaz de enviar símbolos consecutivos con intervalos de símbolos arbitrarios [73, 74]. Este simulador genera el número de moléculas recibidas en cada período de muestreo y los símbolos demodulados finales para el análisis de nivel de señal y símbolo. El simulador de nanoredes MUCIN está a disposición bajo licencia BSD (Berkeley Software Distribution) [73].

En la Figura 2.19 se proporciona un ejemplo de un modelo de sistema de CM en difusión en un entorno 3D [73, 74].

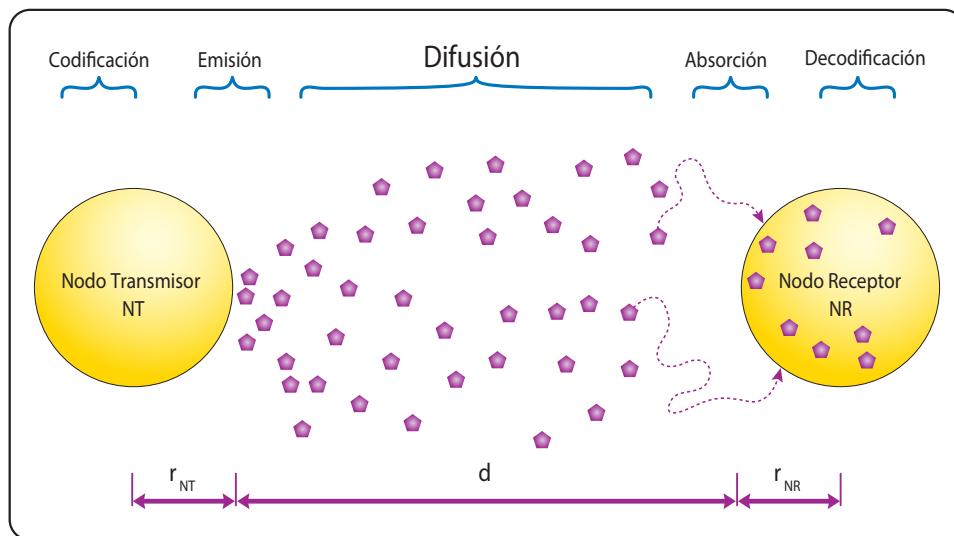


Figura 2.19. Escenario de simulación de un sistema de CM en difusión empleando MUCIN [73].

Las principales características de MUCIN se resumen así [73, 74]:

1. Trabaja en entornos 1D, 2D y 3D.
2. Realiza transferencias de símbolos consecutivos.

3. Cuenta con una probabilidad de reacción en el extremo destino para la recepción de señales vestigiales.
4. Posee fuentes de emisión de moléculas con formas geométricas puntuales y esféricas (reflectoras).
5. Implementa modulaciones fundamentadas en concentración y frecuencia de transmisión molecular.
6. Tiene mecanismos para la mitigación de interferencias tipo ISI (Inter-Symbol Interference) en el receptor comunicacional.

MUCIN admite la recepción molecular vestigial (pérdidas de señal, o distorsiones en la detección y decodificación de moléculas), módulos de filtrado, transmisión de símbolos sucesivos, modulación extendida, etcétera (Figura 2.20) [74].

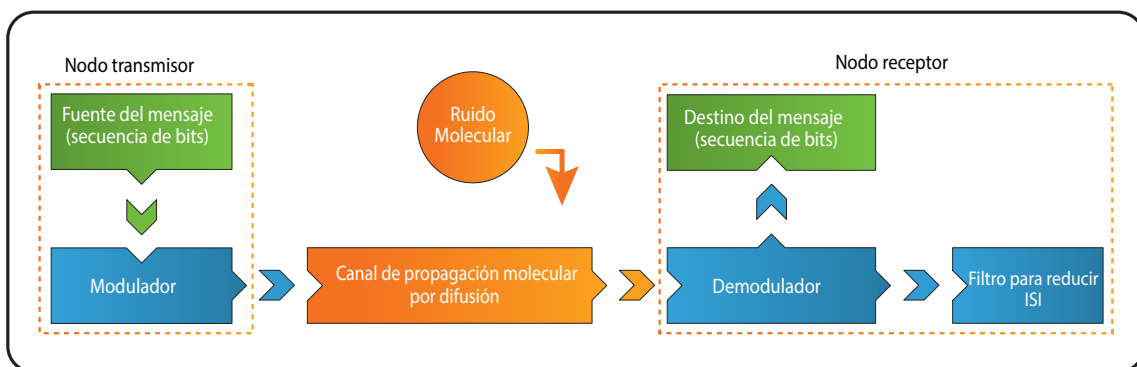


Figura 2.20. Diagrama de bloques de extremo a extremo del simulador MUCIN [73].

2.3.11 AcCoRD (Advanced Communication via Coordination of Rotational Diffusion)

AcCoRD es un proyecto de código abierto de desarrollo activo en Github [75]. En este simulador, los componentes en la comunicación se colocan como fuentes (es decir, transmisores) u observadores (es decir, receptores) de moléculas. Los entornos de propagación se pueden definir con una combinación de regiones microscópicas y mesoscópicas [76-83]. AcCoRD es un simulador que funciona como un solucionador genérico de advección-reacción-difusión estocástica (diseñado en C), para entornos 3D con precisión local flexible para CM. Este simulador genera de manera eficiente estadísticas de las observaciones de moléculas de un receptor (Figura 2.21) [76]. Por otra parte, en la Figura 2.22, se indican algunos resultados de simulaciones realizadas en AcCoRD [75-83].

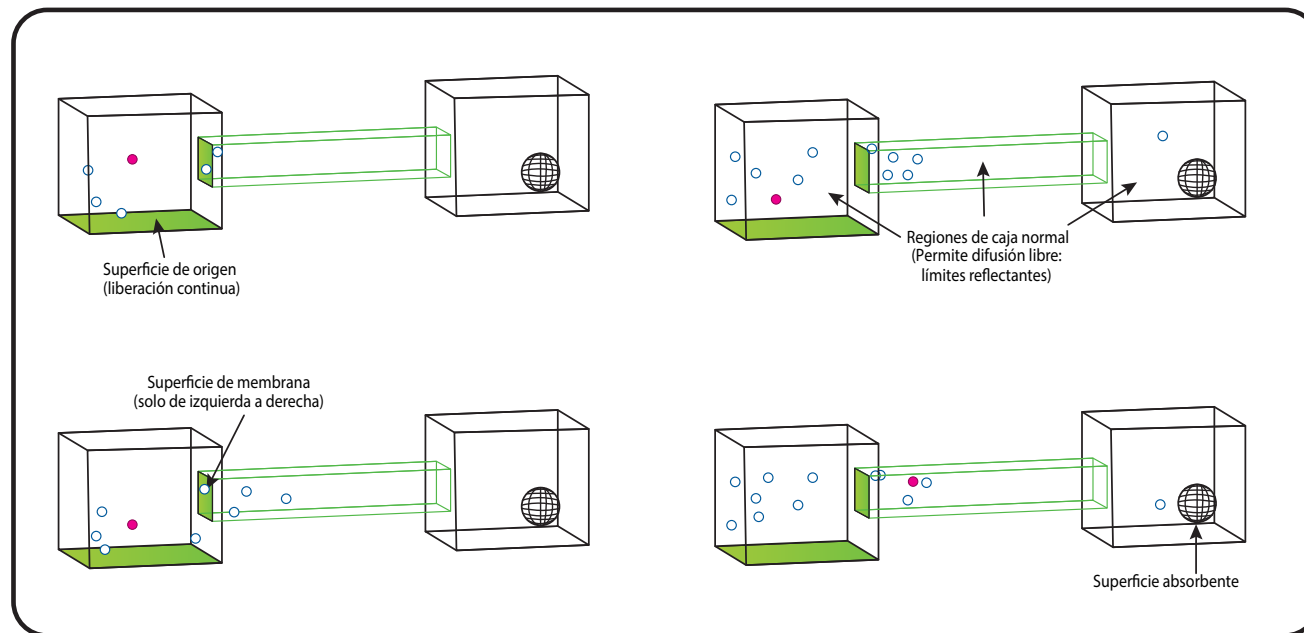


Figura 2.21. Entorno de demostración en una simulación con AcCoRD con múltiples regiones normales y de superficie. Se resalta la trayectoria de una molécula hasta llegar al receptor [76].

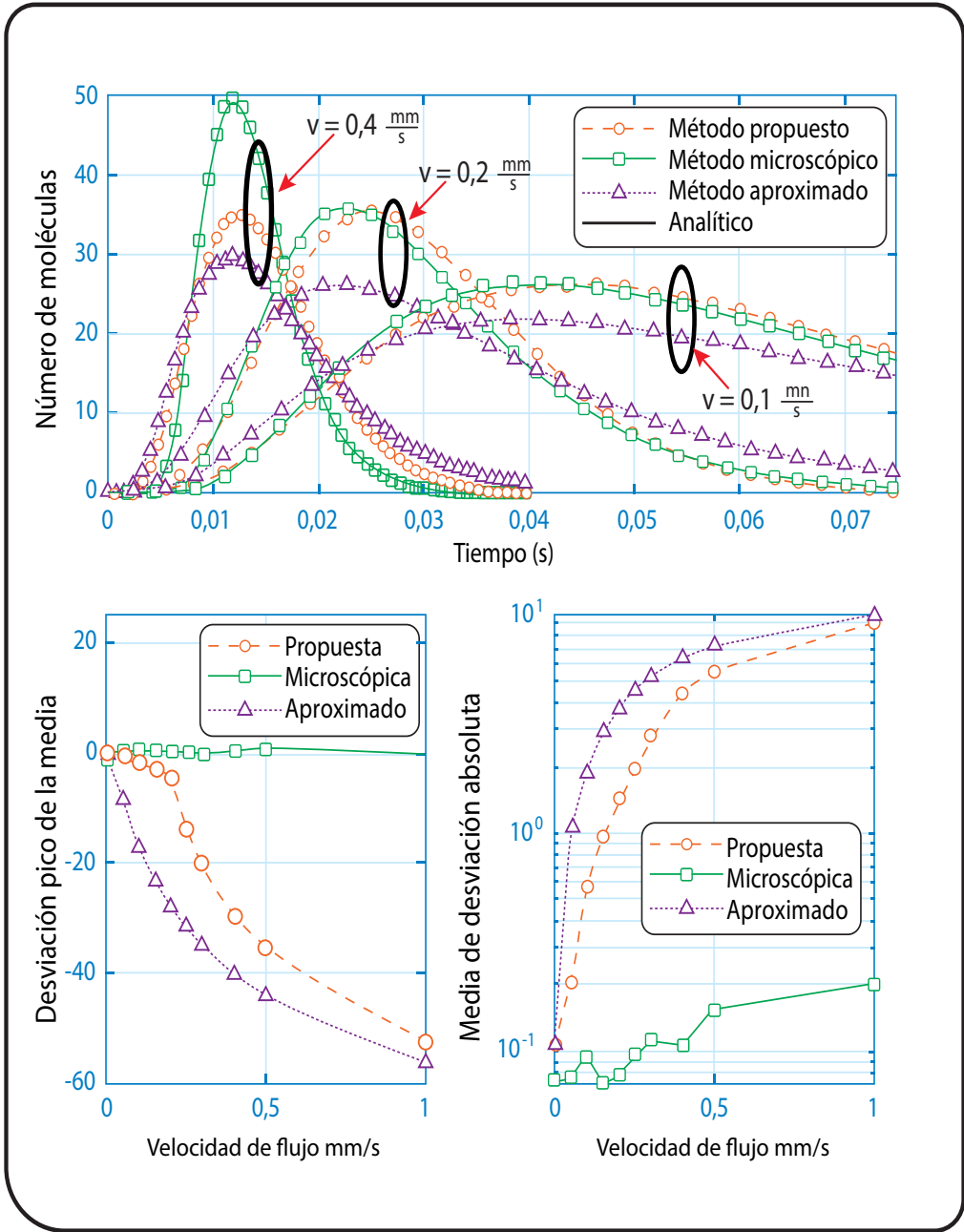


Figura 2.22. Ejemplo de resultados obtenidos en [76] a través de simulaciones en AcCoRD.

AcCoRD posee las siguientes características [75-83]:

1. Las regiones microscópicas establecen la conceptualización para cada molécula individualmente y su evolución en pasos discretos de tiempo. Las regiones mesoscópicas cuentan el número de moléculas en contenedores virtuales seccionados (llamados subvolúmenes) y operan en pasos de tiempo con granularidad continua. Si se incrementa la escalabilidad se ajusta la colocación de regiones mesoscópicas adyacentes con subvolúmenes de diferentes tamaños. Esta característica es una extensión de sistemas de CM en 2D [82].
2. Los componentes pueden distribuirse por todo el entorno como fuentes de moléculas activas (es decir, transmisores) u observadores pasivos (es decir, receptores). Los transmisores liberan moléculas de acuerdo con la modulación de una secuencia de datos binarios aleatorios. Los receptores pueden registrar el número de moléculas y (opcionalmente) sus posiciones en cualquier intervalo especificado. Posteriormente, se podrán combinar estas dos clases de particularidades para habilitar transeptores que se comporten de acuerdo con sus "observaciones". El número exacto de moléculas liberadas y el tiempo de liberación para un intervalo de símbolo dado puede ser determinista o aleatorio.
3. Las reacciones químicas de primer y segundo orden se pueden definir local o globalmente, esto es, en todo el entorno de propagación o en un conjunto específico de regiones. Este marco general puede adaptarse a reacciones tales como la degradación de moléculas, la cinética de enzimas, las uniones superficial reversible o irreversible, las uniones de tipo ligando-receptor, las transiciones a través de los límites de las membranas y el aglutinamiento molecular simplificado. Las reacciones de unión a la superficie incluyen los procesos de absorción, adsorción y desorción, mismos que se relacionan con las acciones de, consumo, adherencia y liberación.
4. AcCoRD implementa el comportamiento microscópico en tiempo continuo. Concretamente, la liberación de moléculas por actores activos, reacciones de orden cero y la mayoría de las reacciones de primer orden pueden ocurrir en cualquier momento. Por lo que, una molécula puede sufrir múltiples reacciones en un solo paso de tiempo microscópico, y la precisión de estos fenómenos es independiente del paso de tiempo considerado.

5. Se puede realizar n número de veces simulaciones independientes en diferentes computadoras y después agregarse al entorno del simulador en conjunto, para determinar el comportamiento promedio y las estadísticas del canal de transmisión.
6. La documentación de AcCoRD incluye instrucciones de instalación y utilización de su última versión, descripciones de todas las opciones de configuración y muchos archivos de configuración de muestra [75]. Las configuraciones de ejemplo se proporcionan para demostrar toda la funcionalidad de AcCoRD.
7. La interfaz de AcCoRD ha sido diseñada para ayudar a los usuarios novatos proporcionando mensajes de salida descriptivos. AcCoRD también incluye herramientas de posprocesamiento en MATLAB. Estas herramientas permiten la agregación de archivos de salida de la simulación, el trazado de observaciones del receptor (ya sea el comportamiento variable en el tiempo o las distribuciones empíricas en momentos específicos) y la visualización del entorno físico.

En las Figuras 2.23 a) - d), se representan cuatro entornos de propagación molecular que ilustran las funciones principales de AcCoRD. Las Figuras 2.23 a) y 2.23 b), revelan entornos híbridos con regiones microscópicas y mesoscópicas. La Figura 2.23 c) consta de un transmisor y distintos receptores. La Figura 2.23 d) denota reacciones moleculares superficiales reversibles [76].

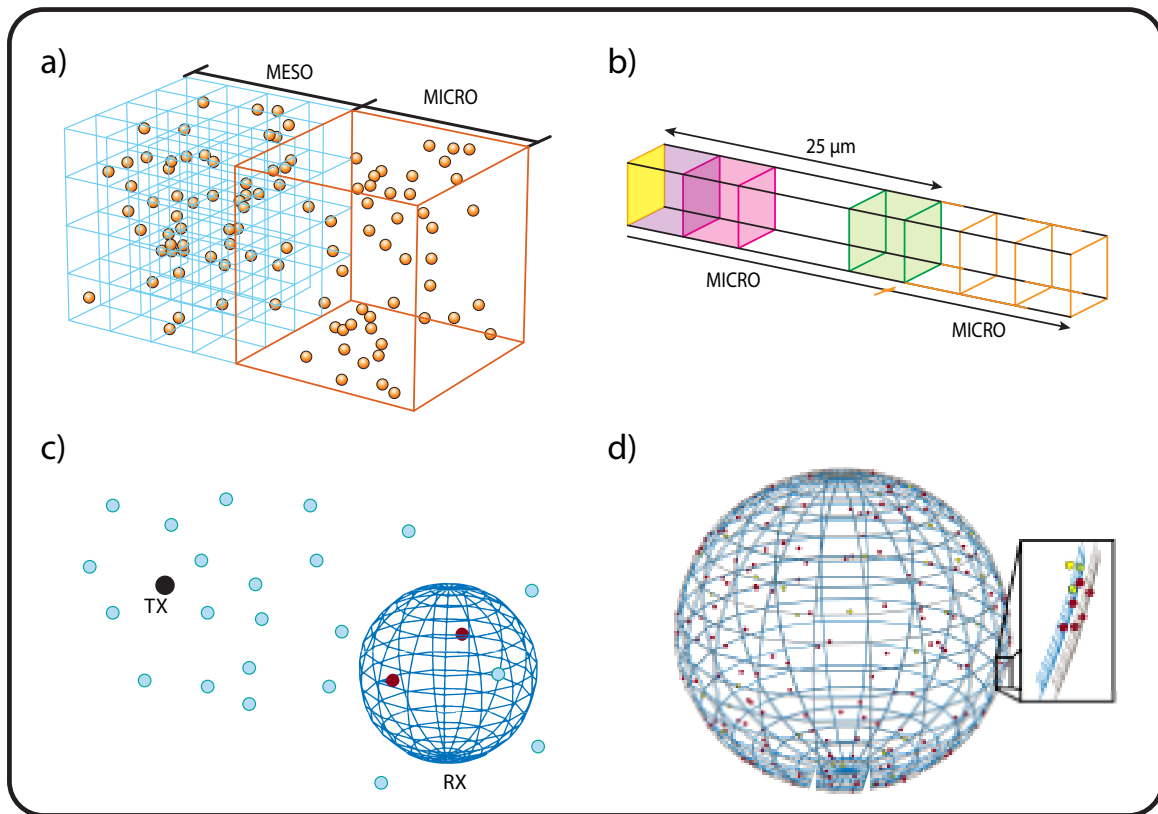


Figura 2.23. Entornos de propagación que ilustran las funciones principales de AcCoRD [76].

2.3.13 N⁴Sim (NaNoNetwork Simulator 4)

Originalmente el simulador N⁴Sim (NaNoNetwork Simulator With Molecular Communications) fue diseñado para el análisis del sistema nervioso en nanoredes de CM, y posteriormente se transformó en N⁴Sim (Nervous NaNoNetwork Simulator With Synaptic Molecular Communications). La naturaleza no convencional de las CM requiere contribuciones de una gran cantidad de campos científicos, lo que hace que el diseño del simulador para tales sistemas sea desafiante [84].

El sistema nervioso del cuerpo humano es una de las nanoredes más grandes e importantes. Existen en la literatura varios simuladores que se sustentan en las interacciones moleculares, y otros, que operan con algunos simuladores de redes neuronales, sin embargo, la mayoría de estos simuladores no son específicos para el sistema nervioso ya que ignoran la difusión sináptica debido a la complejidad computacional requerida para su modelamiento. Por lo que, en [84] se propone un

simulador denominado N⁴Sim, para resolver los problemas antes mencionados en los simuladores existentes. N⁴Sim se emplea para comunicaciones sinápticas de una manera rápida y eficiente. El modelo para el canal de comunicación sináptica es comparable en precisión a los obtenidos mediante las simulaciones Monte Carlo, utilizando una fracción de tiempo y recursos de procesamiento. N⁴Sim potencia un gran conjunto de opciones de diseño para aplicaciones en el sistema nervioso [84].

2.3.13 TouchCom (Touchable-Communication)

El paradigma TouchCom está provisto de un diseño en el que se puede visualizar la interacción de las aplicaciones prácticas de sistemas de comunicación táctil a nanoescala. Este paradigma se centra en las propiedades controlables y rastreables de las portadoras de moléculas de información [85, 86].

TouchCom trabaja en el marco de referencia de los estándares para nanocomunicaciones de IEEE 1906.1 y 1906.1.1, y así se debe ejecutar usando los cinco componentes que se indican en la Figura 2.24, y son: portadora de mensajes, campo, perturbación, movimiento y especificidad. Adicionalmente, en este simulador se establece en forma explícita la estructura del medio de transmisión [85, 86].

En TouchCom los parámetros de ingreso de información incluyen conjuntos de datos para [85, 86]:

1. La generación de transmisiones moleculares en vasos sanguíneos.
2. Determinación de la concentración inicial de un fármaco.
3. Tipificación de diversos fármacos.
4. Estudio de la resistencia y sensibilidad a fármacos.
5. Análisis de la vida útil de los dispositivos nanométricos empleados en la simulación del tratamiento y tamaño de la región visible (es decir, la ocupación de la región visible en toda la región basada en fractales) captada por el ojo humano.

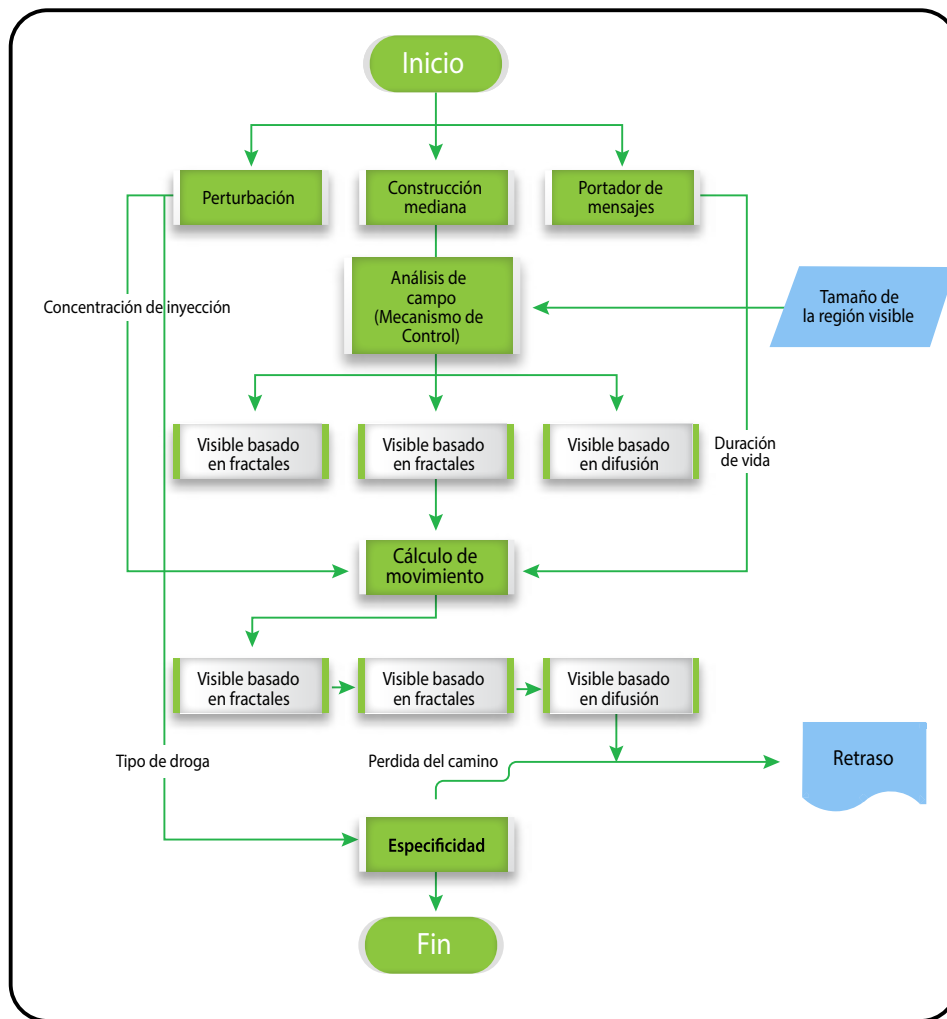


Figura 2.24. Ejemplo del diagrama de flujo de una simulación en TouchCom [85].

Existen portales para la expansión de cada uno de los componentes de TouchCom con la finalidad de que más funciones se puedan conectar convenientemente a la versión actual. La mejora adicional de TouchCom según [85], se centra en la implementación de medios de propagación con difusión y casos con múltiples receptores. TouchCom además considera casos para el uso de sistemas de bobinas 3D y nanopartículas magnéticas artificiales, la tecnología de tipo lab-on-chip y nanopartículas magnéticas para la prueba de movimiento en una red fractal [85]. TouchCom representa un marco unificado que integra la robótica (nanobots) y las telecomunicaciones [85, 86].

2.3.14 Trans-NanoSim

Es una herramienta desarrollada mediante el lenguaje de programación Python, y simula lecturas de información con características técnicas y específicas del transcriptoma aprendidas a partir de la secuenciación de ARN con nanoporos [87].

Trans-NanoSim, se ejecuta como un flujo de procesos biológicos en dos etapas. En la primera se capturan las características específicas de la tecnología que proveen las lecturas de transcriptomas de ONT (Oxford Nanopore Technologies), y en la segunda instancia, se simulan los datos obtenidos de las lecturas de características similares de estos transcriptomas. Por tanto, en la etapa de modelado se usa herramientas de última generación para alinear las lecturas a un transcriptoma de referencia, y genera modelos estadísticos que describen los perfiles de las lecturas como son modos de error y distribuciones de longitud [87].

En la actualidad Trans-NanoSim es un simulador obsoleto, sin embargo, se ha implementado en su totalidad como herramienta en el simulador NanoSim [87].

La secuenciación de ARN es una tecnología que ha ayudado a profundizar el entendimiento de los transcriptomas, en los últimos años se han desarrollado simuladores de lecturas largas genómicas de ONT. DeepSimulator es una de las varias herramientas existentes para esta clase de estudios, este simulador emplea modelado para el aprendizaje profundo en la simulación de señales de corriente eléctrica, que luego son decodificadas en lecturas de secuencia utilizando cualquier método de tipo *base-calling* disponible [87-90].

El estudio realizado por [87], despliega la simulación de transcriptomas en Trans-NanoSim y DeepSimulator, en donde a través de la evaluación comparativa de la similitud entre lecturas experimentales y simuladas, es notable que Trans-NanoSim supera consistentemente al simulador genómico DeepSimulator [87]. Las Figuras 2.25 a) y b) evidencia los resultados estadísticos que se obtuvieron a través de la simulación en los programas mencionados [87].

En la Figura 2.25 b) se puede notar claramente el nivel de confiabilidad en los resultados que proporciona Trans-NanoSim, debido al valor del parámetro estadístico $R^2 = 0,9444$ [87].

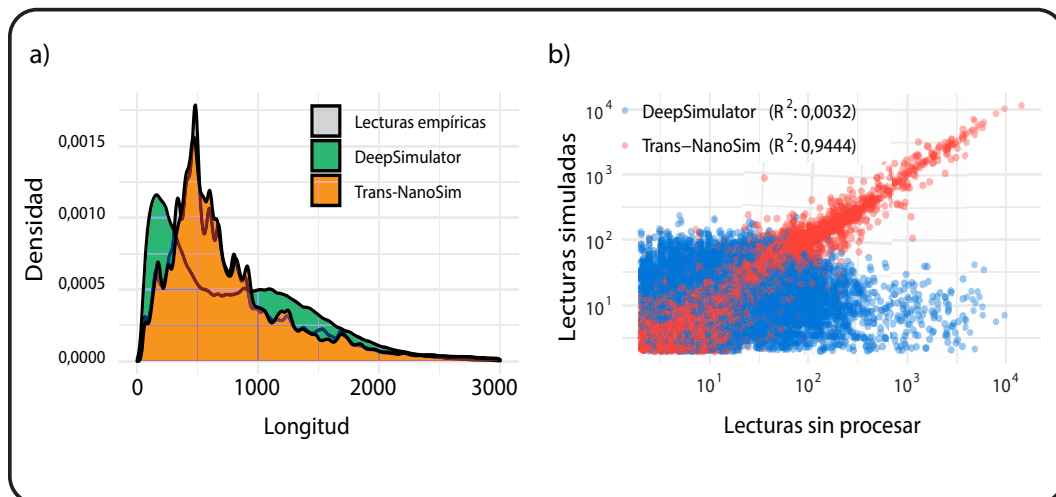


Figura 2.25. Resultados estadísticos comparativos entre los simuladores referidos en [87]. a) Distribución de datos de lecturas de transcriptomas. b) Resultados de coeficiente de determinación para DeepSimulator y Trans-NanoSim en base a R^2 .

Otra característica relevante de Trans-NanoSim es la eficiencia y baja demanda de recursos computacionales necesarios para ejecutar una simulación, como se reporta en [87], Trans-NanoSim desarrolló la simulación durante 2 horas y 11 minutos con un pico de memoria de 526 MB, mientras que DeepSimulator simuló durante 1 día, 8 horas y 32 minutos en total con un pico de memoria de 17,22 GB.



Referencias Bibliográficas

- [1] L. Felicetti, S. S. Assaf, M. Femminella, G. Reali, E. Alarcon, and J. Sole-Pareta, "The molecular communications markup language (Mol-ComML)," *Nano Commun. Netw.*, vol. 16, pp. 12–25, 2018.
- [2] T. Nakano, A. W. Eckford, and T. Haraguchi, "Molecular Communication," Cambridge University Press, Sep. 2013. doi: <https://doi.org/10.1017/CBO9781139149693>
- [3] D. Bi, A. Almpanis, A. Noel, Y. Deng, and R. Schober, "A survey of molecular communication in cell biology: Establishing a new hierarchy for interdisciplinary applications," *IEEE Commun. Surv. Tutor.*, vol. 23, no. 3, pp. 1494–1545, 2021.
- [4] M. Femminella and G. Reali, "Implementation issues of diffusion-based molecular communications receivers based on transcriptional elements," *Digit. Signal Process.*, vol. 124, p. 103160, 2022.
- [5] S. Canovas-Carrasco, A.-J. Garcia-Sanchez, and J. Garcia-Haro, "The IEEE 1906.1 Standard: Nanocommunications as a new source of data," in *2017 ITU Kaleidoscope: Challenges for a Data-Driven Society (ITU K)*, Nov. 2017, pp. 1–7. doi: <https://ieeexplore.ieee.org/document/8247001>
- [6] M. Egan *et al.*, "Toward Interdisciplinary Synergies in Molecular Communications: Perspectives from Synthetic Biology, Nanotechnology, Communications Engineering and Philosophy of Science," *Life*, vol. 13, no. 1, p. 208, 2023.
- [7] "IEEE SA - IEEE 1516-2010." Accessed: Dec. 13, 2022. [Online]. Available: <https://standards.ieee.org/ieee/1516/3744/>
- [8] S. Canovas-Carrasco, A.-J. Garcia-Sanchez, and J. Garcia-Haro, "The IEEE 1906.1 standard: Some guidelines for strengthening future normalization in electromagnetic nanocommunications," *IEEE Commun. Stand. Mag.*, vol. 2, no. 4, pp. 26–32, 2018.
- [9] E. Alarcon *et al.*, "MolComML: The molecular communication markup language," in *Proceedings of the 3rd ACM International Conference on Nanoscale Computing and Communication*, 2016, pp. 1–6.

- [10] C. Mouradian, D. Naboulsi, S. Yangui, R. Glitho, M. Morrow, and P. Polakos, "A Comprehensive Survey on Fog Computing: State-of-the-Art and Research Challenges," 2017, doi: <https://ieeexplore.ieee.org/document/8100873>.
- [11] T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler, F. Yergeau, and J. Cowan, "Extensible markup language (XML) 1.0." W3C recommendation October, 2000.
- [12] D. Eastlake 3rd, J. Reagle, and D. Solo, "(Extensible Markup Language) XML-Signature Syntax and Processing," 2002.
- [13] S. M. Keating *et al.*, "SBML Level 3: an extensible format for the exchange and reuse of biological models," *Mol. Syst. Biol.*, vol. 16, no. 8, p. e9110, 2020.
- [14] S. Konur, P. Gheorghe, and P. Krasnogor, "Verifiable Biology," *Natalio Verifiable Biol.*
- [15] M. Hucka *et al.*, "The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models," *Bioinformatics*, vol. 19, no. 4, pp. 524–531, 2003.
- [16] G. Booch, J. Rumbaugh, I. Jacobson, J. J. G. Molina, and J. S. Martínez, *El lenguaje unificado de modelado: guía del usuario*. Addison-Wesley, 2006.
- [17] M. Catalán Valbuena, "Nuevo lenguaje de especificación de experimentos para el simulador GRO," 2017.
- [18] M. Clerx *et al.*, "CellML 2.0," *J. Integr. Bioinforma.*, vol. 17, no. 2–3, 2020.
- [19] C. M. Lloyd, M. D. Halstead, and P. F. Nielsen, "CellML: its future, present and past," *Prog. Biophys. Mol. Biol.*, vol. 85, no. 2–3, pp. 433–450, 2004.
- [20] S. Crook, P. Gleeson, F. Howell, J. Svitak, and R. A. Silver, "MorphML: Level 1 of the NeuroML standards for neuronal morphology data and model specification," *Neuroinformatics*, vol. 5, no. 2, pp. 96–104, 2007.
- [21] P. Gleeson *et al.*, "NeuroML: a language for describing data driven models of neurons and networks with a high degree of biological detail," *PLoS Comput. Biol.*, vol. 6, no. 6, p. e1000815, 2010.

- [22] J. Birgiolas, V. Haynes, P. Gleeson, R. C. Gerkin, S. W. Dietrich, and S. M. Crook, "NeuroML-DB: Sharing and characterizing data-driven neuroscience models described in NeuroML," *bioRxiv*, 2021.
- [23] S. Panagiotou, H. Sidiropoulos, D. Soudris, M. Negrello, and C. Strydis, "EDEN: A high-performance, general-purpose, NeuroML-based neural simulator," *Front. Neuroinformatics*, vol. 16, 2022.
- [24] M. van der Vlag, S. Diaz, J. Fousek, A. Peyser, M. Woodman, and V. Jirsa, "RateML, a spin-off of the NeuroML and LEMS Domain Specific Languages, tailored to generate rate-based-models suited for simulators such as the Virtual Brain (TVB) featuring high performance computing and parameter sweep capabilities.," in *Bernstein Conference*, Jülich Supercomputing Center, 2020.
- [25] J. Banks, J. S. CARSON II, and L. Barry, "Discrete-event system simulation fourth edition." Pearson, 2005.
- [26] L. Felicetti, M. Femminella, and G. Reali, "A simulation tool for nanoscale biological networks," *Nano Commun. Netw.*, vol. 3, no. 1, pp. 2-18, 2012.
- [27] L. Felicetti, M. Femminella, G. Reali, P. Gresele, and M. Malvestiti, "Simulating an in vitro experiment on nanoscale communications by using BiNS2," *Nano Commun. Netw.*, vol. 4, no. 4, pp. 172-180, 2013.
- [28] I. Llatser *et al.*, "Exploring the physical channel of diffusion-based molecular communication by simulation," in *2011 IEEE Global Telecommunications Conference-GLOBECOM 2011*, IEEE, 2011, pp. 1-5.
- [29] G. E. Uhlenbeck and L. S. Ornstein, "On the theory of the Brownian motion," *Phys. Rev.*, vol. 36, no. 5, p. 823, 1930.
- [30] M. C. Wang and G. E. Uhlenbeck, "On the theory of the Brownian motion II," *Rev. Mod. Phys.*, vol. 17, no. 2-3, p. 323, 1945.
- [31] M. Pierobon and I. F. Akyildiz, "A physical end-to-end model for molecular communication in nanonetworks," *IEEE J. Sel. Areas Commun.*, vol. 28, no. 4, pp. 602-611, 2010.
- [32] J.-P. Rospars, V. Křivan, and P. Lánský, "Perireceptor and receptor events in olfaction. Comparison of concentration and flux detectors: a modeling study," *Chem. Senses*, vol. 25, no. 3, pp. 293-311, 2000.

- [33] I. Llatser, D. Demiray, A. Cabellos-Aparicio, D. T. Altilar, and E. Alarcón, "N3Sim: Simulation framework for diffusion-based molecular communication nanonetworks," *Simul. Model. Pract. Theory*, vol. 42, pp. 210–222, 2014.
- [34] A. Goldbeter, G. Dupont, and M. J. Berridge, "Minimal model for signal-induced Ca²⁺ oscillations and for their frequency encoding through protein phosphorylation.," *Proc. Natl. Acad. Sci.*, vol. 87, no. 4, pp. 1461–1465, 1990.
- [35] M. Pierobon and I. F. Akyildiz, "Diffusion-based noise analysis for molecular communication in nanonetworks," *IEEE Trans. Signal Process.*, vol. 59, no. 6, pp. 2532–2547, 2011.
- [36] B. Atakan and O. B. Akan, "On channel capacity and error compensation in molecular communication," in *Transactions on computational systems biology X*, Springer, 2008, pp. 59–80.
- [37] D. Demiray *et al.*, "Direct: A model for molecular communication nanonetworks based on discrete entities," *Nano Commun. Netw.*, vol. 4, no. 4, pp. 181–188, 2013.
- [38] N. Garralda, I. Llatser, A. Cabellos-Aparicio, and M. Pierobon, "Simulation-based evaluation of the diffusion-based physical channel in molecular nanonetworks," in *2011 IEEE conference on computer communications workshops (INFOCOM WKSHPS)*, IEEE, 2011, pp. 443–448.
- [39] I. Llatser, E. Alarcón, and M. Pierobony, "Diffusion-based channel characterization in molecular nanonetworks," in *2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, IEEE, 2011, pp. 467–472.
- [40] L. Felicetti, M. Femminella, and G. Reali, "Simulation of molecular signaling in blood vessels: Software design and application to atherogenesis," *Nano Commun. Netw.*, vol. 4, no. 3, pp. 98–119, 2013.
- [41] M. T. Barros, S. Balasubramaniam, and B. Jennings, "Comparative end-to-end analysis of Ca²⁺-signaling-based molecular communication in biological tissues," *IEEE Trans. Commun.*, vol. 63, no. 12, pp. 5128–5142, 2015.
- [42] "calcomsim." Accessed: Dec. 16, 2022. [Online]. Available: <https://sites.google.com/site/calcomsimulator/>

- [43] calcomsim, "calcomsimv1.0." Sep. 08, 2015. Accessed: Dec. 16, 2022. [Online]. Available: <https://github.com/calcomsim/calcomsimv1.0>
- [44] D. T. Gillespie, "Exact stochastic simulation of coupled chemical reactions," *J. Phys. Chem.*, vol. 81, no. 25, pp. 2340–2361, 1977.
- [45] X. Cai, "Exact stochastic simulation of coupled chemical reactions with delays," *J. Chem. Phys.*, vol. 126, no. 12, p. 124108, 2007.
- [46] M. T. Barros, "Ca²⁺-signaling-based molecular communication systems: Design and future research directions," *Nano Commun. Netw.*, vol. 11, pp. 103–113, 2017.
- [47] Y. Chahibi, M. Pierobon, S. O. Song, and I. F. Akyildiz, "A molecular communication system model for particulate drug delivery systems," *IEEE Trans. Biomed. Eng.*, vol. 60, no. 12, pp. 3468–3483, 2013.
- [48] E. J. Dickinson, H. Ekström, and E. Fontes, "COMSOL Multiphysics®: Finite element software for electrochemical analysis. A mini-review," *Electrochem. Commun.*, vol. 40, pp. 71–74, 2014.
- [49] O. V. Klymenko, I. Svir, and C. Amatore, "A new approach for the simulation of electrochemiluminescence (ECL)," *ChemPhysChem*, vol. 14, no. 10, pp. 2237–2250, 2013.
- [50] M. Bolhassan, A. Abdali, and M. Kuscu, "Microfluidic molecular communication transmitter based on hydrodynamic gating," *IEEE Transactions on Molecular, Biological and Multi-Scale Communications*, 2024.
- [51] S. F. Bush, J. L. Paluh, G. Piro, V. Rao, R. V. Prasad, and A. Eckford, "Defining communication at the bottom," *IEEE Trans. Mol. Biol. Multi-Scale Commun.*, vol. 1, no. 1, pp. 90–96, 2015.
- [52] E. Gul, B. Atakan, and O. B. Akan, "NanoNS: A nanoscale network simulator framework for molecular communications," *Nano Commun. Netw.*, vol. 1, no. 2, pp. 138–156, 2010.
- [53] B. K. Kamapantula, A. F. Abdelzaher, M. Mayo, E. J. Perkins, S. K. Das, and P. Ghosh, "Quantifying robustness in biological networks using NS-2," in *Modeling, Methodologies and Tools for Molecular and Nano-scale Communications*, Springer, 2017, pp. 273–290.

- [54] G. Wei, P. Bogdan, and R. Marculescu, "Efficient modeling and simulation of bacteria-based nanonetworks with BNSim," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 12, pp. 868–878, 2013.
- [55] Z. Wang, M. Kim, and G. Rosen, "Validating models of bacterial chemotaxis by simulating the random motility coefficient," in *2008 8th IEEE International Conference on Bioinformatics and BioEngineering*, IEEE, 2008, pp. 1–5.
- [56] S. Balasubramaniam, "Multi-hop conjugation based bacteria nanonetworks," *IEEE Trans. Nanobioscience*, vol. 12, no. 1, pp. 47–59, 2013.
- [57] "Make Your Systems Work Together," Pitch Technologies. Accessed: Dec. 13, 2022. [Online]. Available: <https://pitchtechnologies.com/>.
- [58] A. Akkaya, G. Genc, and T. Tugcu, "HLA based architecture for molecular communication simulation," *Simul. Model. Pract. Theory*, vol. 42, pp. 163–177, 2014.
- [59] "Forwardsim HLA Toolbox for MATLAB - Forwardsim." Accessed: Dec. 13, 2022. [Online]. Available: <https://www.forwardsim.com/products/hla-toolbox/>
- [60] "Make Your Systems Work Together," Pitch Technologies. Accessed: Dec. 13, 2022. [Online]. Available: <https://pitchtechnologies.com/>
- [61] EMostafaAli, "HlaListener." May 19, 2022. Accessed: Dec. 13, 2022. [Online]. Available: <https://github.com/EMostafaAli/HlaListener>
- [62] O. Villimann, "HLA Framework," *Dan. Marit. Inst. CTO Proj. Doc.*, 1999.
- [63] J. W. Van Spengen, "Fedef: A high level architecture federate development framework," DEFENCE RESEARCH AND DEVELOPMENT ATLANTIC DARTMOUTH (CANADA), 2010.
- [64] A. Falcone, A. Garro, S. J. Taylor, A. Anagnostou, N. R. Chaudhry, and O. Salah, "Experiences in simplifying distributed simulation: The HLA development kit framework," *J. Simul.*, vol. 11, no. 3, pp. 208–227, 2017.
- [65] A. Falcone, A. Garro, A. Anagnostou, N. R. Chaudhry, O.-A. Salah, and S. J. Taylor, "Easing the development of HLA federates: The HLA

development kit and its exploitation in the SEE project,” in *2015 IEEE/ACM 19th International Symposium on Distributed Simulation and Real Time Applications (DS-RT)*, IEEE, 2015, pp. 50–57.

[66] A. Falcone, A. Garro, A. D’Ambrogio, and A. Giglio, “Engineering systems by combining BPMN and HLA-based distributed simulation,” in *2017 IEEE International Systems Engineering Symposium (ISSE)*, IEEE, 2017, pp. 1–6.

[67] “IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)– Framework and Rules,” *IEEE Std 1516-2010 Revis. IEEE Std 1516-2000*, pp. 1–38, Aug. 2010, doi: <https://ieeexplore.ieee.org/document/5553440>.

[68] A. Falcone and A. Garro, “Using the HLA Standard in the context of an international simulation project: The experience of the ‘SMAS-HTeam,’” in *15th international conference on modeling and applied simulation, MAS*, 2016, pp. 26–28.

[69] A. Garro and A. Falcone, “On the integration of HLA and FMI for supporting interoperability and reusability in distributed simulation,” in *Proceedings of the Symposium on Theory of Modeling & Simulation: DEVS Integrative M&S Symposium*, 2015, pp. 9–16.

[70] A. Akkaya and T. Tugcu, “dMCS: distributed molecular communication simulator,” in *Proceedings of the 8th International Conference on Body Area Networks*, 2013, pp. 468–471.

[71] S. AbouRizk, Y. Mohamed, H. Taghaddos, F. Saba, and S. Hague, “Developing complex distributed simulation for industrial plant construction using High Level Architecture,” in *Proceedings of the 2010 winter simulation conference*, IEEE, 2010, pp. 3177–3188.

[72] S. F. Bush, *Nanoscale communication networks*. Artech House, 2010.

[73] H. B. Yilmaz and C.-B. Chae, “Simulation study of molecular communication systems with an absorbing receiver: Modulation and ISI mitigation techniques” *Simul. Model. Pract. Theory*, vol. 49, pp. 136–150, Dec. 2014, doi: <https://doi.org/10.1016/j.simpat.2014.09.002>.

[74] “MolecUlar CommunicatIoN (MUCIN) Simulator - File Exchange - MATLAB Central.” Accessed: Dec. 14, 2022. [Online]. Available: <https://www.mathworks.com/matlabcentral/fileexchange/46066-molecular-communication-mucin-simulator>

[75] “GitHub - adamjgnoel/AcCoRD: Actor-based Communication via Reaction-Diffusion. A molecular communication simulator. Simulates stochastic reaction-diffusion systems.” Accessed: Dec. 14, 2022. [Online]. Available: <https://github.com/adamjgnoel/AcCoRD/>

[76] A. Noel, K. C. Cheung, R. Schober, D. Makrakis, and A. Hafid, “Simulating with AcCoRD: Actor-based Communication via Reaction-Diffusion,” *Nano Commun. Netw.*, vol. 11, pp. 44–75, Mar. 2017, doi: <https://doi.org/10.1016/j.nancom.2017.02.002>

[77] “AcCoRD Simulator.” Accessed: Dec. 14, 2022. [Online]. Available: https://warwick.ac.uk/fac/sci/eng/people/adam_noel/software/accord/

[78] O. A. Dambri and S. Cherkaoui, “Enhancing signal strength and ISI-avoidance of diffusion-based molecular communication,” in *2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC)*, IEEE, 2018, pp. 1–6.

[79] A. Noel, Y. Deng, D. Makrakis, and A. Hafid, “Active versus passive: Receiver model transforms for diffusive molecular communication,” in *2016 IEEE Global Communications Conference (GLOBECOM)*, IEEE, 2016, pp. 1–6.

[80] Y. Deng, A. Noel, W. Guo, A. Nallanathan, and M. Elkaslan, “3D stochastic geometry model for large-scale molecular communication systems,” in *2016 IEEE Global Communications Conference (GLOBECOM)*, IEEE, 2016, pp. 1–6.

[81] A. Noel and A. W. Eckford, “Asynchronous peak detection for demodulation in molecular communication,” in *2017 IEEE International Conference on Communications (ICC)*, IEEE, 2017, pp. 1–6.

[82] A. Noel, K. C. Cheung, and R. Schober, “Multi-scale stochastic simulation for diffusive molecular communication,” in *2015 IEEE International Conference on Communications (ICC)*, IEEE, 2015, pp. 1109–1115.

- [83] A. Noel, K. C. Cheung, and R. Schober, "On the statistics of reaction-diffusion simulations for molecular communication," in *Proceedings of the Second Annual International Conference on Nanoscale Computing and Communication*, 2015, pp. 1-6.
- [84] N. A. Turgut, B. A. Bilgin, and O. B. Akan, "N4Sim: The first Nervous NaNoNetwork Simulator with Synaptic Molecular Communications," *IEEE Trans. NanoBioscience*, 2021.
- [85] Y. Zhou, Y. Chen, R. D. Murch, R. Wang, and Q. Zhang, "Simulation framework for touchable communication on NS3Sim," *Nano Commun. Netw.*, vol. 16, pp. 26-36, 2018.
- [86] Y. Chen, P. Kosmas, P. S. Anwar, and L. Huang, "A touch-communication framework for drug delivery based on a transient microbot system," *IEEE Trans. Nanobioscience*, vol. 14, no. 4, pp. 397-408, 2015.
- [87] S. Hafezqorani, C. Yang, T. Lo, K. M. Nip, R. L. Warren, and I. Birol, "Trans-nanosim characterizes and simulates nanopore rna-sequencing data," *Gigascience*, vol. 9, no. 6, p. giaa061, 2020.
- [88] Z. Wang, M. Gerstein, and M. Snyder, "Rna-seq: a revolutionary tool for transcriptomics," *Nature reviews genetics*, vol. 10, no. 1, pp. 57-63, 2009
- [89] C. Yang, J. Chu, R. L. Warren, and I. Birol, "Nanosim: nanopore sequence read simulator based on statistical characterization," *GigaScience*, vol. 6, no. 4, p. gix010, 2017
- [90] Y. Li, S. Wang, C. Bi, Z. Qiu, M. Li, and X. Gao, "DeepSimulator1.5: a more powerful, quicker and lighter simulator for nanopore sequencing," *Bioinformatics*, vol. 36, no. 8, pp. 2578-2580, 2020.

**Lenguaje Extensible de
Marcado para Comunicaciones
Moleculares (MolComML-
Molecular Communications
Markup Language)**

03

El Lenguaje Extensible de Marcado para Comunicaciones Moleculares (MolComML-Molecular Communications Markup Language) ofrece un elevado nivel de abstracción, lo que lo convierte en una herramienta extremadamente útil para la transferencia de información molecular. Este sistema aprovecha componentes biológicos, tanto naturales como sintéticos, para facilitar las comunicaciones en el ámbito de la biología.

Una de las ventajas principales de MolComML es su capacidad para fusionar el análisis numérico con la síntesis experimental. Esto se logra a través de su lenguaje de marcado flexible, que no solo permite realizar simulaciones precisas, sino que también facilita la validación cruzada entre los datos experimentales y los resultados teóricos. Gracias a esta interoperabilidad, es posible integrar diferentes tipos de herramientas de software, potenciando así el alcance y la precisión de los experimentos.

En este capítulo, además de detallar las características y funcionalidades de MolComML, se explorarán estudios de casos de simulaciones realizadas con esta plataforma, ilustrando su aplicación práctica.

3.1

PLATAFORMAS DE PROGRAMACIÓN PARA EL ANÁLISIS, SIMULACIÓN Y EXPERIMENTACIÓN EN COMUNICACIONES MOLECULARES

Las Tecnologías de la Información y la Comunicación (TIC) ofrecen un amplio conjunto de herramientas que facilitan el análisis, el diseño y simulación para la transferencia de información en los sistemas de Comunicaciones Moleculares (CM). Aunque las teorías de comunicación desarrolladas para sistemas de transmisión tradicionales se han aplicado a los sistemas biológicos, los esfuerzos para avanzar en esta área han sido dispersos y fragmentados. En todo el mundo, diversas iniciativas han emergido con el objetivo de proporcionar principalmente herramientas de software para la programación y simulación de estos sistemas, y en algunos casos, para el desarrollo de interfaces de hardware orientadas a la experimentación en sistemas de CM [1].

A pesar de la importancia científica que representan estas iniciativas, uno de los principales problemas que ha surgido es la falta de compatibilidad entre los distintos enfoques. La naturaleza independiente de estos esfuerzos ha dificultado el empleo conjunto de herramientas, escenarios y resultados de casos de estudio entre investigaciones, lo que ha generado obstáculos significativos en la usabilidad y reutilización de información en sistemas de CM. Así, aunque existen numerosos lenguajes de programación y simuladores para sistemas de nanocomunicaciones moleculares, la mayoría de ellos no son interoperables. Este problema de falta de estandarización ha evidenciado la necesidad de contar con un lenguaje universal que permita el manejo eficaz de los recursos de simulación y experimentación en sistemas biológicos de comunicación [1].

En respuesta a esta necesidad surge el Lenguaje Extensible de Marcado para Comunicaciones Moleculares (MolComML^x), que es una solución diseñada para promover la repetibilidad y la consistencia de los resultados experimentales. MolComML no solo facilita la interoperabilidad entre diferentes plataformas de simulación, sino que también mejora la compatibilidad y reproducibilidad de los experimentos. Esto facilita

^x MolComML se puede entender como la contraparte europea de NS-3 (Network Simulator-3) para estandarizar las simulaciones de CM.

el intercambio de información entre laboratorios y plataformas, lo cual es fundamental para validar y contrastar los resultados obtenidos en investigaciones biológicas experimentales (Figura 3.1) [1].

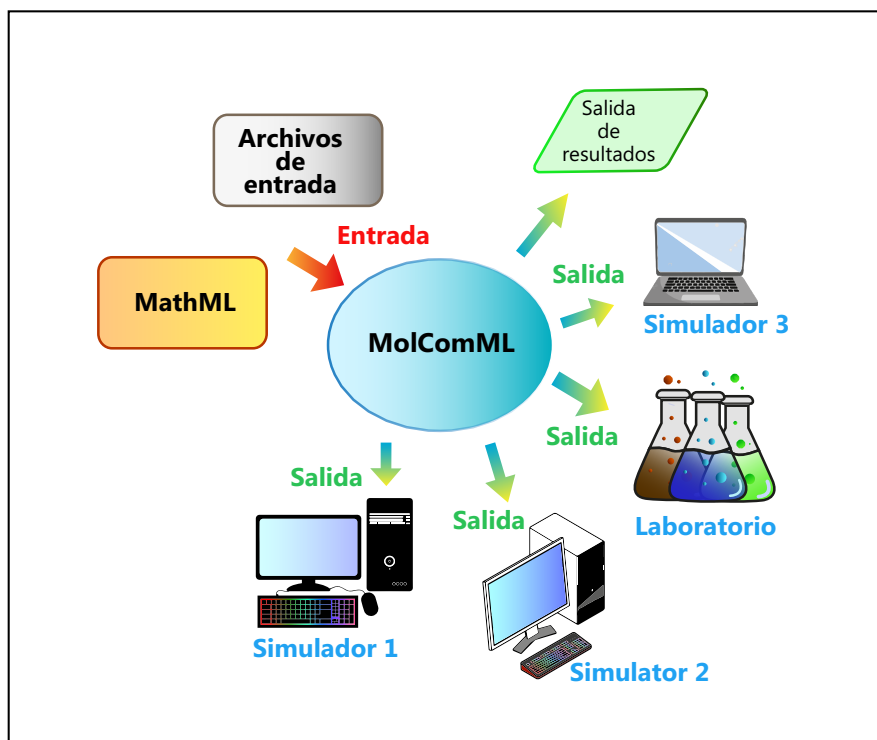


Figura 3.1. Funcionamiento de MolComML [1].

La principal fortaleza de MolComML radica en su capacidad para integrar diferentes lenguajes de marcado utilizados en múltiples disciplinas científicas, lo que lo convierte en una herramienta altamente versátil. Entre los lenguajes más relevantes que MolComML incorpora se encuentran:

- MathML (Mathematical Markup Language), el cual permite la inclusión de funciones matemáticas para la representación precisa de ecuaciones y parámetros complejos [1].
- SBML (Systems Biology Markup Language) [2] y SBOL (Synthetic Biology Open Language) [3], ambos diseñados para describir de manera estructurada procesos biológicos en sistemas *in-vivo* y sintéticos, respectivamente.

En este contexto, es importante mencionar la amplia variedad de programas e interfaces que existen para la simulación y/o experimentación

en nanosistemas de comunicación, lo que destaca la diversidad de enfoques en este campo. A continuación, se resume brevemente algunos lenguajes de marcado que se analizaron el Capítulo 2 [1]:

- SBML es un lenguaje especializado en el modelamiento de procesos biológicos complejos (en la Sección 2.2 del Capítulo 2 se analiza SBML).
- SBOL es un lenguaje estándar para la descripción y simulación de sistemas biológicos sintéticos (en la Sección 2.2 del Capítulo 2 se analiza SBOL).
- CellML (Cellular Modeling Markup Language), es un lenguaje diseñado para el modelado matemático de funciones celulares (en la Sección 2.2 del Capítulo 2 se analiza CellML).
- NeuroML (Neuroscience Markup Language), este es un lenguaje enfocado en el modelado de redes neuronales y sistemas de transferencia de información cerebral (en la Sección 2.2 del Capítulo 2 se analiza NeuroML).

Estos lenguajes y sistemas proporcionan un marco robusto para la simulación de fenómenos biológicos y la experimentación en CM, contribuyendo a la interoperabilidad entre diversas plataformas, facilitando la validación cruzada de resultados.

3.2

ANÁLISIS DEL LENGUAJE EXTENSIBLE DE MARCADO PARA COMUNICACIONES MOLECULARES MolComML

En los últimos años, el campo de las CM se ha desarrollado aceleradamente, apoyado en una amplia gama de plataformas tecnológicas (como se ha estudiado en el capítulo anterior). Este progreso ha sido posible gracias a la colaboración interdisciplinaria entre áreas como la biología, la informática y la ingeniería de telecomunicaciones. Sin embargo, a pesar de la diversidad de enfoques y herramientas disponibles, era imprescindible contar con un lenguaje común que posibilite un análisis coherente y preciso de la información generada por los sistemas de nanocomunicaciones, independientemente del ámbito de estudio que se emplee. Para solventar este requerimiento aparece

MolComML, que se presenta como una solución universal de software estandarizado similar a la alternativa que los estándares IEEE 1906.1 y 1906.1.1 tenían con NS-3 los cuales buscan estandarizar los lenguajes de programación, simuladores y herramientas de experimentación en CM.

El principal objetivo de MolComML es permitir a los investigadores diseñar y configurar experimentos de manera uniforme, definiendo interfaces de entrada/salida, recopilando, compartiendo y validando resultados. Además, este lenguaje facilita la reutilización y modificación de modelos, garantizando la repetibilidad de los experimentos a través de diferentes herramientas de simulación. Para apoyar este esfuerzo global, se creó el proyecto CIRCLE (Coordinating European Research on Molecular Communications), con la participación de organizaciones de diversas regiones como Estados Unidos, Asia y Australia [4]. CIRCLE fue diseñado para impulsar la adopción de MolComML a nivel mundial y destacar los beneficios derivados de su implementación.

MolComML es un lenguaje basado en XML que combina el análisis numérico con la síntesis experimental, brindando una flexibilidad significativa [5]. Su estructura está diseñada para promover la convergencia de diferentes disciplinas científicas, permitiendo la integración de múltiples plataformas. Esto proporciona la interoperabilidad necesaria para representar elementos, interacciones, configuraciones y resultados experimentales en el ámbito de las CM [6].

Entre los principales lineamientos de MolComML para abordar los sistemas de comunicación a nivel molecular, se incluyen [4]:

1. La capacidad de representar distintos tipos de escenarios de CM en todos los niveles de abstracción.
2. La adaptación de formatos de archivos, lo que evita tener que reescribir modelos y permite el uso de diversas herramientas de software.
3. La garantía de conservación de los modelos, independientemente del software empleado.
4. El establecimiento de MolComML como el lenguaje estándar para el análisis de escenarios de CM mediante herramientas de software que sintetizan resultados experimentales.
5. La disponibilidad del lenguaje para su uso en diversos entornos de software, lo que facilita el acceso a modelos y herramientas.
6. La escalabilidad del lenguaje, y su aplicación en diferentes contextos y sistemas.

7. La necesidad de actualizaciones continuas para incorporar nuevas definiciones y modelos, dado el constante avance en el estudio de los sistemas de CM.

En las siguientes secciones, se explorarán en detalle los componentes fundamentales de MolComML y su funcionamiento en los diferentes escenarios de simulación y experimentación.

3.2.1 Elementos principales de MolComML

En la Figura 3.2, se presenta un diagrama UML (Lenguaje Unificado de Modelado) que ilustra algunos de los componentes clave de la estructura de un sistema de CM para MolComML. Este diagrama muestra cómo se organizan los elementos del sistema, donde cada componente incluye un conjunto de parámetros que pueden ser definidos como privados o públicos, según el propósito específico de la simulación que se desea llevar a cabo [7].

La Figura 3.2 destaca dos áreas sombreadas de particular importancia [7]:

- a) Zona sombreada en color celeste. Aquí se encuentran los componentes principales que constituyen la base estructural del lenguaje para sistemas de CM, los cuales se indican en los siguientes párrafos.
 - Elementos. Se denomina de esta manera a aquellas partes constitutivas de un sistema de CM.
 - Topologías e interfaces de red. Configuraciones y conexiones que definen cómo interactúan los nodos (transmisores y receptores) en la red molecular.
 - Protocolos de comunicación. Normas que regulan la transferencia de información entre los elementos.
 - Estados y reglas de interacción. Las condiciones y comportamientos que determinan las dinámicas en el sistema.
 - Canal de propagación. Medio físico por el cual la información se transfiere de un lugar a otro.
 - Tipos de modulación. Técnicas empleadas para codificar la información en las señales moleculares.
- b) Zona sombreada en color coral. En esta área se encuentran los atributos, reglas y parámetros que se relacionan con las fórmulas

matemáticas utilizadas en el simulador. Estos elementos son fundamentales para realizar cálculos precisos y para la correcta implementación de los algoritmos que rigen el comportamiento de CM.

Estas dos zonas destacan la estructura modular y adaptable de un sistema de CM, lo que permite un alto grado de personalización y flexibilidad en las simulaciones. El uso de este modelo facilita tanto el análisis como el diseño de sistemas de CM, integrando aspectos teóricos y experimentales bajo un marco común.

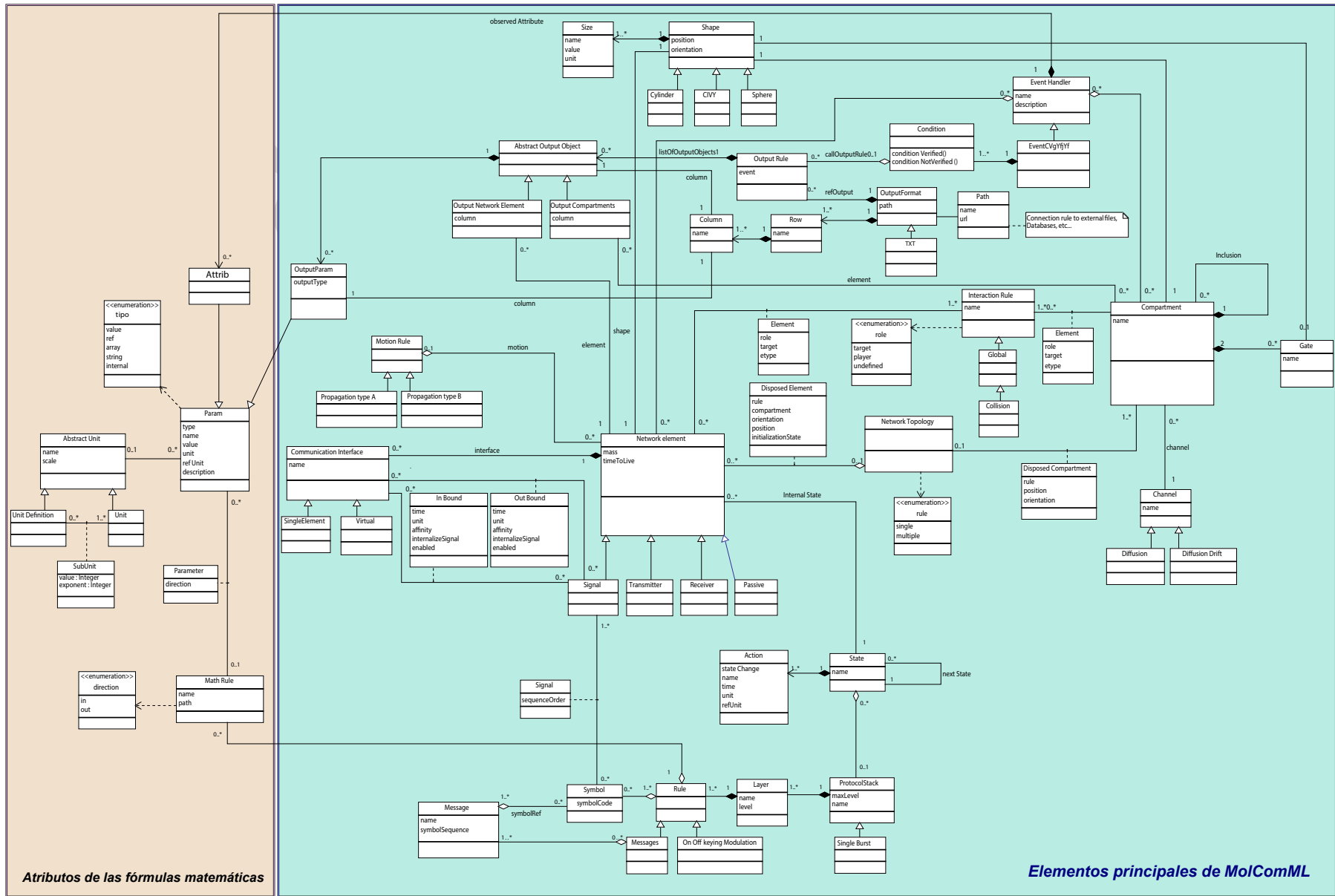


Figura 3.2. Diagrama UML en MolComML [7].

A continuación se detalla en profundidad cada uno de los componentes citados en el diagrama UML de MolComML.

3.2.2 Elementos de red (*Network elements*)

La clase *Network Elements* se emplea para la construcción de una arquitectura de red de nanocomunicaciones donde cada elemento se describe a través de un conjunto de atributos generales. Estos atributos incluyen: forma y tamaño del elemento, masa, tiempo de vida, interfaz de comunicación, reglas de movimiento (*motion rules*) y el proceso de propagación. Estos atributos denotan una visión detallada de cómo cada componente de la red molecular interactúa y se comporta en el sistema [7, 8].

Entre los principales elementos de red, destacan los siguientes objetos: transmisor, receptor y señal a ser emitida. Enseguida, se describen estos elementos, sus características y funciones [5]:

- **Transmisor (*Transmitter*) Tx**

El objeto *Transmitter* es el encargado de codificar la información en forma de señal molecular. Este proceso de codificación incluye diversos parámetros clave, tales como [5, 8]:

- La tasa de creación de moléculas, que determina la velocidad a la que se generan las moléculas de señal.
- La tasa de emisión, que regula cuántas moléculas se libera en un determinado periodo de tiempo.
- El mecanismo de liberación de moléculas, que indica cómo las moléculas se dispersan en el entorno de propagación.

- **Receptor (*Receiver*) Rx**

Por su parte el objeto *Receiver* es el responsable de decodificar la información contenida en la señal molecular recibida. Dicha señal puede verse afectada por fluctuaciones y variaciones en el comportamiento de los canales moleculares, lo que puede alterar su contenido original. Además, se pueden definir múltiples configuraciones para los receptores, como por ejemplo los receptores absorbentes, que permiten diferentes comportamientos según el tipo de interacción deseada [8].

- Señal (*Signal*)

El transporte de información a través de las condiciones físicas del canal de transmisión molecular hasta llegar al receptor se realiza por medio de *Signal*. Existen diferentes formas de representar la información en forma de señales biológicas y químicas, como es el caso de señales de ADN (Ácido Desoxirribonucleico), proteínas, iones y otros tipos de moléculas biológicas. Estas señales son esenciales para garantizar la transmisión efectiva de información entre los componentes de la red molecular [8].

En la Figura 3.3, se visualizan cuatro elementos clave de red, estos son: el transmisor, el receptor y dos tipos distintos de señales. Es importante destacar que los términos en inglés se mantienen en esta descripción debido a que son nombres específicos de objetos y atributos definidos en el diagrama UML de MolComML que se muestra en la Figura 3.2 [4].

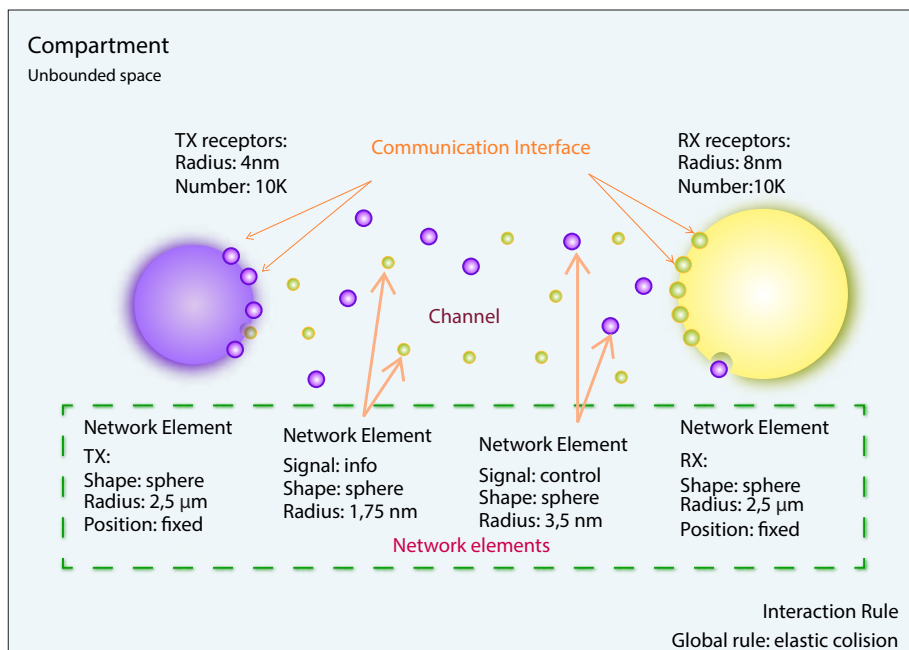


Figura 3.3. Descripción de elementos de red en MolComML [4].

En la Figura 3.4, se ilustra un fragmento de código que ejemplifica cómo se definen diversos elementos de simulación en MolComML, tales como nodos, direccionamiento, límites del entorno de propagación, entre otros. Estos elementos son fundamentales para construir una simulación precisa y detallada, permitiendo modelar con exactitud las interacciones y comportamientos dentro de un sistema de CM [4].

```
<listOfNetworkElement>
  <networkElement id="NE02"    name="RX_Node"  type="receiver">
    <shape      id="SH001"    type="sphere"  />
    <motion    refID=""      name="none"   />

    <child  id="NE02-1"    childOf="NE02"    name="RX"  >
      <size    refShape="SH001"    name="radius" refUnit="U05"  unit="nm"
      value="5000"  />
      <mass    value="100"    refUnit="U07"  unit="pg"   />

      <timeToLive value="10"    refUnit="UD03"  unit="hour"  />

      <interface refID="CI01-02"  type="RX_FullSurface" value="1"  />

    </child>
  </networkElement>
</listOfNetworkElement>
```

Figura 3.4. Definición de elementos de red en MolComML [4].

El código presentado en la figura anterior especifica los atributos y características de los nodos, incluyendo su ubicación espacial y sus propiedades, lo cual resulta esencial para garantizar que las simulaciones se ejecuten de manera coherente con los parámetros establecidos. Al definir aspectos como el direccionamiento y la limitación del entorno de propagación, se asegura que los movimientos y las interacciones de los elementos de red en la simulación sean consistentes con las reglas físicas y biológicas del sistema [4].

3.2.3 Interfaz de comunicación (*Communication Interface*)

La sección de código *Communication Interface* gestiona el intercambio de información entre los elementos de red. Para que este proceso sea viable, se requiere definir interfaces de comunicación externas que garanticen la interacción entre los diferentes componentes de la red. Estas interfaces poseen diversas propiedades que definen aspectos como la direccionalidad de la comunicación, la afinidad (compatibilidad entre elementos) y el tipo de señales que se transmiten y reciben [4, 7].

Los elementos de red pueden intercambiar información usando una interfaz de comunicación, lo que permite una descripción clara y detallada de cada uno de estos elementos. Adicionalmente, se emplea un subconjunto de parámetros específicos para detallar las propiedades de la interfaz, y se precisan las relaciones entre la interfaz de comunicación y los elementos de señalización en la red, los cuales heredan características del elemento de red principal. Para asegurar la compatibilidad entre el par interfaz-elemento de señal, se utilizan dos clases de asociación que son mapeadas mediante el empleo de datos y reglas, tal como se observa en la Figura 3.5 [7].

De acuerdo con la Figura 3.3, es posible modelar las interfaces de comunicación para los nodos transmisor, receptor y elementos virtuales, así como dos tipos distintos de interfaces correspondientes a dos clases de señales. Ambas interfaces son de tipo único, ya que cada una modela un receptor de superficie, lo cual se denota en el código mostrado en la Figura 3.5. Para cada tipo de interfaz de comunicación, el archivo de MolComML establece características como tamaño, afinidad con tipos de señal, número de receptores, entre otros parámetros [7].

```

<listOfCommunicationInterface>
  <CommunicationInterface id="CI01" name="Absorbing surface" type="virtual">

    <child id="CI01-01" childOf="CI01" name="TX_FullSurface" >
      <outbound enabled="yes">
        <signal refID="NE03-1" type="generic carrier" time="0" unit="s"
          refUnit="U03" affinity="1" internalizeSignal="no"/>
      </outbound>

      <inbound enabled="no">
      </inbound>
    </child>

    <child id="CI01-02" childOf="CI01" name="RX_FullSurface" >
      <inbound enabled="yes" >
        <signal refID="NE03-1" type="generic carrier" time="0" unit="s"
          refUnit="U03" affinity="1" internalizeSignal="yes" />
      </inbound>

      <outbound enabled="no">
      </outbound>
    </child>
  </CommunicationInterface>
</listOfCommunicationInterface>

```

Figura 3.5. Definición de interfaces de comunicación en MolComML [4].

3.2.4 Compartimento (*Compartment*)

Un contenedor que alberga elementos específicos en el entorno de simulación corresponde al componente *Compartment*, el cual se caracteriza por un tipo y una forma geométrica particular, y puede acoger desde sustancias químicas hasta células modeladas o elementos de red. En un escenario de simulación, los compartimentos pueden organizarse en una arquitectura jerárquica, con múltiples compartimentos del mismo tipo, ya sea anidados o dispuestos en el escenario de comunicación a estar cerca entre sí. Las conexiones entre compartimentos se establecen mediante entradas, las cuales definen reglas de cruce (es decir, cómo los elementos se desplazan entre los diferentes compartimentos) e interacción [7].

Cada entrada en un compartimento tiene un conjunto de atributos únicos que incluyen un nombre identificativo, posición, tamaño, forma y orientación. Estos detalles precisan cómo los elementos de la simulación interactúan con el entorno de propagación. Cada elemento de red en un escenario de simulación debe estar albergado en un compartimento [8].

En la Figura 3.6, se presenta un ejemplo que especifica el espacio y los límites del compartimento empleado en una simulación. En este caso, se ha definido un compartimento cúbico con lados de 1 mm y paredes absorbentes. Esto significa que cualquier elemento de red móvil que entre en contacto con las paredes del cubo será eliminado de la simulación, emulando un entorno donde las paredes actúan como barreras que absorben elementos [4].

```
<listOfCompartment>
  <Compartment id="C001" name="unbounded domain" >
    <shape id="SH002" type="cubic" >
      <size name="side" value="1000" unit="um" refUnit="U04" />
    </shape>

    <channel name="free diffusion" refID="CH001" />

    <gate name="" refID="" />
  </Compartment>
</listOfCompartment>
```

Figura 3.6. Definición de compartimento en MolComML [4].

3.2.5 Compuertas (*Gates*)

El bloque *Gates* conceptualiza una lista de entradas que permiten la interconexión entre compartimentos adyacentes o anidados en un entorno de simulación. Cada una de estas entradas es idéntica en su estructura y está compuesta por un conjunto de atributos que incluyen: un nombre único, posición espacial, el tamaño, la forma y su orientación espacial. Estas entradas son cruciales ya que facilitan el traslado de elementos entre las superficies de los compartimentos que conectan, viabilizando la comunicación o la transferencia de elementos entre los mismos [4, 7].

3.2.6 Formas (*Shape*)

La sección *Shape* en el archivo de configuración consta de atributos clave como *position* (posición) y *orientation* (orientación). Esta sección está diseñada como un elemento genérico que se puede usar para representar cualquier tipo de objeto geométrico asociado a los elementos de red, como es el caso de transmisores, receptores o puertas de interconexión entre compartimentos. Las formas geométricas que

pueden emplearse incluyen figuras como cilindros, cubos, entre otras, y cada una de ellas hereda de la sección *Shape* sus dimensiones específicas, y su posición en el compartimento [4, 7].

En la Figura 3.3, por ejemplo, se visualiza la utilización de esferas, como representación geométrica para el transmisor y el receptor. Estas esferas se caracterizan por el valor de su radio, por lo que, en este caso, es necesario configurar los valores de 1000 ηm y 5000 ηm para los extremos comunicacionales, respectivamente. De forma similar, si se modela un vaso sanguíneo, podría emplearse la geometría de un cilindro, especificado mediante su radio, longitud y posición en el entorno de simulación [4, 7].

3.2.7 Reglas de Interacción (*Interaction Rules*)

La definición de reglas de interacción a través del uso de expresiones matemáticas se establece por el componente *Interaction Rules*. Estas reglas pueden ser importadas desde modelos externos, como aquellos formulados en el lenguaje MathML. Tales reglas se pueden instaurar de manera global o como reglas específicas, dependiendo de si se aplican a todo el entorno de simulación o solo a un subconjunto particular de elementos de red [4, 9].

Cada elemento de red y compartimento debe estar asociado, al menos, a una regla de interacción. En esta asociación, se asigna un rol que describe cómo interactúan los objetos implicados. Esto es crucial para definir el comportamiento molecular en situaciones que ocurren en la interacción entre moléculas, como por ejemplo en las colisiones (que producen unión, fusión o rebote molecular), que pueden dar lugar a diversos resultados. Consecuentemente, es esencial especificar claramente los comportamientos que se esperan que ocurran durante la interacción de los elementos considerados en la simulación [7].

En el diagrama UML de la Figura 3.2, se exhiben tres atributos en la clase *role*, que son: *target*, *player* y *undefined*. Estos atributos detallan la naturaleza de la interacción entre los componentes con un objetivo definido. El rol *undefined* se utiliza cuando el papel específico de cada elemento en la interacción no es relevante, como en el caso de una colisión elástica. En la Figura 3.2, también se pone de manifiesto que la clase *Interaction Rules* depende directamente de la clase *role* y sus asignaciones correspondientes, es decir [7]:

1. *Target*. Define el objetivo en la interacción.
2. *Player*. Se emplea cuando hay un solo objetivo definido en la interacción.

3. *Undefined*. Se usa cuando el rol de los elementos en la interacción no tiene importancia significativa.

En la Figura 3.7, se describen dos tipos de interacciones:

1. Regla global. Se aplica a situaciones donde no se consideran significativos los elementos de red o compartimentos involucrados no se consideran significativos, como en el caso de una colisión elástica. En este tipo de interacción, los elementos se definen como *undefined* debido a su papel irrelevante en el proceso de simulación [7].
2. Regla específica. Se utiliza en casos especiales, como cuando un receptor transparente actúa como *target* en la interacción. Este receptor es capaz de interactuar solo con ciertos elementos de red, sin contar aquellos que no son importantes para el proceso. En este caso, el receptor transparente solo cuantifica el número de elementos específicos que llegan a él sin realizar otras acciones adicionales [7]. Por ejemplo, en una simulación, estos elementos de red pueden incluir moléculas de señal o elementos pasivos como los glóbulos rojos, que no juegan un papel activo en las colisiones [7].

También, se expone una tercera regla relacionada con las acciones pertinentes a la cuantificación que sucede cuando un receptor transparente (*target*) interactúa con los elementos de red que monitorea (*player*) [7]. Esta acción se puede interpretar como una colisión o un evento cercano a una colisión, comúnmente referida como *asimilación*, en función de las reglas de compatibilidad ligando-receptor a través de la afinidad (*affinity*) en el proceso de captación de información en su arribo al receptor biológico [7].

Un ejemplo de este caso consistiría en la interacción compleja entre una molécula (*player*) y su receptor (*target*) en una situación de colisión molecular [7, 8]. Este enfoque se plasma en la Figura 3.5, que detalla cómo el elemento de red que actúa como *target* que verifica si la interacción *player* produce el contacto con una interfaz de comunicación compatible en la superficie del receptor. Este tipo de interacción en particular también consta en la Figura 3.3, donde se emplea una regla específica para gestionar la interacción entre las señales y sus receptores biológicos ubicados en las superficies del emisor y en el extremo de recepción de comunicaciones [7].

```

<listOfInteractionRule>
  <globalRule id="IR001" name="generalCollisionRule" type="global">
    <!-- This rule could be a set of math equations -->
    <mathRule id="MR001" name="elasticCollision" path=
      "./mathEquations/resolveCollision.xml" >
      <!-- A list of <param> -->
    </mathRule>
    <!-- role = (1) target: the target of the interaction
                (2) undefined: the role is not important
                (3) player: interaction only with the target -->
    <element type="NetworkElement" refID="*" etype="" role="undefined" targetID=
      "*" />
  </globalRule>

  <specificRule id="IR002" name="transparentCollision" type="collision">
    <mathRule id="MR002" name="transparentCollision" path=
      "./mathEquations/transparent.xml" >
      <!-- A list of <param> -->
    </mathRule>
    <element type="NetworkElement" refID="NE02" etype="receiver" role="target"
      />
    <element type="NetworkElement" refID="NE01" etype="passive" role="player"
      targetID="NE02" />
    <element type="NetworkElement" refID="NE03" etype="signal" role="player"
      targetID="NE02" />
  </specificRule>
</listOfInteractionRule>

```

Figura 3.7. Definición de reglas de interacción en MolComML [7].

3.2.8 Canal de comunicaciones (*Communication Channels*)

El canal de comunicación (*Communication Channels*) es el medio por el cual las moléculas se trasladan hasta alcanzar el receptor. Existen diversos tipos de canales moleculares, que pueden variar en función de las propiedades físicas del entorno de propagación y el mecanismo por el cual las señales se transportan. Entre los más comunes se encuentran aquellos que se basan en la difusión simple (*Diffusion*) y los que operan bajo un proceso de difusión en presencia de arrastre (*Diffusion Drift*). Estos modelos de canales se representan como objetos en la clase *Communication Channels* y son de gran importancia en la simulación del entorno molecular [7].

Cada elemento del canal será definido y vinculado a un compartimento que se encuentre incrustado en el entorno de comunicación. La vinculación del canal con el compartimento es esencial para establecer cómo se comportará el canal en relación con las características locales del entorno. Con este proceso se detallan las condiciones de propagación del canal de transmisión con un alto grado de precisión. Dichas condiciones pueden incluir aspectos como la densidad molecular, la viscosidad del medio, o las fuerzas externas que afectan la propagación de las señales. Para conseguir esta precisión, se usan reglas matemáticas que se definen a través de uno o más archivos en formato MathML, los cuales modelan las ecuaciones que gobiernan el comportamiento de las señales de información en el canal molecular (Figura 3.8) [7].

Además, *Communication Channels* inicializa los parámetros específicos que se emplean en las ecuaciones matemáticas importadas, ofreciendo una flexibilidad considerable en la configuración de diferentes escenarios de simulación. De esta manera, el comportamiento del canal puede adaptarse a los requerimientos de cada simulación, según el tipo de señales que se estén transmitiendo y las propiedades del medio de propagación [7].

La definición del canal puede ser compartida entre uno o más compartimentos, lo que implica que múltiples compartimentos pueden utilizar un único canal de comunicación. Esto es particularmente útil en simulaciones donde los elementos de red interactúan a través de un entorno común o similar, como se ejemplifica en las Figuras 3.3 y 3.8. En estos casos, se usa un modelo de canal en difusión libre para ambas señales, lo que simplifica la representación del movimiento molecular en el entorno simulado [7].

```
<listOfChannel>
  <channel id="CH001" name="free diffusion" type="Diffusion">
    <mathRule id="MR010" name="brownian_diffusion" path=
      "./mathEquations/brownian.xml" >
      <!-- A list of <param> -->
    </mathRule>
  </channel>
</listOfChannel>
```

Figura 3.8. Definición de un canal de comunicación en MolComML [7].

3.2.9 Topología de red (*Network Topology*)

La clase *Network Topology* es responsable de detallar el desplazamiento relativo de los elementos de red en el entorno simulado. Esta descripción incluye una lista de elementos que pertenecen a cada compartimento, junto con su posición inicial y su estado al comienzo de la simu-

lación. El objetivo principal de esta clase es administrar la organización espacial de los elementos de red y su interacción en el compartimento, modelando en forma precisa el entorno de la simulación [8].

Para establecer una referencia entre la topología y los elementos de red o compartimentos, esta clase maneja una asociación pormenorizada entre las diferentes clases implicadas. Esta asociación define claramente las relaciones entre los elementos, determinando cómo interactúan entre sí en el modelo de CM a simular. Las reglas que rigen el comportamiento y el movimiento de los elementos de red pueden ser definidas manualmente, especificando los parámetros individuales para cada elemento, o de manera automática, usando fórmulas matemáticas que controlan estos aspectos [7].

En la Figura 3.9, se ilustran dos elementos clave de red: un nodo transmisor y un nodo receptor, ambos ubicados en el compartimento previamente descrito en las secciones anteriores. Esta figura destaca no solo la posición inicial de cada nodo, sino también su estado antes de que la simulación comience. En este caso, los nodos representados están fijos en el compartimento, lo que significa que no cambian su posición durante la simulación. Esto concuerda con lo expuesto en la Figura 3.3, donde los elementos como el transmisor y el receptor son estáticos, mientras que las señales moleculares que representan la información son generadas durante la ejecución de la simulación, lo que añade dinamismo al proceso [7].

```
<networkTopology>
  <disposedCompartment id="DC001" refID="C001" name="unbounded domain" rule="single"
  >
    <position refShape="SH002" name="coords" value="array(x=0; y=0; z=0)"
    unit="" refUnit="" />
    <disposedElement id="DE001" refID="NE01-1" name="transmitter" rule=
    "single" >
      <position refShape="SH001" name="coords" value="array(x=5; y=0;
      z=0)" unit="um" refUnit="U04" />
      <initialization refID="IS01" />
    </disposedElement>

    <disposedElement id="DE002" refID="NE02-1" name="receiver" rule="single">

      <position refShape="SH001" name="coords" value="array(x=0; y=0;
      z=0)" unit="um" refUnit="U04" />
      <initialization refID="IS02" />
    </disposedElement>
  </disposedCompartment>
</networkTopology>
```

Figura 3.9. Definición de una topología de red en MolComML [7].

3.2.10 Estados (*State*)

La clase *state* denota el estado interno de cada uno de los elementos de red, tal como consta en el diagrama UML de la Figura 3.2. Para esto, utiliza la relación *internalState* como referencia. Cada estado incluye una lista de acciones específicas que determinan el comportamiento del elemento en cuestión, junto con el tiempo requerido para completar dichas acciones antes de que se produzca una transición hacia el siguiente estado. Este enfoque permite modelar el comportamiento dinámico de los elementos de red a lo largo del tiempo, capturando cómo interactúan y evolucionan en la simulación [7].

Adicionalmente, cada estado puede estar vinculado a un elemento de una pila de protocolos, lo que se visualiza en la Figura 3.2. Tanto el transmisor como el receptor poseen estados internos que refieren el protocolo general que se ejecuta en ambos nodos. Esto es crucial para garantizar que los procesos de comunicación entre los nodos sigan un conjunto coherente de reglas y comportamientos, independientemente del tipo de señal o canal en uso [7, 10].

3.2.11 Pila de protocolos (*Protocol Stack*)

Los elementos de red se vinculan al objeto *Protocol Stack* a través de la clase *Internal State*. En este contexto, las acciones específicas asociadas a cada estado se enuncian mediante el protocolo Finite State Machine (FSM), que puntualiza detalladamente el protocolo que se ejecuta en cada uno de los elementos de la red. Con esto se puede configurar pilas de protocolos con diferentes organizaciones, personalizadas según la estructura de niveles e identificaciones requeridas [8, 4].

Cada capa en estas pilas de protocolos puede ser mapeada a sistemas de telecomunicaciones convencionales o bien ser definida como una nueva estructura. Cada capa contiene un conjunto de reglas que particularizan el comportamiento del protocolo. Esto es, por ejemplo, la capa física incluye una lista de símbolos asociados, los cuales están definidos por una serie de parámetros tales como: ráfagas de información, tipo de portadora, y duración de la ráfaga. Cada símbolo debe estar vinculado a un mensaje para poder definir la secuencia de símbolos correspondiente [8, 4].

Para poder personalizar la configuración de estos elementos, se inicializan los parámetros pertinentes a través de un archivo XML estándar, como se muestra en la Figura 3.10. En el ejemplo, se proponen dos

capas protocolarias [7]:

1. Capa 1. Es la responsable de la transmisión y recepción de ráfagas que codifican o detectan los símbolos.
2. Capa 2. Es la encargada de la decodificación de los mensajes, además de gestionar el nivel de señal para evitar la congestión en el receptor y enviar señales de retroalimentación (feedback).

Un caso particular que se cita en la Figura 3.3 es el de un protocolo bidireccional, que imita el funcionamiento de TCP (Transmission Control Protocol), pero adaptado a la segunda capa en la arquitectura propuesta. Asimismo, en el marco de MolComML, se permite la definición de algoritmos específicos para cada protocolo, por ejemplo, aquellos diseñados para regular el tamaño de ráfagas de información [7].

```

<listOfProtocolStack>
  <protocolStack id="PS01" name="Emission of a Single Burst of carriers" type=
  "SingleBurst" maxLevel="2">
    <layer level="1" name="Physical">
      <rule id="PR1-01" type="on_off_keying modulation" >
        <mathRule id="MRO20" name="impulse" path=
        ". /mathEquations/impulse2.xml" >
          <!-- A list of <param> -->
        </mathRule>

        <symbol id="SYMO01" symbolCode="1" >
          <signal id="MOL001" type="carrier" refID="NE03-1"
          sequenceOrder="1" >
            <param type="value" name="burst" value="500"
            unit="" refUnit="" />
            <param type="value" name="duration" value="0"
            unit="s" refUnit="U03" />
          </signal>
        </symbol>

        <symbol id="SYMO02" symbolCode="0" >
          <signal id="MOL002" type="carrier" refID="NE03-1"
          sequenceOrder="2" >
            <param type="value" name="burst" value="0" unit=""
            refUnit="" />
            <param type="value" name="duration" value="0"
            unit="s" refUnit="U03" />
          </signal>
        </symbol>
      </rule>
    </layer>
    <layer level="2" name="signal_pattern">
      <rule id="PR2-01" type="messages" >
        <message id="MES001" name="singleImpulse" symbolSequence "[1 0]"
        symbolRef=" [SYMO01 SYMO 02]" />
        <!-- e.g. symbolSequence "[1 0 0 1 01]"-->
      </rule>
    </layer>
  </protocolStack>
</listOfProtocolStack>

```

Figura 3.10. Definición de la pila de protocolos en MolComML [7].

3.2.12 Programador de eventos (*Event Scheduler*)

El programador de eventos (*Event Scheduler*) es un componente clave en la administración de eventos en un entorno de simulación molecular. Este módulo actúa como un controlador que supervisa y regula las actividades relacionadas con los elementos de red. Para llevar a cabo este proceso, este programador usa la clase *EventHandler*, la cual monitorea un conjunto de atributos clave, tales como nombre (*name*) y descripción (*description*). Estos atributos son fundamentales para identificar variaciones significativas en los valores de los elementos de la simulación y, cuando se detectan estos cambios, desencadenan el evento correspondiente [6].

La lógica de estos eventos se establece a través de la clase *Condition*, donde el comportamiento esperado de un evento se detalla en la sección CDATA (*Character DATA*)^{xi}. En esta sección, se emplea pseudocódigo para especificar de forma clara las condiciones bajo las cuales deben activarse ciertos eventos. Este enfoque posibilita que *Event Scheduler* configure el estado inicial de cada elemento de red y, a su vez, regule los eventos particulares que causarán transiciones de estado en un nodo destino, ya sea como resultado de un evento detectado o en momentos preprogramados que ocurrirán más adelante [6].

Por ejemplo, en la Figura 3.3, se constituye un evento programado en el que el receptor (Rx) inicia un proceso de búsqueda para localizar un nodo transmisor (Tx). En este evento específico, se pone en marcha una secuencia de acciones entre un par de nanomáquinas biológicas. Estas nanomáquinas emplean técnicas originalmente diseñadas para el protocolo TCP. Sin embargo, a diferencia de este protocolo, no se utilizan acuses de recibo explícitos en este entorno, ya que estos acuses podrían afectar negativamente el rendimiento de la comunicación debido al retraso significativo que caracteriza a las CM [11].

El programador de eventos *EventScheduler* gestiona una serie de eventos que provocan transiciones de estado en la máquina FSM. Durante la simulación, tanto la información de los elementos de transmisión y recepción y los detalles de los eventos programados, se registran de manera periódica en una base de datos MySQL (My Structured Query Language). Esta exportación constante de datos mantiene un registro preciso de la dinámica de la simulación y facilita su análisis posterior [8].

^{xi} *Character DATA* es una sección que se procesa como bloques de texto que no son examinadas por el analizador, pero si es reconocida por XML.

3.2.13 Elementos de salida

En el contexto de MolComML, se puede configurar un esquema por-menorizado de la información de salida, el cual permite especificar una lista de elementos que se exportarán durante la simulación. Estos elementos incluyen tanto los elementos de red como los compartimentos, y la selección de atributos específicos que serán monitoreados, además de, los intervalos de tiempo en los que se realizará dicho monitoreo. El manejo de los atributos se puede personalizar configurando reglas específicas, lo que otorga una gran flexibilidad al sistema [8].

Este formato de salida en MolComML es fundamentalmente importante pues facilita la interoperabilidad entre diferentes simuladores. Los resultados obtenidos en una simulación pueden ser exportados y usados como información de entrada para otro simulador, debido a la uniformidad en el formato de la información. Este hecho destaca las características modulares e integradoras de MolComML, en el cual diversos programas pueden interactuar y compartir datos de manera eficiente empleando este formato común [8].

La clase *OutputFormat* es responsable de definir el formato que tendrá la información de salida. En esta clase se configuran las acciones de exportación y el tipo de archivo de texto en el que se almacenarán los resultados de la simulación. El esquema estándar de salida suele consistir en una matriz de datos generada a partir de un evento que es activado bajo condiciones específicas, previamente establecidas por *EventScheduler*. Las reglas de salida recogen los resultados provenientes de los elementos de red o la información asociada a los compartimentos. Para administrar este proceso, se utiliza la clase *AbstractOutputObject*, que se vincula directamente con los objetos *OutputNetworkElement* y *OutputCompartments*, como es notorio en la Figura 3.2 [7, 8].

Este esquema no solo optimiza la manera en que se gestionan los resultados de las simulaciones, sino que también asegura que cualquier información relevante pueda ser transferida, procesada y reusada en simulaciones futuras, proveyendo así una mayor continuidad en estudios científicos posteriores que emplean simulaciones moleculares [7, 8].

3.2.14 Elementos de soporte

En el diagrama UML de la Figura 3.2, se puede observar, en el lado izquierdo, un conjunto de clases de apoyo que desempeñan un papel fundamental en la estructura de MolComML. Estas clases describen varios aspectos esenciales que suministran el funcionamiento correcto de los elementos modelados en el sistema. Entre los componentes gestionados por estas clases se encuentran atributos, parámetros, unidades de medida, y fórmulas matemáticas [10, 12].

3.2.14.1 Parámetro (*Param*)

En MolComML, es posible definir una lista de atributos personalizados para cada elemento en un archivo de configuración, lo que otorga una notable flexibilidad para ampliar los atributos predefinidos o introducir otros completamente nuevos. Esta capacidad de personalización es primordial para la adaptación de los requerimientos específicos de la simulación y flexibilizar una representación concreta de las entidades biológicas y su comportamiento en un entorno controlado [10, 12].

La clase *Param* administra estos atributos personalizados. En ella, se establecen parámetros que pueden funcionar tanto como atributos predefinidos como nuevos atributos usados normalmente para la inicialización de entidades biológicas [7, 10, 12]. Los principales atributos de esta clase se incluyen a continuación [10, 12]:

1. *type* y *name*. Determinan el tipo de información y el nombre del parámetro respectivamente.
1. *value* y *unit*. Se refieren al valor del atributo y su unidad de medida correspondiente.
1. *description*. Provee una breve explicación o descripción del parámetro.

El atributo *type* es particularmente importante, ya que puede asumir valores específicos que detallan el tipo de información que se encuentra en el atributo *value*. El atributo *value* debe ser numérico y, dependiendo de la naturaleza de la información, puede clasificarse en varios tipos [12, 13]:

1. *value*. Se utiliza para representar datos numéricos simples.
2. *ref*. Sirve como referencia a otros elementos definidos en el mismo archivo de configuración de MolComML, basándose

en un ID único. Es adecuado para establecer relaciones entre diferentes componentes de la simulación.

3. *array*. Se emplea cuando se necesita representar matrices numéricas, las cuales son utilizadas frecuentemente para describir posiciones en sistemas de coordenadas o conjuntos de datos.
4. *string*. Se destina a descripciones textuales, como aquellas empleadas en el objeto *OutputParam* para exportar información específica relacionada con tal parámetro.
5. *internal*. Se refiere a sub-atributos relacionados con fórmulas matemáticas que se especifican en los archivos MathML adjuntos. Estos atributos internos permiten calcular comportamientos más complejos en la simulación.

Este enfoque estructurado y altamente flexible facilita la creación de configuraciones personalizadas que modelan con precisión una amplia variedad de escenarios en el campo de las CM y su consecuente simulación de procesos biológicos [12, 13].

3.2.14.2 Unidades (*unit*)

En MolComML, se define el objeto *unit* que representa la unidad de medida asociada con el atributo *value* para cada parámetro o atributo numérico. Este proceso es crucial para garantizar la coherencia y precisión de las simulaciones, especialmente cuando se trabaja con magnitudes físicas o biológicas que requieren una correcta representación en unidades [7, 8, 12].

La clase *AbstractUnit* es fundamental en este contexto, ya que sirve como clase base para los objetos *Unit* y *UnitDefinition*, los cuales configuran unidades de medida tanto simples como personalizadas. Estas unidades personalizadas son el resultado de combinaciones de unidades estándar, y se gestionan a través del objeto *SubUnit*, que incluye los atributos *value* y *exponent*. Este enfoque flexibiliza la definición de unidades, posibilitando la representación de magnitudes más complejas cuando es necesario [7, 8, 12].

Para cada unidad definida, se requiere proporcionar [7, 8, 12]:

1. Un ID único. Identifica la unidad en el sistema.
2. Un nombre. Detalla la denominación de la unidad.

3. Una escala. Ajusta el valor de la unidad en relación con otras magnitudes.

Cuando se trata de unidades personalizadas, el formato específico debe incluir el valor y el exponente, lo cual combina adecuadamente diferentes unidades estándar. Esto resulta conveniente, por ejemplo, en ecuaciones matemáticas complejas donde es necesario expresar magnitudes compuestas por múltiples unidades. Un ejemplo de esta personalización se puede observar en la Ecuación 3.1, donde se combinan unidades estándar mediante exponentes para representar magnitudes más complejas. Es importante destacar que se han mantenido los términos en inglés en la ecuación para preservar la correspondencia con los nombres y valores de los objetos tal como se emplean en MolComML [7, 8, 12].

$$\text{customUnit} = (\text{value} \times 10^{\text{scale}})^{\text{exponent}} \quad (3.1)$$

Este enfoque no solo asegura una correcta interpretación de las unidades de medida en la simulación, sino que también permite extender las capacidades del sistema para adaptarse a escenarios específicos, como los que podrían encontrarse en simulaciones de procesos biológicos o físicos altamente estructurados [7, 8, 12].

3.2.14.3 Reglas matemáticas (*Math Rule*)

La clase *Math Rule* se especifica mediante un archivo externo en formato MathML. Seguidamente, para cada regla matemática importada, se define un conjunto de parámetros que describe e inicializa un objeto en la simulación a través de *direction*. La administración de este componente está dada por los siguientes atributos [4]:

1. *direction=out*. Se utiliza para obtener los resultados de una fórmula.
2. *direction=in*. Se usa para inicializar un valor.
3. *direction=internal*. Se define en el mismo archivo MathML.

En algunos casos se requiere inicializar un parámetro con valores definidos en otra sección del fichero MolComML, como es el caso del tamaño de un elemento de red [4].

De acuerdo a la Ecuación 3.2, el atributo *value* se establece por la referencia correspondiente al valor que un elemento adquiere por los parámetros que se explicarán a continuación [7, 8]:

$$NE02.size.name[radius].value \quad (3.2)$$

- **NE02.** Esta cadena representa el ID único del elemento para un componente específico en el archivo MolComML, en este caso, un elemento de red.
- **Size.** Denota al atributo dado por el tamaño del elemento. Es uno de los atributos fijos en la jerarquía de este elemento.
- **Name.** Particulariza a uno de los atributos hijos del elemento extraído, por lo que, hace alusión al nombre puntual del atributo.
- **[radius].** Es el valor asignado al atributo anterior, es decir, indica el radio del elemento.
- **Value.** Este atributo retorna el valor asociado pertinente al radio del elemento.

El resultado de esta notación devuelve el valor del atributo *radius* del elemento de red que está asociado al ID único *NE02* [7].

3.3

EXPERIMENTACIÓN EN MolComML

MolComML, como lenguaje especializado en simulaciones de CM, se basa en el software BiNS2 (Biological and Nano-Scale Communication Simulator, en la Sección 2.2.2 del Capítulo 2 se analiza BiNS2), como su plataforma subyacente para la ejecución de simulaciones. Dado que BiNS2 proporciona la infraestructura requerida para realizar las simu-

laciones, es fundamental garantizar que antes de comenzar el proceso de simulación se hayan instalado correctamente todos los paquetes, métodos, clases y archivos que forman parte del simulador [14].

Para inicializar la simulación se debe contar con un entorno IDE (Integrated Development Environment), para gestionar y organizar los archivos y paquetes concernientes al correcto funcionamiento del simulador. Todos los componentes esenciales, como clases, librerías y métodos de BiNS2, deben descargarse previamente desde los repositorios en línea. Estos archivos, de fácil acceso en la web, están disponibles en fuentes oficiales citadas en [14].

El proceso de instalación de estos paquetes es un paso crítico, ya que asegura que el entorno esté adecuadamente configurado para soportar la ejecución de simulaciones de MolComML. La correcta instalación de los componentes no solo garantiza que el simulador funcione de manera óptima, sino que también permite que el usuario se concentre en la fase de simulación sin inconvenientes técnicos que puedan retrasar el proceso [14].

Para llevar a cabo los experimentos descritos, se ha empleado el software Apache NetBeans en su versión 15. Esta herramienta proporciona un entorno adecuado para la creación y gestión de proyectos, facilitando la implementación del código necesario para las simulaciones [14].

Además, para el almacenamiento y administración de resultados obtenidos durante las simulaciones, se han usado XAMPP (Cross-platform (X), Apache (A), MariaDB (M), PHP (P), Perl (P)) y phpMyAdmin, que actúan como un conjunto de herramientas para la gestión de bases de datos. XAMPP proporciona un servidor web local, mientras que phpMyAdmin facilita la interacción y gestión de las bases de datos mediante una interfaz gráfica de usuario, permitiendo a los investigadores realizar consultas, administrar tablas y visualizar los datos de manera intuitiva [14].

Un aspecto importante del proceso de simulación es la correcta configuración de un archivo XML. Este archivo es esencial para iniciar el simulador y garantizar el acceso conveniente a las bases de datos pertinentes. En este archivo, se incluye una sección específica denominada `<database>`, que alberga los atributos necesarios para establecer la conexión con una base de datos local. Se debe señalar que, en cada simulación, se inicializa una nueva base de datos para almacenar los resultados específicos de dicha ejecución [14].

En la Figura 3.11, se presenta un fragmento de la configuración de cómo se realiza el acceso a la base de datos, brindando una referencia visual para entender la estructura y los componentes del archivo XML [14].

```
<Database>
  <address>localhost</address>
  <port>3306</port><!-- 3307 localhost, 3306 remote -->
  <db>name_of_database</db>
  <db-backup>nano_saved</db-backup>
  <user>name_of_user</user>
  <password>password_of_user</password>
</Database>
```

Figura 3.11. Configuración XML del acceso a la base de datos local en MolComML.

3.4

CASOS DE ESTUDIO DE SIMULACIONES EN MolComML

Una vez que se han identificado y configurado todos los elementos requeridos para el correcto funcionamiento de MolComML, es posible iniciar el proceso de simulación. En este contexto, se han diseñado y ejecutado dos escenarios de simulación utilizando MolComML como herramienta en el simulador BiNS2 [15].

El primer escenario está enfocado en el estudio de la difusión de la información biológica a través de un canal de propagación molecular. Este tipo de simulación es crucial para entender cómo se distribuyen y desplazan las partículas en un entorno determinado. Este análisis es fundamental en sistemas de CM, pues se necesita caracterizar la propagación de moléculas para la transmisión/recepción de información en ambientes como fluidos o tejidos biológicos [15].

El segundo escenario, en cambio, explora una aplicación en el ámbito de la medicina. En este caso, el enfoque es la detección de Células Tumorales (CT) circundantes en vasos sanguíneos. Este tipo de simulación plantea una solución potencial para problemas biomédicos complejos, donde se busca diseñar herramientas de diagnóstico avanzado mediante la implementación de nanomáquinas que interactúan a nivel molecular para identificar células malignas. Estos escenarios de CM reflejan cómo los simuladores pueden ser empleados no solo para

entender los fenómenos de transmisión de información en seres vivos, sino también para desarrollar aplicaciones innovadoras en la detección y tratamiento de enfermedades [15].

- **Caso 1. Escenario de simulación # 1: entornos de propagación molecular**

En la primera simulación, el enfoque se centra en el estudio de la propagación molecular por difusión, un proceso básico en CM. Para este caso, se establecen tres distancias distintas para el enlace de comunicación, cada una permite evaluar cómo varía el comportamiento de las moléculas en función de la distancia entre las nanomáquinas emisoras y receptoras. Las moléculas de información se emiten en ráfagas, bajo un movimiento Browniano libre que describe cómo las partículas se desplazan de manera aleatoria en un medio líquido [15].

La modulación que se usa es OOK (On-Off Keying), técnica en la que el bit-1 se representa mediante la emisión de 1000 moléculas, mientras que para el bit-0 no se transmite ninguna concentración molecular. Este enfoque es adecuado para simulaciones de CM, ya que simplifica el proceso de transferencia de información. En este tipo de comunicación, la emisión o no emisión de moléculas transmite la información de manera efectiva, ajustándose a la naturaleza discreta de los sistemas moleculares [15].

Las nanomáquinas transmisoras (Tx) y receptora (Rx) se modelan como esferas absorbentes en posición fija. Estas condiciones propenden a la interacción entre las moléculas y las nanomáquinas, permitiendo que las moléculas se absorban al llegar al receptor, lo que asegura una correcta recepción de la información. La geometría esférica no solo resuelve los cálculos, sino que también indica cómo las nanomáquinas pueden interactuar en un entorno molecular [15].

Para garantizar la precisión y confiabilidad de los resultados obtenidos con BiNS2 y MolComML, se realizó una comparación con el simulador N3Sim (en la Sección 2.2.3 del Capítulo 2 se analiza N3Sim). N3Sim es una herramienta reconocida para la simulación de la propagación molecular, y su utilización generalizada en la investigación de CM respalda la validez de los resultados obtenidos. Este simulador permite la ejecución de instrucciones de programación mediante comandos en la consola de Windows, lo que potencia su implementación en estudios comparativos [15].

Los valores específicos de los elementos de la transmisión para cada simulador se presentan en la Tabla 3.1, donde se incluyen detalles sobre la descripción de cada parámetro, sus valores numéricos, las unidades correspondientes y los identificadores que se usan en la configuración del archivo XML [15].

Tabla 3.1. Valores configurados en MolComML. Simulación correspondiente a entornos de propagación molecular.

Descripción	Objeto XML	Identificador XML	Valor	Unidad
Tamaño del transmisor	<code><networkElement> <size> type= "sphere"</code>	<code>name= "Tx_node" "radius","value"</code>	50	nm
Posición del transmisor	<code><networkTopology> <position></code>	<code>"coords" "transmitter", "value= array"</code>	$(x=0; y=0; z=0)$	μm
Tamaño del receptor	<code><networkElement> <size> type= "sphere"</code>	<code>name= "Rx_node" "radius","value"</code>	2	μm
Posición del receptor	<code><networkTopology> <position></code>	<code>"coords","receiver" "value= array"</code>	$(x=3; y=0; z=0)$ $(x=5; y=0; z=0)$ $(x=15; y=0; z=0)$	μm
Tamaño de moléculas	<code><networkElement> <size> type= "signal"</code>	<code>"burst", "radius","value"</code>	2	nm
Distancia	---	---	3, 5, 15	μm
Burst size	---	---	1000	u

Una vez que se han completado las configuraciones correspondientes, se ejecuta cada una de las simulaciones. La Figura 3.12 ilustra los resultados obtenidos al gestionar una distancia de propagación corta, de 3 μm , con el empleo de MolComML y N3Sim. Los resultados manifiestan que los dos simuladores se comportan similarmente, evidenciando una correlación en cuanto a la distribución y propagación de las moléculas en este rango de distancia [15].

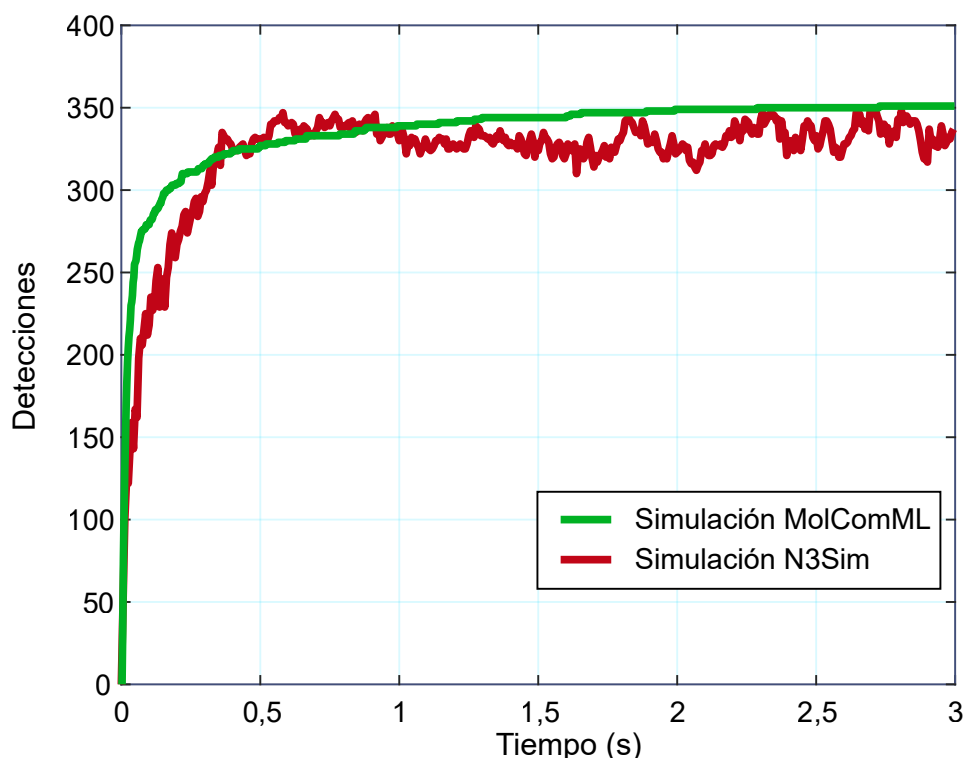


Figura 3.12. Resultado de las simulaciones para el caso de propagación en un entorno de difusión molecular. La distancia del enlace molecular es de 3 μm .

De manera análoga, las Figuras 3.13 y 3.14 revelan los resultados de las simulaciones para las distancias de 5 μm y 15 μm , respectivamente. En estos escenarios, los resultados obtenidos en los simuladores MolComML y N3Sim continúan mostrando una notable semejanza, lo que refuerza la coherencia en los resultados que cada simulador produjo. La correspondencia entre ambos simuladores confirma que, para diferentes rangos de distancia, los modelos de propagación molecular para ambas plataformas siguen siendo robustos y consistentes [15].

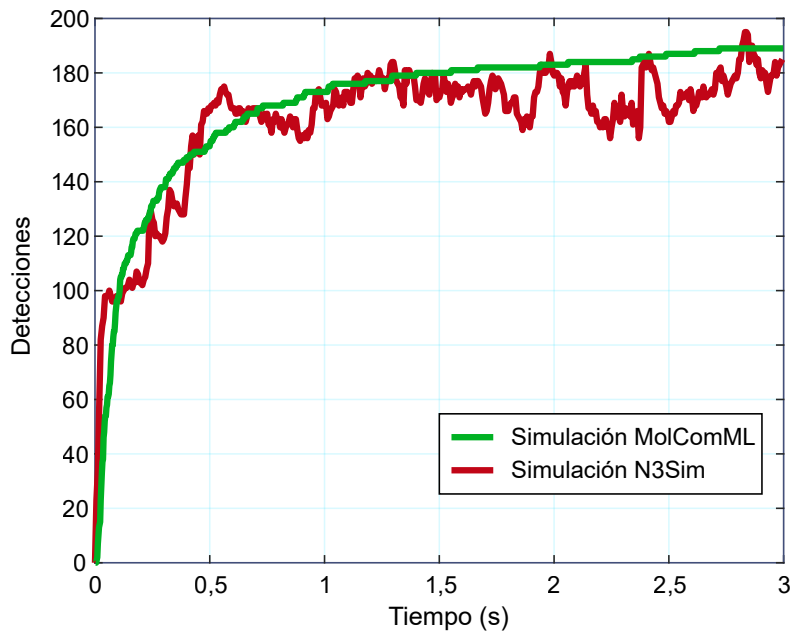


Figura 3.13. Resultados de las simulaciones para el caso de propagación en un entorno de difusión molecular. La distancia del enlace molecular es de 5 μm .

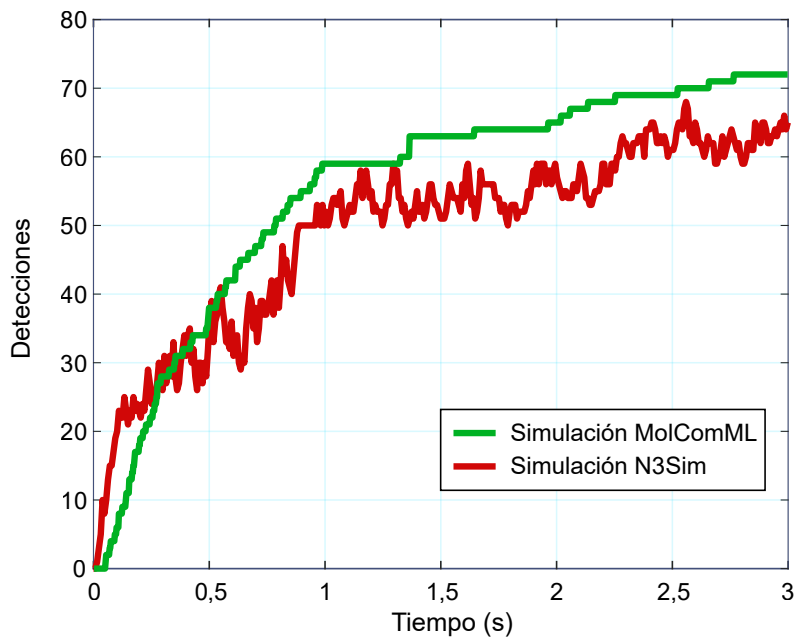


Figura 3.14. Resultados de las simulaciones para el caso de propagación en un entorno de difusión molecular. La distancia del enlace molecular es de 15 μm .

Los resultados de las simulaciones previamente efectuadas corroboran que en un sistema de comunicación biológico, en entornos de difusión molecular, la distancia de separación entre las nanomáquinas Tx y Rx impacta significativamente el proceso de transferencia de información. De este modo, la cantidad de moléculas recibidas en el extremo destino (Rx) disminuye de forma proporcional inversa con la distancia del enlace molecular. Esto se debe a que, cuanto mayor es la distancia, menor es la probabilidad de que las moléculas lleguen al receptor, lo que afecta directamente la eficiencia del sistema de CM [15].

Este fenómeno es crucial en el diseño de sistemas de CM, porque la distancia de propagación es un factor clave en el intercambio de información, y tiene que ser seleccionada en forma adecuada para optimizar la comunicación entre nanomáquinas [15].

Como es notable en las figuras anteriores, las simulaciones indican una considerable similitud en cuanto al comportamiento de las dos herramientas de programación de CM, particularmente en la cantidad de moléculas que establece una comunicación apropiada entre las nanomáquinas en los extremos de comunicación. Durante todo el proceso de la simulación, los patrones de difusión molecular siguen un curso similar en ambos casos, lo que indica que los modelos teóricos subyacentes son coherentes y que el proceso de propagación molecular se lleva a cabo de manera par en ambos simuladores [15].

No obstante, surge una diferencia importante en términos de la eficiencia del muestreo y la calidad de recepción de los elementos depositados en el entorno de simulación cuando se emplea MolComML. Los gráficos generados con MolComML se caracterizan por una mayor suavidad, lo que sugiere que este entorno de simulación maneja los datos con mayor precisión y consistencia. Esto puede atribuirse por cómo MolComML gestiona los procesos de simulación, con una mayor capacidad para modelar la dispersión molecular de manera fluida y continua [15].

En cambio, la irregularidad en los gráficos de N3Sim puede reflejar fluctuaciones en el procesamiento de los eventos moleculares, lo cual podría impactar en la precisión al modelar las interacciones en entornos complejos [15].

La similitud en los resultados sugiere que, MolComML y N3Sim ofrecen un rendimiento comparable, lo que confirma la fiabilidad de MolComML al replicar escenarios moleculares con alta precisión en comparación con simuladores reconocidos como N3Sim. Esto también destaca que el modelo de difusión molecular y los algoritmos utilizados para ambos

simuladores son robustos para este tipo de configuración, proporcionando resultados consistentes y coherentes en cuanto a la propagación y recepción de moléculas en un entorno controlado [15].

- **Caso 2. Escenario de simulación # 2: detección de Células TumORAles (CT) en vasos sanguíneos**

En el segundo caso de análisis, se consideran los estudios realizados por [16, 17]. Estas investigaciones aportan los parámetros fundamentales para configurar una simulación que refleje con la mayor precisión posible un entorno biológico real. Consecuentemente, en base a los estudios en [16, 17], se selecciona como escenario de simulación una sección de vaso sanguíneo de 6 mm de longitud y de radio 30 μm .

En este entorno biológico, se deposita una célula tumoral y un receptor transparente de CT (CTRx), el cual se mantiene en una posición fija durante toda la simulación. Este receptor se comporta como un destino biológico especializado, el cual se puede emplear en aplicaciones médicas para el diseño, identificación y las reacciones biológicas pertinentes por la presencia de CT cercanas que se encuentran en el flujo sanguíneo [15-17].

En el entorno de simulación además, se incluyen otros elementos biológicos, como glóbulos rojos, blancos, y plaquetas. Estos componentes adicionales se incorporan para producir una connotación más real en las interacciones biológicas entre elementos que ocurren en el flujo sanguíneo, ejerciendo influencia en el movimiento y la propagación de las moléculas y células involucradas. La dinámica entre estos elementos y la célula tumoral ofrece un escenario de simulación con alta complejidad biológica, acercando los resultados al comportamiento real del cuerpo humano [15-17].

Esta simulación no solo permite observar la difusión y comportamiento de las moléculas tumorales, sino que también, posibilita el análisis de cómo los diferentes componentes sanguíneos y las características físicas del vaso influyen en el proceso de detección celular. Estos factores son decisivos para la precisión en el comportamiento de una célula tumoral en un entorno vascular y la evaluación correspondiente de la efectividad en la recepción y detección de CT [15-17].

Este segundo caso es muy importante para realizar aproximaciones prácticas que pueden servir en aplicaciones médicas, en las que la detección de CT circulantes es fundamental en diagnósticos tempranos y en el desarrollo de terapias personalizadas [15-17].

En este análisis, se definieron cuatro escenarios de simulación con el objetivo de estudiar el comportamiento de las CT en distintos entornos y configuraciones. Los parámetros indispensables en estos escenarios son, la distancia de separación entre la célula tumoral y el receptor CTRx, y el radio de la célula tumoral [15-17].

Las distancias de separación entre una célula tumoral y el CTRx en los escenarios de simulación son de 6 mm y 3 mm, respectivamente, estableciendo dos casos distintos para evaluar cómo la proximidad entre la célula tumoral y su receptor afecta el tiempo de detección y, en última instancia, la eficiencia del sistema de monitoreo. Asimismo, para el radio de la célula tumoral, se configuran dos valores representativos de diferentes tamaños celulares, 5 μm y 2.5 μm , lo que viabiliza la evaluación de cómo el tamaño de la célula afecta su movimiento dentro del vaso sanguíneo y el tiempo que tarda en llegar al receptor [15-17].

El estudio de este conjunto de simulaciones se centra en el tiempo de tránsito que la célula tumoral requiere para recorrer el vaso sanguíneo hasta llegar al receptor CTRx bajo estas condiciones. El objetivo principal es determinar el impacto que la distancia y el tamaño de la célula ejercen en la capacidad del receptor para identificar la célula tumoral de manera conveniente [15-17].

Los detalles completos de estos parámetros y de los objetos e identificadores XML usados para configurar cada simulación, se proporcionan en las Tablas 3.2 y 3.3. Estas tablas incluyen información clave como, las descripciones de los elementos, sus valores numéricos, dimensiones o unidades de medida e identificadores específicos en el archivo de configuración XML [15].

Tabla 3.2. Parámetros definidos para la simulación de los elementos presentes en el torrente sanguíneo [4, 15].

Descripción	Parámetro	Objeto XML	ID XML	Valor	Unidad
Glóbulos rojos	Concentración	$\langle networkTopology \rangle$ $\langle disposedElement \rangle$ $\langle mathRule \rangle$	name: "RBC concentration"	$5 \cdot 10^6$	U/mm ³
	Radio	$\langle networkElement \rangle$ $\langle size \rangle$ type= "sphere"	name: "RBC" "radius";"-value"	2,9	μm
Glóbulos blancos	Concentración	$\langle networkTopology \rangle$ $\langle disposedElement \rangle$ $\langle mathRule \rangle$	name: "WBC concentration" "value"	$7,5 \cdot 10^3$	U/mm ³
	Radio	$\langle networkElement \rangle$ $\langle size \rangle$ type= "sphere"	name: "WBC" "radius";"value"	3,8	μm
Plaquetas	Concentración	$\langle networkTopology \rangle$ $\langle disposedElement \rangle$ $\langle mathRule \rangle$	name: "PLA concentration"	$2,5 \cdot 10^5$	U/mm ³
	Radio	$\langle networkElement \rangle$ $\langle size \rangle$ type= "sphere"	name: "PLA" "radius";"-value"	1	μm

Tabla 3.3. Parámetros definidos para la simulación de vasos sanguíneos, CT y CTRx [4, 15].

Parámetros del vaso sanguíneo, CT y CTRx				
Descripción	Objeto XML	ID XML	Valor	Unidad
Radio del vaso sanguíneo	<networkElement> <size> type= "tube"	name: "radius tube" "value"	30	µm
Longitud de la sección del vaso		name: "length tube" "value"	6	mm
Radio de CTRx	<networkElement> <size> type= "sphere"	name: "radius CT counter" "value"	5	µm
Radio de CT	<networkElement> <size> type= "sphere"	name: "radius CT" "value"	5; 2,5	µm
Distancia entre CT y CTRx	---	---	6, 3	mm

Al efectuar la configuración y ejecutar el proceso de simulación con los parámetros previamente referidos, se analizan los resultados del escenario propuesto mediante métodos estadísticos. Este análisis confirma que la detección de CT en los vasos sanguíneos depende de manera contundente de dos factores principales: la distancia de separación entre la CT y el receptor CTRx, y el radio o tamaño de la célula maligna. Estos factores ejercen una influencia significativa en el tiempo de detección, o el tiempo que se requiere para que la célula tumoral alcance su destino en el CTRx [15].

Las variaciones en los tiempos de detección de CT se relacionan directamente con la naturaleza dinámica del entorno del vaso sanguíneo, que no es homogéneo. En este entorno existen glóbulos rojos y blancos, y además plaquetas, los cuales generan colisiones que alteran el desplazamiento de la célula tumoral. Debido a esta interacción, las CT tienen un movimiento aleatorio y no lineal, lo que provoca variaciones en el tiempo de detección en el CTRx [15].

La Figura 3.15 a) consta de un diagrama de cajas que confirma que la distancia de separación entre la célula tumoral y su receptor afecta notablemente el tiempo de su detección. Es evidente que, cuando la distancia de separación es menor, el tiempo requerido para la detección

de la célula tumoral también disminuye de modo notable. Este comportamiento manifiesta la importancia de la proximidad entre la célula tumoral y su receptor en el proceso de detección [15].

Por otra parte, en la Figura 3.15 b) se demuestra en el diagrama de cajas cómo el tamaño o radio de las CT influye en el tiempo de su detección. En este caso, al comparar células con radios de 2.5 μm y 5 μm , se observa que la distribución de los valores de tiempo de detección es similar en promedio para ambos radios. Sin embargo, un análisis más detallado revela que la dispersión de los valores es menor cuando el radio de la célula tumoral es de 2.5 μm , lo que indica que el tamaño de la célula también tiene un impacto directo en el tiempo necesario para que la célula tumoral alcance a su CTRx [15].

En última instancia, los resultados de las simulaciones para este segundo caso de escenarios de CM confirman que la distancia y el tamaño de la célula tumoral son factores determinantes en el tiempo de detección de las mismas [15].

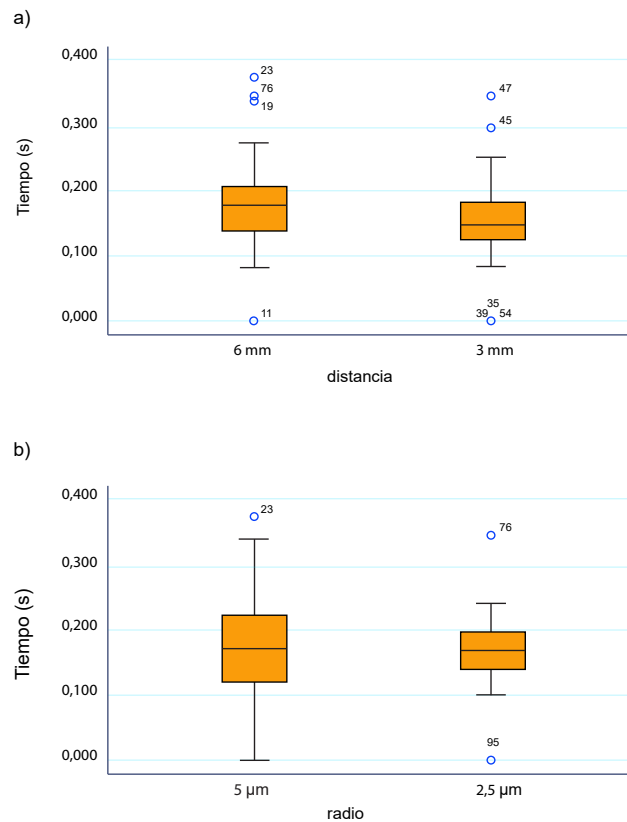


Figura 3.15. a) Diagrama de cajas, distribución de los valores del tiempo de detección de CT en relación con la distancia. b) Diagrama de cajas para los valores del tiempo de recorrido en relación con el radio de las CT [15].

Los dos aspectos preponderantes que influyen en la dinámica del comportamiento del flujo sanguíneo y la correspondiente detección de CT de forma análoga a como se sugiere en [16, 17] son:

- La concentración de glóbulos rojos tiende a ser mayor en el eje central del vaso sanguíneo, lo que indica que las interacciones y colisiones en esta zona son más frecuentes, afectando la movilidad y desplazamiento de las CT.
- La velocidad de propagación del fluido también es mayor en el eje axial del vaso sanguíneo, lo que implica que las células ubicadas cerca de esta región experimentan una mayor fuerza de arrastre y, por tanto, un desplazamiento más rápido hacia su destino.

A partir de estas dos condiciones, se puede profundizar adicionalmente la influencia que tiene la ubicación inicial de la célula tumoral y del receptor CTRx con relación al eje central del vaso sanguíneo. Con lo cual, se concluye que la posición de las células en el vaso modifica el tiempo de detección de CT y las interacciones que tienen lugar en el entorno sanguíneo propiamente [15].

Entonces, la Tabla 3.4 proporciona un esquema de las ubicaciones establecidas para determinar la variación en la interacción de las CT y el CTRx, para lo cual se consideran distintas posiciones a lo largo del vaso sanguíneo, incluyendo aquellas cercanas al eje central y otras ubicaciones de mayor aproximación a las paredes del vaso. Este enfoque analiza si las condiciones de flujo y la concentración de células en diferentes regiones del vaso implican cambios sustanciales en el tiempo de detección de CT [15].

Tabla 3.4. Posiciones definidas para la detección de CT y el CTRx.

Elemento	Coordenadas (x; y; z) μm		
	Posición 1	Posición 2	Posición 3
CT	0; 0 ;0	10; 0 ;10	16,25; 0 ;19
CTRx	0; 6; 0	10; 6; 10	16,25; 6; 19

En la primera parte de la simulación que se ejecuta para verificar la dependencia posicional de las CT y el CTRx, se ubica a la nanomáquina CTRx en el eje central del vaso sanguíneo, tal como se especifica en la Tabla 3.4. En tanto, la célula tumoral se posiciona en el origen referencial durante todo el tiempo de simulación. Esta ubicación inicial de la célula tumoral tiene como propósito estudiar su comportamiento en el vaso y cuantificar el tiempo que tarda en alcanzar la nanomáquina CTRx. Con estas características de simulación, el énfasis en la obtención de resultados radica en evidenciar las trayectorias aleatorias de la célula tumoral en este entorno biológico [15].

Los resultados de las interacciones en este sistema de CM revelan que solo en el 10% de las simulaciones la nanomáquina CTRx detecta la célula tumoral. Esta baja tasa de detección se relaciona directamente con la presencia de elementos biológicos en el vaso debido a la alta concentración de células (glóbulos rojos, blancos, y además plaquetas) y el intenso movimiento molecular que ocurre en el eje central del vaso sanguíneo. Las colisiones y las dinámicas de interacción entre estos elementos impiden que las CT obedezcan una trayectoria lineal hacia su CTRx [15].

Para facilitar la interpretación de estos resultados, las figuras correspondientes al vaso sanguíneo se proyectan en el plano (x, z). En estas figuras, el radio total del vaso está representado por una línea discontinua de color naranja, y una línea de puntos que indica la zona endotelial. Adicionalmente, un círculo de color negro identifica la ubicación del CTRx, situada directamente sobre la posición inicial de la célula tumoral (Figura 3.16) [15].

En la Figura 3.16, se puede observar el desplazamiento aleatorio que presentan las células malignas en el interior del vaso (colores variados). En la mayoría de las simulaciones, la célula tumoral permanece relativamente cerca del eje central del vaso; no obstante, en algunas instancias, la célula se desplaza hacia los extremos sin regresar al centro, donde se encuentra el CTRx. Esto ocurre debido a la mayor concentración de elementos celulares en el eje del vaso y a la mayor velocidad de propagación en esa región. Las colisiones entre las células del entorno provocan una repulsión que empuja a las CT hacia las paredes del vaso, disminuyendo por tanto, la probabilidad de que sean detectadas por el CTRx en esta porción simulada del vaso sanguíneo [15].

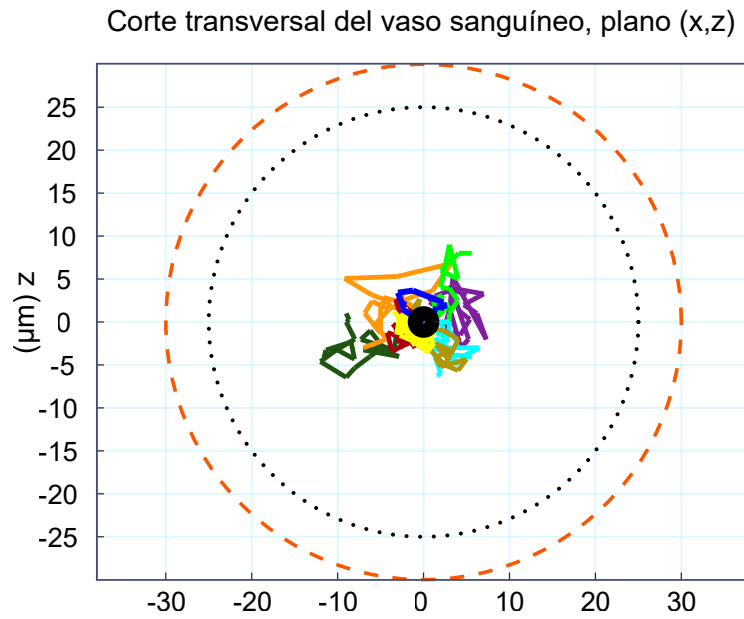


Figura 3.16. Simulación basada en la ubicación de la célula tumoral y el CTRx en el eje central del vaso sanguíneo [15].

En contraste, la Figura 3.17, traza las trayectorias de simulaciones independientes, donde se comparan los movimientos de las CT que fueron interceptadas por el CTRx con aquellas que no establecieron comunicación [15]. Además, es notable el tiempo que la célula tumoral tarda en recorrer el vaso hasta llegar al CTRx. En esta figura, la trayectoria interceptada se representa en color rojo, destacando que la célula tumoral fue detectada en un tiempo de 72 ms. Sin embargo, hay que señalar que la duración del tiempo de detección varía entre simulaciones, dependiendo de las diferentes trayectorias que la célula tumoral establezca a través del movimiento aleatorio que la gobierna. Por esta razón, el tiempo de interceptación podría ser mayor o menor en otros escenarios simulados [15].

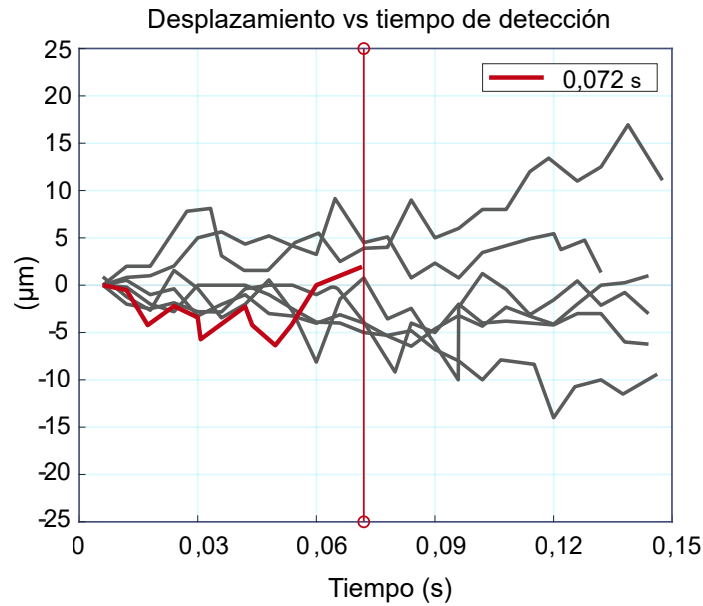


Figura 3.17. Trayectorias descritas por las CT en varias simulaciones. La línea de color rojo representa el recorrido y el tiempo de detección de una célula tumoral.

En la segunda fase de la simulación en la que se varía la posición de las CT y el CTRx, se puede apreciar una mejor detección que la nanomáquina CTRx efectúa sobre las CT. En estas 30 nuevas simulaciones, la tasa de detección de las CT se incrementa aproximadamente en 30%, lo que supone un aumento considerable en este segundo proceso de detección en comparación con el primero. Sin embargo, se mantiene la tendencia predominante en el comportamiento dinámico de las CT mientras se desplazan hacia el endotelio del vaso sanguíneo. Esta tendencia puede explicarse por la ubicación inicial de las CT, que aunque se posicionen más alejadas del eje central, todavía se pueden encontrar en zonas de alta concentración de células más grandes, como glóbulos rojos y blancos, lo que genera colisiones que proyectan las CT hacia las paredes del vaso sanguíneo [15].

En la Figura 3.18 se verifica este fenómeno, es decir, los resultados de varias simulaciones expresan que el 30% de los casos corresponden a trayectorias detectadas por el CTRx. En todas estas simulaciones, las CT constan de un patrón de desplazamiento hacia el endotelio, reafirmando la hipótesis de que las colisiones con otras células son un factor preponderante en el movimiento de las CT [15].

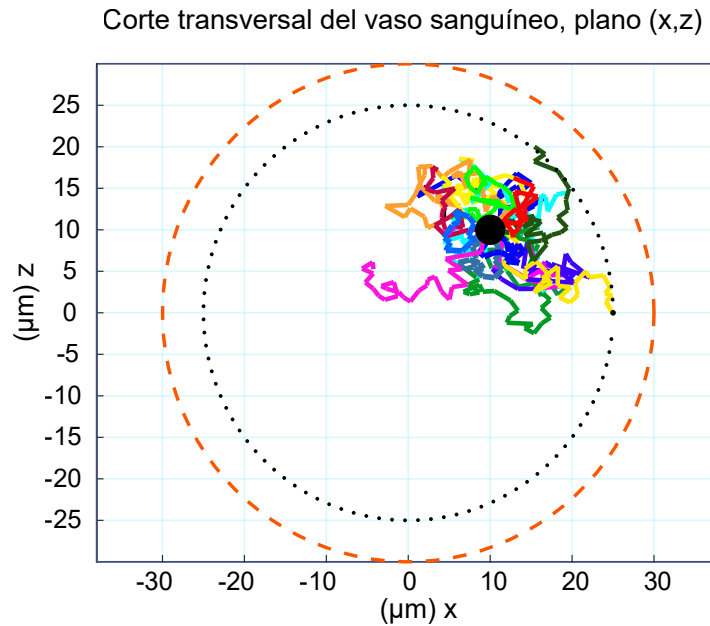


Figura 3.18. Simulaciones realizadas en base a la ubicación desplazada de las CT en relación al eje central del vaso sanguíneo [15].

Concomitantemente, la Figura 3.19 plasma el desplazamiento de las CT (analizado en las simulaciones previas) y los tiempos de detección asociados a cada una de ellas. Como se esperaba, los tiempos en los que el CTRx detecta la célula tumoral varían entre simulaciones, con valores de 97 ms, 116 ms y 125 ms. Esta variabilidad en los tiempos de detección es ocasionada en gran medida, por las diferentes trayectorias de las CT, las que recorren rutas más largas o cortas en función de su interacción con el entorno celular [15].

Otro aspecto importante por señalar es que, dependiendo de la ubicación de las CT y el CTRx, los tiempos de detección tienden a ser mayores en comparación con la primera fase de simulación. Este hecho ocurre porque, en la zona más alejada del eje central del vaso sanguíneo, la concentración y el movimiento celular son significativamente menores, lo que ralentiza el desplazamiento de las CT. La reducción en el número de colisiones permite que las CT tengan trayectorias más estables, pero también incrementa el tiempo necesario para que el CTRx las intercepte [15].

En la Figura 3.19, las líneas de color verde, cian y rojo describen las trayectorias de las CT en función del tiempo requerido para detectarlas y asociarlas al CTRx [15].

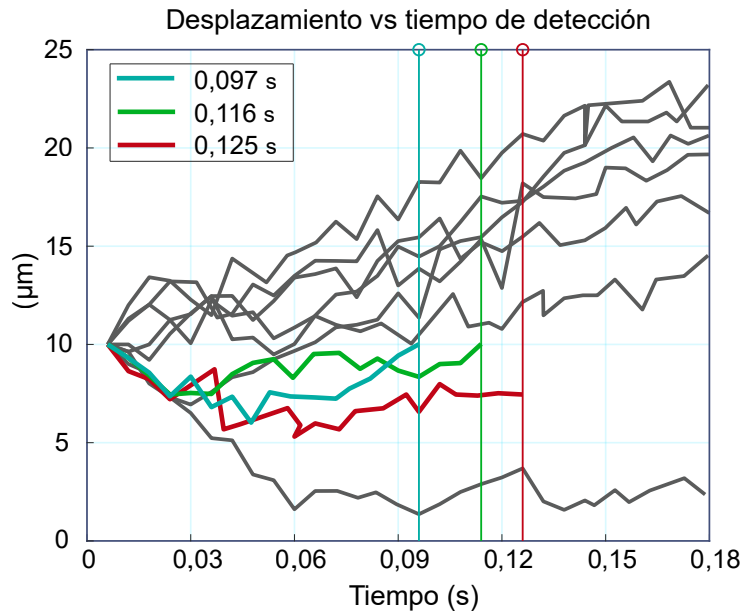


Figura 3.19. Trayectorias descritas por CT en varios escenarios de CM. Las líneas de color cian, rojo y verde representan el recorrido y el tiempo de detección de las CT, y las líneas de color gris determinan su no detección [15].

Después de completar las simulaciones e interpretar los resultados para los sistemas de CM propuestos, y teniendo en cuenta que las CT tienden a desplazarse hacia las zonas alejadas del eje central del vaso sanguíneo, se puede modificar la posición de la nanomáquina receptora CTRx hacia la región cercana al endotelio y también la posición original de las CT, según se expone en la Tabla 3.4. Estas variaciones en la configuración de la simulación pretenden mejorar la detección que realiza el CTRx, debido a la tendencia de las CT a acercarse al endotelio por lo que, nuevas ubicaciones del CTRx pueden incrementar las probabilidades de interceptación [15].

Consecuentemente, las simulaciones efectuadas sugieren que las CT se orientan en forma continua hacia el endotelio. Este fenómeno permite que la detección que lleva a cabo el CTRx sea más efectiva. En esta configuración, la tasa de detección aumenta aproximadamente en 40% comparada con las simulaciones anteriores, logrando que el CTRx identifique una célula tumoral en el 70% de los casos [15].

La Figura 3.20 indica que en la mayoría de las simulaciones, las CT permanecen próximas al endotelio, donde la actividad celular es considerablemente menor, facilitando su detección. Sin embargo, en algunos casos, se advierte un ligero desplazamiento radial de las CT, lo que provoca la desviación de su trayectoria hacia el CTRx, impidiendo su detección [15].

Corte transversal del vaso sanguíneo, plano (x, z)

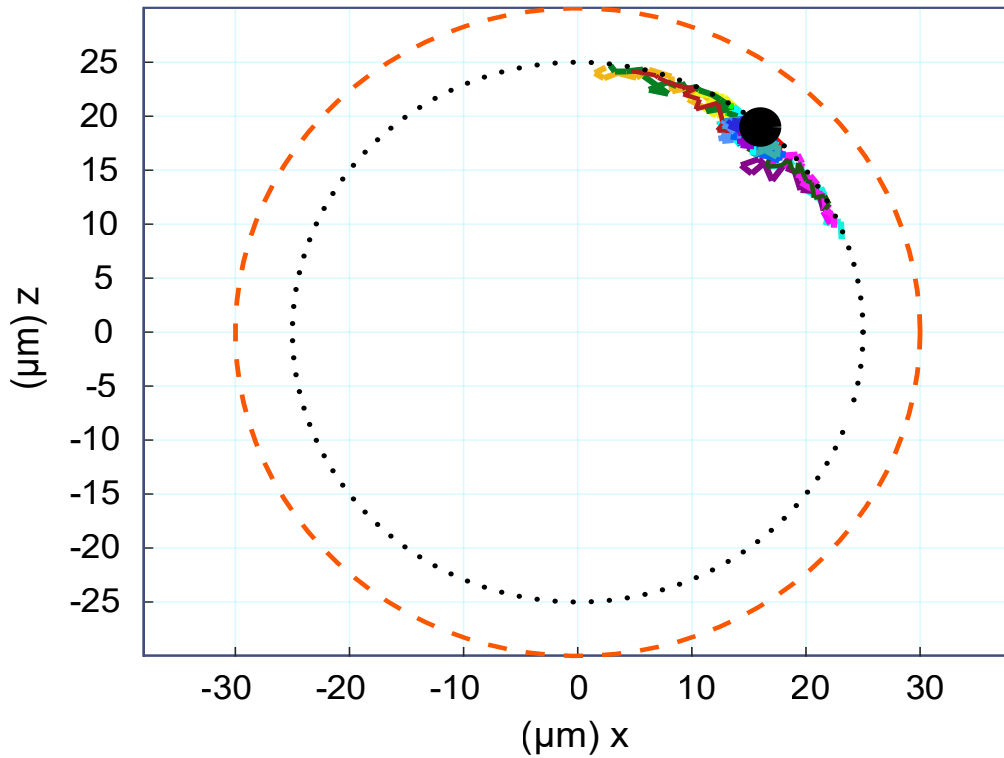


Figura 3.20. Simulaciones obtenidas al posicionar las CT y el CTRx cercanos al endotelio [15].

Finalmente, la Figura 3.21 expone una serie de trayectorias de desplazamiento de las CT al ser detectadas en el destino molecular; también, se pone de manifiesto el tiempo que tardan en recorrer el vaso sanguíneo hasta alcanzar el CTRx. En esta figura se pueden visualizar la variación en los tiempos de detección de las CT en las diferentes simulaciones, lo cual es coherente con la naturaleza aleatoria del movimiento de este tipo de células en el entorno biológico. A pesar de que los tiempos de recepción son diferentes, el nuevo posicionamiento del CTRx en el vaso, resulta ser más efectivo para interceptar las CT, especialmente en comparación con las simulaciones anteriores, donde la nanomáquina receptora estaba ubicada en el eje central del vaso [15].

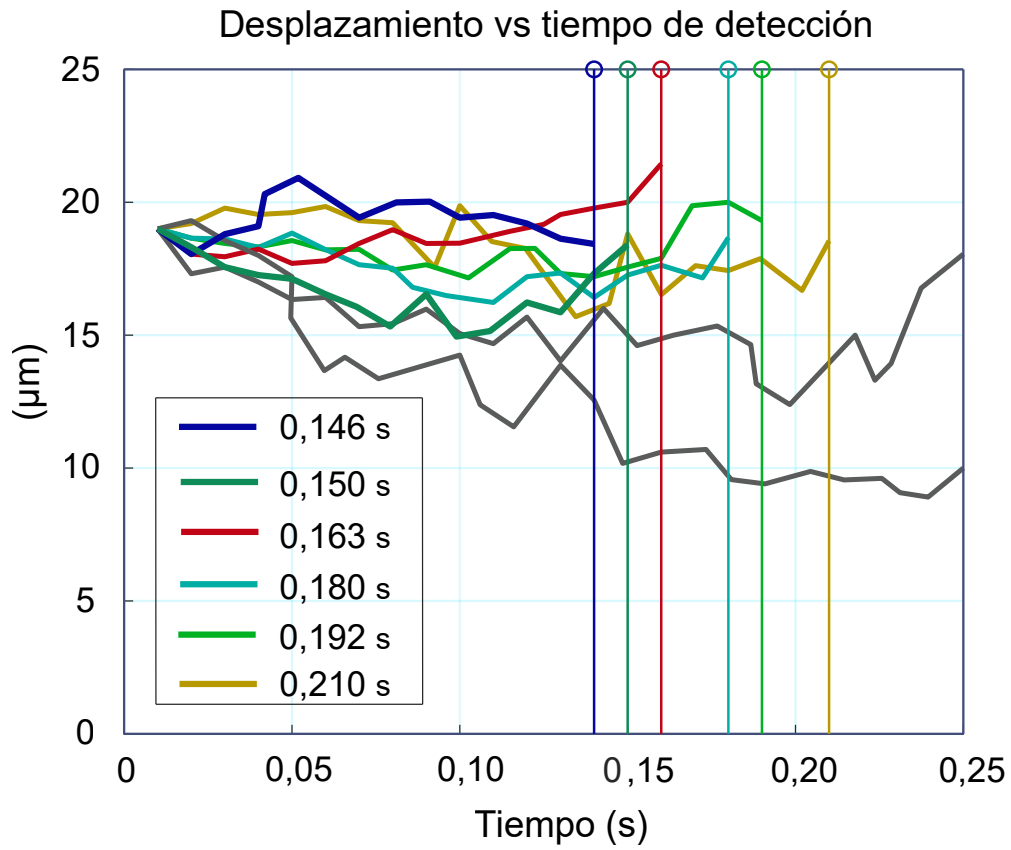


Figura 3.21. Trayectorias resultantes de las simulaciones en las que se modifican las posiciones de los elementos nanomáquina transmisora y nanomáquina receptora aproximándolas al endotelio. Las líneas en distintos colores señalan detecciones de las CT[15].

Referencias bibliográficas

- [1] G. Reali *et al.*, “Simulation tools for molecular communications,” *TCSIM Newsl.*, no. 22, pp. 8–12, 2016.
- [2] M. Galdzicki *et al.*, “The Synthetic Biology Open Language (SBOL) provides a community standard for communicating designs in synthetic biology,” *Nat. Biotechnol.*, vol. 32, no. 6, pp. 545–550, 2014.
- [3] S. Neymotin, “Integrating Systems Biology Markup Language (SBML) with NEURON”.
- [4] L. Felicetti, S. S. Assaf, M. Femminella, G. Reali, E. Alarcon, and J. Sole-Pareta, “The Molecular Communications Markup Language (MOL-COMML),” *Nano Communication Networks*, vol. 16, pp. 12–25, Jun. 2018, doi: <https://doi.org/10.1016/j.nancom.2018.03.001>.
- [5] F. Achard, G. Vaysseix, and E. Barillot, “XML, bioinformatics and data integration,” *Bioinformatics*, vol. 17, no. 2, pp. 115–125, 2001.
- [6] H. B. Yilmaz, A. C. Heren, T. Tugcu, and C.-B. Chae, “Three-dimensional channel characteristics for molecular communications with an absorbing receiver,” *IEEE Commun. Lett.*, vol. 18, no. 6, pp. 929–932, 2014.
- [7] L. Felicetti, S. S. Assaf, M. Femminella, G. Reali, E. Alarcon, and J. Sole-Pareta, “The molecular communications markup language (Mol-ComML),” *Nano Commun. Netw.*, vol. 16, pp. 12–25, 2018.
- [8] S. S. Assaf, “Synthetic biology guidelines for diffusion based molecular communication,” 2019.
- [9] N. Farsad, H. B. Yilmaz, A. Eckford, C.-B. Chae, and W. Guo, “A comprehensive survey of recent advancements in molecular communication,” *IEEE Commun. Surv. Tutorials*, vol. 18, no. 3, pp. 1887–1919, 2016.
- [10] I. Llatser *et al.*, “Exploring the physical channel of diffusion-based molecular communication by simulation,” in *2011 IEEE Global Telecommunications Conference-GLOBECOM 2011*, 2011, pp. 1–5.
- [11] L. Felicetti, M. Femminella, G. Reali, T. Nakano, and A. V Vasilakos, “TCP-like molecular communications,” *IEEE J. Sel. Areas Commun.*, vol. 32, no. 12, pp. 2354–2367, 2014.

[12] C. Schölzel, V. Blesius, G. Ernst, and A. Dominik, "Characteristics of mathematical modeling languages that facilitate model reuse in systems biology: A software engineering perspective," *NPJ Syst. Biol. Appl.*, vol. 7, no. 1, pp. 1-20, 2021.

[13] A. Akkaya, G. Genc, and T. Tugcu, "HLA based architecture for molecular communication simulation," *Simul. Model. Pract. Theory*, vol. 42, pp. 163-177, 2014.

[14] «Networks and Services Lab.,» Faculty of Engineering at the University of Perugia, [Online]. Available: <http://conan.diei.unipg.it/lab/index.php/product/biological-nanoscale-simulator-bins2>. [Último acceso: 15 septiembre 2024].

[15] E. Pilligua and Y. Cevallos, "Simulación de sistemas de comunicación biológicos en entornos de difusión y de células tumorales en vasos sanguíneos mediante la herramienta de nanocomunicaciones a nivel molecular MolComML," Universidad Nacional de Chimborazo, 2024. [Online]. Available: <http://dspace.unach.edu.ec/handle/51000/12908>

[16] D. A. Haber and V. E. Velculescu, «Blood-based analyses of cancer: circulating tumor cells and circulating tumor DNA» *Cancer discovery*, vol. 4, nº 6, pp. 650-661, 2014.

[17] K. G. Phillips, C. R. Velasco, J. Li, A. Kolatkar, M. Luttgen, K. Bethel, B. Duggan, P. Kuhn and O. J. McCarty, «Optical quantification of cellular mass, volume, and density of circulating tumor cells identified in an ovarian cancer patient. *Frontiers in Oncology*. 2012 ;2:72. doi: <https://doi.org/10.3389/fonc.2012.00072>



cedia

El sello editorial de la Corporación Ecuatoriana para el Desarrollo de la Investigación y la Academia - CEDIA, nace con la finalidad de apoyar a la creación y la publicación de resultados, investigaciones y procesos académicos, que fomenten el desarrollo de la ciencia y la innovación a nivel nacional e internacional.



ISBN: 978-9942-7317-4-6



9 789942 731746